The Raymond and Beverly Sackler
Faculty of Exact Sciences

Quantum Clustering and its Application to Asteroid Spectral Taxonomy

Thesis submitted in partial fulfillment of the requirements
for the M.Sc. degree at Tel Aviv University
School of Physics and Astronomy

by

Lior Deutsch

Under the supervision of

Prof. David Horn

Prof. Shay Zucker

February 2017

Acknowledgements

I would like to thank David Polishook for providing comments and feedback.

All of the data utilized in this publication were obtained and made available by the the MIT-UH-IRTF Joint Campaign for NEO Reconnaissance. The IRTF is operated by the University of Hawaii under Cooperative Agreement no. NCC 5-538 with the National Aeronautics and Space Administration, Office of Space Science, Planetary Astronomy Program. The MIT component of this work is supported by NASA grant 09-NEOO009-0001, and by the National Science Foundation under Grants Nos. 0506716 and 0907766.

Table of Contents

Abstract

Finding patterns in data is a main task in data mining and data exploration. Clustering algorithms find patterns in the form of a partition into groups of data points, where a proximity measure governs the partitioning. Quantum clustering (QC) is a clustering algorithm, inspired by quantum mechanics, which models the probability distribution function (pdf) of data points as a wave function of a particle in a potential, and identifies minima of the potential as clusters. The members of each cluster are all data points which lie in the basin of attraction of the minimum.

We present two new formulations of QC. The first is an entropy formulation, showing that potential minimization is equivalent to maximization of the difference between the pdf and the entropy field. The entropy field is associated with the probabilities to assign a point in feature space to a data point. The entropy field can be viewed as a transformation on the pdf that levels out its peaks and results in the potential. The entropy formulation also leads to a new clustering algorithm, similar to QC, but where the objective is to maximize the entropy. The second formulation shows that QC is related to the fuzzy c-means algorithm. Whereas in the latter the optimization is performed over the locations of cluster centers, QC is shown to be equivalent, under a certain initialization, to an optimization process over the data points while cluster centers remain constant.

QC depends on one free parameter, σ, which determines the scale of the clusters. Running QC repeatedly, at different values of σ, can be used for data sets which exhibit patterns at various scales, or when the scales of clusters are not known in advance. We introduce an alternative approach, of hierarchical quantum clustering (HQC). In HQC, as σ is gradually increased, clusters are merged into larger ones. This ensures that clusters at different scales follow an agglomerative structure, which is easy to interpret. The branches of the hierarchical tree can be cut at different scales to obtain various clustering assignments.

We then apply HQC to the problem of asteroid spectral taxonomy. The data set consists of measurements of the reflectance spectra of asteroids in the visual and near infrared ranges. The reflectance spectrum of an asteroid is an indication of its surface composition. For example, S-type asteroids are stony and show two absorption features in their spectra. We use 365 measurements, of 286 unique asteroids. We first examine the hierarchical clustering at a scale large enough to merge multiple measurements of the same asteroid into one cluster, and show that at this scale the clusters are very heterogeneous, leading to the conclusion that we should cluster spectra and not asteroids. We then turn to a smaller scale, and find that HQC leads to 26 clusters, some of them of flat spectra and some of wavy waveforms. 101 spectra remain singletons. We compare the results to the Bus-DeMeo taxonomy, and show that the proposed HQC taxonomy is based on clusters with smaller variances.

I. OVERVIEW OF CLUSTER ANALYSIS

A. Introduction

Cluster analysis, or simply clustering, is the task of partitioning a set of objects into groups, such that the objects that belong to each group are similar to each other, and objects that belong to different groups are less similar. These groups are called clusters. The precise meaning of "similar" and the required magnitudes of similarity and dissimilarity depend on the specific problem being solved. The set of objects is called a data set, and is commonly[1] a finite subset of $\mathbb{R}^d$. The integer d is the dimension of the data. Each dimension represents a feature of the data, and $\mathbb{R}^d$ is called the feature space. The members of the data set are data points. The goal of clustering can be seen as finding patterns, or structure, in the form of clusters[2], in the data set. It is a main task of data mining, exploration and analysis.

Clustering algorithms differ in their precise objective, and in the steps done to achieve the objective. No single clustering algorithm is suitable for all clustering problems, and an algorithm should be chosen based on the problem specifications, such as the metric of similarity. A common problem encountered by practitioners is that the problem specification is often ill-defined: A dataset is given in which it is not clear beforehand what metric and objective should be chosen. A good choice is one which would result in clusters which are meaningful, but the practitioner may not know in advance what structure he/she is looking for. This may happen especially when the data set doesn't have a good generative model, when the data set is large or has a high dimension, or when it has structures with different scales. In these situations, the practitioners should handle the problem by trying out various clustering algorithms with various parameters.

Clustering algorithms perform unsupervised learning – their input is unlabeled data and their task is to find structure in the data or an efficient representation of the data. Semi-supervised clustering algorithms, in which some prior information about whether some pairs of objects should be grouped together, or where this information is provided by a human agent at some stages of the algorithm, also exist[1].

B. Examples of Clustering Algorithms

Clustering algorithms differ by the metric they use on the data, how they define a cluster, the steps they perform to find clusters, and the assumptions they make on the data set. In the next paragraphs, we'll give some examples of clustering algorithms and clustering algorithms families.

1. *Hierarchical Clustering*

Agglomerative hierarchical algorithms build clusters in a bottom-up fashion. They initialize each data point to be a singleton cluster (that is, a cluster composed of a single point), and proceed by iteratively merging closest clusters into one cluster. The process ends either when all data points

---

[1] In this work I shall talk only about clustering of data in $\mathbb{R}^d$, but other forms of data exist, such as categorical data.

[2] A data set may have structure which is not in the form of clusters, such as symmetry or manifoldness. Detecting these forms of structure are the goals of other tasks

are merged into one cluster or when the smallest distance between clusters crosses a threshold. The result is a hierarchy of clusters, which has the form of a rooted binary tree (also called a dendrogram) whose leaves are the singleton clusters, and every node represents the clusters obtained by merging its two children. Edges of this tree may then be removed (manually or otherwise) to obtain a final clustering of the data.

The simplest agglomerative hierarchical algorithms are single-linkage[2], in which the distance between two clusters is defined as the smallest distance between a pair of elements, one from each cluster; complete-linkage[2], in which the distance between two clusters is defined as the largest distance between a pair of elements, one from each cluster; centroid-linkage[2], in which the distance between two clusters is defined as the distance between the clusters' centroids; and group-average[2], in which the distance between two clusters is defined as the average distance between pairs of elements, one from each cluster. All of these algorithms need a metric to be specified on $\mathbb{R}^d$.

More sophisticated agglomerative hierarchical algorithms were developed to overcome shortcomings of the previously mentioned algorithms. For example, CURE[3] is a compromise between single-linkage and centroid-linkage, in which each cluster is represented by a small number of representative points, and the distances between two clusters is the smallest distance between a pair of elements, one of each set of representative points of the clusters. Thus CURE is more immune to outliers than single-linkage, and to non-spherically shaped clusters than centroid-linkage. Another example is CHAMELEON[4], which defines the distance between clusters as a combination of their "closeness" – a measure of the smallest width[3] along the "seam" joining the two clusters – and "interconnectivity" – a measure of the total width of the seam. CHAMELEON copes better with clusters of various shapes and densities.

2. *Partitional Clustering Algorithms*

In contrast to hierarchical methods, which form a hierarchical set of clusters, partitional clustering algorithms yield only a single set of clusters. The algorithms include iterative steps, but only the final set of clusters is considered valid. An example of such an algorithm is k-means[5], which seeks to locate k clusters such that the total sum of squared distances between each data point and its cluster's centroid is minimized. k is a parameter chosen by the user. An optimal solution to this optimization problem is hard, but various heuristics exist that find a local minimum, the most standard being the periodic iterations of these steps, until convergence: (1) assign each data point to the closest centroid, (2) update the centroid to be the mean of the data points assigned to it. An initialization of the centroids is required, and the algorithm is known to be sensitive to the initialization. A variation of k-means is k-medoids[6], in which clusters are not represented by their centroids (means), which may not be part of the cluster, but rather by a member of the cluster which is the most similar to the other points of the cluster. The similarity measure does not have to be the squared distance as in k-means, and is typically chosen to be an absolute distance instead,

[3] A better choice of words would be "hyper-width", for large dimensions.

being more robust to outliers.

3. *Fuzzy Clustering*

k-means can be generalized into fuzzy c-means[7], which belongs to the fuzzy clustering algorithms family. In these algorithms, each data point is not assigned uniquely to one cluster, but rather gets assigned to all clusters with a certain probability (weight). The probability reflects the degree to which the data point is a member of a cluster. Fuzzy c-means is similar to k-means in that it minimizes the sum of squared distances between data points and clusters, the difference being that the squared distances are average[4] squared distances between each data point and all the clusters it may belong to. Thus, the optimization algorithm optimizes over the probabilities and the centroids' locations.

4. *Probability Distribution-Based Clustering Algorithms*

An algorithm related to fuzzy c-means, is the expectation-maximization (EM) algorithm for a Gaussian mixture model (GMM)[2]. In GMM, it is assumed that the data set can be generated by repeating the following steps: (1) Choose one of k Gaussians by some probability. (2) Sample a point from the chosen Gaussian. k is a predefined number. The means of the Gaussians are not known in advance. The covariances of the Gaussians and the probability to choose each Gaussian may or may not be assumed to be known. The unknown parameters are then estimated using EM, which is an algorithm for obtaining the maximum likelihood estimation: The chosen estimated parameter values give the highest probability to have obtained the data set. Once these parameters are estimated, cluster assignment can be fuzzy, based on posterior probability of each point to have been sampled from each Gaussian; Or a non-fuzzy assignment can be chosen, based on the maximum posterior probability.

Assuming unknown covariance matrices for the all the Gaussians in GMM means that there are a lot of parameters to estimate, and this implies heavy computation and slow convergence. A simpler, more tractable, approach is to take a diagonal covariance matrix, with a known, pre-determined, constant variance σ^2, and use the same covariance matrix for all Gaussians. This means that all Gaussians produce data points with independent components and same variance. The problem with this approach is that the choice of σ^2 is in fact a choice of scale for the problem. Clusters with characteristic sizes much smaller or larger than σ may go undiscovered. Also, the number of Gaussians, which is the number of clusters, is arbitrarily chosen while it could actually vary in different scales. One possible solution is to take a scale-space approach[8]. In this approach, the number of Gaussians is taken to be the number of data points, such that each Gaussian is located on a data point. The sum of all of these Gaussians is an estimation of the probability density function (pdf) that the data points were sampled from. The Gaussians' locations are now not considered unknown, and therefore maximum likelihood estimation is not needed. Clusters are not

[4] More precisely, it does not work to use the average, since this leads upon optimization to a "hard" (as opposed to fuzzy) probability distribution. Therefore, in the expression for the average the probabilities are taken to some constant power greater than 1

defined by these Gaussians, but rather by the peaks of the estimated pdf. Each data point is assigned to the nearest peak. Clustering is performed for various values of σ, and the result is a hierarchy of clusters, spanning different scales. The "correct" scales to use, according to this method, are stable scales, that is, scales in which the number of clusters remains constant along wide range of values of σ.

Another mode-finding method is the mean-shift algorithm[9], where each point is moved in the direction of the highest density of points. Data points that converge to the same final location are grouped into a cluster. The density of points is defined by summing a kernel function over each point, as in the scale-space approach. The kernel does not have to be Gaussian, and it can also be truncated at a certain radius. There are two versions of the mean-shift algorithm: (1) The density is taken constant while the points are moving. (2) As the points are moving, the density of points is updated by the new locations of the points. This second version is called blurring.

5. *Density-Based Clustering Algorithms*

Instead of finding locations with high density, as in the previously described approaches, density based algorithms try to detect connected regions of $\mathbb{R}^d$ which have a higher density of data points, compared to their surrounding. The data points in each such region are considered a cluster, and points falling out of these regions are considered outliers. An algorithm that follows these lines is DBSCAN[10]. It finds dense sets of points by finding "core points"- points that have at least k neighbors within distance ϵ. A cluster is defined as a maximal set of core points which can be pairwise connected by a sequence of core points such that each element of the sequence is within distance ϵ of the previous elements. A cluster also includes non-core points that are within distance ϵ of a core point in the cluster. k and ϵ are parameters of the algorithm that determine the minimal density of the region. A similar algorithm, OPTICS[11], uses just k as a parameter, and finds clusters that correspond to different values of ϵ, thus allowing clusters with different densities. The result is a hierarchy of clusters.

6. *Support Vector Clustering*

Support vector clustering[12] takes a different approach, based on cluster boundaries rather than densities. The idea is to map all data points to a very high dimensional space, possibly infinite, and to surround the high-dimensional points with a hyper-sphere with the smallest possible radius. When the hyper-sphere is mapped back to the original feature space, it still surrounds the points, but it is no longer a sphere. It turns into a set of separated closed boundary surfaces, and each such surface encloses a cluster.

7. *Clustering Using Neural Networks*

Another framework for clustering is based on artificial neural networks. These algorithms are partially inspired by the information processing and learning mechanisms in the brain. One example is the self-organizing map[13]. Here, a grid of "neurons" is spread in feature space. The grid is regular and defines a neighbor for each neuron. The goal is to update the locations of the neurons so as to get a good representation of the data set. The update is performed iteratively as follows: In each iteration, one data point is chosen. The neurons that are close to the data point get

pulled in its direction, but there is also a resistive force applied from the neighbors of the neurons. After convergence, clusters can be identified as the regions where the neurons converged to.

8. *Physics-Inspired Clustering Algorithms*

Another source of inspiration for clustering algorithms is physical systems. Physics-inspired algorithms include the maximal entropy clustering[14], which is a form of fuzzy clustering, similar to fuzzy c-means. The algorithm repeats the following steps, until convergence: (1) Assign probabilities of cluster-membership based on the maximum entropy principle, subject to the constraint of a given total "energy", which is a sum of average squared distances between data points and cluster centers, multiplied by some factor. (2) Update cluster centers to become the average locations of data points, weighted by their membership probability. The first step gives the Boltzmann-Gibbs distribution, and the second step is in essence a mean-shift update. Decreasing the factor in the energy constraint amounts to increasing the temperature of the systems, which makes the clustering fuzzier. In the process of increasing the temperature, clusters may merge. These are identified as phase transitions. The process of increasing the temperature is equivalent to the process of increasing σ in the scale-space approach.

Another physics-inspired clustering algorithm is Quantum Clustering (QC), which is the subject of the next chapter.

II. QUANTUM CLUSTERING (QC)

A. Method Introduction

1. *The Quantum Mechanical Basis*

Quantum clustering[15] is motivated by the quantum mechanical system of a single particle in d-dimensional space. The Hamiltonian operator of such a system is:

$$\hat{H} = \frac{\hat{p}^2}{2m} + V(\hat{\mathbf{x}}) \ , \tag{1}$$

where $\hat{p}$ is the operator of the d-dimensional momentum, $\hat{\mathbf{x}} = (\hat{x}_1 \ , \hat{x}_2 \ , \dots , \hat{x}_d)^T$ is the vector of position operators, and m is the particle's mass. The momentum operators are given by:

$$\hat{p}_i = -i\hbar \frac{\partial}{\partial x_i} \ . \tag{2}$$

Inserting equation (2) into equation (1) gives the differential form of the Hamiltonian:

$$\hat{H} = -\frac{\hbar^2}{2m}\nabla^2 + V(\mathbf{x}) \ . \tag{3}$$

The ground state of this system is described by a wave function $\psi(\mathbf{x})$ which is an eigenfunction of the Hamiltonian:

$$\hat{H}\psi(\mathbf{x}) = E\psi(\mathbf{x}) \ , \tag{4}$$

where E is the lowest eigenvalue of the Hamiltonian. We can assume that $E = \frac{d}{2}$, as in the ground state of a harmonic oscillator, since any constant shift in energy can be absorbed into the definition of $V(\mathbf{x})$. Thus, the eigenvalue equation is:

$$-\frac{\hbar^2}{2m}\nabla^2\psi(\mathbf{x}) + V(\mathbf{x})\psi(\mathbf{x}) = \frac{d}{2}\psi(\mathbf{x}) \ . \tag{5}$$

2. *The QC Algorithm*

The starting point for QC is constructing the wave function $\psi(\mathbf{x})$ out of the data set $\{\mathbf{x}_i\}_{i=1}^n \subset \mathbb{R}^d$, where n is the number of data points. This is done using the Parzen window method[16], which convolves the data points with a fixed Gaussian kernel with covariance $\sigma^2\mathbf{I}$, where $\mathbf{I}$ is the $d\times d$ unit matrix:

$$\psi(\mathbf{x}) = c\sum_{i=1}^{n} \exp\left(-\frac{(\mathbf{x}-\mathbf{x}_i)^2}{2\sigma^2}\right) \ , \tag{6}$$

where c is a constant factor required for $\psi(\mathbf{x})$ to have a unit L^2 norm. This expression is usually used as an estimation of the probability density function (pdf) that the data points were sampled from. In quantum mechanics, Born's rule states the pdf of measuring a particle in a certain location is given by the squared modulus of the wave function, and not by the wave function itself (which may be complex-valued). However, in QC this detail is usually ignored, and $\psi(\mathbf{x})$ is viewed both as a wave function and as a probability distribution. The main reason for this is that the mathematics becomes more cumbersome if $\psi(\mathbf{x})$ is taken to be the square root of the Parzen estimator. Also, the fact that $\psi(\mathbf{x})$ is always real, whereas wave functions are generally complex-valued, is consistent with choice of $\psi(\mathbf{x})$ being the ground state of the Hamiltonian, since a ground state always has a constant phase, which makes it equivalent to a real wave function.

In a typical problem of quantum mechanics, a given Hamiltonian is solved to yield its eigenfunctions. In QC, the reverse is done: $\psi(\mathbf{x})$ is given by equation (6), and $V(\mathbf{x})$ is sought for.

This turns out to be a much easier problem, since all it amounts to is eliminating $V(\mathbf{x})$ from equation (5). The solution is:

$$V(\mathbf{x}) = \frac{\hbar^2}{2m}\frac{\nabla^2\psi(\mathbf{x})}{\psi(\mathbf{x})} + \frac{d}{2} \ . \tag{7}$$

The physical constants $\hbar$ and m have no significance in the QC setting, so they can be dropped. Instead, the parameter σ^2 will be used, to make the potential unit-less:

$$V(\mathbf{x}) = \frac{\sigma^2}{2}\frac{\nabla^2\psi(\mathbf{x})}{\psi(\mathbf{x})} + \frac{d}{2} \ . \tag{8}$$

An explicit expression for the potential is obtained by plugging equation (6) into equation (8):

$$V(\mathbf{x}) = \frac{1}{\psi(\mathbf{x})}\sum_{i=1}^{n}\frac{(\mathbf{x}-\mathbf{x}_i)^2}{2\sigma^2}\exp\left(-\frac{(\mathbf{x}-\mathbf{x}_i)^2}{2\sigma^2}\right) \ . \tag{9}$$

The idea behind QC is that the minima points of the potential $V(\mathbf{x})$ can be thought of as the locations where a physical attractive force originates. For example, if $V(\mathbf{x})$ were the potential of a harmonic oscillator, then the fixed end of the "spring" attached to the particle would be the minimum of $V(\mathbf{x})$. A classical lowest energy particle state would be constantly located at the minimum of $V(\mathbf{x})$, which would correspond to a delta function expression for $\psi(\mathbf{x})$. But in quantum mechanics, a consequence of the uncertainty relations is that such a solution has an infinite momentum and therefore it cannot be a ground state. The ground state is more spread-out, the spreading caused by the Laplacian operator in equation (3), and therefore there is a non-zero probability of observing the particle at locations different from the minima of $V(\mathbf{x})$.

This quantum description can be thought of as the model that generated the data points. For comparison, in a Gaussian mixture model, the points are modeled to have been generated by a pdf which is the weighted sum of a few Gaussians. The variance of the data points around each cluster is caused by the non-zero variance of the Gaussians. In contrast, in QC the data points were generated by a quantum mechanical system with potential $V(\mathbf{x})$, and the variance of the data points around each cluster is caused by the quantum effect of a non-localized wave function.

In QC, each data point is associated with a close minimum of $V(\mathbf{x})$, where the point would presumably have been located if there were no spreading of the wave function. This is achieved by moving the point down the potential[5], using gradient descent[6], until convergence to a *local* minimum. Unlike many other optimization problems, in QC it is desired that the minimum is local and not global, since the former is taken as the cluster location. A few examples of $V(\mathbf{x})$, for synthetically generated two-dimensional[7] data and various values of the Gaussians' width σ, are shown in Figure 1.

---

[5] Interestingly, similar dynamics are described in the de-Broglie-Bohm formulation of quantum mechanics [47]. In this formulation, a particle's motion is dictated by the gradient of the quantum potential, which is given by an expression similar to equation (8), where ψ is replaced by $|\psi|$.

[6] Other gradient methods, such as Newton's method, can also be used, as long as they find the local minimum whose basin of attraction contains the data point.

[7] Caution should be taken when drawing conclusions from two-dimensional examples, about the generality of clustering algorithms in higher dimensions.

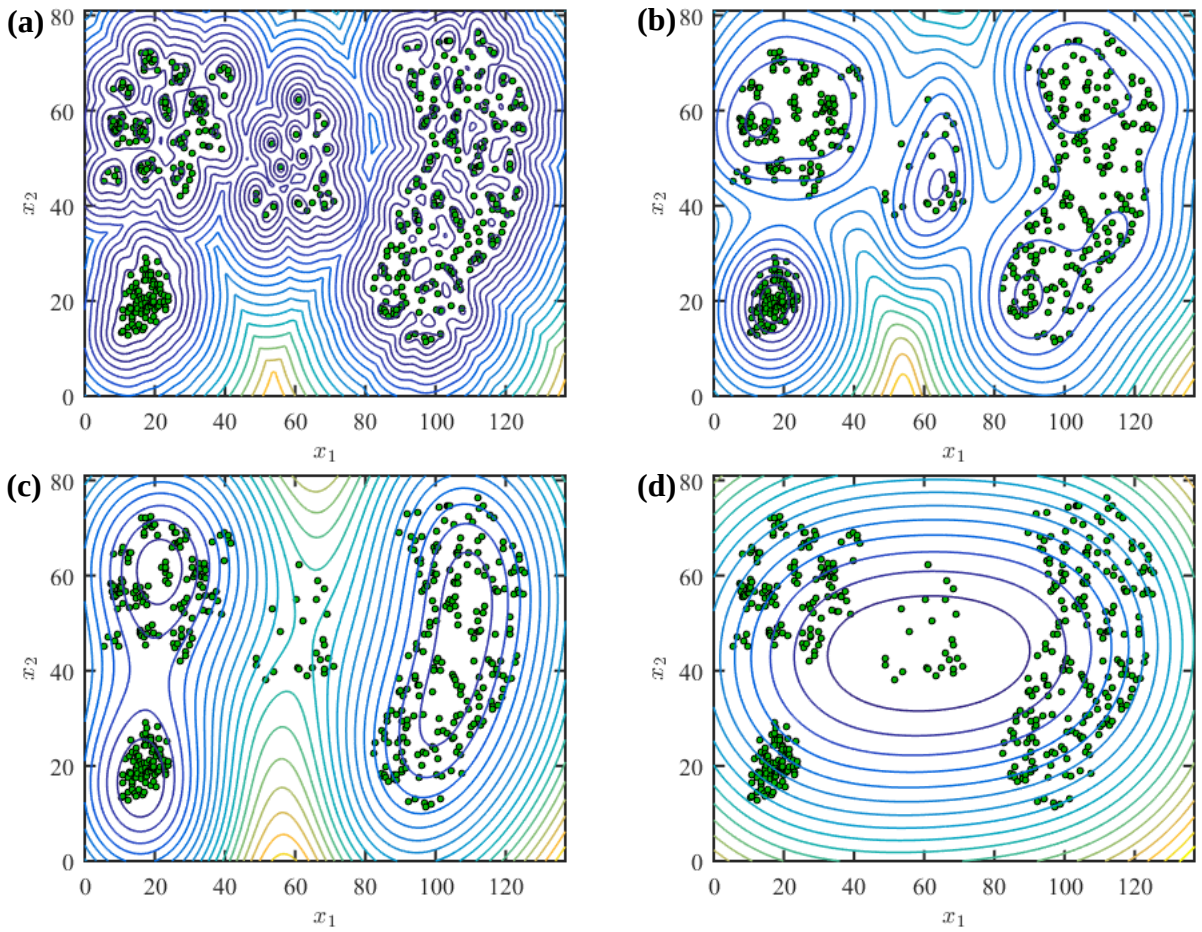


Figure 1: The potential $V(x_1, x_2)$ for synthetically generated data in two dimensions, for various values of σ. The data consists of 500 points. (a) $\sigma = 2$ (b) $\sigma = 10$ (c) $\sigma = 20$ (d) $\sigma = 60$

Quantum Clustering

(QC1) Repeat for each data point $\mathbf{x}_i$:

(QC1.1) Create a "replica" of data point $\mathbf{x}_i$, which will be denoted $\mathbf{x}'_i$, to be located on $\mathbf{x}_i$.

(QC1.2) Repeat the gradient descent step until convergence:

(QC1.2.1) $\mathbf{x}'_i \leftarrow \mathbf{x}'_i - \eta(\boldsymbol{\nabla} V)(\mathbf{x}'_i)$

(QC2) Group replica points that fell into the same minimum as a cluster.

Box 1: The QC algorithm

The QC algorithm steps are depicted in Box 1. In the algorithm, $(\boldsymbol{\nabla} V)(\mathbf{x}'_i)$ denotes the gradient of the potential, evaluated at point $\mathbf{x}'_i$. The gradient has an analytic expression which can easily be derived from equation (9). The parameter η determines the step size of the descent. It may be taken constant or adaptive, and in particular it may depend on the norm of the gradient, thus allowing the gradient to be normalized to unity.

Hierarchical Quantum Clustering

(HQC1) Initialize a small value for σ. If data points have errors assigned to them, this σ should preferably be larger than these errors.

(HQC2) Run QC.

(HQC3) Repeat until σ is high enough:

(HQC3.1) From each resulting cluster, take one representative replica point and discard the rest.

(HQC3.2) For each replica point, perform the gradient descent of QC, as in (QC1.2).

(HQC3.3) Group replica points into clusters as in (QC2).

(HQC3.4) Increase σ by a small amount.

Box 2: The HQC algorithm

B. The Significance of σ, and Hierarchical QC

The parameter σ – the width of the Gaussian in the Parzen window estimator - is a hyper parameter of the algorithm that is chosen by the user. The value of σ influences the resulting clusters. If σ is smaller than the distance between points in the data set, then each data point is already on a minimum of its own Gaussian, the other Gaussians being too far to have any influence. Thus, the resulting clusters will be singletons, with each data point its own cluster (unless there are identical points in the data set, in which case they will form a non-singleton cluster). On the other extreme, if σ is larger than the domain size of all points, then all points will fall under gradient descent to the same location, and will all be grouped into one cluster. An intermediate value of σ will give clusters on a corresponding scale.

A data set may have structures at different scales, and we cannot expect one σ to reveal all these structures. Figure 1 showcases such a situation. It shows four main clusters, bust some of these clusters are composed of smaller clusters, and each cluster has a different size and density of points. The obvious way to deal with this difficulty is by running QC multiple times for a range of σ values, representing different scales, and aggregating the results. The main problem with this approach is the inefficiency of the process. A more efficient approach is using Hierarchical QC (HQC), which is described in Box 2. The algorithm performs QC between successive increments of σ, and whenever replica points fall into a cluster they are merged into one replica point that continues to be moved by QC replica dynamics. The potential $V(\mathbf{x})$ remains defined on the basis of all original data points and the current σ.

There are two advantages to using HQC. The first is computational: As σ grows, there are less and less replicas that undergo the process of gradient descent, hence clustering at higher scales demands less computational steps.

The second advantage of HQC is conceptual: Clusters obtained from all values of σ form a hierarchical tree, in which the leaves are the initial, singleton clusters, and each node in the tree represents a cluster which is the union of the clusters which are its children. The tree is

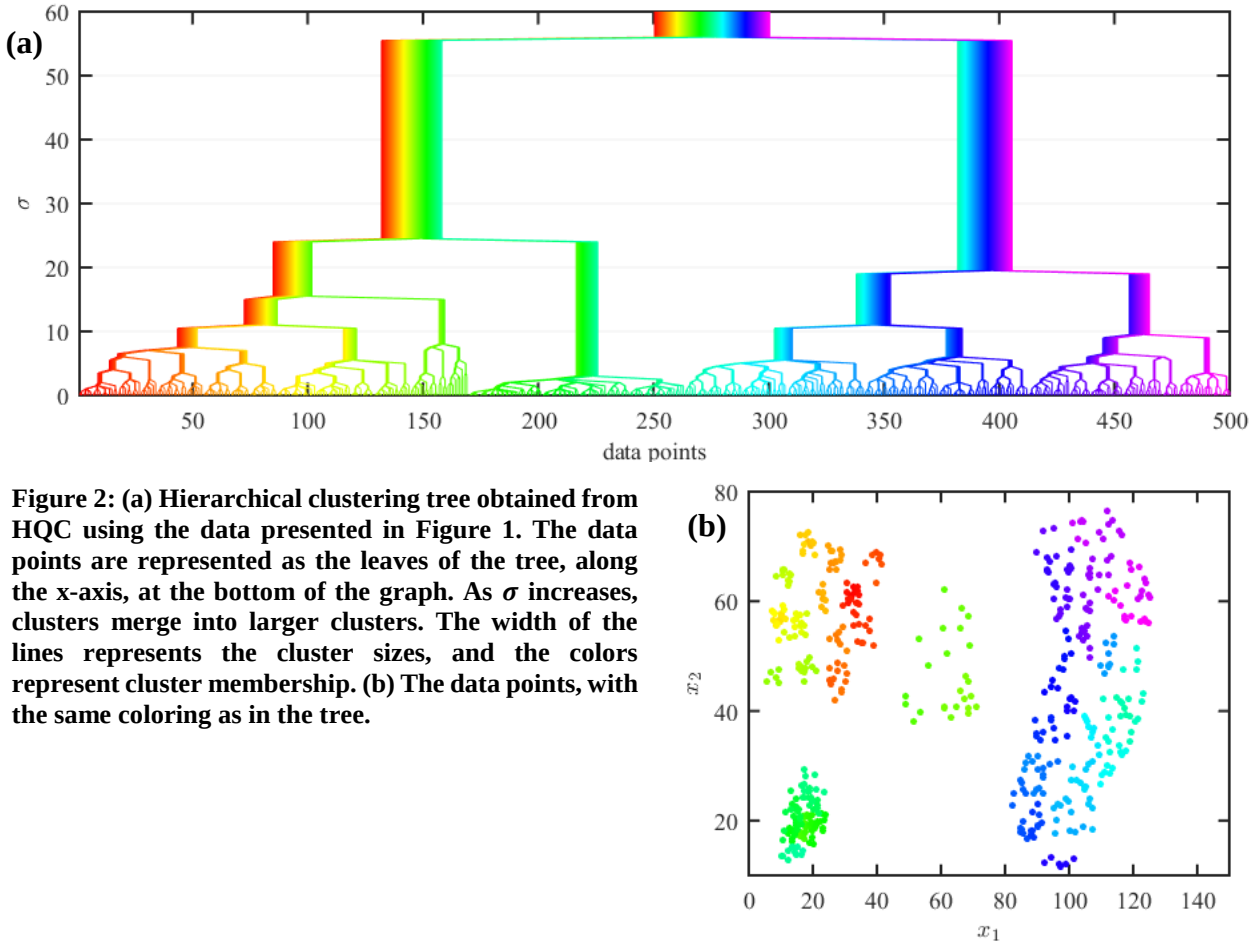


Figure 2: (a) Hierarchical clustering tree obtained from HQC using the data presented in Figure 1. The data points are represented as the leaves of the tree, along the x-axis, at the bottom of the graph. As σ increases, clusters merge into larger clusters. The width of the lines represents the cluster sizes, and the colors represent cluster membership. (b) The data points, with the same coloring as in the tree.

topologically equivalent to the trajectories of replicas under the QC dynamics. This is an obvious outcome of the merging of the replicas. The hierarchy is not mathematically guaranteed when performing *ab initio* QC on a range of values of σ. The advantage of a hierarchy in the set of clusters is that it is more consistent with the idea of scale. If at a small scale two data points are members of the same cluster, then we would like them to stay in the same cluster also on larger scales. Also, if there is a requirement for a single set of mutually disjoint clusters, then this can be done consistently by removing edges in the tree of clusters, the results being clusters which have different scales.

An illustration of the clustering tree of the synthetic data from Figure 1, generated by HQC, is displayed in Figure 2. It shows that clusters form at multiple scales. Some clusters show stability over a wide range of σ values. The evolution of replica points is demonstrated in Figure 3. The most general way to obtain a final partitioning of the data is to choose a cutoff σ for each branch of the tree. The cutoff can be chosen either manually or automatically. A reasonable cutoff value for a cluster (branch) can be chosen, for example, based on a combination of these conditions: (1) The cluster is stable, in the sense that it hasn't changed for a large range of σ values (this is in the spirit of scale-space clustering). (2) The cluster is not the result of a merging of two relatively large clusters. (3) The cluster is not the result of a merging of two clusters which have very different characteristic sizes, where the characteristic size of a cluster is the latest value of σ that caused the cluster to change.

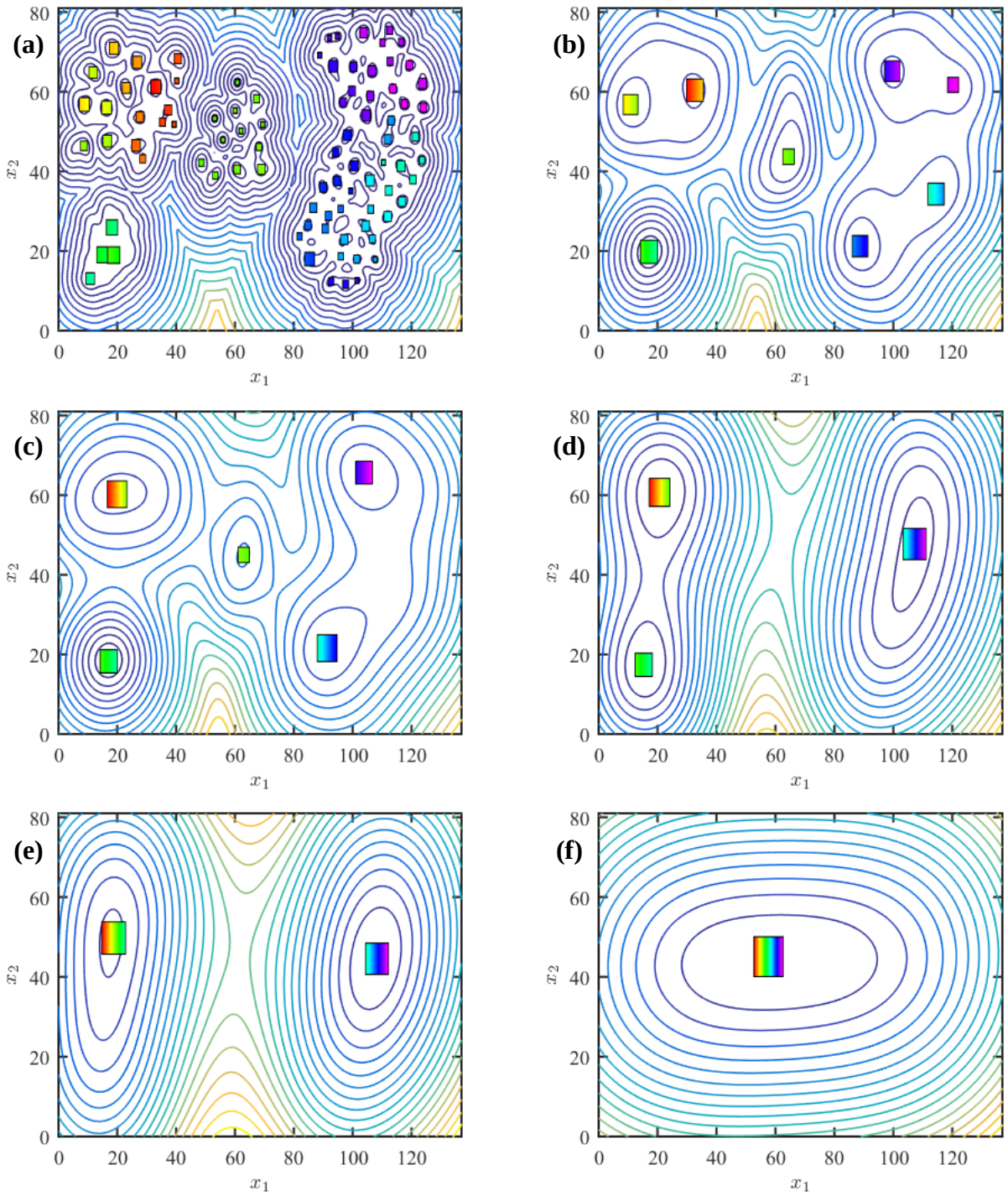


Figure 3: Evolution of replicas under HQC, using the data presented in Figure 1 and Figure 2. Each figure corresponds to a different value of σ. Each square represents a cluster at the corresponding scale. The size of the square corresponds to the number of members of the cluster. The colors of each square indicate its members; refer to Figure 2(b) to the color encoding. (a) $\sigma = 2$ (b) $\sigma = 9$ (c) $\sigma = 12$ (d) $\sigma = 21$ (e) $\sigma = 27$ (f) $\sigma = 58$

HQC is bottom-up – starting with clusters of size one and merging them repeatedly. A top-down hierarchical version of QC has also been proposed in [17]. In Top-Down QC (TDQC), QC is applied to the dataset, and then the dataset is divided into two separate data sets, the first one consisting of all data points that fell into the cluster with minimal value of the potential $V(\mathbf{x})$, and

the second one consists of the rest of the data points. The procedure is then repeated reclusively on the two new data sets. A main difference between TDQC and HQC is that TDQC uses a single value of σ while HQC increases σ to find multiscale clustering.

C. Entropy Formulation

QC employs the minima of $V(\mathbf{x})$ for cluster assignments. An alternative algorithm could have assigned data points to clusters based on the maxima of $\psi(\mathbf{x})$, as is done in the scale state approach. Although similar, these two methods can yield different results (see for example [15]), and understanding the source of the differences can help deciding which algorithm is better for a given problem. In this section, we describe the entropy formulation of QC which relates $V(\mathbf{x})$ and $\psi(\mathbf{x})$ by an entropy term. This formulation can shed light on the differences between minimizing $V(\mathbf{x})$ and maximizing $\psi(\mathbf{x})$.

The Parzen estimation of the pdf, equation (6), can also be viewed as a Gaussian mixture model (GMM). The process of sampling a point $\mathbf{X} \in \mathbb{R}^d$ from the underlying distribution is equivalent to a two-step process: (1) Sample uniformly a number N for the set $\{1,2,\dots,n\}$ (2) sample a point $\mathbf{X}$ from the Gaussian distribution centered at $\mathbf{x}_N$ with covariance $\sigma^2\mathbf{I}$, where $\mathbf{I}$ is the $d\times d$ identity matrix. Under this description, the pdf $\psi(\mathbf{x})$ can be written as:

$$\psi(\mathbf{x}) = \sum_{i=1}^{n} \mathbb{P}(\mathbf{X}=\mathbf{x} \mid N=i)\mathbb{P}(N=i) \quad . \tag{10}$$

$\mathbb{P}(\cdot)$ is used to denote both the probability of an event and the probability density of an event, and it should be clear which one by its argument. It follows that:

$$\mathbb{P}(N=i \mid \mathbf{X}=\mathbf{x}) = \frac{\mathbb{P}(\mathbf{X}=\mathbf{x} \mid N=i)\mathbb{P}(N=i)}{\psi(\mathbf{x})} \quad . \tag{11}$$

From uniformity, $\mathbb{P}(N=i) = \frac{1}{n}$, and this expression becomes:

$$P(i|\mathbf{x}) \equiv \mathbb{P}(N=i \mid \mathbf{X}=\mathbf{x}) = \frac{\exp\left(-\frac{(\mathbf{x}-\mathbf{x}_i)^2}{2\sigma^2}\right)}{\sum_{j=1}^{n}\exp\left(-\frac{(\mathbf{x}-\mathbf{x}_j)^2}{2\sigma^2}\right)} \quad . \tag{12}$$

$P(i|\mathbf{x})$ is the probability that a given point $\mathbf{x}$ was sampled from the i'th Gaussian. In a GMM where the number of Gaussians is small, the maximum value of $P(i|\mathbf{x})$ can be used for cluster assignment of the point $\mathbf{x}$. In the current setting, the number of Gaussians is the number of data points, and each data point coincides with a Gaussian center, so there is no sense in making such an assignment.

Using equation (12), we can rewrite the explicit expression for $V(\mathbf{x})$ in equation (9) as:

$$V(\mathbf{x}) = \mathbb{E}\left[\frac{(\mathbf{x}-\mathbf{X})^2}{2\sigma^2} \mid \mathbf{x}\right] \quad , \tag{13}$$

where $\mathbb{E}$ is the expectation function, $\mathbf{X}$ is a random variable whose outcome can be any of the data points $\{\mathbf{x}_i\}_{i=1}^n$, where the probability of outcome $\mathbf{x}_i$ is $P(i|\mathbf{x})$. This expression can be thought of as describing $V(\mathbf{x})$ as an average energy at point $\mathbf{x}$.

A point $\mathbf{x}$ (not necessary from the data set) has an uncertainty as to which Gaussian it was sampled from. This uncertainty can be quantified using the entropy:

$$S(\mathbf{x}) = -\sum_{i=1}^{n} P(i|\mathbf{x}) \log P(i|\mathbf{x}) \quad . \tag{14}$$

A lower bound on the value of $S(\mathbf{x})$ is 0, which corresponds to a situation in which the first nearest neighbor of $\mathbf{x}$ is much closer to $\mathbf{x}$ than the rest of the data points. An upper bound is $\log n$, which describes a situation in which all the data points are equidistant from $\mathbf{x}$. It can be shown that the following relation holds between the quantum potential, the entropy, and the Parzen estimation of the pdf (wave function):

$$V(\mathbf{x}) = S(\mathbf{x}) - \log \psi(\mathbf{x}) \quad . \tag{15}$$

This relation is analogous to the following relation in the statistical mechanical description of a canonical ensemble:

$$U = TS - k_B T \log Z \quad . \tag{16}$$

Here, U is the internal energy of the system, given by the average energy taken upon all accessible microstates of the system; S is the entropy of the system; Z is the partition function; k_B and T are Boltzmann's constant and the temperature respectively.

The probability that the system is in a microstate with energy E is, by Boltzmann's distribution, proportional to $e^{-E/k_B T}$. The partition function is given by the sum $\sum e^{-E_i/k_B T}$ over all possible energies of microstates. Obviously U, S, Z in statistical mechanics are analogous to $V(\mathbf{x})$, $S(\mathbf{x})$ and $\psi(\mathbf{x})$ in the QC setting, respectively, and $P(i|\mathbf{x})$ is the Boltzmann distribution. A difference should be noted, though: in statistical mechanics these quantities describe the state of an entire system, while in the QC setting, these are functions of location $\mathbf{x}$ in feature space.

Equation (15) suggests that the minima of $V(\mathbf{x})$ may be in different locations than the maxima of $\psi(\mathbf{x})$, and are shifted by the entropy $S(\mathbf{x})$. In the trivial case where all n data points are located in the same location, $S(\mathbf{x}) = \log n$ and the extrema coincide. Another trivial situation is the limit $\sigma \to 0$, such that the distances between data points become much larger than σ, and therefore $S(\mathbf{x})$ is almost 0 in the neighborhood of each data point. Otherwise, $S(\mathbf{x})$ can change the gradients of $\psi(\mathbf{x})$, such that the basins of attraction of maxima in $\psi(\mathbf{x})$ are different from the basins of attraction of minima in $V(\mathbf{x})$, thus providing different clustering schemes.

Another enlightening way to look at equation (15) is to write it as:

$$e^{-V(\mathbf{x})} = \frac{\psi(\mathbf{x})}{e^{S(\mathbf{x})}} \quad . \tag{17}$$

$\psi(\mathbf{x})$ is a pdf, with values proportional to the density of data. In particular, $\psi(\mathbf{x})$ has high values in regions of high density. The value of $e^{S(\mathbf{x})}$ is dominated by the highest probabilities in equation (14), that is, by the closest data points to $\mathbf{x}$. Therefore, $e^{S(\mathbf{x})}$ can be thought of as a measure of the number of nearest data points to the point $\mathbf{x}$. Thus, like the pdf, $e^{S(\mathbf{x})}$ is also high in regions of high density. In this view, $e^{-V(\mathbf{x})}$ is obtained from the pdf by locally normalizing by the number of nearest neighbors. The effect of this is an attenuation of high peaks to the values of the lower peaks.

To see this, we show in Figure 4 an example of the three functions $e^{-V(\mathbf{x})}$, $\psi(\mathbf{x})$, and $e^{S(\mathbf{x})}$ for a synthetic data set in one dimension, for three value of σ. For some σ, $S(\mathbf{x})$ has a maximum in the regions where $V(\mathbf{x})$ has a minimum and where $\psi(\mathbf{x})$ has a maximum. This suggests that $S(\mathbf{x})$ can also be used as a target function for clustering, just as $V(\mathbf{x})$ is used in QC, and $\psi(\mathbf{x})$ is used in

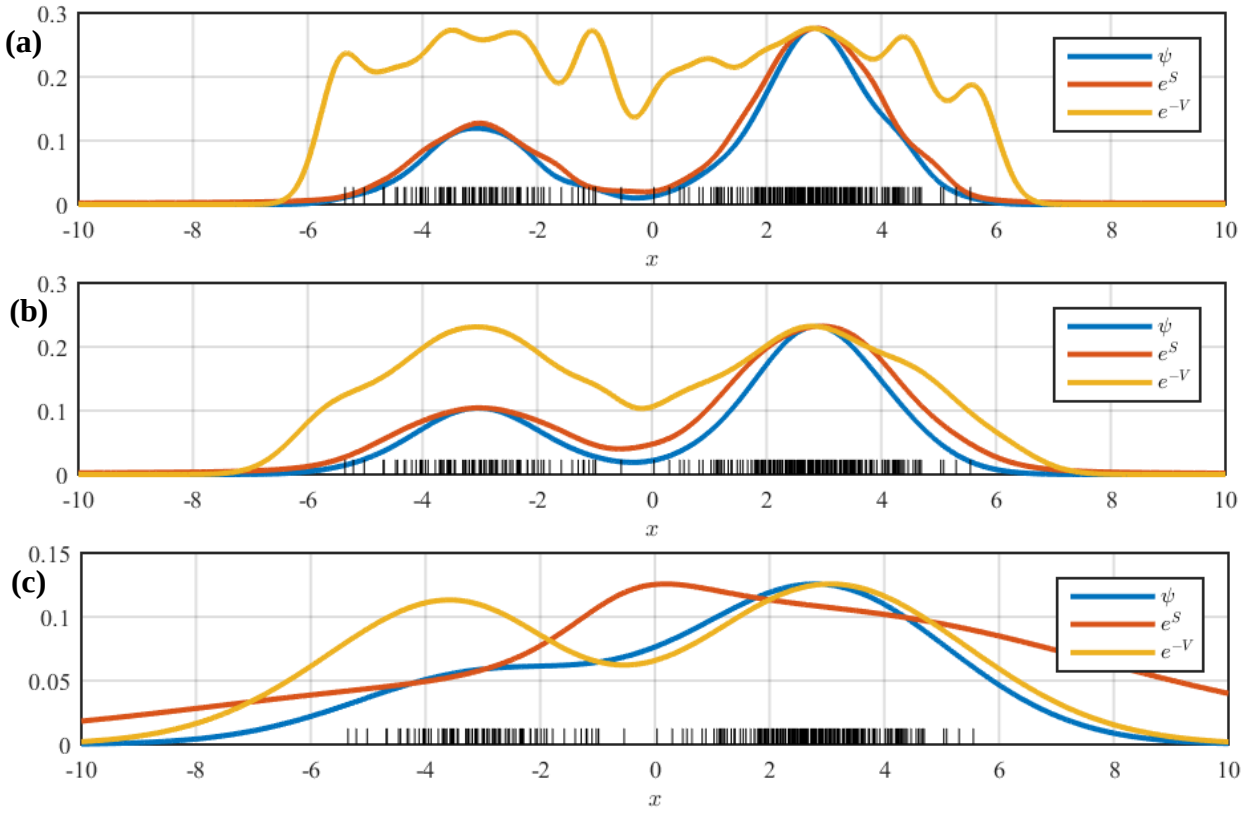


Figure 4: The values $e^{-V(\mathbf{x})}$, $e^{S(\mathbf{x})}$ and $\psi(\mathbf{x})$ for a one dimensional synthetic data set of 300 points and for three values of σ. The points are marked as vertical lines along the x-axis. For ease of comparison between the three functions, the functions are normalized to have the same maximum value. (a) $\sigma = 0.4$ (b) $\sigma = 0.7$ (c) $\sigma = 2$

the scale-space approach. These three algorithms all follow the same lines of flow of replica points in feature space, but with different target functions: QC, described in Box 1; Maximal Entropy Clustering (MEC), which is obtained by replacing $-\eta(\vec{\nabla}V)(\mathbf{x}_i')$ with $+\eta(\vec{\nabla}S)(\mathbf{x}_i')$ in Box 1; and Maximal Probability Clustering (MPC), which is obtained by replacing $-\eta(\vec{\nabla}V)(\mathbf{x}_i')$ with $+\eta(\vec{\nabla}\log\psi)(\mathbf{x}_i')$ in Box 1. All three algorithms also have hierarchical versions, analogous to HQC described in Box 2.

D. Blurring Dynamics

In the dynamics described by QC, MEC and MPC, the replica points move in the corresponding fields - $V(\mathbf{x})$, $\psi(\mathbf{x})$ or $S(\mathbf{x})$ – which are determined by the original data points. Alternatively, we could have defined the points $\mathbf{x}_i$ in equation (9) to be the replica points themselves, such that the function $V(\mathbf{x})$ changes on each replica update. This requires the replica updates to be performed concurrently, and not to wait for convergence of each replica point before moving the next one as described in Box 1. In [9], a similar process is called blurring, thus we call this algorithm Blurring Quantum Clustering (BQC) and it is described in Box 3. Blurring versions of MEC and MPC can similarly be described. A motivation for the blurring algorithm is that after each replica point has been updated, it is a bit closer to its "source" and therefore may serve as a better estimator for the actual pdf of the data. Disadvantages of the blurring process are: (1) The dynamics does not necessarily converge. (2) In regular QC, it is possible to assign novel data points

Blurring Quantum Clustering

(BQC1) Repeat until convergence into clusters:

(BQC1.1) $\{\mathbf{x}'_i\}_{i=1}^n \leftarrow \{\mathbf{x}_i\}_{i=1}^n$

(BQC1.2) For each replica point $\mathbf{x}_i$:

(BQC1.2.1) perform one gradient descent update:

$\mathbf{x}_i \leftarrow \mathbf{x}_i - \eta\left(\vec{\nabla}V\right)(\mathbf{x}_i)$, where V is formed by the replica points $\{\mathbf{x}'_i\}_{i=1}^n$

(BQC2) Group replica points that fall into the same minimum as a cluster.

Box 3: QC version with blurring

to previously derived clusters using the original potential $V(\mathbf{x})$ formed by the initial data points. This cannot be done in BQC, since $V(\mathbf{x})$ becomes a dynamic field that depends on the instantaneous positions of the replicas. (3) The dynamics and final positions of the replica depend on the precise prescription used in gradient descent, thus adding more degrees of freedom to the algorithm. For example, the step size η can be chosen to be constant, or to be proportional to the inverse of the gradient's norm, or based on a line search, or based on Newton's method or on a Quasi-Newton method [18]. Each choice may result in a different update for each replica and therefore in different dynamics of the field $V(\mathbf{x})$.

E. Relation to Fuzzy c-means

In k-means, the objective is to minimize the loss function

$$L(\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_k) = \sum_i \left(\mathbf{x}_i - \mathbf{c}(i)\right)^2 \ , \tag{18}$$

where $\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_k$ are cluster centers and $\mathbf{c}(i) \in \{\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_k\}$ is the cluster center closest to the data point $\mathbf{x}_i$. Cluster centers are initialized by some specific scheme, and the dynamics updates the cluster centers so as to minimize the loss.

In Fuzzy c-means, the "hard" assignment of a data point to a cluster is replaced by a "soft" assignment, such that data point $\mathbf{x}_i$ is assigned to the cluster with center $\mathbf{c}_j$ with a probability (or weight) $p(j|\mathbf{x}_i)$, such that $\sum_j p(j|\mathbf{x}_i) = 1$. Thus the loss function of the fuzzy algorithm is:

$$L(\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_k) = \sum_i \left(\mathbf{x}_i - \mathbf{c}(i)\right)^2 p(j|\mathbf{x}_i) \ . \tag{19}$$

The relation between this approach and QC can be demonstrated as follows: choose[8] the assignment probability as in equation (12). Choose the number of cluster centers k to be equal to the number of data points n, and the initial locations of the clusters $\mathbf{c}_i$ to be the locations of the data points $\mathbf{x}_i$. In other words, we take the fuzzy c-means setting, with initial cluster centers to lie exactly on all data points. In this setting, it can be seen that the loss function becomes:

$$L(\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_n) = 2\sigma^2 \sum_i V(\mathbf{x}_i) \ , \tag{20}$$

where $V(\mathbf{x})$ is given by equation (9) (or, equivalently, equation (13)), and where for each i, the

[8] This is different from the probability assignment in conventional fuzzy c-means, see [7]

initial location of $\mathbf{c}_i$ is $\boldsymbol{x}_i$. It is important to emphasize that although we choose initially $\mathbf{c}_i = \mathbf{x}_i$, we treat the variables $\mathbf{c}_i$ and $\mathbf{x}_i$ as independent. The optimization performed by fuzzy c-means is over the variables $\mathbf{c}_i$, keeping the data points $\mathbf{x}_i$ fixed.

Choosing as many cluster centers as there are data points is not in the spirit of the k-means or fuzzy c-means algorithms, since the number of clusters is expected to be much smaller than the number of data points. On the other hand, placing a cluster center on each data point has the following appealing property: a-priori, the best candidate locations for cluster centers are on data points, hence *all* candidate locations for cluster centers are considered, albeit with redundancy. Furthermore, if the number of data points is not very small, we can assume there is no need to actually move the cluster centers, since they are probably already located in good positions. In Fuzzy c-means this means that all there is left to do is to assign probabilities. But, an alternative approach could be the following: Instead of moving the cluster centers while keeping the data points fixed, move the data points while keeping the cluster centers fixed. Thus, in equation (20), instead of optimizing over $\mathbf{c}_i$, we optimize over $\mathbf{x}_i$. This means that each term in equation (20) can be optimized independently. The optimization should be done locally, so that each data point $\boldsymbol{x}_i$ is driven to its local minima, which is determined mainly by the local distribution of the cluster centers.

The dynamics described in the previous paragraph is identical to the dynamics suggested by QC. We see that in this context, QC can be seen as a dual algorithm to fuzzy c-means, where instead of moving cluster centers while keeping data points fixed, it moves replicas of the data points while keeping cluster centers fixed. QC is recovered when the initial cluster centers are taken to be exactly the data points.

Another interesting perspective is obtained by rewriting equation (20) using equation (15), within the conventional fuzzy c-means setting, where the number of clusters and their locations are initialized at will:

$$L(\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_k) = 2\sigma^2 \textstyle\sum_i S(\mathbf{x}_i) - 2\sigma^2 \log \prod_i \psi(\mathbf{x}_i) \ . \qquad (21)$$

It follows then that the conventional fuzzy c-means seeks to minimize a loss that is comprised of two terms: The first is a total entropy of the system, and the second is the (negation of) the log-likelihood of the data, given that it was generated with a Gaussian mixture model.

F. Extending QC to Big Data

A data set can be "big" in two respects: (1) the number n of data points. (2) the dimension d of feature space. In an era of big data sets, a very desirable property of a clustering algorithm is its ability to run on such data sets in reasonable time.

The time complexity of the QC algorithm is $O(n^2 dt)$, where t is the number of steps required for the gradient descent to converge[9]. t is highly dependent on the data set structure and on the type of gradient descent algorithm used. The reason for the n^2 term is that the gradient on each of

[9] Each data point may need a different number of gradient descents steps. Thus, t should be thought of as an average quantity.

the n replica points is calculated using all of the n data points. The d term is there because calculating distances and differences between two points in d dimensions requires summing over d numbers.

Dealing with the t term can be done using an efficient gradient-based algorithm. In [19], for example, the use of the Broyden-Fletcher-Goldfarb-Shanno algorithm [18] was suggested. The d term can be dealt with using dimensionality reduction techniques, such as principal component analysis (PCA) [20] or diffusion maps [21]. Reducing the dimensions should be done with care, since the objective of some of these techniques is to reduce dimensions with small global variance, but these dimensions may have valuable information for the local structure of clusters. Also, the time complexity of reducing dimensions needs to be taken into account.

The most important factor on the performance of QC on big data is the n^2 term. In the following, we describe some strategies that can be used to cope with a big number of data points. These strategies can be combined. Some of them are ideas that haven't been tested yet and should be further researched.

Parallelism: QC is "embarrassingly parallel", in the sense that the gradient descent of each replica can be calculated completely in parallel to all others. This calls for a multi-threaded or cluster-distributed implementation of QC. It does require, though, that all data points are stored in the memory of each cluster.

Hierarchical QC: As described above, in HQC the number of replica points is reduced as the algorithm proceeds. This can help, but the initial clustering, for low σ, is still using n replicas. The performance of HQC depends strongly on the choice of the grid of σ values.

Approximate QC (AQC): As suggested in [22], the wave function $\psi(\mathbf{x})$ is replaced by an approximation $\hat{\psi}(\mathbf{x})$, constructed using a small subset of the data points $\mathbf{x}_i$. These representative points are chosen so as to give a good representation of the original $\psi(\mathbf{x})$. When constructing the approximate $\hat{\psi}(\mathbf{x})$, the Gaussian of each representative point is multiplied by a coefficient representing the density of data points around its location.

QC with Stochastic Gradient Descent: In this approach, the gradient of a replica point at a certain iteration is calculated using a small, random subset of the data points $\mathbf{x}_i$. This random batch is sampled anew on every iteration and for any replica point. The gradient calculated using the batch can be thought of an estimation of the true gradient. This approach removes the need to perform any $O(n^2)$ calculations. However, there is a tradeoff between the size of the batch and the size of the gradient step. A small batch means that the estimation of the gradient will have high variance, thus a small step size is needed to ensure that replica points don't deviate from their path. Another important issue that should be looked at is the bias of the estimate. If it is biased, repeated use can lead a replica point in wrong directions. A disadvantage of this approach is that small clusters, or outliers, may be overlooked since a random batch has a high chance of missing the data points in the small cluster, and therefore they won't generate any gradient force on the replica.

Using nearest neighbors: This is similar to the previous method, the difference being that the

gradient is calculated using just a set of k-nearest data points. For low dimensions, a repeated query of k-nearest neighbors can be done efficiently using the k-d tree data structure [23]. For higher-dimensions, an approximate nearest-neighbor calculation can be used, such as Locality-Sensitive Hashing [24].

QC on a graph: Calculating a new gradient for each replica point on each iteration of the algorithm is what takes the most compute time. If the movement of replica points were restricted only to the set of data points, then there would be no need for a new calculation of the gradients for each iteration, since each replica will always be in a location where the gradients have already been calculated. A prescription is needed for updating the location of a replica point, given a gradient. This approach has two more beneficial properties: (1) if two replica points meet at one location, they may be fused into one. (2) If a replica point a arrives at a location which was earlier occupied by a previous replica point b, then the future dynamics of a is identical with the dynamics of b when it was at that location, so there is no need to progress a any more since we know its future.

III. APPLICATION TO ASTEROID SPECTRAL TAXONOMY

A. Introduction to Asteroids

1. *Asteroids and their Distribution and Formation*

Asteroids are small solid objects that orbit the sun. They can have a diameter as small as tens of meters, or as big as thousands of kilometers. The millions of asteroids in the solar system are mainly distributed among asteroid belts. The largest asteroid belt, in terms of the number of asteroids, is the Main Belt[10]. It contains asteroids whose elliptical orbits have a semi major axis in the range 1.52 AU to 5.2 AU, between Mars and Jupiter. More than 99% of the solar-system asteroids are contained in the Main Belt. Figure 5 shows the distribution of asteroid mass within the Main Belt. It can be further divided into families based on semi-major axis values.

The fact that most asteroids reside in the Main Belt can be explained by the process of asteroid formation[11] [25]. Asteroids, like planets, were formed from solar nebula, containing gas and dust left over from the sun's formation. Clumping of the solar nebula, along with collisions between these clumps, led to objects of sizes of about 10 km over the course of millions of years. As the process continued, larger objects were formed, some of which eventually, merged into the terrestrial planets. But objects at distances of around 2 AU to 4 AU from the sun had a different history. When Jupiter was formed, some of these objects were in resonance with it. This means that the ratio of the orbital period of Jupiter around the sun to the orbital period of these objects is close to a ratio of small integers, and therefore the gravitational influence of Jupiter on these objects adds up coherently in time. Saturn also formed resonances with some of these objects. The resonances caused an increase in the velocity of the objects, which upon collisions would shatter, rather than clump, into smaller objects, which became the asteroids. The gravitational interaction of Jupiter with other solar system objects caused it to migrate inwards towards the sun, and as this migration occurred, the resonating regions were swept along the asteroid belt, exciting more objects to higher speeds, thus forming more asteroids.

Another, smaller, group of asteroids is the Near-Earth Objects (NEO), which consists of asteroids (and comets[12]) whose semi-major axes are close to earth's semi-major axis.

2. *Asteroid Designation*

By the International Astronomical Union (IAU) standards, asteroids can be identified both by a name and by a numerical designation. For example, Ceres is the name of the asteroid whose designation is 1. To get a name and a permanent numerical designation, the asteroid must be observed a number of times at different oppositions. Before it gets a permanent designation, a provisional designation is used, which includes the year of observation and some additional characters, for example 2001-VS78. Asteroids are given names only after approved by a committee

[10] The Kuiper belt is a larger asteroid belt that lies beyond Neptune, the farthest planet from the sun.

[11] I give a simplistic description here.

[12] Comets are objects similar to asteroids, the main difference being that comets are also composed from ice

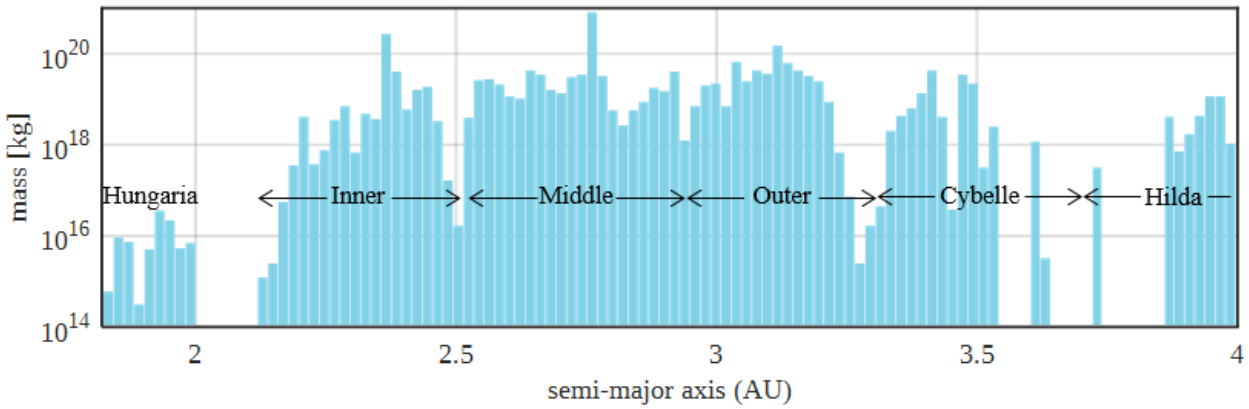


Figure 5: Total mass of Main Belt asteroids per 0.02 AU bins, and names of asteroid families. Only asteroids larger than 5 km are considered here. Note that the Trojan family is not shown in this figure. The figure was reproduced from [48]

of the IAU.

B. Asteroid Composition and Their Taxonomy

1. *Composition*

A main source for the study of asteroid composition is the observation, using ground-based telescopes, of the colors of the sun light reflected off the asteroids, both in the visual and near infrared domains. Analysis of the reflected spectrum teaches us about the presence of absorbing material on the surface of the asteroid. Other, less common, methods to gather information about asteroid composition are by performing close-up measurements by a spacecraft orbiting the asteroid (such as the space probe Dawn, launched by NASA, which was in orbit around the asteroid Vesta during 2011 and 2012, and as of 2017 it is in orbit around Ceres), and by analyzing the composition of meteorites that have fallen onto earth.

Asteroids come in three major types[26], based on their composition: S-type asteroids are stony and consist mainly of silicates, that are manifested in the spectrum by absorption features at around 1μm and 2μm. C-type asteroids are carbonaceous. Their spectra show absorption at the ultraviolet end, bellow 0.5μm. The third group consists of metallic asteroids, whose spectra are featureless. Yet the classification of asteroids into these three types is by no means clear-cut, since asteroids can be made up of mixtures of these materials, or of other materials. Therefore, finer taxonomies have been proposed over the years. In the following paragraphs, we'll describe two of the proposed asteroid spectral taxonomies.

2. *Tholen Taxonomy*

The taxonomy suggested by Tholen [27] in 1984 is based on the reflected spectrum in eight wavelengths in the range 0.337μm to 1.041μm. This range overlaps with the visual range (VIS) and with some of the near-infrared range (NIR). In addition, albedo measurements are used. A total of 589 asteroids have been measured. The data was preprocessed to have unit variance for each wavelength and was normalized to have unit reflectance at wavelength 0.55μm, thus reducing the dimension by one. The normalization is needed since the measured reflectances represent relative magnitudes, not absolute ones.

The 589 measurements were divided into 405 “high-quality data” and 184 “low-quality data”, based on their given measurement errors. Using the high-quality data, a minimal spanning tree was constructed. The vertices of the tree are the asteroids and the value on each edge was taken as the Euclidian distance between the asteroids’ seven dimensional spectra. Edges with large values were manually removed, and the resulting connected components were each considered to be a cluster. Some of the clusters were further partitioned to smaller clusters using the albedo data. At this point, a repeated process of optimizing the clusters has been performed, where on each step, all data points (high-quality and low-quality) were attributed to clusters based on their three nearest neighbors from within the high-quality data points. Cluster centers were computed by the mean over all asteroids whose three nearest neighbors belong to the cluster. The repeated process terminates when cluster centers converge. Finally, albedo data was used to refine the clusters.

The Tholen taxonomy comprises of 14 classes. The carbonaceous classes **B**, **C**, **F** and **G** all have an ultra-violet feature, whose main absorption wavelength is just bellow the smallest measured wavelength. The depth of the feature and the sign of the slope at higher wavelengths differ between the four classes.

The stony classes **S**, and **A** have features around 1μm. In the **A** class, the feature is more prominent.

The classes **E**, **M** and **P** are featureless, and differ in their albedo. **T** and **D** are also featureless. These classes are distinguished by the behavior of their slope.

Three more classes - **Q**, **R** and **V** were used for asteroids with unique spectra, making them singleton classes (in Tholen’s data set). All of these spectra show absorption features around 1μm.

3. *Bus-DeMeo Taxonomy*

An improved taxonomy was suggested by DeMeo *et al.*[28] in 2009, which we refer to as the Bus-DeMeo taxonomy. The range of measured wavelengths is larger than Tholen’s: 0.45μm to 2.45μm, but albedo data wasn’t used. The 371 asteroid spectrum measurements were spline-interpolated to a wavelength grid of 0.05μm. The reflectance was normalized to 1 at the wavelength 0.55μm.

A guiding principle in the construction of the Bus-DeMeo taxonomy was to be consistent with an older taxonomy, by Bus and Binzel [29]. The Bus taxonomy was performed in VIS, and the Bus-DeMeo taxonomy is its extension into NIR. Thus, in accordance with the Bus taxonomy, the slopes of the spectra were computed and removed. The slope of a spectrum is defined as the number γ which makes the line $r = 1 + \gamma(\lambda - 0.55\mu\text{m})$ closest as possible to the spectrum (in least-squares sense). When the slope is found, the spectrum is divided by the line. The reason the slope is removed is that it was found that consecutive measurements give rise to large variations in the slope. The slope depends on factors such as humidity and clouds, and also on angle of observation.

After the slope was removed, PCA was applied to the data, and the spectra were examined in six dimensions which include projection on the five first principal components, and the parameter

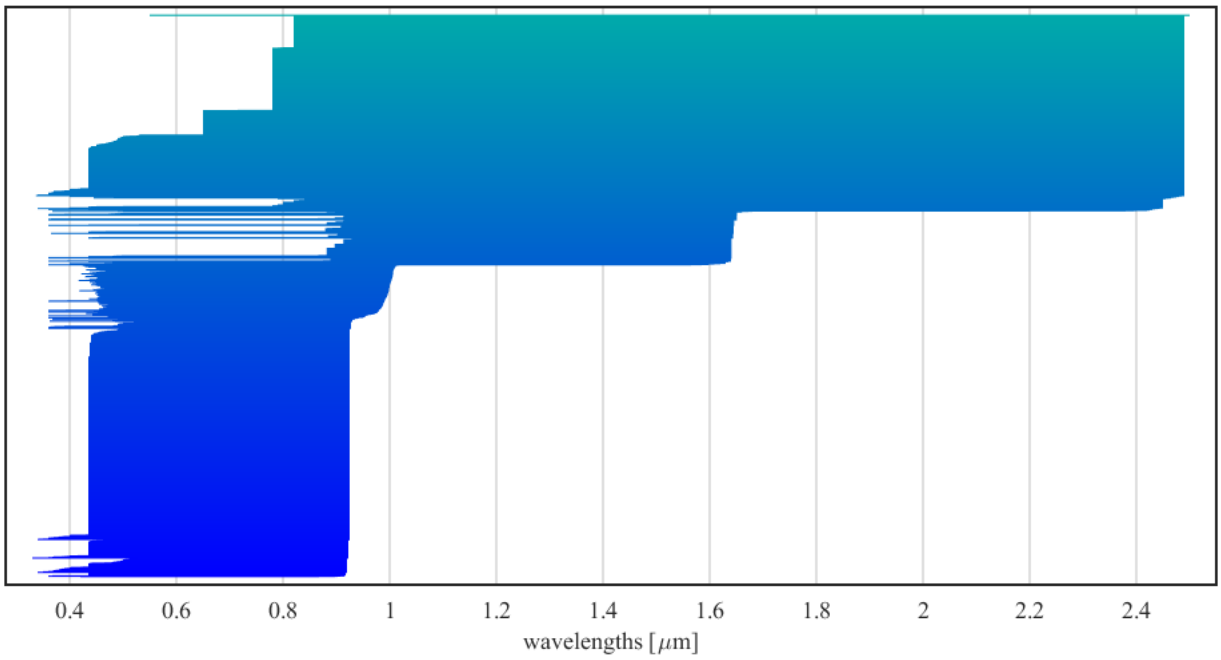


Figure 6: Wavelength ranges of data. Each horizontal slice of this figure represents one asteroid measurement. The wavelength spanned by the slice are the wavelengths measured. This figure does not represent the resolution of measurements.

γ. Thus, although the slope was removed from the spectra, γ is still reinstated as a feature.

Clustering was performed manually by examining the data in planes spanned by pairs of the six dimensions. Classes were defined by dividing the space into regions with linear boundaries, with the Bus classes as guidelines. 27 classes were formed in this taxonomy.

C. The Data

1. *Data Acquisition*

The reflected spectrum of an asteroid is measured using ground-based telescopes with a charge-coupled device (CCD) sensor for VIS[30], or a NIR spectrograph for NIR[31]. The measured quantity is the intensity of sun light that is reflected from the asteroid's surface, and then passes through the earth's atmosphere before being detected. Atmospheric absorption and scattering cause attenuation of the measured signal, and the spectral shape of the attenuation depends on atmospheric conditions, such as humidity and clouds. Also, the attenuation can depend on the direction of observation, since in each direction the light travels through different air masses. To correct for this distortion, the measured spectrum is divided by the measured spectrum of an analog solar star in a direction close to that of the asteroid. This is a star with a spectrum similar to that of the sun.

Noise in the sensor, atmospheric attenuation and solar analog correction all introduce errors to the resulting spectrum. The amount of uncertainty in the measurement can be calculated based on the sensor gain, noise and on the correction.

It is important to note that the measured spectrum represents only an average characteristic of the surface of the asteroid over the disk illuminated by the sun light that is reflected to earth. It is

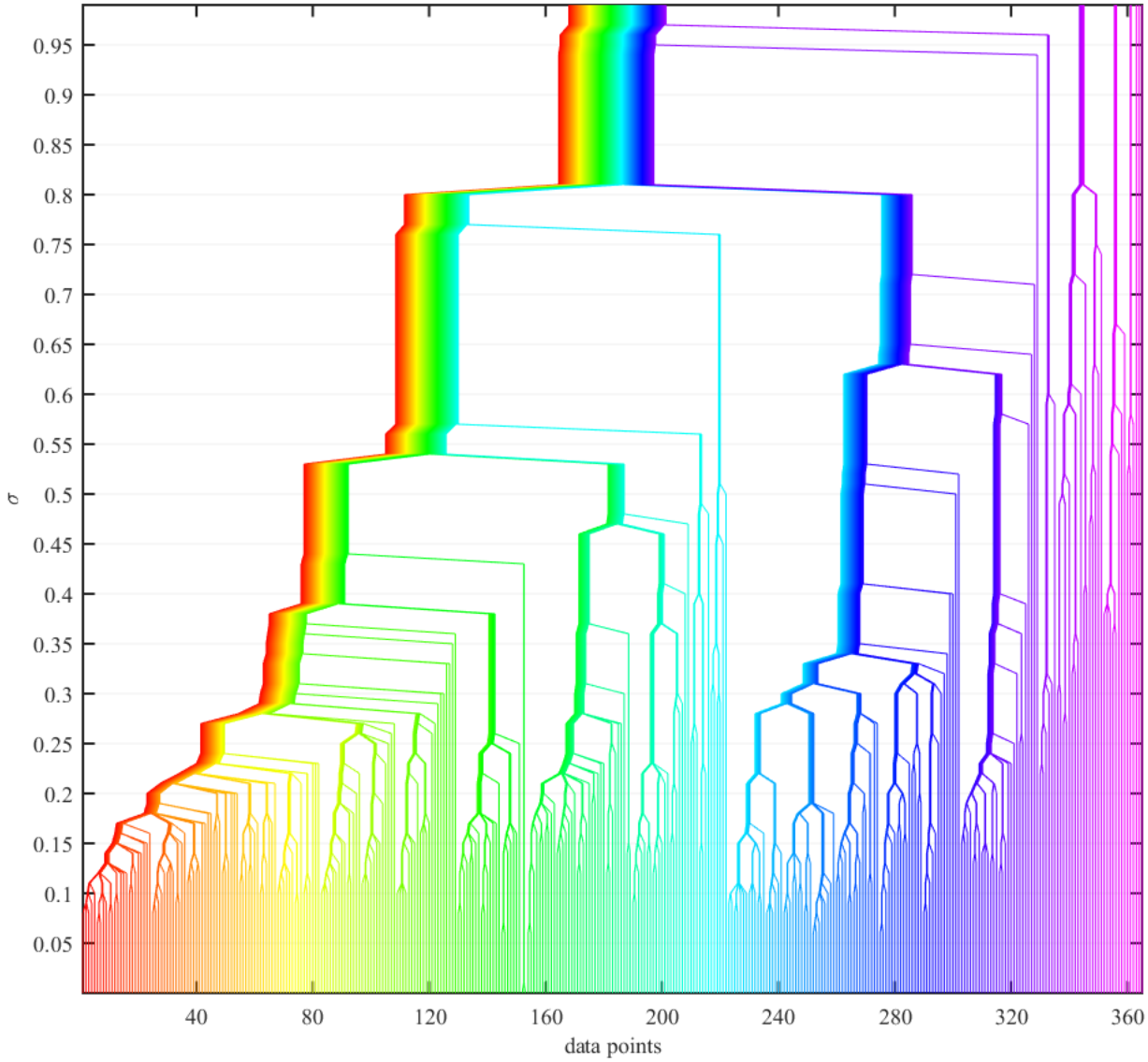


Figure 7: The hierarchical clustering tree obtained from applying HQC to the asteroid spectrum data.

usually assumed that the bulk of the asteroid has a similar composition to the surface, but grain size, temperature, exposure to radiation and viewing angle can affect the spectral properties.

2. *Data Source*

The data was downloaded from the MIT Planet Spectroscopy group website [32], which contains data from multiple sources[29] ([29],[34]-[45]), some of them unpublished. The data consists of spectra of asteroids, primarily from the Main Belt but also from the NEO family. At the time of access, the website contained measurements of 2659 asteroids, in VIS and NIR.

The measurements in NIR were performed mainly using the spectrograph SpeX with the NASA Infrared Telescope Facility in Mauna Kea, Hawaii [31]. It has two operation bands, $0.8 - 2.5\mu$m and $2.5 - 5.5\mu$m. The former band, which is less corrupted by thermal background noise from the telescope and sky, was used for the asteroid measurements. SpeX includes a dispersive prism that splits the different wavelength onto an array of InSb pixels. This allows the simultaneous

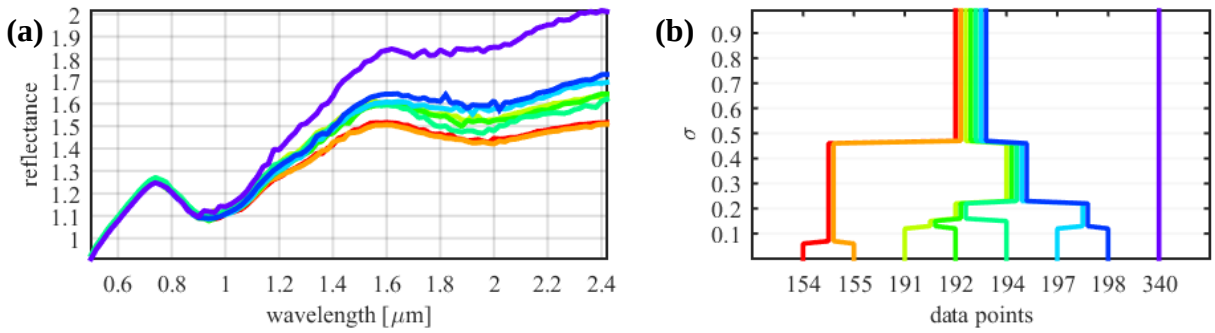


Figure 9: (a) Eight different spectra of Eros. (b) The sub-tree of the HQC hierarchical tree spanned by the eight Eros measurements, with colors corresponding to the spectrum graphs . Data point numbers correspond to Figure 7.

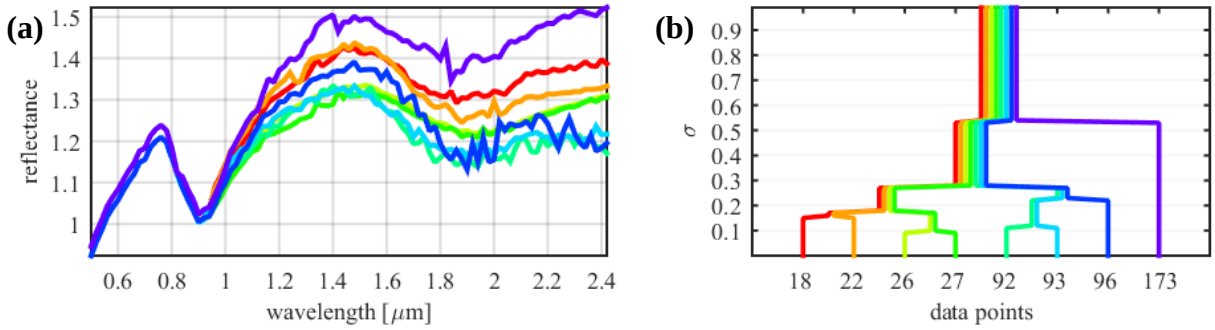


Figure 8: (a) Eight different spectra of Ganymed. (b) The sub-tree of the HQC hierarchical tree spanned by the eight Ganymed measurements, with colors corresponding to the spectrum graphs. Data point numbers correspond to Figure 7.

capturing of the entire spectrum, and it is therefore guaranteed that spectra taken at the same time come from the same position. In particular, this means that the effect of the asteroid rotation on its instantaneous intensity can be neglected.

For VIS, one of the instrument used was the Mark III spectrograph with the Hiltner telescope, located at the Michigan Dartmouth MIT Observatory in Arizona. Like SpeX, this spectrograph can also capture the entire spectrum in a single exposure, onto a CCD pixel array.

Each measurement file consists of a list of wavelengths, and for each wavelength there exists a reflectance value and a measurement error. The reflectance values were normalized such that the reflectance at wavelength 0.55μm equals 1.

The total number of asteroid measurements is 3518, as some asteroids were measured more than once. Spectra of asteroids with multiple measurements are not always independent, though, since in some cases two different NIR measurements are joined with the same VIS measurement.

The asteroid measurements don't all share the same wavelength range and resolution. Some asteroids were measured only in VIS and some only in NIR. Figure 6 shows the wavelength range of each measurement. In our clustering analysis we demand all data points to be defined on the same set of features. This means that we shall disregard all measurements which don't include both VIS and NIR ranges. In particular, we use only measurements which include the wavelength range of 0.5μm to 2.43μm, and which don't have a gap larger then 0.1μm where the spectrum

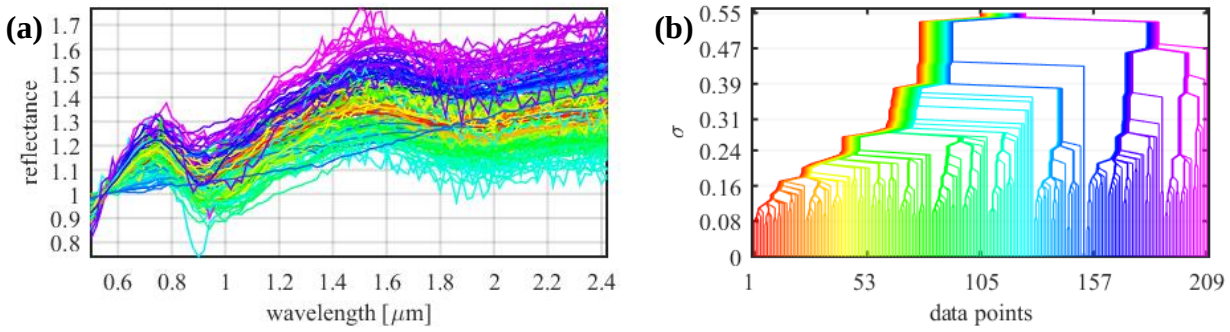


Figure 10: (a) The largest cluster obtained from $\sigma = 0.55$. (b) The sub-tree of the HQC hierarchical tree that leads to this cluster. Data point numbers correspond to Figure 7.

was not measured. This leaves us with 286 asteroids and 365 measurements. This is about 10% of the original data. Most measurements have a resolution of about 0.05μm.

3. *Preprocessing*

The data was linearly interpolated to a grid with 0.02μm resolution, with the smallest wavelength being 0.5μm. Since the resolution is smaller than the measured resolution, prior to the interpolation each spectrum has been smoothed with a triangular filter with a width of 0.02μm. Thus, the data after preprocessing consists of 365 measurements in 97 dimensions.

We decided not to remove slope of the spectra. Although it is considered a less reliable feature of the spectra, it may still convey valuable information about physical properties of asteroids. In particular, exposure of the asteroid to radiation from the sun and to meteoroids – a phenomenon known as space weathering - can cause the slope to increase, thus reddening the asteroids. A young asteroid which is the product of older asteroids colliding may have faces which were exposed to space weathering for just small amount of times. Also, the Bus-DeMeo taxonomy removes the slope to be similar to the Bus taxonomy, which itself tried to preserve the Tholen taxonomy. But it may be a good idea to start a taxonomy afresh, without resorting to older taxonomies and assumptions, and to examine the results.

D. Results of Applying HQC

HQC was applied to the data. The starting σ was chosen to be 0.01, which is small enough to cause hardly any motion of replicas. σ was then increased by 0.01 at each step of HQC. The code implementing HQC can be found in [33].

The hierarchical tree obtained from the clustering is presented in Figure 7. The most general way to obtain a final clustering of the data is to choose a cutoff value for σ separately for each branch of the tree. Here we will focus on a constant cutoff applied to the whole tree.

One may think that a guiding principle in the selection of the final σ value could be that the spectra of different measurements of the same asteroid should all fall into the same cluster. The asteroids Eros (433) and Ganymed (1036) both have eight measurements in our data set, more than any other asteroid. These are shown, respectively, in Figure 9 and in Figure 8, along with the sub-tree that shows when these measurements merge. From these figures it is seen that for Eros, if we

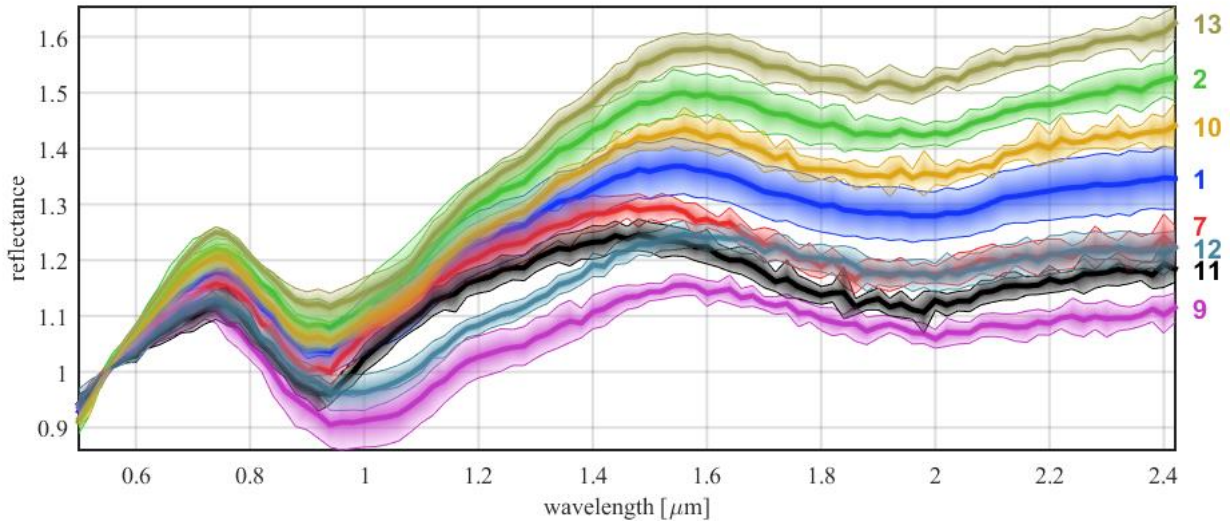


Figure 11: Cluster means for the eight largest clusters, among clusters that have two absorption features, obtained for $\sigma = 0.22$. The shade around each cluster represents the standard deviation of the cluster members for each wavelength. The numbers on the right are the HQC cluster designations.

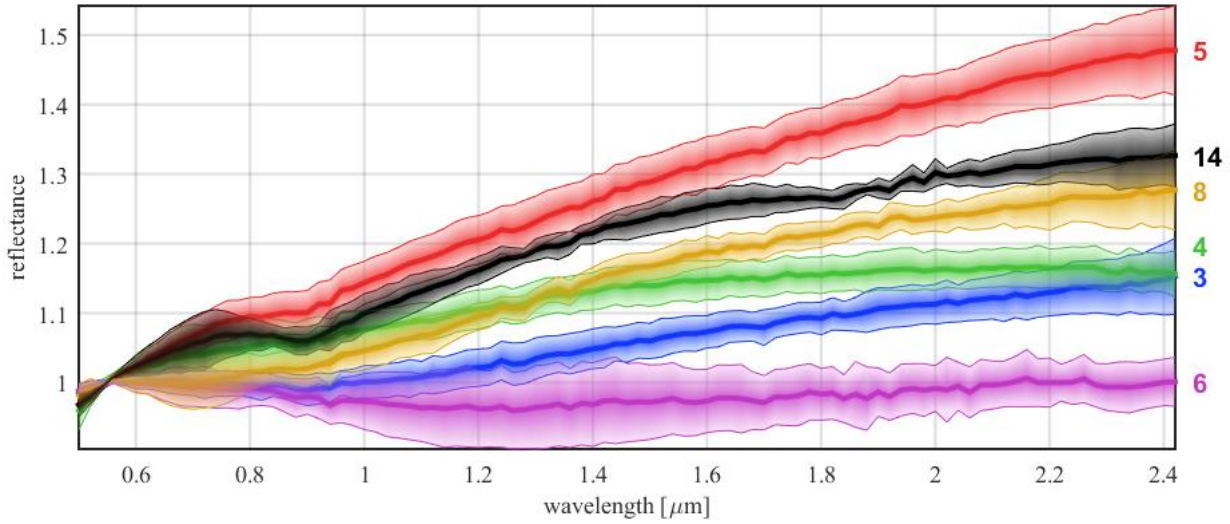


Figure 12: Cluster means for the six largest clusters, among clusters that have a flat waveform, obtained for $\sigma = 0.22$. The shade around each cluster represents the standard deviation of the cluster members for each wavelength. The numbers on the right are the HQC cluster designations

ignore the one spectrum which is quite different from all others, the value of σ which merges the Eros measurements is $\sigma = 0.5$. For Ganymed, the value is $\sigma = 0.55$, although already for $\sigma = 0.3$ all but one measurement merge.

Using $\sigma = 0.55$ as a cutoff, the largest cluster obtained is shown in Figure 10. This cluster is quite broad and is very heterogeneous in the spectra it contains, including both flat waveforms and wavy waveforms with absorption features. The conclusion is that clustering at scale of $\sigma = 0.55$ gives clusters which are too coarse. Another conclusion is that we shouldn't necessarily seek for a unique cluster designation for each asteroid. The variance between different measurements of the same asteroid gives rise to waveforms which may be significantly different from each other. The variance is probably a result of viewing the asteroid from different directions and at different

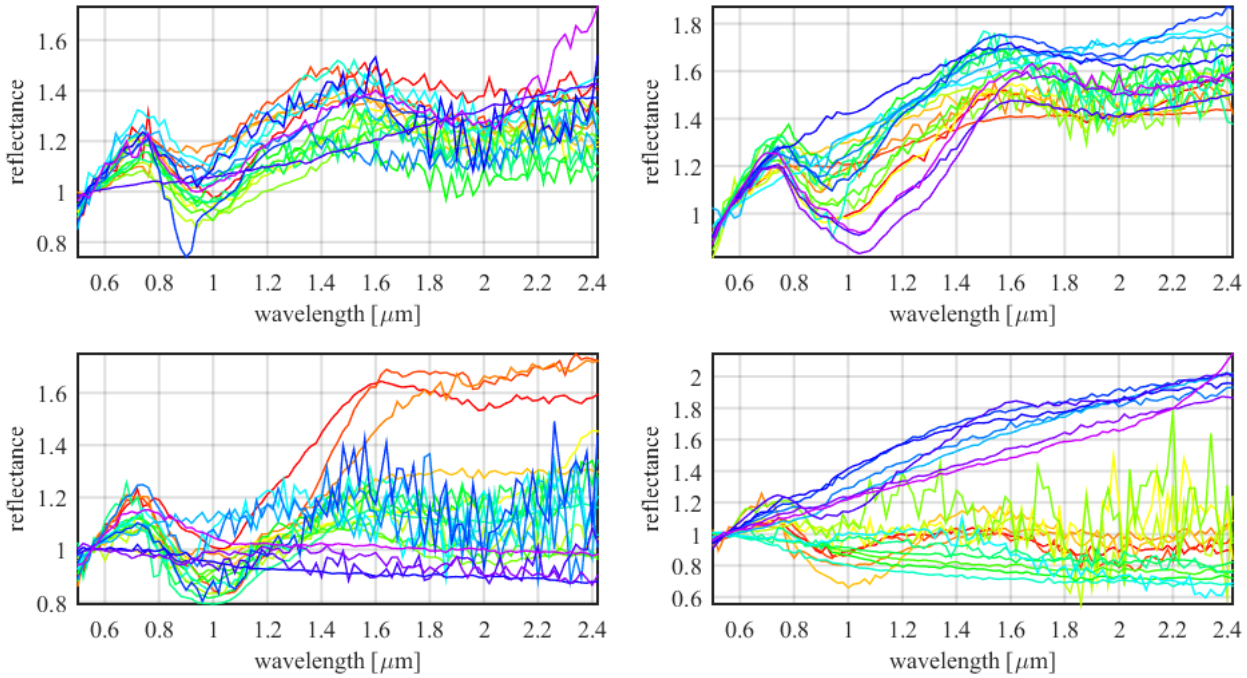


Figure 13: All spectra that fell into HQC singleton clusters at $\sigma = 0.22$.

angles. The asteroid's surface is not necessarily homogenous in its grain size, temperature and exposure to radiation, and this may be the reason for the differences. The clusters we find, then, are classes of spectral types and not of asteroids. This is on par with the conclusion presented in [30]:

> *"The classification assigned to an asteroid is only as good as the observational data. If subsequent observations of an asteroid reveal variations in its spectrum, whether due to compositional heterogeneity over the surface of the asteroid, variations in viewing geometry, or systematic offsets in the observations themselves, the taxonomic label may change. When this occurs, we should not feel compelled to decide which label is "correct" but should rather accept these distinct labels as a consequence of our growing knowledge about that object."*

Following this conclusion, we look at smaller values of σ. We choose $\sigma = 0.22$, which by visual inspection gives tight clusters which are different from each other. Figure 11 and Figure 12 show the means of the largest clusters. We designate the clusters by consecutive integers, sorted by cluster size, starting with **1** for the biggest cluster. The complete association of each asteroid spectrum to a cluster is given in Table 1. The spectra of each cluster are shown in Figure 19 to Figure 44.

The results show that that some clusters, such as **2** and **13**, have minute differences that lead to their differentiation into separate clusters. The question of whether these clusters should actually be merged has no obvious answer. This merger will occur for a larger value of σ, which can be set either globally or locally to this branch of the hierarchical tree. Alternatively, large clusters

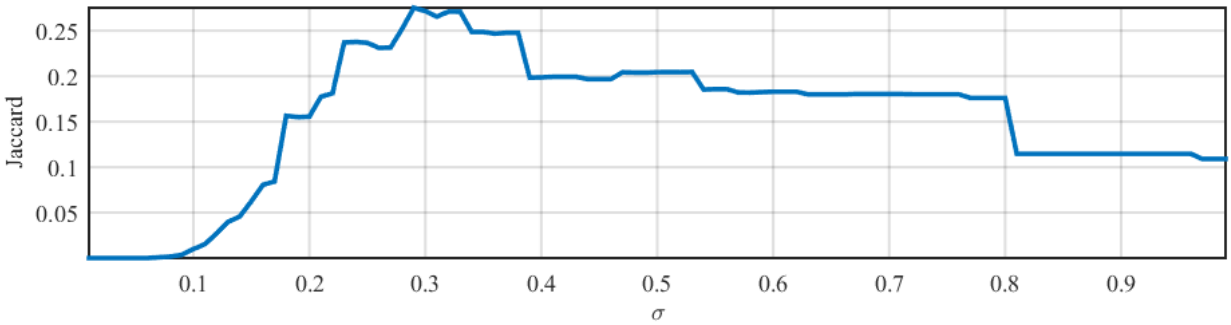
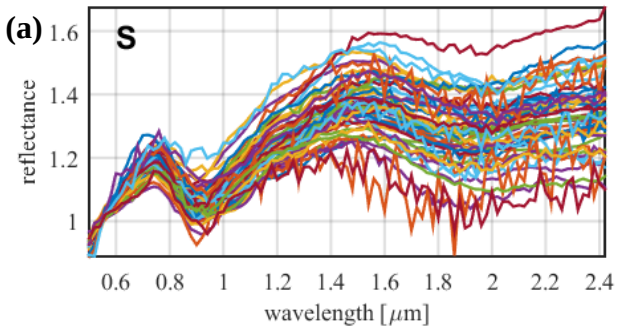
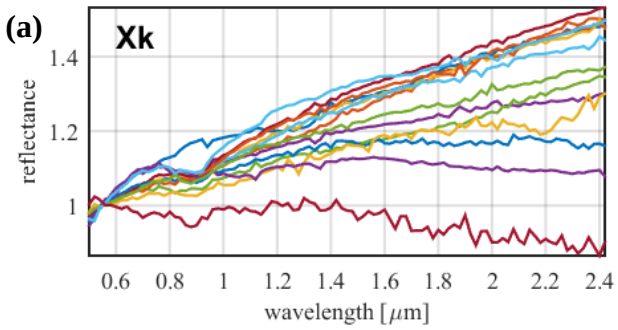


Figure 14: Jaccard similarity score between the Bus-DeMeo taxonomy and the HQC clustering for various values of σ.



(b) **Members**: 5, 7, 14, 17, 20, 25, 26, 27, 29, 30, 37, 40, 57, 61, 67(1), 67(2), 73, 82, 101, 103, 119, 158, 264, 288, 371, 389, 403, 532, 631, 699, 719, 793, 925, 1036(1), 1036(2), 1036(3), 1036(4), 1036(5), 1036(6), 1036(7), 1036(8), 1065, 1131, 1565, 1620(1), 1620(2), 1620(3), 1640, 1660, 2335, 3402, 6585, 8444, 20790, 22771, 2002-AA

Figure 15: (a) The cluster S in the Bus-DeMeo taxonomy. (b) Cluster members.



(b) **Members**: 16, 55(1), 55(2), 56, 65, 69(1), 69(2), 99, 110, 160, 181, 201, 250, 25330

Figure 16: (a) The cluster Xk in the Bus-DeMeo taxonomy. (b) Cluster members.

such as **1** may be separated into smaller clusters by decreasing σ for the cluster.

We obtain, for $\sigma = 0.22$, 101 singleton clusters, displayed in Figure 13. Singletons and small clusters may indicate that an asteroid has unique properties which should further be investigated by experts. Also, as measurements accumulate in the future, additional spectra might be added to the singletons forming new clusters.

E. Comparison with the Bus-DeMeo Taxonomy

The Jaccard similarity score [46] is a measure for comparing two partitions of the same data set. It is defined as

$$J = \frac{n_{11}}{n_{01}+n_{10}+n_{11}}\ , \tag{22}$$

where n_{11} is the number of data point pairs that belong to the same cluster in both partitions, n_{10}

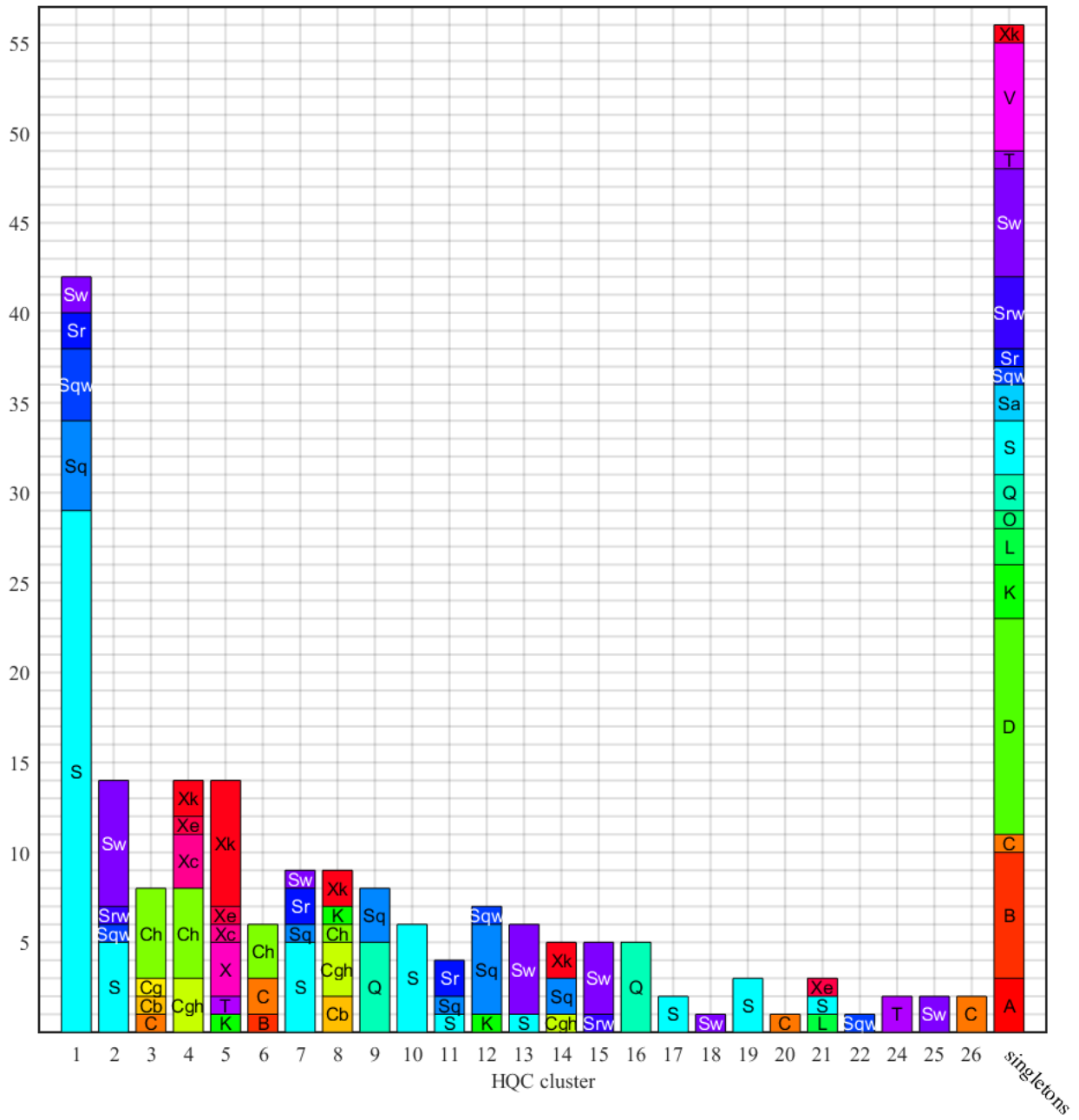


Figure 17: Comparison of the HQC results with the Bus-DeMeo taxonomy. Each vertical stack of columns represents the distribution among the Bus-DeMeo classes of a single HQC cluster. The letters in the columns are the Bus-DeMeo class designations.

is the number of data point pairs that belong to same cluster in the first partition but not in the second partition, and n_{01} is the number of data point pairs that belong to same cluster in the second partition but not in the first partition. The value of J is always between 0 and 1, where higher values are obtained when the two partitions are more similar to each other.

The Jaccard score for the Bus-DeMeo taxonomy and the HQC clustering, for each value of σ, is shown in Figure 14. The score is based on a comparison between 235 spectra which are in the intersection of the data set used in both the Bus-DeMeo taxonomy and HQC analysis. The highest score is 0.28, which is quite low. In particular, for our proposed clustering at $\sigma = 0.22$, the score

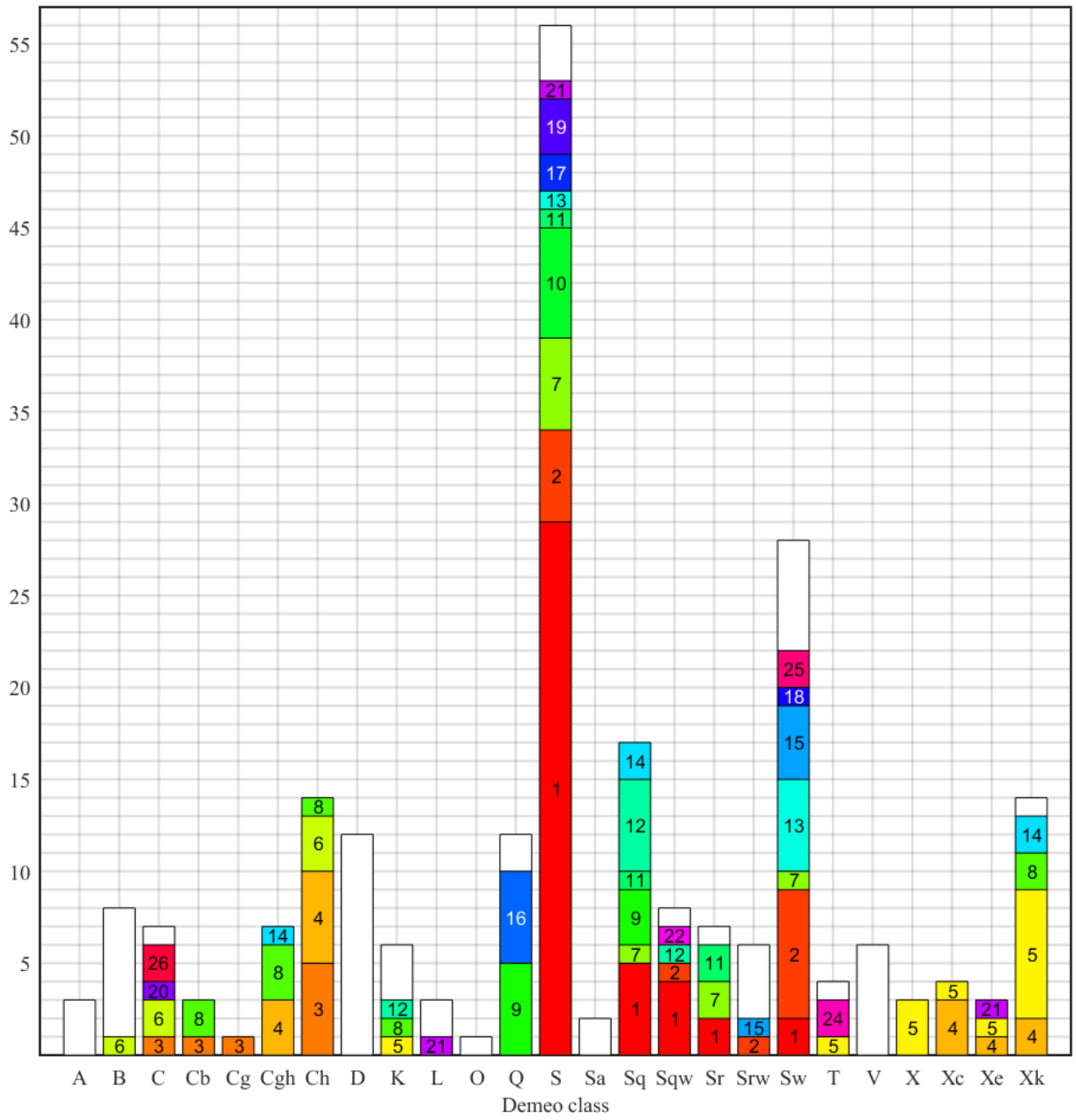


Figure 18: Comparison of the HQC results with the Bus-DeMeo taxonomy. Each vertical stack of columns represents the distribution among the HQC clusters of a single Bus-DeMeo class. The numbers in the columns are the HQC class designations. White columns, with no number, are singleton clusters

is 0.18. This means that the clustering we obtain is substantially different from the Bus-DeMeo taxonomy. This is evident by visual inspection of the classes in the Bus-DeMeo taxonomy. Since in the initial stage of the Bus-DeMeo procedure the slope is removed, some clusters of the Bus-DeMeo taxonomy have high variance in terms of the spectra when viewed before slope removal. Examples are shown in Figure 15 and Figure 16 for two particular classes[13].

[13] Class assignment for each asteroid in the Bus-DeMeo taxonomy was taken from [28]. Since our data set consists of multiple measurements for some asteroids, all of these measurements get assigned to the same Bus-DeMeo taxonomy

Figure 17 and Figure 18 show the sizes of the intersections between clusters in HQC at $\sigma = 0.22$, and classes in the Bus-DeMeo taxonomy.

class. This causes the Bus-DeMeo taxonomy classes as presented here to have slightly larger variances.

IV. Discussion

The QC algorithm was originally motivated by the quantum mechanical system of a particle in a potential. In this work, we have shown that there are more ways to gain insight about the algorithm's workings: The entropy formulation relates the quantum potential to the information theoretical concept of entropy, and it was also shown that the algorithm can be understood as a sort of a dual algorithm to fuzzy c-means. Using these alternative formalisms can lead into new insights about the algorithm. For example, the entropy relation suggests that QC belongs to a family of algorithms which are based on replica flow in feature space. Another algorithm in this family is obtained by maximizing the entropy, instead of minimizing the quantum potential.

Another reason to adopt multiple formulations of QC is to bridge the gap between the physics community and other communities that study cluster analysis, since the quantum mechanical model may not be accessible to researchers or practitioners without background in quantum physics. The concept of entropy, on the other hand, is more widely known.

It was shown the QC can be turned into an agglomerative hierarchical algorithm, HQC. This has the conceptual benefit that clusters which are obtained at larger scales are always disjoint unions of clusters obtained at any smaller scale. The result of HQC is a tree representing the merger of clusters as the scale gets larger. This tree can then be cut at different branches to obtain a final clustering for the problem. Further work could find schemes of performing the cut automatically, based on principles such as the stability of clusters for a wide range of scales.

HQC was applied to the problem of asteroid spectral taxonomy in the VIS and NIR ranges. It seems to provide good results – tightly packed clusters, as compared to the Bus-DeMeo taxonomy. HQC produces a large number of singleton clusters. This may be an indication that the space of asteroid spectral types is very rich, and that the size of the present data set is too small to give a good representation of this entire space. Thus, as more data will become available, HQC should be run again to update the taxonomy.

The large number of singletons can also mean that the value of σ that was chosen does not apply uniformly to the entire data set. If larger values of σ were used at different locations in feature space, some of the singletons could fall into clusters.

We have seen that choosing large enough values of σ, such that multiple measurements of the same asteroid all fall into the same cluster, yields clusters which are very large and are not meaningful. This demonstrates that the spectrum of an asteroid can vary considerably. It also shows where HQC ceases to give a good multiscale clustering for the problem. We saw that clusters obtained for large values of σ contain both flat and wavy spectra. The physical understanding of the problem suggests that all wavy spectra and all flat spectra should be merged into two separate clusters. HQC fails to do this for large values of σ since it is based on the L^2 distance between the waveforms. Had we performed feature engineering on the problem, performing clustering not on the raw waveforms but rather on crafted vector representations of the waveforms, with features such as the locations of extrema and the slopes between extrema, we could have obtained a more natural clustering at larger scales. But this approach has the

disadvantage that by choosing the features we use prior assumptions about the spectra. This may obscure structure that is present in the data and which we do not expect. Another approach can be to separate spectra first, by inspection, into two groups of wavy and flat spectra, and to apply HQC to each group separately.

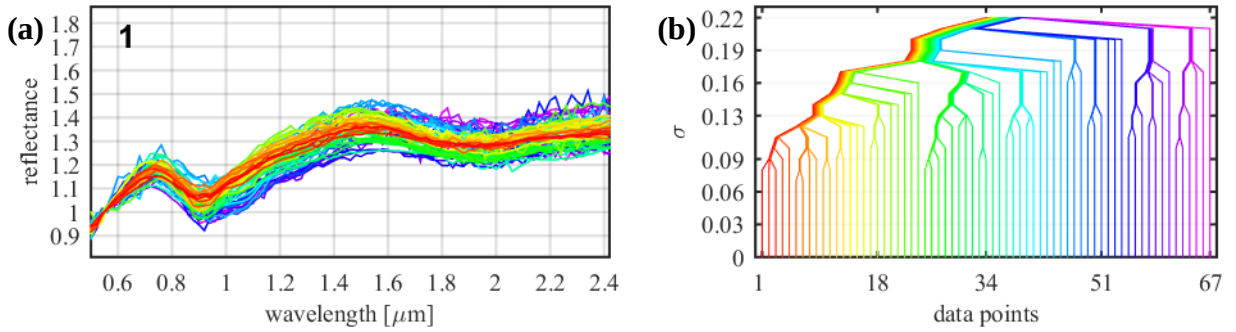
Table 1: The clusters associated with each asteroid by HQC, $\sigma = 0.22$. Singleton clusters don't have a designation. Some asteroids have multiple clusters associated with different measurements.

Designation	Name	Clusters	Designation	Name	Clusters	Designation	Name	Clusters
1	Ceres	**6**	36	Atalante	**8**	79	Eurynome	**1**
2	Pallas		37	Fides	**1**	81	Terpsichore	**3**
3	Juno	**1**	38	Leda	**8**	82	Alkmene	**11**
4	Vesta		39	Laetitia	**1**	87	Sylvia	**5**
5	Astraea	**1**	40	Harmonia	**1**	90	Antiope	**26**
6	Hebe	**1**	41	Daphne	**3,4**	96	Aegle	**5**
7	Iris	**1**	43	Ariadne	**1**	97	Klotho	**4**
10	Hygiea	**3,6**	44	Nysa	**6**	99	Dike	**4**
11	Parthenope	**14**	45	Eugenia	**3**	101	Helena	**1**
13	Egeria	**3**	47	Aglaja	**3**	103	Hera	**10**
14	Irene	**1**	48	Doris	**6**	105	Artemis	**4**
15	Eunomia	**12**	51	Nemausa	**14**	108	Hecuba	**2**
16	Psyche	**5**	52	Europa	**26**	110	Lydia	**5**
17	Thetis	**19**	55	Pandora	**5**	111	Ate	**6**
19	Fortuna	**4,8**	56	Melete	**14**	113	Amalthea	
20	Massalia	**1**	57	Mnemosyne	**1**	114	Kassandra	**5**
21	Lutetia	**4,5**	61	Danae	**1**	118	Peitho	**1**
22	Kalliope	**5**	62	Erato	**6**	119	Althaea	**13**
23	Thalia	**10**	63	Ausonia	**2**	121	Hermione	**8**
24	Themis	**20**	65	Cybele	**8**	131	Vala	**8**
25	Phocaea	**2**	66	Maja	**4,6**	132	Aethra	**4**
26	Proserpina	**1**	67	Asia	**1**	141	Lumen	**4**
27	Euterpe	**10**	69	Hesperia	**5**	142	Polana	**20**
29	Amphitrite	**21**	73	Klytia	**19**	145	Adeona	**3**
30	Urania	**2**	76	Freia		151	Abundantia	
32	Pomona	**1**	77	Frigga	**5**	153	Hilda	**5**
34	Circe	**3**	78	Diana	**4**	158	Koronis	**1**

Designation	Name	Clusters	Designation	Name	Clusters	Designation	Name	Clusters
160	Una	**8**	512	Taurinensis	**2**	1494	Savo	**22**
175	Andromache	**3**	532	Herculina	**1**	1508	Kemi	**6**
180	Garumna	**11**	534	Nassovia	**1**	1542	Schalen	
181	Eucharis	**4**	570	Kythera		1565	Lemaitre	
191	Kolga	**8**	584	Semiramis	**13**	1620	Geographos	**1,2**
192	Nausikaa	**13**	596	Scheila		1627	Ivar	**2,10,22**
201	Penelope	**14**	600	Musa	**1**	1640	Nemo	**1**
208	Lacrimosa	**1**	631	Philippina	**10**	1660	Wood	**1**
210	Isabella	**3**	679	Pax	**21**	1685	Toro	**1,9,12**
234	Barbara		699	Hela	**7**	1768	Appenzella	**3**
236	Honoria		706	Hirundo	**8**	1862	Apollo	**16**
237	Coelestina	**1**	719	Albert	**1**	1864	Daedalus	**9**
244	Sita	**13**	773	Irmintraud	**24**	1865	Cerberus	**7**
245	Vera	**1**	776	Berbericia	**4**	1916	Boreas	**15**
246	Asporina		785	Zwetana	**8**	1917	Cuyo	
250	Bettina	**5**	793	Arizona	**17**	1943	Anteros	**1,2**
261	Prymno	**14**	808	Merxia	**7**	1951	Lick	
264	Libussa	**1**	845	Naema	**3**	1980	Tezcatlipoca	**15**
266	Aline	**3**	863	Benkoela		2062	Aten	
269	Justitia		908	Buda		2063	Bacchus	**9**
279	Thule		925	Alphonsina	**1**	2064	Thomsen	**1**
288	Glauke	**1**	944	Hidalgo		2074	Shoemaker	
301	Bavaria	**3**	984	Gretia		2078	Nanking	**11**
308	Polyxo	**24**	1021	Flammario	**6**	2107	Ilmari	**2**
339	Dorothea		1036	Ganymed	**1,2,7**	2246	Bowell	
345	Tercidina	**3**	1065	Amundsenia	**10**	2335	James	**2**
354	Eleonora		1076	Viola	**6**	2340	Hathor	**9**
371	Bohemia	**1**	1131	Porzia	**1**	2850	Mozhaiskij	**18**
377	Campania	**6**	1139	Atami	**2**	2956	Yeomans	**1**
389	Industria	**7**	1143	Odysseus		3102	Krok	**1**
403	Cyane	**10**	1300	Marcelle	**4**	3103	Eger	**21**
416	Vaticana	**18**	1374	Isora	**12**	3122	Florence	
433	Eros	**2,13,15**	1406	Komppa		3199	Nefertiti	

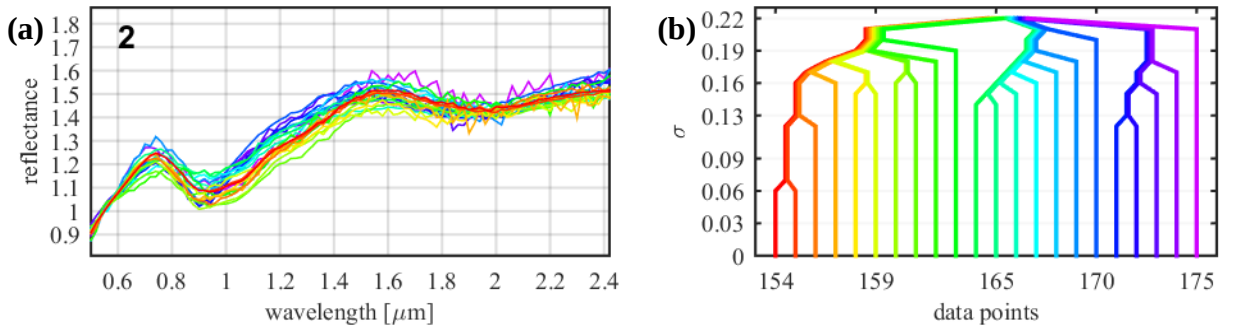
Designation	Name	Clusters	Designation	Name	Clusters	Designation	Name	Clusters
3200	Phaethon		5604			22771		**19**
3248	Farinella		5626		**1**	24445		**1**
3288	Seleucus	**2**	5641	McCleese	**25**	24475		**18**
3317	Paris		5646			25330		
3352	McAuliffe	**2**	5660		**16**	32906		**7**
3402	Wisdom	**10**	5817	Robertfrazer	**1**	35107		**1,12**
3552	Don Quixote		5836		**1,17**	36017		
3628	Boznemcova		6239	Minos	**1,12**	36284		
3635	Kreutz	**2**	6249	Jennifer	**4**	37336		**7**
3671	Dionysus	**6**	6411	Tamaga		65679		**20**
3674	Erbisbuhl	**11**	6455		**15**	66146		**9**
3691	Bede	**4**	6585	O'Keefe	**7**	68278		
3753	Cruithne		6611			68548		
3819	Robinson		7304	Namiki		85818		
3833	Calingasta	**3**	7336	Saunders	**23**	85989		**4**
3858	Dorchester		7341		**9,16**	85990		
3873	Roddy	**7**	7358	Oze	**9**	99942	Apophis	**1**
3908	Nyx		7482		**1**	100926		**1**
4055	Magellan		7822			137799		
4142	Dersu-Uzala		8444	Popovich	**1**	138524		**7,11,17**
4179	Toutatis	**1,7,12**	9400		**1**	139622		**12**
4183	Cuno	**23**	11066	Sigurd	**1**	141052		
4197	Morpheus	**11**	11398		**1**	143624		
4558	Janesick	**7**	11500	Tomaiyowit	**1**	152931		
4688			14402		**3**	163000		
4744	Rovereto		15745	Yuliya	**1**	175706		**3,6**
4954	Eric	**2**	16834		**1**	194268		
5131			16960			194386		**11**
5159	Burbine	**1**	17274			219071		
5261	Eureka		19127	Olegefremov		283460		**1**
5379	Abehiroshi		197127			337866		**1**
5392	Parker		20786		**14**	385186		
5587		**11**	20790		**17**	422686		

Designation	Name	Clusters	Designation	Name	Clusters	Designation	Name	Clusters
1997-AE12			2000-co101		**5**	2002-AL14		
2000-CK33			2001-VS78		**1**			
2000-GK137			2002-AA		**6**			



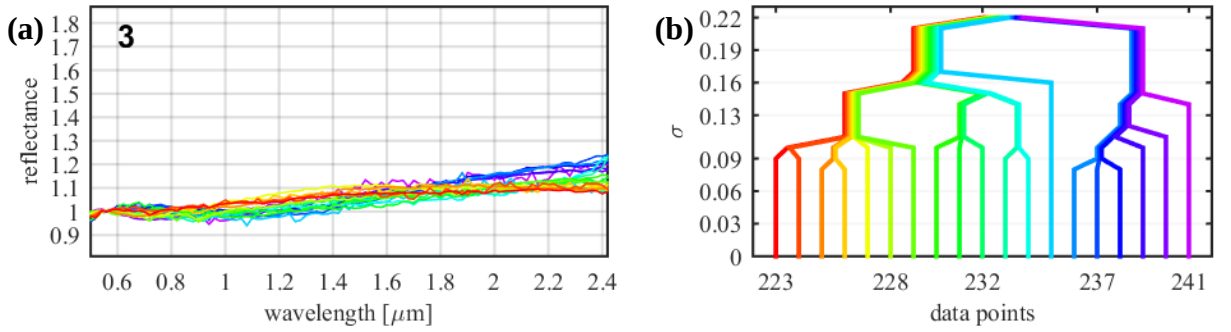
(c) **Cluster size**: 67
Members: 3, 5, 6(1), 6(2), 7, 14, 20, 26, 32, 37, 39, 40, 43, 57, 61, 67(1), 67(2), 79, 101, 118, 158, 208, 237, 245, 264, 288, 371, 532, 534, 600, 719, 925, 1036(2), 1036(3), 1036(5), 1036(7), 1131, 1620(1), 1620(2), 1640, 1660, 1685(3), 1943(2), 2064, 2956, 3102, 4179(2), 5159, 5626, 5817, 5836(2), 6239(1), 7482, 8444, 9400(2), 11066, 11398, 11500, 15745, 16834(2), 24445, 35107(1), 99942, 100926, 283460, 337866, 2001-VS78

Figure 19: (a) The spectra of cluster 1, colored by the hierarchical tree. (b) The hierarchical tree of the cluster. (c) The members of the cluster



(c) **Cluster size**: 22
Members: 25, 30, 63, 108, 433(1), 433(7), 512, 1036(4), 1139, 1620(3), 1627(4), 1627(7), 1943(1), 2107, 2335, 3288(1), 3288(2), 3288(3), 3352(2), 3635, 4954(1), 4954(2)

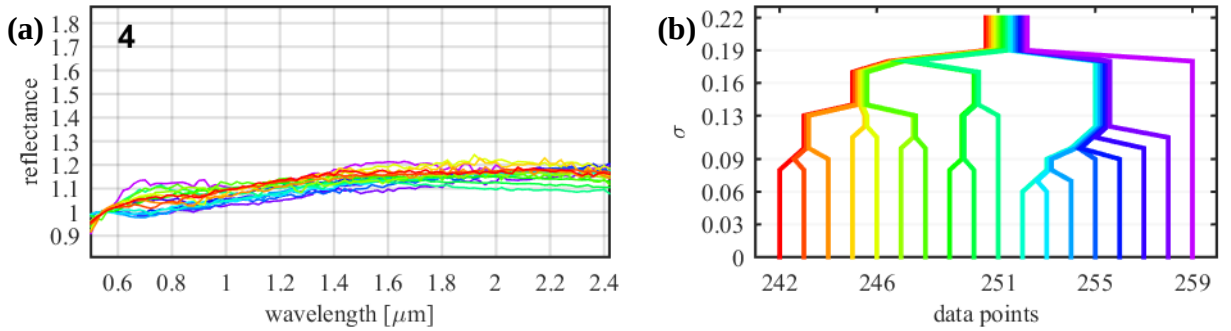
Figure 20: (a) The spectra of cluster 2, colored by the hierarchical tree. (b) The hierarchical tree of the cluster. (c) The members of the cluster



(c)

Cluster size: 19
Members: 10(2), 13, 34, 41(2), 45, 47, 81, 145, 175, 210, 266, 301, 345, 845, 1768, 3833, 14402, 175706(1), 175706(2)

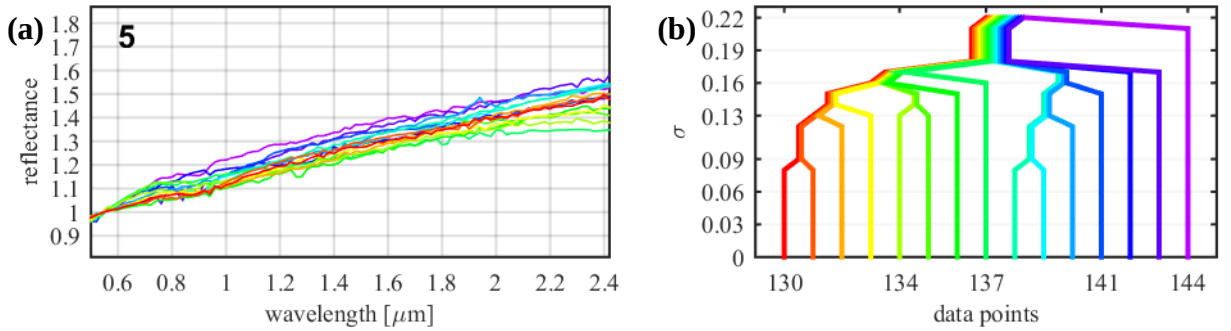
Figure 21: (a) The spectra of cluster 3, colored by the hierarchical tree. (b) The hierarchical tree of the cluster. (c) The members of the cluster



(c)

Cluster size: 18
Members: 19(1), 21(1), 21(3), 41(1), 66(2), 78, 97, 99, 105, 132, 141, 181, 776, 1300(1), 1300(2), 3691, 6249, 85989

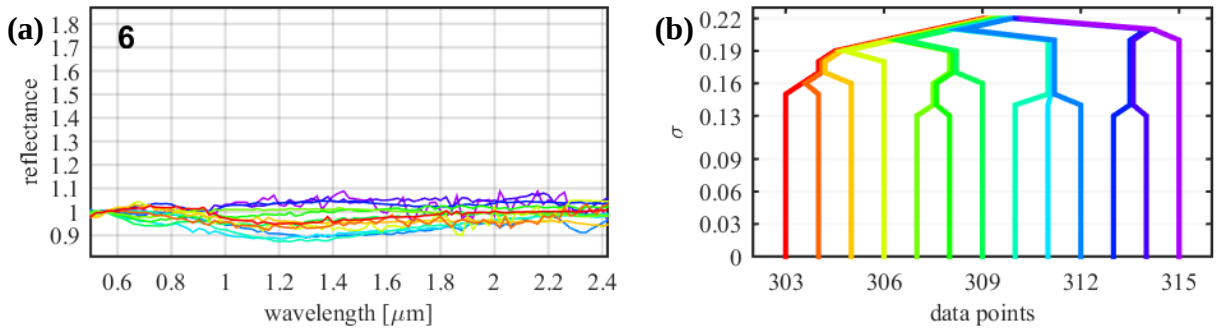
Figure 22: (a) The spectra of cluster 4, colored by the hierarchical tree. (b) The hierarchical tree of the cluster. (c) The members of the cluster



(c)

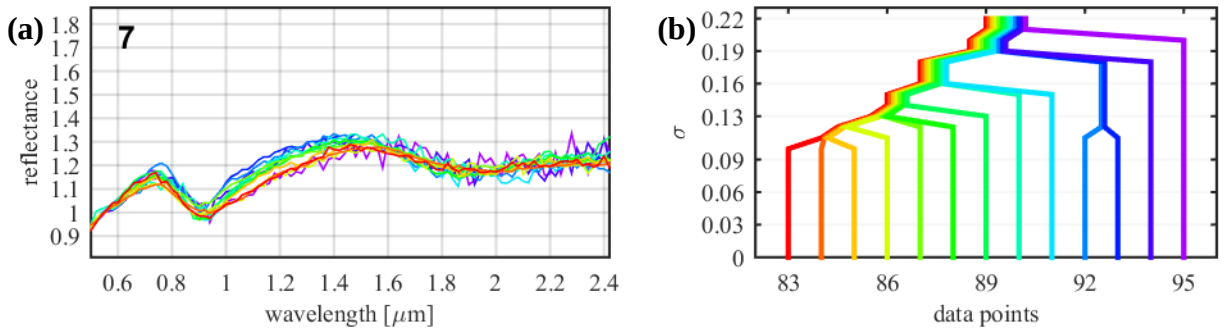
Cluster size: 15
Members: 16, 21(2), 22, 55(1), 55(2), 69(1), 69(2), 77, 87, 96, 110, 114, 153, 250, 2000-co101

Figure 23: (a) The spectra of cluster 5, colored by the hierarchical tree. (b) The hierarchical tree of the cluster. (c) The members of the cluster



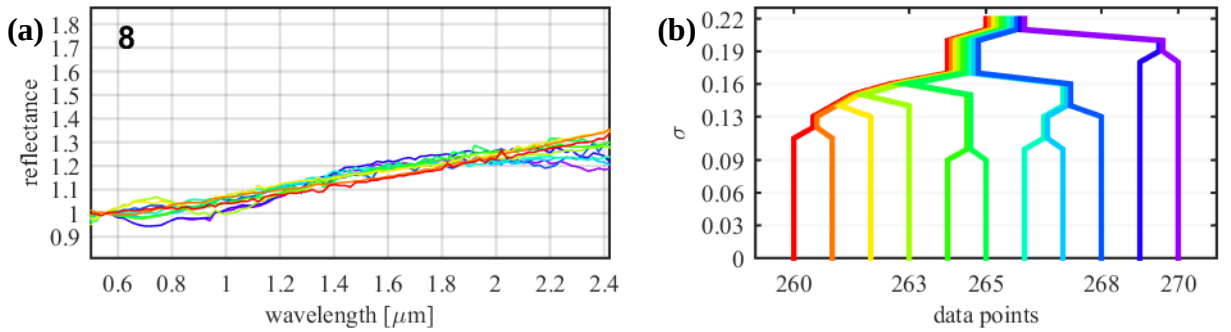
(c) **Cluster size**: 13
Members: 1, 10(1), 44, 48, 62, 66(1), 111, 377, 1021, 1076, 1508, 3671, 175706(4)

Figure 24: (a) The spectra of cluster 6, colored by the hierarchical tree. (b) The hierarchical tree of the cluster. (c) The members of the cluster



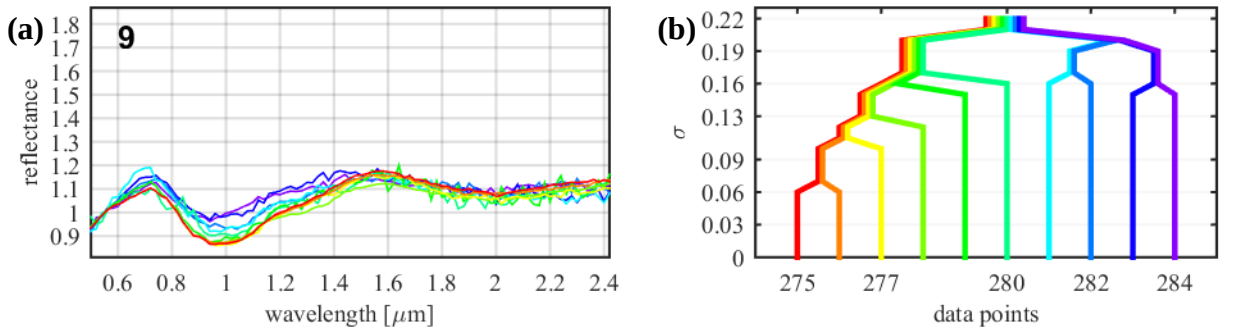
(c) **Cluster size**: 13
Members: 389, 699, 808, 1036(1), 1036(6), 1865, 3873, 4179(1), 4558, 6585, 32906, 37336, 138524(2)

Figure 25: (a) The spectra of cluster 7, colored by the hierarchical tree. (b) The hierarchical tree of the cluster. (c) The members of the cluster



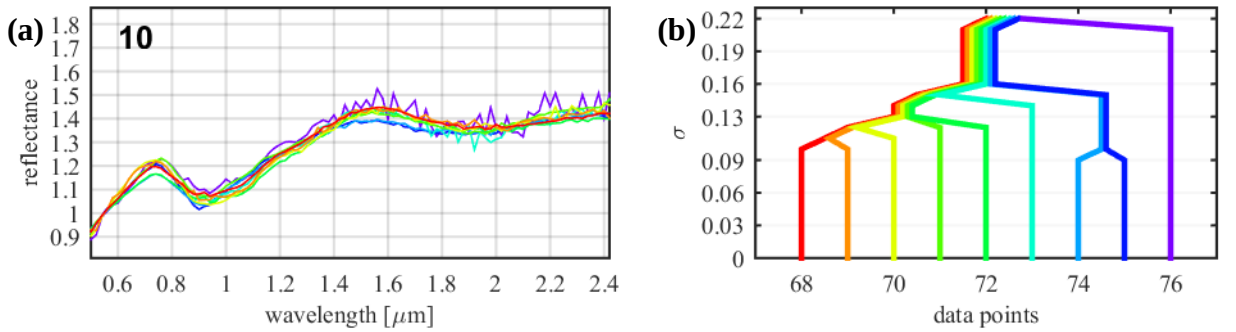
(c) **Cluster size**: 11
Members: 19(2), 36, 38, 65, 121, 131, 160, 191, 706(1), 706(2), 785

Figure 26: (a) The spectra of cluster 8, colored by the hierarchical tree. (b) The hierarchical tree of the cluster. (c) The members of the cluster



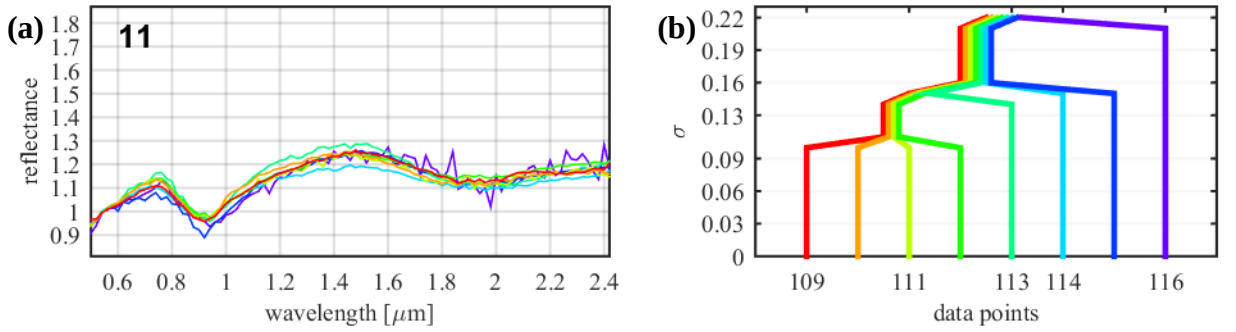
(c) **Cluster size**: 10
Members: 1685(1), 1864, 2063, 2340, 7341(2), 7358, 66146(1), 66146(2), 66146(3), 66146(4)

Figure 27: (a) The spectra of cluster 9, colored by the hierarchical tree. (b) The hierarchical tree of the cluster. (c) The members of the cluster



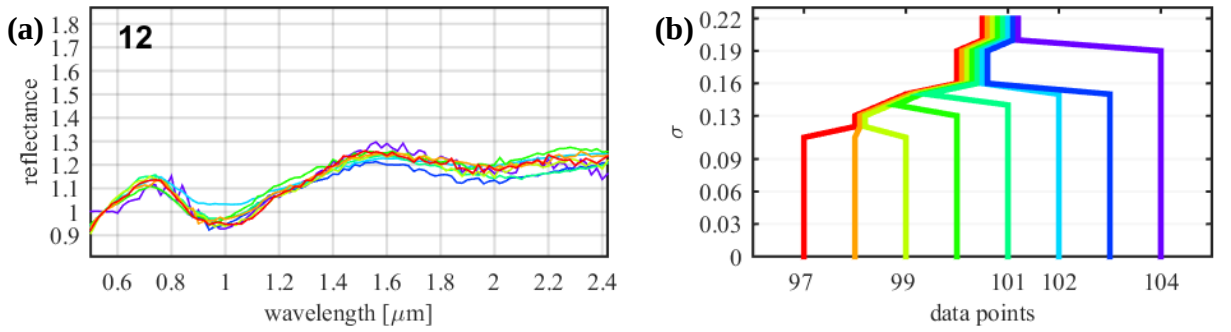
(c) **Cluster size**: 9
Members: 23, 27, 103, 403, 631, 1065, 1627(1), 1627(5), 3402

Figure 28: (a) The spectra of cluster 10, colored by the hierarchical tree. (b) The hierarchical tree of the cluster. (c) The members of the cluster



(c) **Cluster size**: 8
Members: 82, 180, 2078, 3674, 4197, 5587, 138524(3), 194386

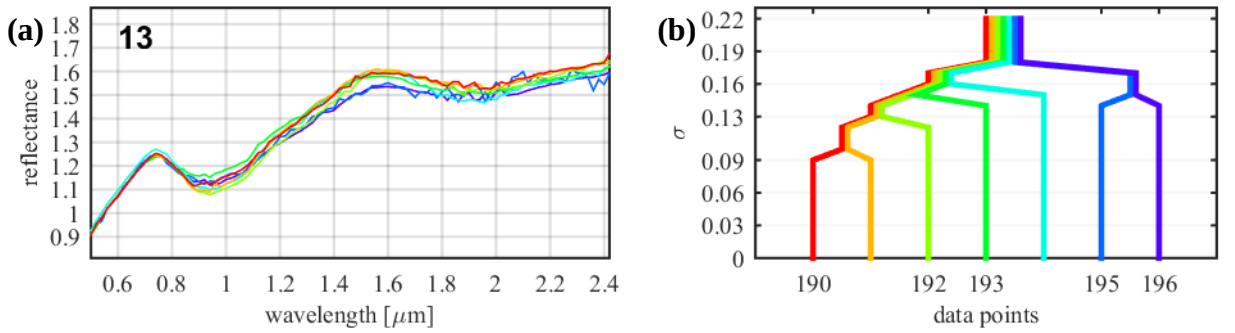
Figure 29: (a) The spectra of cluster 11, colored by the hierarchical tree. (b) The hierarchical tree of the cluster. (c) The members of the cluster



(c)

Cluster size: 8
Members: 15, 1374, 1685(2), 4179(3), 4179(4), 6239(2), 35107(2), 139622

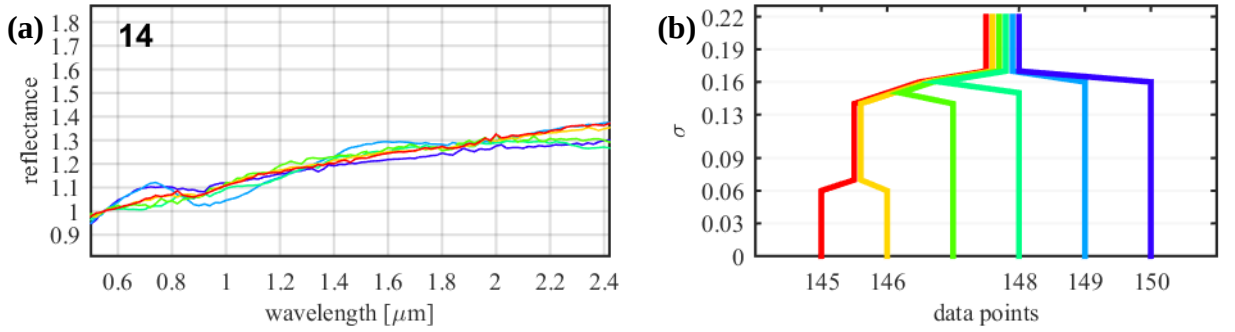
Figure 30: (a) The spectra of cluster 12, colored by the hierarchical tree. (b) The hierarchical tree of the cluster. (c) The members of the cluster



(c)

Cluster size: 7
Members: 119, 192, 244(1), 433(3), 433(6), 433(8), 584

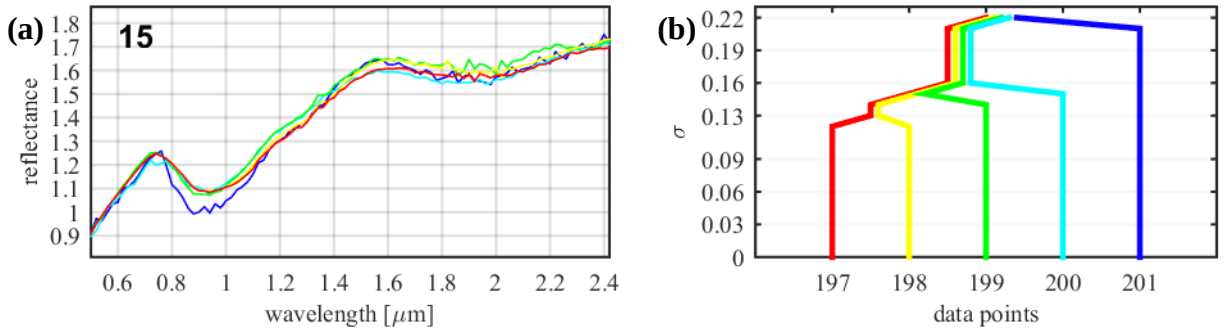
Figure 31: (a) The spectra of cluster 13, colored by the hierarchical tree. (b) The hierarchical tree of the cluster. (c) The members of the cluster



(c)

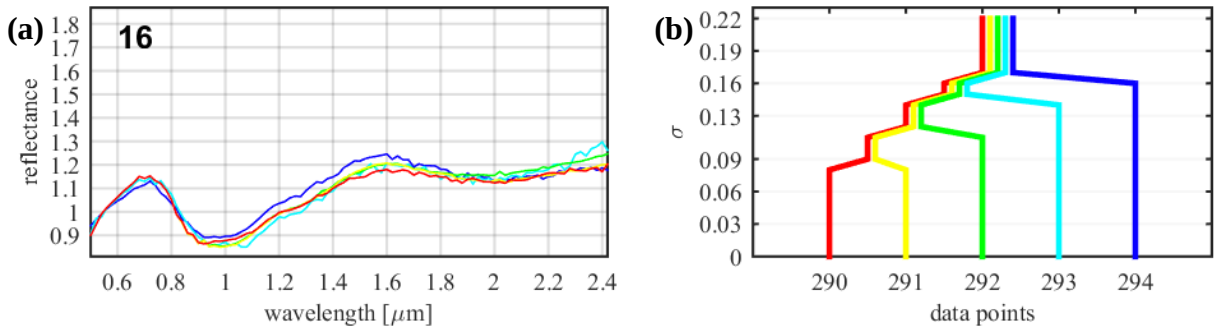
Cluster size: 6
Members: 11, 51, 56, 201, 261, 20786

Figure 32: (a) The spectra of cluster 14, colored by the hierarchical tree. (b) The hierarchical tree of the cluster. (c) The members of the cluster



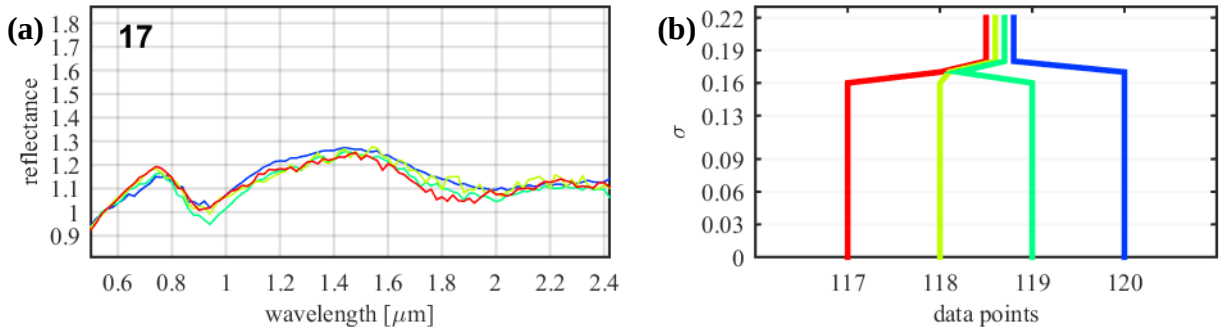
(c) **Cluster size**: 5
Members: 433(2), 433(4), 1916, 1980, 6455(3)

Figure 33: (a) The spectra of cluster 15, colored by the hierarchical tree. (b) The hierarchical tree of the cluster. (c) The members of the cluster



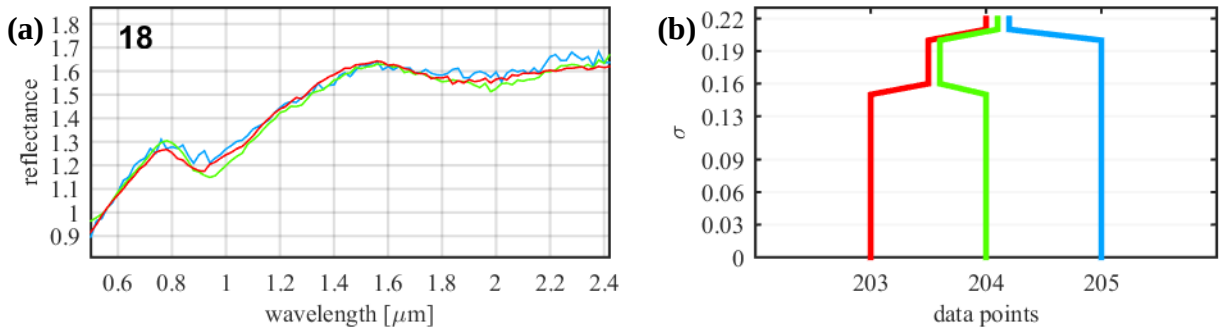
(c) **Cluster size**: 5
Members: 1862(1), 1862(2), 1862(3), 5660, 7341(1)

Figure 34: (a) The spectra of cluster 16, colored by the hierarchical tree. (b) The hierarchical tree of the cluster. (c) The members of the cluster



(c) **Cluster size**: 4
Members: 793, 5836(1), 20790, 138524(1)

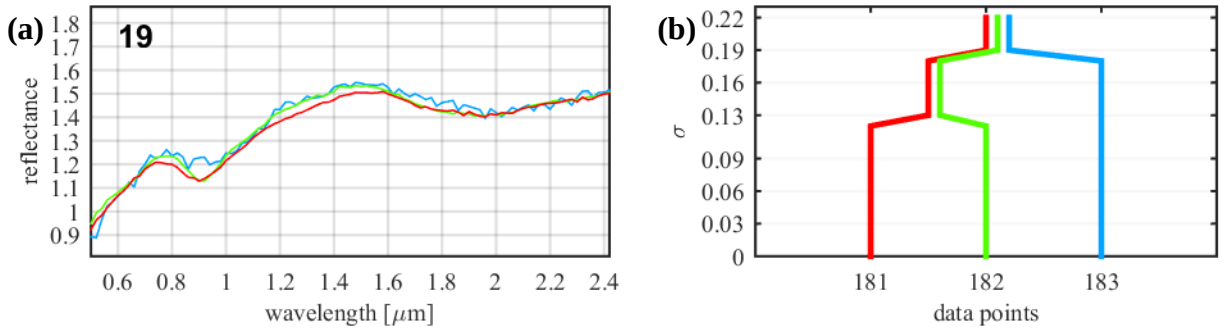
Figure 35: (a) The spectra of cluster 17, colored by the hierarchical tree. (b) The hierarchical tree of the cluster. (c) The members of the cluster



(c)

Cluster size: 3
Members: 416, 2850, 24475(2)

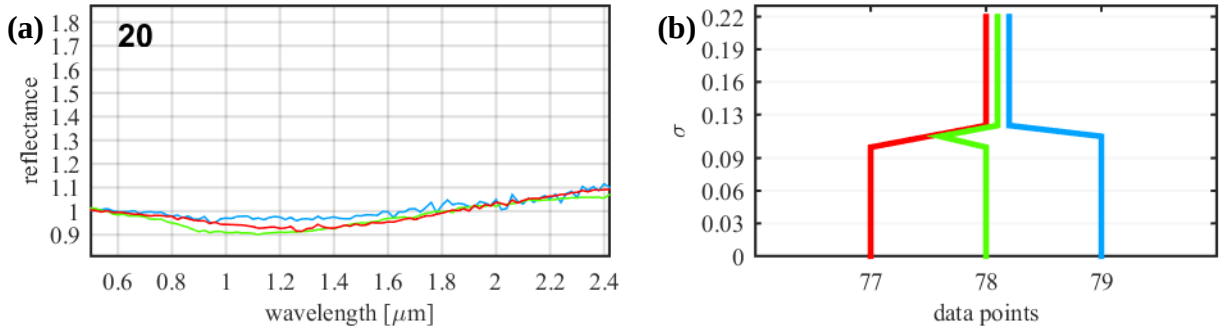
Figure 36: (a) The spectra of cluster 18, colored by the hierarchical tree. (b) The hierarchical tree of the cluster. (c) The members of the cluster



(c)

Cluster size: 3
Members: 17, 73, 22771

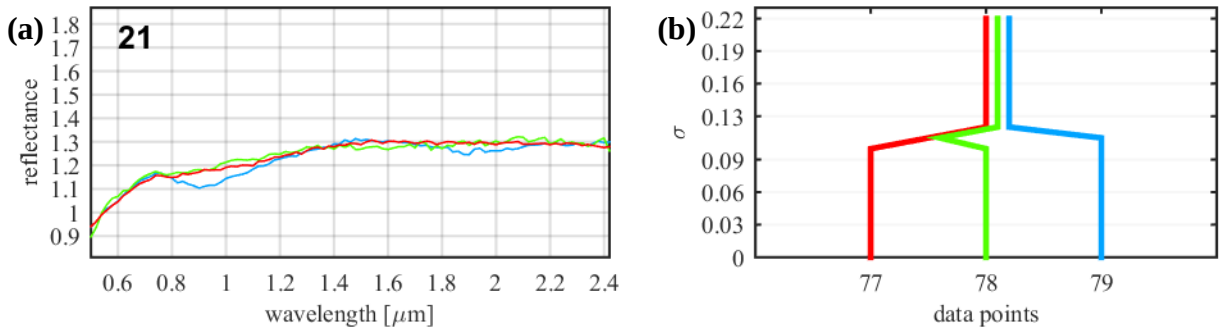
Figure 37: (a) The spectra of cluster 19, colored by the hierarchical tree. (b) The hierarchical tree of the cluster. (c) The members of the cluster



(c)

Cluster size: 3
Members: 24, 142, 65679

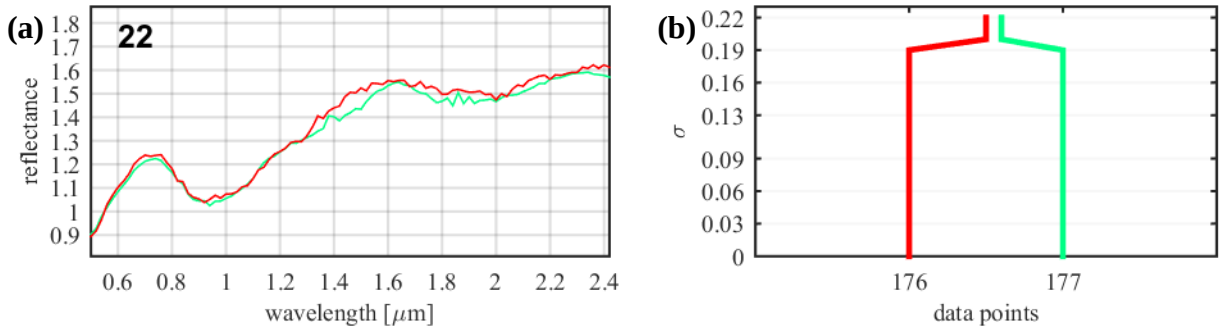
Figure 38: (a) The spectra of cluster 20, colored by the hierarchical tree. (b) The hierarchical tree of the cluster. (c) The members of the cluster



(c)

Cluster size: 3
Members: 29, 679, 3103

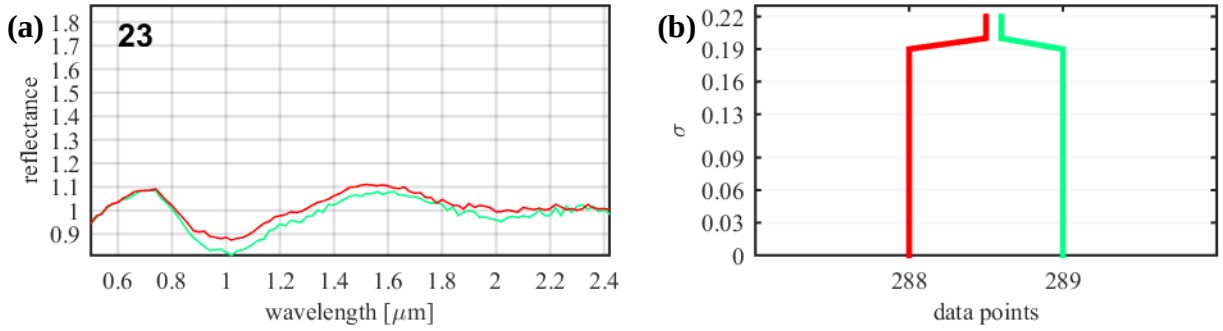
Figure 39: (a) The spectra of cluster 21, colored by the hierarchical tree. (b) The hierarchical tree of the cluster. (c) The members of the cluster



(c)

Cluster size: 2
Members: 1494, 1627(3)

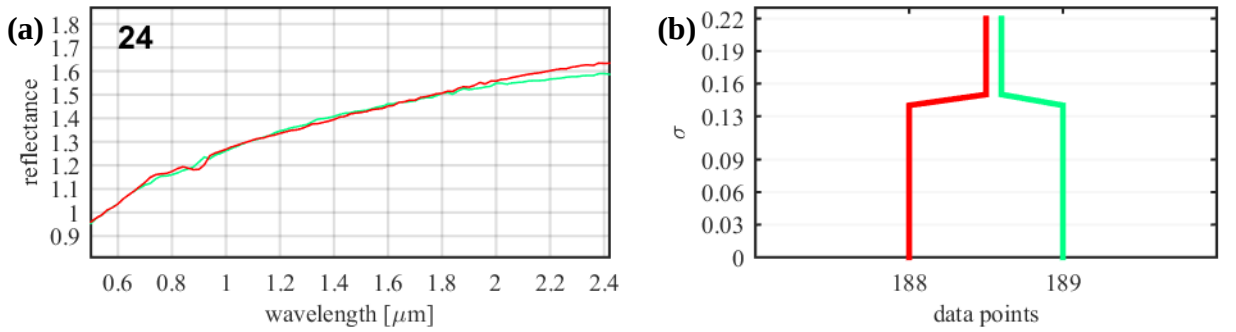
Figure 40: (a) The spectra of cluster 22, colored by the hierarchical tree. (b) The hierarchical tree of the cluster. (c) The members of the cluster



(c)

Cluster size: 2
Members: 4183, 7336

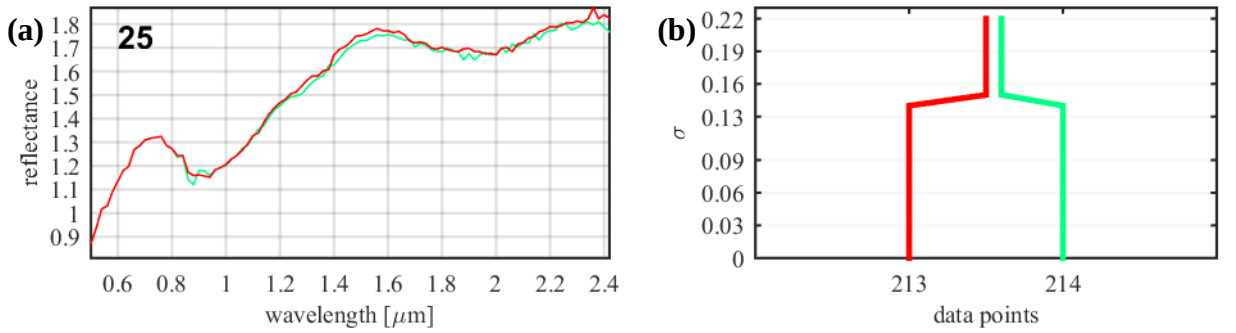
Figure 41: (a) The spectra of cluster 23, colored by the hierarchical tree. (b) The hierarchical tree of the cluster. (c) The members of the cluster



(c)

Cluster size: 2
Members: 308, 773

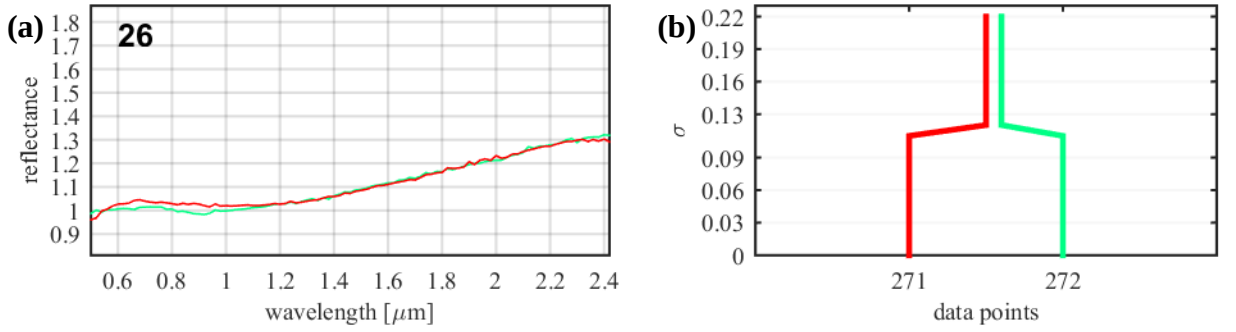
Figure 42: (a) The spectra of cluster 24, colored by the hierarchical tree. (b) The hierarchical tree of the cluster. (c) The members of the cluster



(c)

Cluster size: 2
Members: 5641(2), 5641(3)

Figure 43: (a) The spectra of cluster 25, colored by the hierarchical tree. (b) The hierarchical tree of the cluster. (c) The members of the cluster



(c)

Cluster size: 2
Members: 52, 90

Figure 44: (a) The spectra of cluster 26, colored by the hierarchical tree. (b) The hierarchical tree of the cluster. (c) The members of the cluster

V. References

[1] Grira, Nizar, Michel Crucianu, and Nozha Boujemaa. “Acive semi-supervised fuzzy clustering for image database categorization”. *Proceeding of the 7th ACM SIGMM international retrieval*. ACM, 2005.

[2] Friedman, Jerome, Trevor Hastie, and Robert Tibshirani. "The Elements of Statistical Learning: Data Mining, Inference, and Prediction." *Springer Series in Statistics (* (2009).

[3] Guha, Sudipto, Rajeev Rastogi, and Kyuseok Shim. "CURE: an efficient clustering algorithm for large databases." *ACM Sigmod Record*. Vol. 27. No. 2. ACM, 1998.

[4] Karypis, George, Eui-Hong Han, and Vipin Kumar. "Chameleon: Hierarchical clustering using dynamic modeling." *Computer* 32.8 (1999): 68-75.

[5] MacQueen, James. "Some methods for classification and analysis of multivariate observations." *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*. Vol. 1. No. 14. 1967.

[6] Kaufman, Leonard, and Peter Rousseeuw. *Clustering by means of medoids*. North-Holland, 1987.

[7] Bezdek, James C., Robert Ehrlich, and William Full. "FCM: The fuzzy c-means clustering algorithm." *Computers & Geosciences* 10.2-3 (1984): 191-203.

[8] Roberts, Stephen J. "Parametric and non-parametric unsupervised cluster analysis." *Pattern Recognition* 30.2 (1997): 261-272.

[9] Cheng, Yizong. "Mean shift, mode seeking, and clustering." *IEEE transactions on pattern analysis and machine intelligence* 17.8 (1995): 790-799.

[10] Ester, Martin, et al. "A density-based algorithm for discovering clusters in large spatial databases with noise." *Kdd*. Vol. 96. No. 34. 1996.

[11] Ankerst, Mihael, et al. "OPTICS: ordering points to identify the clustering structure." *ACM Sigmod record*. Vol. 28. No. 2. ACM, 1999.

[12] Ben-Hur, Asa, et al. "Support vector clustering." *Journal of machine learning research* 2.Dec (2001): 125-137.

[13] Vesanto, Juha, and Esa Alhoniemi. "Clustering of the self-organizing map." *IEEE Transactions on neural networks* 11.3 (2000): 586-600.

[14] Rose, Kenneth, Eitan Gurewitz, and Geoffrey C. Fox. "Statistical mechanics and phase transitions in clustering." *Physical review letters* 65.8 (1990): 945.

[15] Horn, David, and Assaf Gottlieb. "Algorithm for data clustering in pattern recognition problems based on quantum mechanics." *Physical review letters* 88.1 (2001): 018702.

[16] Parzen, Emanuel. "On estimation of a probability density function and mode." *The annals of mathematical statistics* 33.3 (1962): 1065-1076.

[17] Varshavsky, Roy, David Horn, and Michal Linial. "Recursive Top-Down Quantum Clustering of Biological Data

[18] Bonnans, Joseph-Frédéric, et al. *Numerical optimization: theoretical and practical aspects*. Springer Science & Business Media, 2006.

[19] Scott, Tony C., Madhusudan Therani, and Xing M. Wang. "Data Clustering with Quantum

Mechanics." *Mathematics* 5.1 (2017): 5.

[20] Pearson, K. "On lines and planes of closest fit to systems of point in space." *Philosophical Magazine* 2.11 (1901): 559-572.

[21] Coifman, Ronald R., and Stéphane Lafon. "Diffusion maps." *Applied and computational harmonic analysis* 21.1 (2006): 5-30.

[22] Shaked, Guy. “Quantum clustering of large data sets.” *M.Sc. thesis, Tel Aviv University* (2013)

[23] Bentley, Jon Louis. "Multidimensional binary search trees used for associative searching." *Communications of the ACM* 18.9 (1975): 509-517.

[24] Datar, Mayur, et al. "Locality-sensitive hashing scheme based on p-stable distributions." *Proceedings of the twentieth annual symposium on Computational geometry*. ACM, 2004.

[25] De Pater, Imke, and Jack J. Lissauer. *Planetary sciences*. Cambridge University Press, 2015.

[26] Chapman, Clark R., David Morrison, and Ben Zellner. "Surface properties of asteroids: A synthesis of polarimetry, radiometry, and spectrophotometry." *Icarus* 25.1 (1975): 104-130.

[27] Tholen, David James. "ASTEROID TAXONOMY FROM CLUSTER ANALYSIS OF PHOTOMETRY." (1984).

[28] DeMeo, Francesca E., et al. "An extension of the Bus asteroid taxonomy into the near-infrared." *Icarus* 202.1 (2009): 160-180.

[29] Bus, Schelte J., and Richard P. Binzel. "Phase II of the small main-belt asteroid spectroscopic survey: A feature-based taxonomy." *Icarus* 158.1 (2002): 146-177.

[30] Bus, S. J., F. Vilas, and M. A. Barucci. "Visible-Wavelength Spectroscopy of Asteroids." Asteroids III, WF Bottke Jr., A. Cellino, P. Paolicchi, and RP Binzel (eds), University of Arizona Press, Tucson, p. 169-182 (2002): 169-182.

[31] Rayner, J. T., et al. "SpeX: A medium-resolution 0.8-5.5 micron spectrograph and imager for the NASA Infrared Telescope Facility." *Publications of the Astronomical Society of the Pacific* 115.805 (2003): 362.

[32] http://smass.mit.edu/, accessed May 23, 2016.

[33] https://github.com/sliorde/quantum-clustering-masters-thesis

[34] Xu, Shui. *CCD photometry and spectroscopy and small main-belt asteroids*. Diss. Massachusetts Institute of Technology, 1994.

[35] Xu, Shui, et al. "Small main-belt asteroid spectroscopic survey: Initial results." *Icarus* 115.1 (1995): 1-35.

[36] Bus, Schelte John. "Compositional structure in the asteroid belt: Results of a spectroscopic survey." (1999).

[37] Bus, Schelte J., and Richard P. Binzel. "Phase II of the small main-belt asteroid spectroscopic survey: The observations." *Icarus* 158.1 (2002): 106-145.

[38] Burbine, Thomas Hewey. *Forging asteroid-meteorite relationships through reflectance spectroscopy*. Diss. Massachusetts Institute of Technology, 2000.

[39] Xu, S., et al. "Small main-belt asteroid spectroscopic survey." *Bulletin of the American*

Astronomical Society. Vol. 25. 1993.

[40] Binzel, Richard P., et al. "Spectral properties of near-Earth objects: Palomar and IRTF results for 48 objects including spacecraft targets (9969) Braille and (10302) 1989 ML." *Icarus* 151.2 (2001): 139-149.

[41] Binzel, Richard P., et al. "MUSES-C target asteroid (25143) 1998 SF36: a reddened ordinary chondrite." *Meteoritics & Planetary Science* 36.8 (2001): 1167-1172.

[42] Binzel, Richard P., et al. "Spectral observations for near-Earth objects including potential target 4660 Nereus: Results from Meudon remote observations at the NASA Infrared Telescope Facility (IRTF)." *Planetary and Space Science* 52.4 (2004): 291-296.

[43] Binzel, Richard P., et al. "Dynamical and compositional assessment of near-Earth object mission targets." *Meteoritics & Planetary Science* 39.3 (2004): 351-366.

[44] Binzel, Richard P., et al. "Observed spectral properties of near-Earth objects: results for population distribution, source regions, and space weathering processes." *Icarus* 170.2 (2004): 259-294.

[45] Rivkin, A. S., et al. "Infrared spectroscopic observations of 69230 Hermes (1937 UB): Possible unweathered endmember among ordinary chondrite analogs." *Icarus* 172.2 (2004): 408-414.

[46] Jaccard, Paul. Distribution de la Flore Alpine: dans le Bassin des dranses et dans quelques régions voisines. Rouge, 1901.

[47] Bohm, David. "A suggested interpretation of the quantum theory in terms of" hidden" variables. I." *Physical Review* 85.2 (1952): 166.

[48] DeMeo, F. E., and Benoit Carry. "The taxonomic distribution of asteroids from multi-filter all-sky photometric surveys." *Icarus* 226.1 (2013): 723-741.