**International Journal of Data Science**

ISSN online: 2053-082X - ISSN print: 2053-0811
https://www.inderscience.com/ijds

# Review of spectral clustering algorithms used in proteomics

Shraddha Kumar, Anuradha Purohit, Sunita Varma

**DOI:** 10.1504/IJDS.2023.10054559

**Article History:**

| | |
|---|---|
| Received: | 09 January 2022 |
| Last revised: | 10 March 2022 |
| Accepted: | 28 March 2022 |
| Published online: | 09 March 2023 |

# Review of spectral clustering algorithms used in proteomics

Shraddha Kumar*, Anuradha Purohit
and Sunita Varma

Department of Computer Engineering,
Shri G.S. Institute of Technology and Science,
Indore, 452003, MP, India
Email: shraddhakumar@sgsits.ac.in
Email: apurohit@sgsits.ac.in
Email: sverma19@sgsits.ac.in
*Corresponding author

**Abstract:** Tandem mass spectrometry (MS/MS) generates a large number of spectra showing the signal intensity of detected ions as a function of mass-to-charge ratio. Spectral clustering in proteomics is a powerful but under-utilised technique. Based on the similarity of spectra, the spectral clustering algorithms systematically and unerringly classify large numbers of spectra, such that all spectra in a given cluster belong to the same peptide. The data points in the spectral clustering approach are connected and do not require having convex boundaries. Spectral clustering therefore reduces the running time and computation requirements of spectral library and database searches. It enhances peptide identification process and has fuelled the development of many new proteomics algorithms recently. The goal of this review is to provide a clear overview of the most popular spectral clustering algorithms used in proteomics. It describes a systematic analysis of these spectral clustering algorithms, evaluating the benefits and limitations of each approach.

**Keywords:** proteomics; tandem mass spectrometry; spectral clustering; consensus spectrum; scoring function; mass spectra; data points; spectral similarity; cluster purity; spectral library; normalised dot product.

**Reference** to this paper should be made as follows: Kumar, S., Purohit, A. and Varma, S. (2023) 'Review of spectral clustering algorithms used in proteomics', *Int. J. Data Science*, Vol. 8, No. 1, pp.16–38.

**Biographical notes:** Shraddha Kumar is a PhD scholar in the Department of Computer Engineering. Her area of research includes soft computing, deep learning and software engineering.

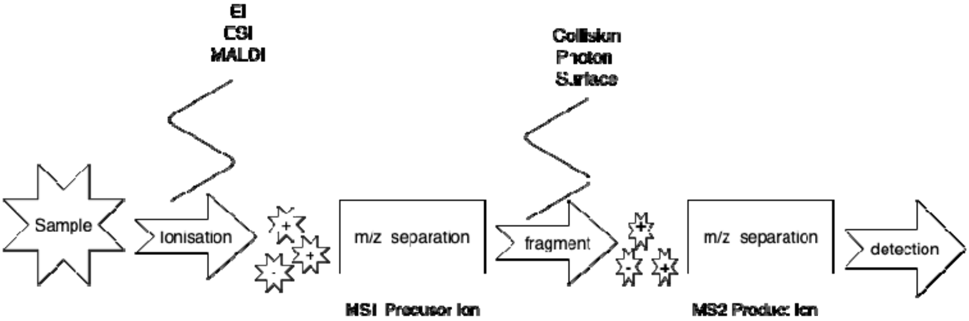
Anuradha Purohit is working as an Associate Professor in the Department of Computer Engineering. Her area of research includes soft computing, machine learning, theory of computation and software engineering.

Sunita Varma is working as a Professor and Head in the Department of Information Technology from February 2017. Her research area is cloud computing, big data analytics and related areas.

## 1 Introduction

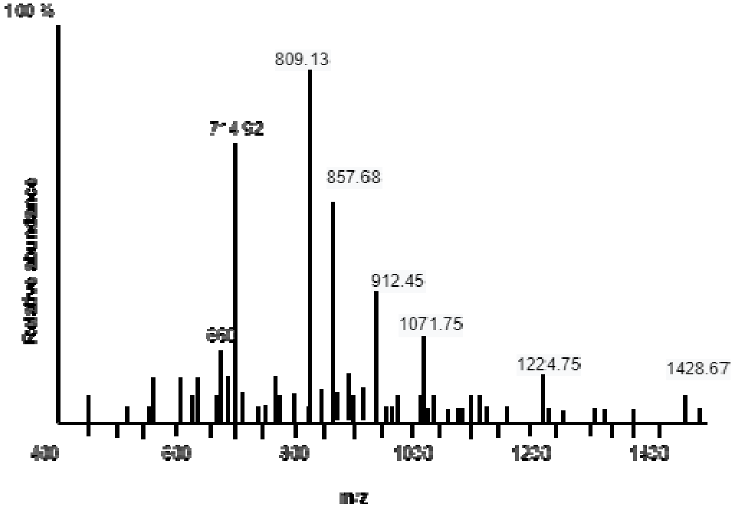
Proteins are the large, complex molecules made up of long chains of amino acids. They are the fundamental blocks of all cellular processes and their detection, quantification and characterisation is often confounded by their many proteoforms and complex chemistry. The standard technology that provides fast, high-throughput characterisation of complex protein mixtures is mass spectrometry-based shotgun proteomics (also known as Tandem Mass Spectrometry or MS/MS or MS2). The schematic of Tandem Mass Spectrometry is shown in Figure 1 (Nationalmaglab.org).

**Figure 1** Schematic of tandem mass spectrometry (MS/MS)



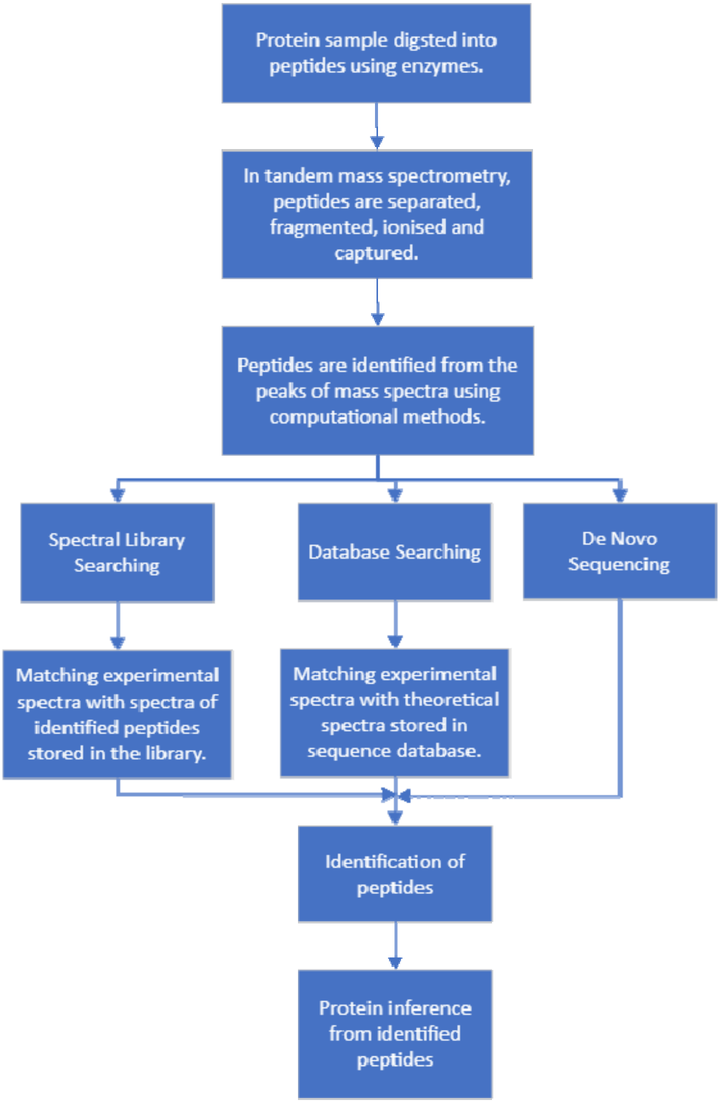
Initially, proteins are digested (cleaved) into smaller amino acid chains known as peptides that are then passed into a mass spectrometer. This instrument is used to measure the intact mass-to–charge (m/z) ratio of peptides (protein fingerprint); the peptides are then isolated in the mass spectrometer and fragmented into shorter chains to generate mass spectrum signatures for each peptide as shown in Figure 2.

**Figure 2** Mass spectra generated from tandem mass spectrometry (MS/MS). To provide clarity minor lines with peak heights of 2% or less of the base peak (the tallest peak) are omitted



Peptides can be identified by matching the experimental and computational simulated spectra using peptide database searching algorithms, spectral library searching or Denovo sequencing. Finally, the peptide profiles are aggregated to report which proteins were more likely to produce the observed peptide set. Overall steps involved in the process of protein identification from tandem mass spectra starting from sample digestion using enzymes to protein inference is illustrated in Figure 3.

**Figure 3** A classical workflow model for protein inference from MS/MS data (see online version for colours)


Protein sample digsted into peptides using enzymes.
In tandem mass spectrometry, peptides are separated, fragmented, ionised and captured.
Peptides are identified from the peaks of mass spectra using computational methods.
Spectral Library Searching
Database Searching
De Novo Sequencing
Matching experimental spectra with spectra of identified peptides stored in the library.
Matching experimental spectra with theoretical spectra stored in sequence database.
Identification of peptides
Protein inference from identified peptides

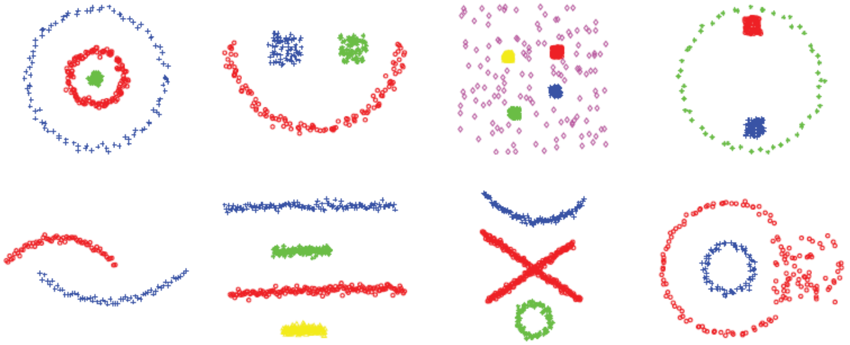
Thousands of peptide fragment ion spectra are produced in each proteomics experiment analysing complex protein mixtures from biological samples. The tandem mass spectra generated from bottom-up proteomics consist of mass-to-charge ratios and relative abundances of a set of fragment ions generated from digested peptides as shown in Figure 2. The patterns of these fragment ions are useful for the identification and quantification of proteomes in the sample. More precisely, the goal of protein investigation is to accurately and quantitatively infer the proteins (output) that give rise to the peptides observed in the sample (Kim et al., 2017). However, the most challenging part in the process is the identification of ms/ms datasets, that is assigning peptides to the mass spectra.

The remainder of this paper is as follows: Section 2 presents the background of spectral clustering. In Section 3, the significance of spectral clustering in proteomics is highlighted. Section 4 lays out a broad review of spectral clustering algorithms in proteomics. Section 5 provides a short discussion on the reviewed algorithms. Finally, Section 6 contains concluding remarks about spectral clustering algorithms.

## 2 Background

Many datasets can be notoriously difficult to cluster with traditional methods. Figure 4 demonstrates a few toy datasets (Zelnik-Manor et al., 2004), which are difficult for traditional clustering algorithms. On such datasets, algorithms which implicitly assume specific shapes of clusters cannot achieve good results. For example, the Euclidian distance metrics assume a convex shape to the underlying clusters. Obviously, such assumptions can impact the quality of the clustering in arbitrary datasets.

**Figure 4** Examples of datasets which cannot be clustered using traditional clustering algorithms (see online version for colours)



*Source*: Zelnik-Manor et al. (2004)

Another disadvantage of the traditional algorithms is related to the inherent challenges in the expectation maximisation (EM) framework, which is often used to learn a mixture model for clustering. This framework is essentially an iterative process of finding local minima, and therefore multiple restarts are required to find a good solution. On the other hand, spectral clustering can solve problems in much more complex scenarios, such as intertwined spirals, or other arbitrary nonlinear shapes, because it does not make assumptions on the shapes of clusters.

The history of spectral clustering can be traced back to Wilm, and Donath (1973) in which it was suggested that the eigenvectors of the adjacency matrix could be used in order to determine the underlying partitions. Typically, this matrix is derived from a set of pairwise similarities between the points to be clustered. This task is called similarity-based clustering or graph clustering. The main difference among spectral clustering algorithms is whether they use normalised or unnormalised ‘graph Laplacian’ methods. Different versions of spectral clustering have been successfully applied to image segmentation (Shi and Malik, 2000), text mining (Inderjit, 2001), speech processing (Francis and Bach, 1963), and general-purpose methods for data analysis and clustering (Inderjit et al., 2004, Ding et al., 2005, Ng et al., 2001, Zelnik-Manor et al., 2004). This success of spectral clustering has encouraged researchers to use it in proteomics for the identification of peptides. An excellent review on the history of spectral clustering can be found in Daniel and Spielman (1996). The spectral clustering process can be viewed as a three-step algorithm as shown in Figure 5:

> “One exceptional advantage of spectral clustering is its ability to cluster ‘points’ which are not necessarily vectors, and to use for this a “similarity”, which is less restrictive than a distance. A second advantage of spectral clustering is its flexibility; it can find clusters of arbitrary shapes, under realistic separations” (Washington.edu.)

Spectral clustering algorithms are mainly implemented as unsupervised machine learning algorithms that systematically and unerringly classify large numbers of spectra, such that all spectra in each cluster belong to the same peptide (Perez-Riverol et al., 2018). The basis of any spectral clustering algorithm relies on three main components (Nationalmaglab.org):

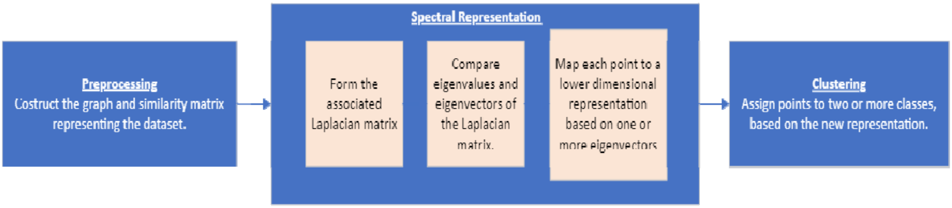
i assessing the similarity between spectra (distance function)

ii creating clusters of related spectra on the basis of pairwise similarities

iii constructing a representative or consensus spectrum for each resulting cluster.

The differences between algorithms and tools depend on how these principles are implemented and which preprocessing steps are used prior to the actual clustering step. Spectral clustering can use a connectivity approach to clustering, wherein data points that are connected to each other form a cluster-graph. The data points are then mapped to intrinsic dimensions so that it retains some meaningful properties of the original data.

**Figure 5** Typical steps in spectral clustering algorithm (see online version for colours)


Preprocessing
Costruct the graph and similarity matrix representing the dataset.
Spectral Representation
Form the associated Laplacian matrix
Compare eigenvalues and eigenvectors of the Laplacian matrix.
Map each point to a lower dimensional representation based on one or more eigenvectors
Clustering
Assign points to two or more classes, based on the new representation.

As stated earlier, spectral clustering uses information from the eigenvalues (spectrum) of special matrices (i.e., Affinity Matrix, Degree Matrix and Laplacian Matrix) derived from the graph or the dataset (Luxburg, 2007). Spectral clustering approaches are flexible and allow the grouping of non-graphical data also. There is no prior deduction about the depiction of the clusters. Other clustering techniques, like K-Means, assume that all data points are at the same distance from the cluster center. Conventional clustering techniques are based on the compactness of data points; therefore, they require the data points to have convex boundaries. Data points in the spectral clustering approach should be connected and do not require having convex boundaries. Also, in the K-means clustering algorithm we have to initially specify the number of clusters to be created, however, the final number of clusters is unknown when performing spectral clustering (Absolutdata.com, 2019). Spectral clustering algorithms can range from simple linear models to highly complex deep learning approaches. However, they can be implemented efficiently by standard linear algebra software, which often outperforms traditional clustering algorithms such as the K-means algorithm.

## 3 Significance of spectral clustering in proteomics

The analysis of large amounts of data that result from the mass spectrometry process is a demanding task in terms of computing power, storage requirements and human inspection capabilities. Mass spectrometers naturally have limited resolution, accuracy, mass range, and sensitivity. Moreover, background noise, resulting from the presence of other substances in the mixture, may produce spurious peaks. These attributes can lead to misidentification or the inability to decide among numerous possible identifications (Nationalmaglab.org).

Most of the proteomics work has focused on the downstream aspects of peptide and protein identification and quantification or post-identification results management; whilst the serious problems of data size, preprocessing of raw data and quality issues are often neglected by most tools. To address this, we need methods that can attack the problem closer to the core where it is created, at the raw data level, before peptide identification. These methods should take a global view of the gathered data such as clustering MS/MS spectra generated through tandem mass spectrometry to remove redundancy and improve spectral quality.

Spectral clustering methods are attractive because they are easy to implement and are reasonably fast (for sparse datasets up to several thousands). They do not intrinsically suffer from the problem of local optima. It can allow comparing differences in data generated between different instruments or labs etc. Large scale proteomics data generated from tandem mass spectra can be clustered based on similarity of spectra. It is then common to generate a consensus spectrum for each cluster which can be used for further spectral library or database searching and identification of new experimental data. Spectral library searches are significantly faster and more accurate than sequence database searching (Käll et al., 2008). However, spectral library searches are limited to only identify previously identified spectra and can often be instrument specific or biased towards particular experimental setups. Using consensus spectra generated through clustering of multiple MS experiments can help elevate these limitations and improve coverage of all peptide-forms. After clustering spectra from multiple experiments some large clusters will form without any known peptide identification. These clusters of dark peptides are due to modifications, peptide variants and novel proteins that were not considered in the original identification of the spectra. Spectral clustering enables mining and identification of these dark peptides by examining relative mass shifts and consensus similarities to other spectral clusters (Deutsch et al., 2018; Horlacher et al., 2016).

The spectral methods for clustering usually involve taking the top eigen-vectors of special matrices (i.e., Affinity Matrix, Degree Matrix and Laplacian Matrix) based on the distance between points (or other properties) and then using them to cluster the various points.

The input data for any clustering algorithm consists of:

A mass spectra data in large proteomics repositories (unidentified, correctly identified, and/or incorrectly identified spectra)

B identified spectra from smaller-scale experiments or curated databases e.g., NIIST.

The primary output that is expected after the spectral clustering process is spectral archives. Spectral archives consist of two cluster categories: clusters with identified spectra (spectral libraries) and clusters of unidentified spectra.

## 4 Review of spectral clustering algorithms in proteomics

Large numbers of papers have been written on spectral clustering and there is a lot of literature available on various protein clustering approaches. The first commercially used clustering algorithm is Basic Local Alignment Search Tool clustering (BlastClust) (Altschul et al., 1990). BlastClust is a hierarchical clustering method that uses single-linkage clustering technique and is functional for single-domain proteins. Systers (SYSTEmatic Re-Searching) (Krause et al., 2000) and ProtoMap (Yona et al., 1999) are not stand-alone applications and cannot be installed and run locally. GeneRAGE (Enrightand Ouzounis, 2000) is a stand-alone application but requires long running time. TribeMCL (Enright et al., 2002) uses Markov model to cluster larger proteome-scale datasets (>20,000 proteins). ProClust (Pipenbacher et al., 2002) uses graph-based techniques and can cluster small-scale proteomics data. FORCE (Wittkop et al., 2007) spectral clustering is stand-alone graph-based clustering technique and is used for small-scale proteomics projects.

It has been observed that most of these clustering algorithms require the user to specify several parameters, and it is not always clear what are the best values for these parameters. The results may highly depend on such parameters (Tang et al., 2009). For example, in a method such as GeneRAGE it is crucial to set the threshold to a value that will provide useful groupings. Notice that if the threshold is set to a value which is too conservative it is likely to generate many singleton clusters. On the other hand, a relaxed threshold would have the opposite effect of including many unrelated proteins into the same cluster. Also, the methods such as SYSters, ProtoMap, ProClust and GeneRAGE are ‘local’, in the sense that they assign a protein to a cluster considering only the distances between that protein and the other proteins in the set. Spectral methods differ from the ones described above in the sense that they are ‘global’, since they assign a protein to a cluster considering all the distances between every pair of proteins in the set. Spectral methods use the leading eigenvectors of a matrix derived from the distance matrix between the points. And we know that the eigenvectors of a matrix depend on the whole matrix: change one value in the matrix, and its eigenvectors will be different. This fact ensures the globality of the method. (Paccanaro et al., 2006).

Considering a broad range of different algorithms, we are particularly interested in systematic review of spectral clustering algorithms that were particularly designed to cluster proteomics data generated from tandem mass spectrometry. The review is based on certain criteria such as preprocessing of datasets, working structure, similarity function, reduction in dataset size, speed, accuracy and cluster purity. The algorithms selected at this date are among the most popular of the published ones.

The algorithms are:

a MS2Grouper (2004)

b PepMiner (2005)

c MS Cluster (2007)

d Improved MS Cluster (2011)

e PRIDE Cluster (2013)

f PRIDE Cluster extended (2016)

g MaRaCluster (2016)

h msCRUSH (2018)

These spectral clustering algorithms are reviewed on the basis of three main components:

i assessing the similarity between spectra (distance function)

ii creating clusters of related spectra on the basis of pairwise similarities

iii constructing a representative or consensus spectrum for each resulting cluster.

a *MS2Grouper*

Tabb et al. (2005) proposed the MS2Grouper clustering algorithm which has three main tasks: to detect similarity between spectra, use pairwise similarities to assess groups of related spectra, and construct a representative spectrum for each similar group. MS2Grouper is a software written in C++ programming language.

*Preprocessing*

MS2Grouper reads in all spectra, sorts (according to their precursor m/z value) and writes a new set of spectra (in MS2 file format) to disk.

*Functioning*

1 Detect similarity between spectra by:

  a Comparing each ms/ms spectra to others within 3Da.

  b Removing singletons.

  c Isolating the next set of spectra.

2 For clustering, MS2Grouper uses a pairwise similarity approach to assess groups of related spectra.

  a Finds a clique of at least 3 ms/ms spectra.

  b Extends paraclique to include neighbours.

3 *Consensus representation*: After the synthesis of a summary spectrum, the most intense spectrum is selected as the representative spectrum.

4 Remove spectra in paraclique.

5 Re-compute similarity within the subgraph.

*Benefits*

i Reduce spectral counts by up to 20%, as compared to un-clustered spectral datasets.

ii Reduce database search times without reduction in identified peptides.

iii Higher signal-to-noise.

iv Use of synthetic representative spectrum in MS2Grouper eliminated the need for 're-centroiding' the data points.

*Limitations*

i Normalised dot products used in this algorithm have more error rates than probability-based assessments of similarity.

ii A synthesised representative often produces spectra that do not score as well as the most intense spectra in the same group.

iii It is possible to improve detection of precursor charge state, quality of filtering, and de novo sequence inference.

b *Pepminer*

Beer et al. (2004) suggest a method that takes a global view of the gathered data by clustering the MS/MS spectra of an entire project consisting of multiple LC-MS/MS runs. Spectra that express similar characteristics, and are therefore believed to represent the same peptide, are grouped in one cluster. A representative spectrum is generated for each of the clusters, replacing the raw spectra.

*Preprocessing*

None

*Functioning*

1 First, similarity between two MS/MS spectra is calculated using normalised dot product. Similarity score is the cosine of the angle between two vectors, it is between 0 and 1.

2 To perform clustering, compute the transitive closures of MS/MS spectra whose pairwise similarity is above 0.6 and whose parent masses differ no more than by 2.5 atomic mass unit. To avoid unjustifiably coupled clusters, other clustering algorithms are also applied.

3 *Consensus representation*: Generation of cluster representative spectrum is accomplished by summing the intensities of all peaks, in all cluster members, across the mass axis.

4 *Retention time normalisation*: Two LC-MS/MS runs, R and S, share k clusters. Retention times of all spectra of S are normalised to the time scale of R.

*Benefits*

i Analysis is faster and less costly.

ii Eases and improves data management.

iii Improves peptide identification.

iv Facilitates the comparison of peptide mixture.

v Allows retention times of different LC-MS/MS runs to be correlated.

*Limitation*

i Peptides fragmented only once escape clustering and be ignored.

ii The data acquisition methodology may also influence the results.

iii Number of identifiable peptides that are not clustered decreases as the number of LC-MS/MS runs in the project grows.

iv Spectra of the same peptide that have different charge states do not join the same cluster because their fragmentation patterns often differ.

c *MS Cluster*

Frank et al. (2008) developed a practical MS-Clustering algorithm capable of handling large datasets using a single desktop PC. Instead of joining the clusters with maximum similarity, it joins the first ones it encounters that have a similarity above a threshold. As compared to traditional hierarchical clustering algorithms, MS-Cluster uses a heuristic approach which enables it to reduce the number of spectral similarity computations.

*Preprocessing*

The algorithm first filters the MS/MS datasets to remove low quality spectra that cannot result in reliable peptide identifications.

*Functioning*

1 *Spectral similarity*: MS Cluster uses a normalised dot-product approach to find similarity among spectra.

2 *Cluster representatives*: The consensus spectrum is constructed by consolidating the peak of all spectra in the cluster. Each consensus peak is assigned a mass that equals the weighted average of the joined peak's masses and an intensity that equals the sum of the peak's intensities.

3 *Clustering algorithm*: It uses a 'bottom-up' approach like incremental hierarchical clustering, which would start with clusters containing single spectra and build the clusters up by merging clusters with similar spectra.

*Benefits*

i Rapidly process large MS/MS datasets (~10 million) while ensuring high quality of resulting clusters.

ii Reduces the number of spectra by up to 90% (as compared to un-clustered data) without reducing the number of identified peptides and proteins.

iii False database identifications that occur due to low-quality of spectra are also reduced.

*Limitations*

i Using this algorithm for smaller datasets may lead to small loss of peptide identifications.

ii Spectra of previously unidentified peptides are difficult to identify due to the use of spectrum libraries search method in this algorithm.

d *Improved MS Cluster*

Frank et al. (2011) revised the MS Cluster algorithm in 2010. They propose a single consensus spectrum approach that clusters MS/MS datasets. For this purpose they use spectral archives, which extend spectral libraries by analysing both identified and unidentified spectra and also information about peptide spectra that are common across species is maintained.

*Preprocessing*

Remove low quality spectra using a regression model that relies on features that distinguish spectra of non-peptide material or poorly fragmented peptides. Typically 40-50% of the spectra are discarded at this stage.

*Functioning*

1 *Clustering algorithm*: Next, bottom up heuristic hierarchical clustering approach is used to join similar spectra that are likely to have originated from the same peptide.

2 Spectral similarity

   a First reduce each spectrum to vectors.

   b Then restrict the dimensionality of these vectors.

3 Constructing consensus spectra is generated by aggregating the spectra in the cluster. It involves several steps: peak list merging, intensity normalisation and peak filtering.

4 Finally, spectral archives are created and updated by the MS-Cluster algorithm.

*Benefits*

i Peptide identifications with spectral archives.

ii Identification of peptides conserved across species.

iii Short peptide identification.

iv The time required to generate an archive is practically the same as the time required to cluster the dataset.

*Limitations*

i Focuses on ion-trap.

ii Large computational time.

e *PRIDE Cluster*

Griss et al. (2013) proposed a clustering algorithm called PRIDE cluster, based on the MS-Cluster algorithm. It was optimised to increase the quality of the generated clusters at the cost of reducing its speed. The PRIDE cluster algorithm can split clusters and always assigns spectra to the cluster with the most similar consensus spectrum. It uses methods that can be used on data from any type of mass spectrometer. These changes considerably increased the execution time of the algorithm but were necessary to make the algorithm usable for heterogeneous datasets.

*Preprocessing*

*Spectrum normalisation*: Spectrum intensities are normalised so that the sum of intensities of all peaks is 1000.

*Functioning*

1 Spectra Similarity is calculated by using normalised dot product method.

For comparison of two spectra only K highest peaks are taken into consideration. K is calculated by dividing the precursor m/z by 50.

2 Spectrum quality assessment: roughly assess a spectrum's signal to noise ratio. Advantage of this simpler approach is that it is applicable to spectra originating from virtually any mass spectrometer platform.

3 Consensus spectrum building:

Same as used in MSCluster

a Add all peaks from all spectra to the consensus spectrum (CS).

b Merge identical peaks.

c Adapt peak intensities.

d Filter CS, keep only top 5 peaks.

4 Spectra Clustering:

a Sort Spectra.

b *Clustering Spectra*: if similarity is above threshold t, add spectra to the cluster.

c *Merging Spectra*: if a cluster's CS is similar (above threshold t) to another cluster's CS the clusters are merged.

d Remove non-fitting spectra.

e Goto (b) Until all spectra fit their cluster or a maximum of N iterations is reached.

*Benefits*

i Annotations in the PRIDE cluster are reliable.

ii Methods can be used on data from any type of mass spectrometer.

iii Algorithm usable for heterogeneous datasets.

iv Improve the quality of the generated spectra.

v Spectra are not added to the first fitting cluster, but to the best fitting cluster.

*Limitations*

i Increased execution time of the algorithm, as compared to MS Cluster algorithm.

f *PRIDE Cluster Extended*

Griss et al. (2016) extended the PRIDE cluster algorithm. The PRIDE cluster extended algorithm uses Apache Hadoop framework thereby increasing spectrum-clustering accuracy and scalability to handle the exponential data increase in the PRIDE Archive. Instead of the normalised dot product that is commonly used, authors used a probabilistic scoring approach to assess the similarity between two spectra.

*Functioning*

1 To assess the similarity between two spectra:

a First, precursor peaks are removed from the MS/MS spectrum and 70 highest peaks per spectrum are kept.

b Hypergeometric distribution is used to model the probability that the number of matched peaks occurred randomly.

c Only the peaks that have at least 50% of the total ion current (of the prefiltered spectrum) or at least the 25 highest peaks are used for spectra comparison.

d The algorithm uses all peaks to build the consensus spectrum.

2 For clustering, a probabilistic spectrum comparison method is used instead of a normalised dot product.

   a First, peak filtering is performed in a pure mapping job.

   b Five successive rounds of clustering are performed with decreasing similarity thresholds to reach a final accuracy of 99%.

   c Depending on the precursor's mass-to-charge ratio value, spectra are segregated into bins.

3 Consensus spectra are identified from reliable human spectral clusters, using all peaks.

*Benefit*

i Cluster large amounts of unidentified spectra without incurring a high degree of false positive matching.

ii Able to identify roughly 20% of the originally unidentified spectra in the PRIDE archive.

iii More accurate than the MSCluster clustering algorithm.

*Limitation*

i Computational complexity is more as compared to its previous counterparts.

g *MaRaCluster*

The and Käll (2016) present a scheme for hierarchical clustering of fragment spectra, MaRaCluster. This approach gives more weight to rare peaks while lowering the contribution of frequently present peaks. To counteract cluster contamination through chimeric spectra, it employs complete-linkage hierarchical clustering.

*Preprocessing*

The spectra are converted to MS2 format and assigned accurate precursor masses. Subsequently spectra are split into separate files based on these precursor masses to accommodate parallel processing.

*Functioning*

1 Only N most intense fragment peak locations in the spectra are considered and their mass-to-charge ratio is registered as a function to find the background frequency of the fragments.

2 A scoring function is used to calculate pairwise distances between spectra.

3 For clustering of spectra, a bottom-up hierarchical clustering is applied using a memory constrained complete linkage algorithm.

4 The consensus spectrum is generated for each cluster using the merging procedure employed by MS-Cluster.

*Benefits*

i As compared to its previous counterparts, 40% more peptides are identified for the same number of consensus spectra.

ii Independent of cluster size, MaRaCluster generates more purer clusters.

iii Only fewer spectra are left unclustered. Size of the clusters generated by MaRaCluster is relatively small as compared to the size of clusters generated by MSCluster.

iv Require less runtime as compared to typical runs of MS-Cluster.

v Rarity based distance measure is superior to the cosine distance and its complete linkage is better than single linkage.

vi Rarity based scoring function is a useful alternative to spectral dot product.

*Limitation*

i Rarity based scoring function increases complexity of MaRacluster algorithm.

ii Careful consideration between sensitivity and specificity is needed.

iii When processing large datasets, the number of comparisons will most likely become too large.

h *msCRUSH (mass spectrum ClusteRing Using locality Sensitive Hashing)*

Wang et al. (2019) present msCRUSH software, which uses locality sensitive hashing (LSH) technique to implement spectral clustering algorithm. msCRUSH is implemented in C++ and is released as open source software. The algorithm can significantly speed up clustering by selecting a subset of highly similar spectra through one-time processing of all spectra while retaining comparable or higher sensitivity and accuracy.

*Preprocessing*

a First putative noise peaks in each input ms/ms spectrum is removed.

b Vector conversion is done in which each input MS/MS spectrum is embedded into a numerical vector.

*Functioning*

1 *Random projection*

The numerical vectors are then randomly projected into buckets by utilising the selected similarity Hash functions

2 *Merge*

a Within each bucket, Locality Similarity Hashing (LSH) instead of cosine similarity is used between each pair of spectra of the same charge and close precursor mass is calculated.

b Two spectra are merged into a consensus spectrum if their pairwise similarity is higher than the specific threshold; otherwise, they will remain separate.

c After merging and replacement, the new spectrum (i.e., consensus spectrum) vector will proceed into the next iteration of vector conversion, random projection and merge.

3 *Consensus generation*: After a maximum number of iterations, msCRUSH will generate consensus spectra as the final clustering report.

*Benefits*

i Faster than the PRIDE cluster algorithm.

ii Higher clustering sensitivity.

iii Comparable accuracy.

iv Identify 1% – 3% more unique peptides.

v msCRUSH outputs fewer singleton clusters.

*Limitations*

msCRUSH generates clusters with slightly lower purity values as compared to PRIDE clusters.

Table 1 shows the comparison of all the spectral clustering algorithms discussed in this paper.

**Table 1** Comparison of the spectral clustering algorithms

| | *Pre-processing* | *Pairwise Similarity* | *Dot Product Similarity* | *Probabilistic Model* | *Local similarity hashing* | *Representative Spectra* | *Merged Consensus spectra* |
|---|---|---|---|---|---|---|---|
| MS2 Grouper (2004) | ✓ | ✓ | | | | ✓ | |
| PepMiner (2005) | | | ✓ | | | | ✓ |
| MS Cluster (2007) | ✓ | | ✓ | | | | ✓ |
| Improved MS (2011) | ✓ | | ✓ | | | | ✓ |
| PRIDE cluster (2013) | ✓ | | ✓ | | | | ✓ |
| PRIDE cluster extended (2016) | ✓ | | | ✓ | | | ✓ |
| MaRaCluster (2016) | ✓ | ✓ | | | | | ✓ |
| msCRUSH (2018) | ✓ | | | | ✓ | | ✓ |

**Table 2** Summary of characteristics of spectral clustering algorithms used in proteomics

| *S. no.* | | *MS2 Grouper* | *PepMiner* | *MS Cluster* | *Improved MS Cluster* | *PRIDE Cluster* | *PRIDE Cluster extended* | *MaRaCluster* | *MsCRUSH* |
|---|---|---|---|---|---|---|---|---|---|
| 1 | Dataset size | Smaller datasets | Smaller datasets | Larger mass spectrometry datasets and not useful small datasets | Larger mass spectrometry datasets | Larger mass spectrometry datasets | Larger mass spectrometry datasets | Larger mass spectrometry datasets | Larger mass spectrometry datasets |
| 2 | Preprocessing | Requires preprocessing | No preprocessing | Requires preprocessing | Requires preprocessing | Requires preprocessing | Requires preprocessing | Requires preprocessing | Requires preprocessing |
| 3 | Similarity detection | One-to-one comparison of the generated spectra. (Cosine distance method) | Using normalised dot product | Using normalised dot product | Using normalised dot product | Using normalised dot product | Use probabilistic spectrum comparison | uses pairwise distances between spectra. (Rarity based method) | Uses locality sensitive hashing (LSH) to compare two spectra |
| 4 | Clustering method | Complete-link hierarchical clustering | Approximate-link hierarchical clustering | Bottom-up incremental hierarchical clustering | Bottom-up heuristic hierarchical clustering | Single- link hierarchical clustering | Complete- link hierarchical clustering | Complete-link hierarchical clustering | Iterative greedy strategy to obtain hierarchical clustering |

**Table 2** Summary of characteristics of spectral clustering algorithms used in proteomics (continued)

| *S. no.* | | *MS2 Grouper* | *PepMiner* | *MS Cluster* | *Improved MS Cluster* | *PRIDE Cluster* | *PRIDE Cluster extended* | *MaRaCluster* | *MsCRUSH* |
|---|---|---|---|---|---|---|---|---|---|
| 5 | Consensus representative Spectra (CS) | Select most intense spectrum as consensus spectrum | CS is generated by summing the intensities of all peaks in all cluster members | CS is generated by consolidating the peaks | CS is generated by consolidating the peaks | CS is generated by consolidating the peaks | CS is generated by consolidating the peaks | Involves several steps such as: peak list merging, intensity normalisation and peak filtering to find CS | Consolidation of peaks is followed by vector conversion |
| 6 | Reduction of dataset size | Reduce spectral counts by up to 20% | Like that of MS2Grouper | Reduces the number of spectra by up to 90% without reducing the number of identified peptides | Like that of MS Cluster | Like that of MS Cluster | More effective as compared to previous algorithms | Like that of MS Cluster | More effective as compared to MaRaCluster algorithm |
| 7 | Cluster purity | Produces spectra that do not score as well as the most intense spectra in these groups’ | Peptides fragmented only once escape clustering and be ignored | Like that of PepMiner | Like its previous counterparts | Like its previous counterparts | Generate more purer clusters as compared to msCRUSH | 98% more average purity, independent of cluster s`4e | Cluster purity is less than MaRaClusters |

**Table 2** Summary of characteristics of spectral clustering algorithms used in proteomics (continued)

| *S. no.* | | *MS2 Grouper* | *PepMiner* | *MS Cluster* | *Improved MS Cluster* | *PRIDE Cluster* | *PRIDE Cluster extended* | *MaRaCluster* | *MsCRUSH* |
|---|---|---|---|---|---|---|---|---|---|
| 8 | Speed | Reduce database search times without reduction in identified peptides, as compared to unclustered data | Analysis is faster as compared to MS2Grouper | Rapidly process large ms/ms datasets (~10 million) | Large computational time | Increased execution time of the algorithm | Like its previous counterparts | Requires less execution time as compared to typical runs of MS-Cluster | Faster than PRIDE cluster algorithm |
| 9 | Accuracy | This algorithm has more error rates than probability-based assessments of similarity | Improved accuracy as compared to MS2Grouper | Reduces the number of false database identifications with low-quality spectra | Like its previous counterparts: MS2Grouper and PepMiner | Like that of MSCluster | More accurate than two previous clustering algorithms, MSCluster and MaRaCluster | Rarity based distance measure is superior to the cosine distance | Comparable accuracy |
| 10 | Identification of unique clusters | Better peptide identification as compared to unclustered data | Improve peptide identification as compared to MS2Grouper | It is not useful for smaller datasets since this usually leads to small loss of peptide identifications | Short peptide identification | Like that of MSCluster | Able to identify roughly 20% of the originally unidentified spectra in the PRIDE archive | Same as that of MS Cluster | Identify 1% – 3% more unique peptides |

## 5 Discussion

Eight of the most prominent spectral clustering algorithms (MS2 Grouper, PepMiner, MS Cluster, Improved MS Cluster, PRIDE Cluster, PRIDE Cluster extended, MaRaCluster and msCRUSH) have been discussed in detail as shown in Table 2. Among these MS2grouper and Pepminer algorithms are suitable for smaller datasets, whilst all other work efficiently with larger datasets. Except for Pepminer, all other algorithms perform preprocessing of the raw proteomics data. To detect similar spectra, MS2Grouper performs one-to-one comparison of the generated spectra. Pepminer, MSCluster, Improved MSCluster and PRIDE cluster use normalised dot product in similarity function. PRIDE cluster extended use probabilistic spectrum comparison and MaRaCluster uses pairwise distances between spectra, and msCRUSH uses locality similarity hashing (LSH) to compare two spectra. In MS2Grouper, the consensus representative spectra is selected as the one with highest intensity, in PepMiner a representative spectra is generated by summing the intensities of all peaks in all cluster members, whereas in MSCluster, PRIDE Cluster, PRIDE Cluster Extended and MaRaCluster it is generated through consolidation of the MS/MSpeaks. In msCRUSH consolidation is followed by vector conversion. Improved MSCluster involves several steps such as peak list merging, intensity normalisation and peak filtering to find representative spectra. Each of the discussed algorithms has its own contribution in reducing the size of the spectral dataset, but PRIDE Cluster Extended and msCRUSH turn out to be most effective. PRIDE Cluster Extended generates cleaner clusters compared to msCRUSH. However, msCRUSH is faster, with better sensitivity, accuracy and identifies more unique clusters compared to PRIDE Cluster Extended. These approaches show an ongoing trend in the improvement of spectral clustering algorithms, further advanced approaches such as deep learning with neural networks could be explored for further improvement of spectral clustering.

## 6 Conclusion

In this paper, a thorough review of the spectral clustering algorithms used in proteomics is presented. The algorithms discussed are the most popular and most referenced up to the current date. The paper provides a deeper dive into these spectral clustering algorithms and highlights their benefits, shortcomings, functioning and improvements that are provided through different approaches. A comparison on important features and their availability in the various spectral clustering algorithms has also been done, summarised and presented in the paper. It has been observed that there is tremendous opportunity for exploitation of current spectral clustering algorithms and further exploration to solve various problems related to peptide identification. Spectral clustering is currently constrained by accuracy and identification of unique clusters. If we can overcome these challenges, spectral clustering will accelerate peptide and protein identification.

## References

adt_admin (2019) *Spectral Clustering*, Absolutdata.com, Available at https://www.absolutdata.com/learn-analytics-whitepapers-webinars/spectral-clustering/ (Accessed 23 April, 2021).

Altschul, S.F., Gish, W., Miller, W., Myers, E.W. and Lipman, D.J. (1990) 'Basic local alignment search tool', *Journal of Molecular Biology*, Vol. 215, No. 3, pp.403–410, doi: 10.1016/S0022-2836(05)80360-2.

Beer, I., Barnea, E., Ziv, T. and Admon, A. (2004) 'Improving large-scale proteomics by clustering of mass spectrometry data', *Proteomics*, Vol. 4, No. 4, pp.950–960.

Daniel, A. and Spielman, S-H. (1996) 'Spectral partitioning works: planar graphs and finite element meshes', *IEEE Symposium on Foundations of Computer Science*, pp.96–105.

Deutsch, E.W., Perez-Riverol, Y., Chalkley, R.J., Wilhelm, M., Tate, S., Sachsenberg, T., … Röst, H. (2018) 'Expanding the use of spectral libraries in proteomics', *Journal of Proteome Research*, Vol. 17, No. 12, pp.4051–4060.

Ding, C., He, X. and Simon, H.D. (2005) 'On the equivalence of nonnegative matrix factorization and spectral clustering', *Proceedings of the 2005 SIAM International Conference on Data Mining*, Society for Industrial and Applied Mathematics, Philadelphia, P.A., pp.606–610.

Enright, A.J. and Ouzounis, C.A. (2000) 'GeneRAGE: a robust algorithm for sequence clustering and domain detection', *Bioinformatics* (*Oxford*, *England*), Vol. 16, No. 5, pp.451–457, doi: 10.1093/bioinformatics/16.5.451.

Enright, A.J., Van Dongen, S. and Ouzounis, C.A. (2002) 'An efficient algorithm for large-scale detection of protein families', *Nucleic Acids Research*, Vol. 30, No. 7, pp.1575–1584, doi: 10.1093/nar/30.7.1575.

Francis, R. and Bach, I. (1963) 'Learning spectral clustering, with application to speech separation', *Journal of Machine Learning Research*, p.7.

Frank, A.M., Bandeira, N., Shen, Z., Tanner, S., Briggs, S.P., Smith, R.D. and Pevzner, P.A. (2008) 'Clustering millions of tandem mass spectra', *Journal of Proteome Research*, Vol. 7, No. 1, pp.113–122.

Frank, A.M., Monroe, M.E., Shah, A.R., Carver, J.J., Bandeira, N., Moore, R.J., … Pevzner, P.A. (2011) 'Spectral archives: extending spectral libraries to analyze both identified and unidentified spectra', *Nature Methods*, Vol. 8, No. 7, pp.587–591.

Griss, J., Foster, J.M., Hermjakob, H. and Vizcaíno, J.A. (2013) 'PRIDE cluster: building a consensus of proteomics data', *Nature Methods*, Vol. 10, No. 2, pp.95–96.

Griss, J., Perez-Riverol, Y., Lewis, S., Tabb, D.L., Dianes, J.A., del-Toro, N., … Viz-caíno, J.A. (2016) 'Recognizing millions of consistently unidentified spectra across hundreds of shotgun proteomics datasets', *Nature Methods*, Vol. 13, No. 8, pp.651–656.

Horlacher, O., Lisacek, F. and Müller, M. (2016) 'Mining large scale tandem mass spectrometry data for protein modifications using spectral libraries', *Journal of Proteome Research*, Vol. 15, No. 3, pp.721–731.

Inderjit, S. (2001) 'Co-clustering documents and words using bipartite spectral graph partitioning', *Proceedings of the Seventh ACMSIGKDD International Conference on Knowledge Discovery and Data Mining*, pp.269–274.

Inderjit, S., Dhillon, Y. and Guan, B. (2004) 'Kernel k-means: spectral clustering and normalized cuts', *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp.551–556.

Käll, L., Storey, J.D., MacCoss, M.J. and Noble, W.S. (2008) 'Posterior error probabilities and false discovery rates: two sides of the same coin', *Journal of Proteome Research*, Vol. 7, No. 1, pp.40–44.

Kim, M., Eetemadi, A. and Tagkopoulos, I. (2017) 'DeepPep: deep proteome inference from peptide profiles', *PLoS Computational Biology*, Vol. 13, No. 9, e1005661.

Krause, A., Stoye, J. and Vingron, M. (2000) 'The SYSTERS protein sequence cluster set', *Nucleic Acids Research*, Vol. 28, No. 1, pp.270–272, doi: 10.1093/nar/28.1.270.

Luxburg, U. (2007) 'A tutorial on spectral clustering', *Statistics and Computing*, Vol. 17, No. 4, pp.395–416.

Ng, A.Y., Jordan, M. and Weiss, Y. (2001) 'On spectral clustering: analysis and an algorithm', *Advances in Neural Information Processing Systems*, Vol. 14, pp.849–856.

Paccanaro, A., Casbon, J.A. and Saqi, M.A.S. (2006) 'Spectral clustering of protein sequences', *Nucleic Acids Research*, Vol. 34, No. 5, pp.1571–1580, doi: 10.1093/nar/gkj515.

Perez-Riverol, Y., Vizcaíno, J.A. and Griss, J. (2018) 'Future prospects of spectral clustering approaches in proteomics', *Proteomics*, Vol. 18, No. 14, e1700454.

Pipenbacher, P., Schliep, A., Schneckener, S., Schönhuth, A., Schomburg, D. and Schrader, R. (2002) 'ProClust: improved clustering of protein sequences with an extended graph-based approach', *Bioinformatics*, Vol. 18, Suppl 2, pp.S182–S191, doi: 10.1093/bioinformatics/18.suppl_2.s182.

Shi, J. and Malik, J. (2000) 'Normalized cuts and image segmentation', *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 22, No. 8, pp.888–905, doi: 10.1109/34.868688.

Tabb, D.L., Thompson, M.R., Khalsa-Moyers, G., VerBerkmoes, N.C. and McDonald, W.H. (2005) 'MS2Grouper: group assessment and synthetic replacement of duplicate proteomic tandem mass spectra', *Journal of the American Society for Mass Spectrometry*, Vol. 16, No. 8, pp.1250–1261.

Tang, D.M., Zhu, Q.X. and Yang, F. (2009) 'A comparative study of clustering algorithms for protein sequences', *2009 Fourth International on Conference on Bio-Inspired Computing*, IEEE.

The, M. and Käll, L. (2016) 'MaRaCluster: A fragment rarity metric for clustering fragment spectra in shotgun proteomics', *Journal of Proteome Research*, Vol. 15, No. 3, pp.713–720.

Wang, L., Li, S. and Tang, H. (2019) 'MsCRUSH: fast tandem mass spectral clustering using locality sensitive hashing', *Journal of Proteome Research*, Vol. 18, No. 1, pp.147–158.

Wilm, E. and Donath, A.J. (1973) 'Lower bounds for the partitioning of graphs', *IBM Journal of Research and Development*, Vol. 17, No. 5, pp.420–425.

Wittkop, T., Baumbach, J., Lobo, F.P. and Rahmann, S. (2007) 'Large scale clustering of protein sequences with FORCE – a. layout-based heuristic for weighted cluster editing', *BMC Bioinformatics*, Vol. 8, No. 1, Vol. 396, doi: 10.1186/1471-2105-8-396.

Yona, G., Linial, N. and Linial, M. (1999) 'ProtoMap: Automatic classification of protein sequences, a hierarchy of protein families, and local maps of the protein space', *Proteins*, Vol. 37, No. 3, pp.360–378, doi: 10.1002/(sici)1097-0134(19991115)37: 3<360: aid-prot5> 3.0.co; 2-z

Zelnik-Manor, L. and Perona, P (no date) *Self-Tuning Spectral Clustering*, Neurips.cc, Available at. https://proceedings.neurips.cc/paper/2004/file/40173ea48d9567f1f393b20c855bb40b-Paper.pdf (Accessed 28 February, 2022).

## Websites

"Nationalmaglab.org", Available at: https://nationalmaglab.org/user-facilities/icr/techniques/tandem-ms (Accessed 12 April, 2021).

(No date) Washington.edu. Available at: https://sites.stat.washington.edu/mmp/Papers/ch2.2.pdf (Accessed 1 March, 2022).