A Practical Guide to Sparse Clustering for Studying Molecular Development of the Human Brain

Justin L. Balsor[1] Keon Arbabi[1] Dezi Ahuja[2] Ewalina Jeyanesan[1] Kathryn M. Murphy[1,2]

[1] McMaster Integrative Neuroscience Discovery and Study (MiNDS) Program, McMaster University, Hamilton, ON, L8S 4K1, Canada

[2] Department of Psychology, Neuroscience & Behavior, McMaster University, Hamilton, ON, L8S 4K1, Canada

Correspondence should be addressed to Kathryn M. Murphy; kmurphy@mcmaster.ca

Kathryn Murphy, Department of Psychology, Neuroscience & Behavior, McMaster University, McMaster University, 1280 Main Street West, Hamilton, ON L8S 4K1, Canada.

Authors' contributions: JB Designed research, performed research, analyzed data, wrote/revised the paper; KA designed research, analyzed data, wote/revised the paper; DA performed research, analyzed data, wrote/revised the paper; EJ analyzed data, revised the paper; KM designed research, analyzed data, wrote/revised the paper.

Acknowledgements: We thank Brendan Kumagai for help with some of the coding. The human tissue samples were obtained from the NIH NeuroBioBank. The proteomic data were from previous studies from our lab (Pinto et al., 2010, 2015; Williams et al., 2010; Siu et al., 2015, 2017) and the transcriptomic data were from (Kang et al., 2011) (GSE25219).

Funding Sources: NSERC Grant RGPIN-2015-06215 and RGPIN-2020-06403 awarded to KM, Woodburn Heron OGS awarded to JB and KA, and NSERC CGS-M awarded to EJ. The funder had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Data and Code Availability: The data used to support the findings in this guide and example code are available here: https://osf.io/6vgrf/

Conflict of Interest Statement: The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Abstract

Studying the molecular development of the human brain presents unique challenges for selecting the best data analysis approach. The rare and valuable nature of human postmortem brain samples, especially for studies examining development, means that those studies have small sample sizes (n) but often include measurements (p) for a large number of genes or proteins for every sample. Thus, most of those data sets have a structure that is $p >> n$, which introduces the problem of sparsity. Here we present a guide to analyzing human brain development data by focusing on sparsity-based clustering methods developed for small sample sizes. We test different methods and identify an application of sparse K-means clustering called Robust Sparse K-means Clustering (RSKC) that does a good job revealing clusters of samples that reflect lifespan stages from birth to aging. The algorithm adaptively selects a subset of the genes or proteins that contributes to generating clusters of samples that are spread across the lifespan. This approach addresses a problem in current studies that were unable to identify postnatal clusters. The guide illustrates that careful selection of the clustering method is essential to reveal meaningful aspects of human brain development.

Background

As molecular tools have become integrated with human neuroscience, there has been a renewed focus on mapping human brain development. Many of those studies have binned data into age groups that approximate the lifespan stages described by neuroanatomical and behavioural maturation (Law et al., 2003; Duncan et al., 2010; Pinto et al., 2010; Siu et al., 2015, 2017). However, age-binning assumes that the stages defined by structural anatomy, as well as cognitive and social-emotional development, are a good fit for molecular brain development. Instead of age-binning, other studies are using exploratory approaches such as unsupervised clustering to analyze those high dimensional data sets. A wide choice of clustering algorithms exists, but there is a need for guidance about selecting appropriate ones for partitioning samples into age-related clusters that reflect the molecular development of the human brain.

When cluster analyses have been applied to studying human brain development, those studies have found regional and pre- versus postnatal groupings but have failed to find postnatal clusters that match human development (Colantuoni et al., 2011; Kang et al., 2011; Carlyle et al., 2017; Li et al., 2018). As a result, it has been challenging to link the stages of cognitive, perceptual, or social-emotional development with the maturation of molecular mechanisms in the human brain. High throughput molecular tools have accelerated the amount of information measured from each postmortem sample to generate a long list of gene or protein expression values. Those values represent a vector that precisely describes where each sample exists in the high dimensional space that defines the molecular complexity of human brain development. The challenge is to determine how samples cluster in that space and if those clusters are a proxy for age-related stages of the human lifespan. Although cluster analysis is not new in biology (Eisen et al., 1998; Tamayo et al., 1999; Hastie et al., 2000, 2001), applying it to studying human brain development presents unique problems because of the small sample sizes found in those studies.

The rare and valuable data collected from human postmortem brain samples hold a special place in neuroscience research because they bridge the gap between animal models and human brain development. Here we provide a practical guide to applying cluster analysis for studying the molecular development of the human brain that focuses on overcoming the small

sample size problem. We introduce the idea of sparsity-based clustering, and one approach, in particular, sparse *K*-means clustering, developed to address the problem of data sets with a large number of genes but a small number of samples (Witten and Tibshirani, 2010).

Box 1. Glossary

	Term	Definition
1	*Feature*	A measurable piece of the collected data, such as the expression of an individual gene or protein. The number of features in a data set is often denoted with *p.*
2	*Sparse/Sparsity*	Describes data sets where observations have been spread across many features/dimensions, and grow increasingly distant from one another; common when the number of features *(p)* is approximately equal to, or greater than, the number of observations *(n).*
3	*Dimension reduction*	Methods that aim to find features which adequately characterize the original high-dimensional data in a lower-dimensional space.
4	*Subspace*	A subset of features within the data set.
5	*k*	A parameter that is used to specify a desired number of clusters.
6	*Sparsity-based clustering methods*	Clustering algorithms that use all *p* features in a data set but either subset or reweigh features according to how well they contribute to cluster formation. These methods are further divided into '*Subspace clustering methods*' and '*Sparse k-means clustering methods*'.
7	*Subspace clustering methods*	An extension of traditional clustering that attempts to find clusters in different subspaces of the same data set.
8	*Sparse k-means clustering methods*	An extension of traditional clustering that addresses sparsity by adaptively searching both the local and global features.
9	*Medoid*	The data point (x_i) within a cluster (Y) with the smallest average dissimilarity between all other data points in *Y.*
10	*Unsupervised*	Data-driven and independent of external classifications (e.g. age, sex, experimental condition).
11	*Supervised*	Dependent on external classifications (e.g. age, sex, experimental condition).
13	*Curse of Dimensionality*	As the number of dimensions in a data set grows, the space needed to represent each dimension increases exponentially. This leads to sparsity within the data and makes it difficult to draw meaningful statistical conclusions.

14	*Feature Weight*	A measure that quantifies the extent to which a feature contributes to cluster formation. Large weights denote that a feature is important to forming clusters

Why consider unsupervised clustering to analyze human brain development?

Modern high throughput molecular tools are allowing neuroscientists to measure the expression of thousands of genes or proteins leading to larger data sets and a need for better computational tools to provide new insights into human brain development. Unsupervised clustering is one of the computational approaches that is being used to analyze those large data sets. Clustering partitions samples into distinct clusters based on similar molecular features rather than a predetermined grouping of the cases such as age or sex. Furthermore, this data-driven approach is a powerful exploratory tool that can be used for discovering novel aspects of the molecular development of the human brain.

Selecting the best clustering algorithms for studying brain development is difficult. One of the biggest challenges is that many clustering algorithms depend on data sets with thousands of samples, but studies using human postmortem tissue samples have a small number of cases, especially when examining development. In this paper, we illustrate the problem with a commonly used clustering method (tSNE - t-distributed stochastic neighbour embedding)(Maaten and Hinton, 2008) and then introduce two approaches to sparsity-based clustering, subspace clustering (Agrawal et al., 1998; Aggarwal et al., 1999) and sparse *K*-means clustering designed for small data sets (Witten and Tibshirani, 2010; Kondo et al., 2016). We explore these methods to test which ones identify age-related clusters for the development of the human visual cortex (V1). We show that sparse *K*-means clustering can reveal robust age-related clusters for human V1 development, and the approach scales from studies with 10s of proteins to 1000s of genes.

What challenges are there in clustering small sample sizes?

Currently, transcriptomic, proteomic and other omics data sets of human brain development include measurements of many more molecular features than samples. The combinatorial nature of those data makes it challenging to use traditional statistical comparisons to understand the many molecular changes that happen in the developing brain. Instead, high dimensional analyses using all of the data are needed to classify the biological features that differentiate the human brain across the lifespan. However, even when clustering is used, the complexity of the findings can still be challenging to interpret, and studies may need to group the data into predefined age categories to describe the spatiotemporal dynamics of the developing brain (Li et al., 2018).

In the mathematical notation used for clustering algorithms, the genes or proteins are called features or observations and are represented by p, while the number of samples is represented by n. Most human brain development data sets are either $p \approx n$ *or* $p > n$ and are best described as sparse data sets. Here sparsity refers to there being many more features than samples. When clustering those data, algorithms can *borrow strength* from the large number of features to represent each sample in the high dimensional space. However, if only a subset of the features contributes to partitioning the samples into clusters, then the analyses may run into the *curse of dimensionality* (Bellman, 1983). For brain development, this means that developmentally relevant features become obscured as more and more genes or proteins that do not contribute to developmental changes are included in the data set. The central problem in analyzing these *large p small n* data sets is identifying the molecular features associated with specific lifespan stages from a very large set of candidate genes. Two approaches to focusing on the relevant features include preprocessing the data using dimension reduction methods (e.g. Principal Component Analysis (PCA), tSNE) or sparsity-based clustering algorithms that retain all of the features but subset or reweight them during clustering.

Some of the common approaches to unsupervised cluster analysis in neuroscience, like PCA and tSNE, work well to reduce the number of dimensions and separate data points into clusters in low-dimensional space, especially if there are large differences in features that fall on orthogonal sets of dimensions. For example, tSNE analysis of transcriptomic data identified

separate clusters for cortical and cerebellar development (Kang et al., 2011; Carlyle et al., 2017), and PCA has shown that age can explain a large fraction of the variation in protein expression during cortical development (Pinto et al., 2015; Breen et al., 2018). Those approaches represent linear combinations of genes or proteins and focus on reducing dimensionality by identifying correlated features. Problems arise when the features that differentiate clusters are not orthogonal, then linear methods like PCA breakdown (Chang, 1983). Thus, traditional dimension reduction and clustering methods are prone to pruning off too much information and, thereby, may miss subtle but significant changes in the human brain's molecular development that can impact neurobiological functions. In contrast, sparsity-based clustering methods keep all of the features.

We test two approaches to sparsity-based clustering that aim to uncover more subtle clustering by focusing on only some of the features. First, we examine subspace clustering methods (CLIQUE, PROCLUS)(Agrawal et al., 1998; Aggarwal et al., 1999) that can reveal hidden clusters based on only a subset of the features. Second, we test sparse *K*-means clustering methods (SPARCL, Robust and sparse k-means clustering-RSKC)(Witten and Tibshirani, 2010; Kondo et al., 2016) developed to handle sparsity in smaller data sets by adaptively searching the local, non-orthogonal sets of dimensions as well as the global, orthogonal sets of dimensions.

What are some approaches to sparsity-based clustering?

Because traditional dimension reduction methods may prune off too much information or miss more subtle changes in the human brain's molecular development, we tested a set of sparsity-based clustering algorithms. Here, sparsity refers to the idea that not all 30,000 genes play a role in brain development. So analyzing brain development can be simplified if the clustering algorithm can identify the smaller number of molecular mechanisms that reflect the spatiotemporal dynamics of neurodevelopment.

In this section, we introduce and compare four clustering methods designed to handle data sparsity but it is not an exhaustive review of sparsity-based clustering. High-dimensional clustering is an active area of statistics research with many new algorithms and approaches published yearly (Wierzchoń and Kłopotek, 2018). This paper presents a practical guide to some

of the sparsity-based clustering methods that are useful for studying molecular brain development. The commonly used subspace clustering methods are reviewed by Parsons et al., (Parsons et al., 2004) and more recently by Oktar and Turkan (Oktar and Turkan, 2018). Also, Hastie et al (Hastie et al., 2015) describe approaches to data analysis that exploit sparsity to uncover signals in the data.

The first two methods, CLIQUE (Agrawal et al., 1998) and PROCLUS (Aggarwal et al., 1999), represent a family of subspace clustering algorithms that use projected clustering to discover dense regions or subspaces of correlated points and partition the data in the subspaces. These algorithms were selected because they represented opposite approaches to subspace clustering. CLIQUE is a bottom-up or agglomerative approach that starts with the leaves at the bottom of a hierarchical tree and moves up to a higher dimensionality subspace. In contrast, PROCLUS is divisive starting at the top and splitting the branches into smaller groups as it moves down the tree.

The agglomerative approach of CLIQUE does not strictly partition points into unique clusters and may result in a data point being assigned to more than one cluster. It is also prone to classifying points as outliers and excluding them from the analysis. The divisive clustering of PROCLUS is based on medoids and uses a 3-step top-down approach to projected clustering. The steps involve initializing the number of clusters (k) and the subspace search size (number of dimensions to consider), then iteratively assigning medoids to find the best clusters for the local dimensions, and a final pass to refine the clusters. Typically, PROCLUS has better accuracy than CLIQUE in partitioning points into clusters, but the *a priori* selection of cluster size is not easy and demands an iterative approach to finding clusters. Furthermore, by restricting the subspace search size, some essential features may be omitted from the analysis.

Both CLIQUE and PROCLUS were developed for data sets with many more samples (n), 2-3 orders of magnitude larger than most data sets of human brain development. Although those algorithms are accurate for large data sets with thousands of samples, they are less well suited for discovering clusters in small sample sizes. So we needed to test sparsity-based clustering designed for small data sets and that criteria led us to select two more approaches to sparse hierarchical clustering, SPARCL and RSKC (Kondo, 2016; Witten and Tibshirani, 2018).

SPARCL was developed by Witten & Tibshirani (2010) to adaptively select and reweigh the subset of features during clustering thus eliminating the need for data reduction preprocessing. The algorithm uses a lasso-type penalty to solve the problem of clustering samples that differ on a small number of features. The reweighted variables then become the input to *K*-means hierarchical clustering. The adaptive feature selection of SPARCL focuses on the subset of genes or proteins that underlie differences among clusters, so this process is similar to removing noise from the data. Thus, SPARCL simultaneously clusters the samples and identifies the dominant features thereby making it easier to determine the subset of proteins or genes responsible for partitioning samples into different clusters.

The Witten & Tibshirani sparse clustering algorithm has many strengths for analyzing data sets with $p \approx n$ or $p > n$; however, it can form clusters containing just one observation (Witten and Tibshirani, 2010). A more recent extension of the algorithm, Robust and Sparse *K*-means Clustering (RSKC), addresses small clusters by assuming that outlier data points cause that problem. RSKC uses the same clustering framework as SPARCL, except that it is *'robust'* to outliers (Kondo et al., 2016). RSKC iteratively identifies clusters in the data, then identifies clusters with a small number of data points (e.g., n=1) and flags those data points as potential outliers. The outliers are temporarily removed from the analysis, and clustering proceeds as outlined above for SPARCL. Once all clusters have been identified, the outliers are re-inserted in the high-dimensional space and grouped with the nearest neighbour cluster. Thus, RSKC identifies meaningful clusters in the data and includes all of the data points.

Methods

What databases were used?

Our lab has been studying development of human V1 by quantifying expression of synaptic and other neural proteins using a library of postmortem tissue samples (n=31, age range 21 days - 79 years, male/female=18/13) (Supplementary Material Table 1). In addition, genome-wide exon-level gene expression data for postnatal human V1 that was collected by Kang et al (2011) (n=48, age range 4 months - 82 years, male/female=27/21)(Supplementary Material Table 2) was used to test the scalability of the sparsity-based clustering. The

preprocessed data from Kang et al. (2011) were downloaded from the Gene Expression Omnibus (GSE25219). The exon-summarized expression data for 17,656 probes were extracted, and probe identifiers were matched to genes. If a gene was matched by two or more probes then probe expressions were averaged if highly correlated (Pearson correlation, $r \geq 0.9$).

The clustering methods were tested using three groups of protein or gene data. The first group of protein data was from a series of studies using the Murphy lab samples to examine the development of synaptic and nonsynaptic proteins in human V1 (Murphy et al., 2005; Pinto et al., 2010, 2015; Williams et al., 2010; Siu et al., 2015, 2017). Western blotting was run using each sample (2-5 times) to probe for 23 different proteins (Supplementary Material Table 3). The tissue preparation and Western blotting methods have been described in detail previously (Siu et al., 2017, 2018). The initial clustering tests used a subset of 7 proteins (GluN1, GluN2A, GluN2B, GluA2, $GABA_A\alpha1$, $GABA_A\alpha3$, Synapsin) to explore age-related clustering with tSNE, ProCLUS, Clique, SPARCL and RSKC.

Next, the sparsity-based clustering using RSKC was explored using all 23 proteins to determine how adding more features changed the age-based clustering. Then the reliability of the age-related clustering was explored by running 100 iterations of RSKC with the 23 proteins. A heatmap illustrating the number of times each sample was partitioned into a cluster was made to visualize the reliability.

The scalability of RSKC was tested using a larger protein database and the much larger gene database. These tests included clustering a matrix with 95 proteins collected from the Murphy lab postmortem tissue samples. This time the samples were probed with a high density ELISA array (RayBiotech Quantibody Human Cytokine Array 4000) and an additional 72 proteins were measured for a total of 95 proteins (p=95)(Supplementary Material Table 3). Finally, RSKC clustering was done using the genes in the Kang database by selecting the ones that match the SynGO genes (n=988) (Koopmans et al., 2019) and then clustering using the full set of genes in the database (n=17,237).

Results

What are the basic steps to sparsity-based clustering using R?

Here we describe four sparsity-based high-dimensional clustering approaches (PROCLUS, CLIQUE, SPARCL, RSKC) for analyzing the development of human V1 using 7 or 23 proteins. Then we explore the scalability of the RSKC method using two larger data sets with 95, 988, or 17,237 proteins or genes.

All of the analyses were done in the R programming language using the integrated development environment RStudio (version 1.3.1093). The basic steps in the workflow used to examine each of the clustering methods is illustrated in Figure 1. The text refers to the R packages that were used and R Markdowns are included in the Supplementary Material.

Figure 1 illustrates the basic steps to testing various sparsity-based clustering methods to examine if they produce an age-related progression in the median age of the clusters. The data were prepared in an *nxp* matrix with each sample forming a row and the features, proteins or genes, arranged in columns. Those data were used as the input to the clustering algorithms. Here the sparsity-based algorithms tested were ProClus and Clique from the *subspace* package (Hassani, 2015), SPARCL (Witten and Tibshirani, 2018) and RSKC (Kondo, 2016). For all of the algorithms a range of *k* values from 2 to 9 were tested to explore the types of clusters produced.

The results of a tSNE dimension reduction was used to visualize clusters across all of the methods tested. However, clustering was not done on the tSNE data itself even though that is a commonly used approach. We used tSNE strictly as a visualization tool because it does a good job projecting points from high dimensional space onto 2D so that neighboring points reflect their similarity.

The Elbow method was used to determine the best number of clusters. Finally, the quality of the age-related clustering of the samples was evaluated by making a boxplot to visualize the progression of the median ages.

This workflow was used for all of the clustering methods described in the next section and an example R Markdown of the analysis is included in Supplementary Material.

Evaluating which sparsity-based clustering finds age-related clusters?

We began evaluating clustering methods by exploring if tSNE visualization produced an age-related organization of the samples using the human V1 data set with 7 proteins and all runs as the input to the *tsne* package (Donaldson and Donaldson, 2010) (Fig 2A). Color-coding the samples by their age showed a global progression in the ages with younger samples mapped to the bottom right and older to the top left in the 2D tSNE space. Next, we used a commonly used agglomerative hierarchical clustering algorithm, *agnes* in the *cluster* package (Maechler, 2019), to test if this cluster approach would reveal age-related groupings of the samples. There was some indication that clustering with this method could partition samples into clusters representing an age group of human development (Fig. 2B). For example, 4 and 6 clusters had some age progression, but there were 2 clusters with a very similar median age in each case. These findings show that high dimensional visualization and clustering of human V1 postnatal samples using measurements from just 7 proteins can find age-related progression, but the overlapping ages suggest that standard hierarchical clustering methods might miss subtle aspects of molecular development of human V1.

Next, we tested the two density projection sparsity-based clustering methods that use either top-down (PROCLUS) or bottom-up (CLIQUE) clustering methods with all of the observations (n=31) and 7 of the proteins from the human visual cortex development data set. The outputs were plotted in 2D using tSNE, and the data points were colour-coded according to the clusters identified by each method. Finally, to determine if the clusters represented developmental changes in the data set, we plotted boxplots showing each cluster's median age.

PROCLUS: The PROCLUS clustering method was implemented in RStudio using the *ProClus* function in the *subspace* package version 1.0.4 (Hassani, 2015). We explored clusters between k=2-9, and Figure 3 shows the results for 2, 4, 6, and 8 clusters for the human V1 data with 7 proteins and all runs included.

Visualizing the clusters found with PROCLUS (Fig. 3A) showed a mixing of the samples, but the boxplots illustrating the ages of the samples in the clusters suggested an

age-related progression, especially for 4 or 6 clusters (Fig. 3B). The PROCLUS clusters' age progression was somewhat better than the hierarchical clusters but still had clusters with very similar median ages. More importantly, some clusters had only one or two data points, and many samples were tagged as outliers (small grey dots) and excluded from the clusters. Thus, PROCLUS's iterative top-down feature identification and cluster border adjustments performed poorly for identifying age-related stages of human V1 development.

CLIQUE: The bottom-up clustering method CLIQUE was tested to determine how well this iterative approach to building clusters performed using 7 proteins to group the human V1 samples into age-related clusters.

The CLIQUE function from the *subspace* package (Hassani, 2015) was used to test clustering. CLIQUE requires an input value for the interval setting because the intervals divide each dimension into equal-width bins that are searched for dense regions of data points. Here we tested a range of intervals (xi=2-8) and cluster numbers (k=2-9) (Fig. 3C&D).

CLIQUE allows data points to be in more than one cluster, so to visualize the multi-cluster identities, we plotted the data points using concentric colour-coded rings. CLIQUE placed all of the data into multiple overlapping clusters, which was true for all interval settings (xi=2-8). That poor partitioning of samples resulted in no progression in the clusters' median age (Fig. 3D). Thus the iterative bottom-up clustering of CLIQUE performed poorly for clustering the samples into age-related groups.

Comparing these top-down PROCLUS and bottom-up CLIQUE density methods for sparsity clustering showed that neither algorithm was a good fit for producing age-related clustering of the samples to reveal stages of the lifespan. PROCLUS performed somewhat better because some of the parameters resulted in clusters with a progression in the median cluster age; but, the number of data points treated as outliers was unacceptably high.

SPARCL: Next, we tested a sparsity-based clustering algorithm, sparse K-means clustering, optimized for small sample sizes (Witten and Tibshirani, 2010). The SPARCL package (version 1.0.4) was used (Witten and Tibshirani, 2018) to cluster the human V1 samples using 7 proteins. This approach adaptively finds subsets of variables that capture the different

dimensions and includes all samples in the clusters. SPARCL searches across multiple dimensions in the data and adjusts each variable's weight based on their contribution to the clustering. Thus, the term 'sparse' in this method refers to selecting different subsets of proteins to define each cluster.

To implement sparse *K*-means clustering, we used the *Kmeans.sparsecluster* function in the sparcl package (Witten and Tibshirani, 2018). We explored a range of *k* clusters between k=2-9. The sparcl package also includes a function to help determine other input variables, such as the boundaries for reweighting the variables (*wbounds*) to produce optimal clustering.

Visualizing the clusters created by SPARCL showed useful partitioning of the samples into clusters (Fig. 3E) that moved from the bottom right to the top left in the tSNE plot. Also, the boxplots illustrate a good progression of the median cluster age for 4 and 6 clusters. However, SPARCL is prone to making clusters with only 1 sample, and that was the case in this example for *k*=4-9 clusters. To address that problem, we tested another sparse *K*-means cluster algorithm that is robust to making clusters of n=1.

RSKC: Finally, we tested a modified version of the SPARCL algorithm called Robust and Sparse *K*-means clustering (RSKC)(Kondo et al., 2016). The RSKC algorithm was designed to be robust to the influence of outliers that can drive other algorithms to create clusters of n=1. RSKC operates by iteratively omitting outliers from cluster analysis, assigning all remaining samples to clusters, and then reinserting outliers to the analysis by grouping them into the nearest-neighbouring cluster.

Using the *RSKC* package in R (Kondo, 2016) we explored clustering for a range of *k* values (*k*=2-9) using the human V1 data set with 7 proteins and all runs (Fig. 4). The visualization of the clusters on the tSNE plot showed good grouping of the samples into spatially separated clusters. The boxplots illustrate good progression in the median ages of the clusters, especially for 4 or 6 clusters (Fig. 4B). In addition, this algorithm adaptively reweighted the proteins to identify the most robust clusters and we plotted the weights for each of the 7 proteins (Fig. 4C). This component of RSKC identified the lifespan variations in GluN2B, Synapsin, and GluN2A as having the greatest impact on the clustering of the samples.

To explore the scalability of RSKC we clustered the full data set with 23 proteins for the human V1 samples (Murphy lab)(Fig. 5). In this example, the average expression value for each protein was used. For this analysis we used the elbow plot method (see below) to select the number of clusters and with this data set 6 clusters was the best. Figure 5A shows a tight progression of age-related clustering from cluster A with the youngest median age to cluster F with the oldest age. The addition of more proteins to the RSKC clustering provided greater precision for identifying the subtle changes that represent the temporal dynamics of human V1 development. The weights for the proteins (Fig. 5B) showed that all of the proteins contributed to this high dimensional clustering. Next, we used the weights to improve the visualization of the clusters in a tSNE plot. The protein values were transformed by multiplying them with the weight for each protein and the transformed data were visualized using tSNE (Fig. 5C). This analysis showed distinct spatial separation of the clusters in the 2D tSNE plot.

Since the starting conditions for clustering can affect which clusters samples end up in, we tested how robust RSKC was by running the algorithm 100 times with different starting conditions. We then plotted a boxplot showing the overall age-related clusters and a heatmap showing the number of times each sample fell into the different clusters (Fig. 6). This analysis showed that the progression in the median age of the clusters was robust to the starting conditions. Furthermore, the heatmap showed that clusters B and C were the least stable but the other clusters had strong consistency for which samples were partitioned into those clusters.

Does RSKC scale to large numbers of proteins or genes?

So far, we have shown that RSKC does a good job of partitioning samples into age-related clusters with data sets that have fewer than 25 proteins. Here we examine if RSKC scales to larger data sets with 2 to 3 orders of magnitude more features.

We tested the RSKC clustering using data from the Murphy lab human V1 samples with measurements for 95 proteins (Supplementary Material Table 4)(Fig. 7A). Once again, 100 iterations of RSKC clustering was used to ensure that the clusters were robust to the starting condition. This analysis found strong age-related clustering of the samples showing 6 well-defined clusters that step across the lifespan.

Clustering with RSKC was extended to the Kang et al. (2011) data set (Supplementary Material Table 2) to examine how well it performed on a data set with measurements of thousands of genes. The first clustering example was done using 988 genes that overlap with the SynGO database (Koopmans et al., 2019) to assess clustering defined by synaptic genes. That analysis showed age-related progression of the median age of the clusters (Fig. 7B), although compared with the protein clusters, there was a large jump in the median age between clusters B and C, suggesting that gene and protein expression's maturation may follow different timescales. Finally, using RSKC with all 17,237 genes in the Kang data set resulted in very similar clusters to the SynGO gene clusters. There was a progression of median ages for the clusters, but with this larger set of genes, cluster B spanned the gap from childhood to adults (Fig. 7C). Thus, RSKC cluster analysis of 95 proteins revealed the tightest age-related clusters, but the gene data also resulted in the partitioning of samples into age-related clusters. This finding contrasts with the analyses used by Kang et al. (2011) that did not find postnatal age-related clustering. Thus, the optimization of sparse *K*-means cluster analysis (RSKC) for small sample sizes provides another approach for analyzing the human brain's molecular development that is sensitive to subtle changes that occur across the postnatal lifespan.

A note about how to select the number of clusters.

An essential step in *k*-means clustering is selecting *k*, which denotes the number of groups to classify observation into. The correct choice of k is often ambiguous, as there are many different approaches for making this decision. Intuitively, an optimal k lies in between maximum generalization of the data using a single cluster and maximum accuracy by assigning each observation to its own cluster. One of the most common heuristics for determining k is the elbow plot method, where the sum of squared distances of observations to the nearest cluster center is plotted for various values of k. As k increases, the sum of squared distances tends towards zero. The “elbow” occurs at the point of diminishing returns for minimizing the sum of squared distances, and the k value at this point is selected as the optimal number of clusters (Thorndike. 1953).

To tailor the selection of k to RSKC, we applied the elbow method to the Weighted Within Sum of Squares (WWSS), the objective function maximized by the algorithm. WWSS

was calculated for various values of *k* and averaged over 100 iterations. The elbow can be identified using the elbowPoint function in the *akmedoids* package (version 0.1.5) (Adepeju et al., 2020), which uses a Savitzky-Golay filter to smooth the curve and identify the x where the curvature is maximized. This method found that $k = 6$ was the optimal number of clusters for all of the applications of RSKC used in this paper.

There are more than 30 methods to determine the optimal values for *k* and a large number of journal papers (e.g. (Tibshirani et al., 2001)) and web resources (e.g. Cluster Validation Essentials) that can be used to learn more. The R packages *NbClust* (Charrad et al., 2015) and optCluster (Sekula, 2020) are particularly helpful tools for choosing the number of clusters because they test various methods for selecting *k* (Charrad et al., 2014).

What can you do after clustering the samples?

Clustering the data collected from human postmortem tissue samples to reveal the age-related progression in the molecular complexity of the brain is just the start of using high dimensional analyses. The application of these modern exploratory data-driven approaches are revealing novel aspects of human brain development. Once the appropriate clustering technique is found then there are many other analyses that can be used to interrogate the molecular makeup of the different clusters. An important benefit of clustering with RSKC is that it outputs the feature weights. Those weights reveal the impact of specific proteins or genes on differentiating the molecular environment of the brain during the progression of lifespan stages. Those proteins and genes can be used as the input to GO analysis to catalog the molecular processes, cellular components and biological processes that dominate the stages. They can also be used for differential gene expression analysis to highlight which features are enriched during different stages. The top weighted proteins or genes are also useful for creating a phenotype that characterize higher dimensional changes occuring different stages of the lifespan (Balsor et al., 2020).

Conclusions

The last decade has seen remarkable growth in the number of studies examining the human brain's molecular features. In parallel, high throughput tools have dramatically increased

the amount of data collected for every sample. The complexity and high dimensional nature of those data sets have spurred the need for more guidance in selecting a appropriate tools to analyze those big data. Some studies are now collecting data from 100s or 1000s of human brain postmortem samples (e.g. PsychENCODE), but studies of development still have many fewer samples, and the ages of the samples are spread across the lifespan. The small sample sizes of the developmental data sets make it difficult to apply many commonly used cluster methods. Those methods lack the sensitivity needed to reveal robust clusters defined by the subtle differences in genes or proteins that occur across the postnatal lifespan. At the same time, sparsity-based clustering algorithms designed for small samples size have emerged. In this guide, we explored the application of sparsity-based clustering and showed that one algorithm, Robust Sparse *K*-means Clustering, is a good fit for revealing the subtle and gradual stages of human brain development that occurs from birth to aging. In the next decade, the amount of data collected from each postmortem brain sample will only continue to grow as single-cell RNA sequencing methods are applied to studying human brain development. Neuroscientists will continue to face many challenges in analyzing those sparse and very high dimensional data sets. Nevertheless, careful selection of analysis techniques designed for small sample sizes can be expected to have a meaningful impact on discovering novel aspects of human brain development.

References

Adepeju, M., Langton, S., and Bannister, J. (2020). *akmedoids: Anchored Kmedoids for Longitudinal Data Clustering*. Available at: https://CRAN.R-project.org/package=akmedoids.

Aggarwal, C. C., Wolf, J. L., Yu, P. S., Procopiuc, C., and Park, J. S. (1999). Fast algorithms for projected clustering. *Sigmod Rec* 28, 61–72. doi:10.1145/304181.304188.

Agrawal, R., Gehrke, J., Gunopulos, D., and Raghavan, P. (1998). Automatic subspace clustering of high dimensional data for data mining applications. Sigmod Rec 27, 94–105. doi:10.1145/276304.276314.

Balsor, J. L., Ahuja, D., Jones, D. G., and Murphy, K. M. (2020). A Primer on Constructing Plasticity Phenotypes to Classify Experience-Dependent Development of the Visual Cortex. *Front Cell Neurosci* 14, 245. doi:10.3389/fncel.2020.00245.

Bellman, R. (1983). Mathematical Methods in Medicine. *Math Methods Medicine*, 249–252. doi:10.1142/9789814503013_bmatter.

Breen, M. S., Ozcan, S., Ramsey, J. M., Wang, Z., Ma'ayan, A., Rustogi, N., et al. (2018). Temporal proteomic profiling of postnatal human cortical development. *Transl Psychiat* 8, 267. doi:10.1038/s41398-018-0306-4.

Carlyle, B. C., Kitchen, R. R., Kanyo, J. E., Voss, E. Z., Pletikos, M., Sousa, A. M. M., et al. (2017). A multiregional proteomic survey of the postnatal human brain. *Nat Neurosci* 20, 1–15. doi:10.1038/s41593-017-0011-2.

Chang, W. C. (1983). On Using Principal Components before Separating a Mixture of Two Multivariate Normal Distributions. *Journal of the Royal Statistical Society: Series C (Applied*

Statistics) 32, 267–275. doi:10.2307/2347949.

Charrad, M., Ghazzali, N., Boiteau, V., and Niknafs, A. (2014). NbClust : An R Package for Determining the Relevant Number of Clusters in a Data Set. *J Stat Softw* 61. doi:10.18637/jss.v061.i06.

Charrad, M., Ghazzali, N., Boiteau, V., and Niknafs, A. (2015). *NbClust: Determining the Best Number of Clusters in a Data Set*. Available at: https://CRAN.R-project.org/package=NbClust.

Eisen, M. B., Spellman, P. T., Brown, P. O., and Botstein, D. (1998). Cluster analysis and display of genome-wide expression patterns. *Proc National Acad Sci* 95, 14863–14868. doi:10.1073/pnas.95.25.14863.

Hassani, M. (2015). *Package "subspace" - Interface to OpenSubspace.* Available at: https://CRAN.R-project.org/package=subspace.

Hastie, T., Tibshirani, R., Botstein, D., and Brown, P. (2001). Supervised harvesting of expression trees. *Genome Biol* 2, research0003.1. doi:10.1186/gb-2001-2-1-research0003.

Hastie, T., Tibshirani, R., Eisen, M. B., Alizadeh, A., Levy, R., Staudt, L., et al. (2000). "Gene shaving" as a method for identifying distinct sets of genes with similar expression patterns. Genome Biol 1, research0003.1. doi:10.1186/gb-2000-1-2-research0003.

Hastie, T., Tibshirani, R., and Wainwright, M. (2015). Statistical Learning with Sparsity. 257–284. doi:10.1201/b18401-11.

Kang, H. J., Kawasawa, Y. I., Cheng, F., Zhu, Y., Xu, X., Li, M., et al. (2011). Spatio-temporal transcriptome of the human brain. *Nature* 478, 483–9. doi:10.1038/nature10523.

Kondo, Y. (2016). *RSKC: Robust Sparse K-Means*. Available at: https://CRAN.R-project.org/package=RSKC.

Kondo, Y., Salibian-Barrera, M., and Zamar, R. (2016). RSKC: An RPackage for a Robust and

Sparse K-Means Clustering Algorithm. J Stat Softw 72, 1–26. doi:10.18637/jss.v072.i05.

Koopmans, F., Nierop, P., Andres-Alonso, M., Byrnes, A., Cijsouw, T., Coba, M., et al. (2019). SynGO: An Evidence-Based, Expert-Curated Knowledge Base for the Synapse. *Neuron* 103, 217-234.e4. doi:10.1016/j.neuron.2019.05.002.

Li, M., Santpere, G., Kawasawa, Y. I., Evgrafov, O. V., Gulden, F. O., Pochareddy, S., et al. (2018). Integrative functional genomic analysis of human brain development and neuropsychiatric risks. *Science* 362, eaat7615. doi:10.1126/science.aat7615.

Maaten, L. van der, and Hinton, G. (2008). Visualizing Data using t-SNE. *Journal of Machine Learning Research* 9, 2579–2605.

Murphy, K. M., Beston, B. R., Boley, P. M., and Jones, D. G. (2005). Development of human visual cortex: a balance between excitatory and inhibitory plasticity mechanisms. *Dev Psychobiol* 46, 209–221. doi:10.1002/dev.20053.

Oktar, Y., and Turkan, M. (2018). A review of sparsity-based clustering methods. *Signal Process* 148, 20–30. doi:10.1016/j.sigpro.2018.02.010.

Parsons, L., Haque, E., and Liu, H. (2004). Subspace clustering for high dimensional data: a review. *Acm Sigkdd Explor Newsl* 6, 90–105. doi:10.1145/1007730.1007731.

Pinto, J. G. A., Hornby, K. R., Jones, D. G., and Murphy, K. M. (2010). Developmental changes in GABAergic mechanisms in human visual cortex across the lifespan. Front Cell Neurosci 4, 16. doi:10.3389/fncel.2010.00016.

Pinto, J. G. A., Jones, D. G., Williams, C. K., and Murphy, K. M. (2015). Characterizing synaptic protein development in human visual cortex enables alignment of synaptic age with rat visual cortex. *Front Neural Circuit* 9, 3. doi:10.3389/fncir.2015.00003.

Sekula, M. (2020). *optCluster: Determine Optimal Clustering Algorithm and Number of*

Clusters. Available at: https://cran.r-project.org/web/packages/optCluster/index.html.

Siu, C. R., Balsor, J. L., Jones, D. G., and Murphy, K. M. (2015). Classic and Golli Myelin Basic Protein have distinct developmental trajectories in human visual cortex. *Front Neurosci-switz* 9, 138. doi:10.3389/fnins.2015.00138.

Siu, C. R., Beshara, S. P., Balsor, J. L., Mancini, S. J., and Murphy, K. M. (2018). "Use of Synaptoneurosome Samples to Study Development and Plasticity of Human Cortex," in Synaptosomes. (Humana Press, New York, NY), 269–286. doi:10.1007/978-1-4939-8739-9_15.

Siu, C. R., Beshara, S. P., Jones, D. G., and Murphy, K. M. (2017). Development of Glutamatergic Proteins in Human Visual Cortex across the Lifespan. *J Neurosci* 37, 6031 6042. doi:10.1523/jneurosci.2304-16.2017.

Tamayo, P., Slonim, D., Mesirov, J., Zhu, Q., Kitareewan, S., Dmitrovsky, E., et al. (1999). Interpreting patterns of gene expression with self-organizing maps: Methods and application to hematopoietic differentiation. Proc National Acad Sci 96, 2907–2912. doi:10.1073/pnas.96.6.2907.

Tibshirani, R., Walther, G., and Hastie, T. (2001). Estimating the number of clusters in a data set via the gap statistic. *J Royal Statistical Soc Ser B Statistical Methodol* 63, 411–423. doi:10.1111/1467-9868.00293.

Wierzchoń, S., and Kłopotek, M. (2018). Modern Algorithms of Cluster Analysis. *Stud Big Data*. doi:10.1007/978-3-319-69308-8.

Williams, K., Irwin, D. A., Jones, D. G., and Murphy, K. M. (2010). Dramatic Loss of Ube3A Expression during Aging of the Mammalian Cortex. *Front Aging Neurosci* 2, 18. doi:10.3389/fnagi.2010.00018.

Witten, D. M., and Tibshirani, R. (2010). A framework for feature selection in clustering. *J Am*

Stat Assoc 105, 713–726. doi:10.1198/jasa.2010.tm09415.

Witten, D. M., and Tibshirani, R. (2018). *sparcl: Perform Sparse Hierarchical Clustering and Sparse K-Means Clustering*. Available at: https://CRAN.R-project.org/package=sparcl.

Figure Legends

Figure 1. *The workflow for studying age-related molecular development of the brain.* First, arrange the data into an *n* x *p* matrix, where features (*p*) are represented as columns and samples (*n*) as rows. Then, select the desired clustering algorithm (ex. CLIQUE, PROCLUS, SPARCL, RSKC) and test its performance along a range of clusters (*k*). Lastly, determine the optimal *k* value using the elbow method and compare the median age of clusters with boxplots.

Figure 2. *Age-related organization of cases and initial clustering results.* A) A 2-Dimensional tSNE scatter plot colour-coded according to individual cases' age. B) tSNE scatter plots and box plots showing the results of *agnes* for *k* = 2, 4, 6, and 8 case clusters. The tSNE plots display individual cases as points. Points are colour-coded according to their designated cluster determined by *agnes*. Box plots display the distribution of ages for each cluster.

Figure 3. *Comparison of various sparse clustering methods.* Top-down PROCLUS subspace method across range of cluster numbers (2,4,6,8). The clusters are visualized in tSNE representations of the data by color-coding each data point with its cluster identity (Ai-iv) and in boxplots showing the median age of the samples in each cluster (Bi-iv). C-D) Bottom-up Clique subspace clustering method for a range of 'intervals'. Different clusters are visualized as coloured dots in a tSNE representation of the data (Ci-iv) and as box plots depicting the mean age of the samples (Di-iv). E-F) Sparse clustering after varying the inputted k cluster number (2,4,6,8). Different clusters are visualized as coloured dots in a tSNE representation of the data (Ei-iv) and as box plots depicting the mean age of the samples (Fi-iv). The colours in scatter plots and boxplots represent the cluster designation for all plots.

Figure 4. *Age-related clustering of 7 synaptic proteins for a single iteration.* Expression data from seven synaptic proteins were input into RSKC and used to identify k = 2, 4, 6 and 8 case clusters. For each k-value, three plots were constructed: A) a 2-Dimensional tSNE scatter plot showing samples colour-coded by their cluster designations, B) Box plots displaying the distribution of ages for each cluster, and C) a bar graph representing the RSKC weights for all seven proteins.

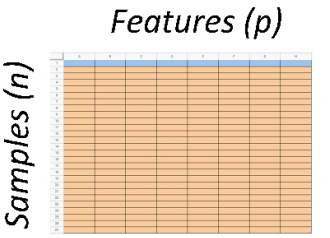
Figure 5. *Age-related clustering of 23 synaptic proteins for a single iteration.* A) Expression data from 23 synaptic proteins was used to identify six case clusters. Boxplots of cluster age ordered from youngest (red) to oldest (grey) median age. B) Bar plot visualizing RSKC feature weights for proteins. C) tSNE plot of the protein data scaled by RSKC weights and color-coded by RSKC cluster. In both A) and C), sample ages were reduced to samples averages to reduce crowding.

Figure 6. *Robust, age-related clustering of 23 synaptic proteins for 100 iterations.* A) Expression data from 23 synaptic proteins was used to identify six case clusters. The cluster designation of each case over 100 iterations of RSKC was used to visualize the distribution of case ages (in years). Boxplots denote the median and interquartile range of ages in each cluster, and points denote outliers. B) Heatmap visualizing the number of times each case was assigned to each cluster over 100 iterations of RSKC.

Figure 7. *Age-related clusters across different numbers of proteins and genes .* Expression data from A) 95 synaptic and immune-related proteins, B) 988 synaptic genes and, C)

17,237 protein-coding genes was used to identify six case clusters. The cluster designation of each case over 100 iterations of RSKC was used to visualize the distribution of case ages (in years). Boxplots denote the median and interquartile range of ages in each cluster, and points denote outliers.

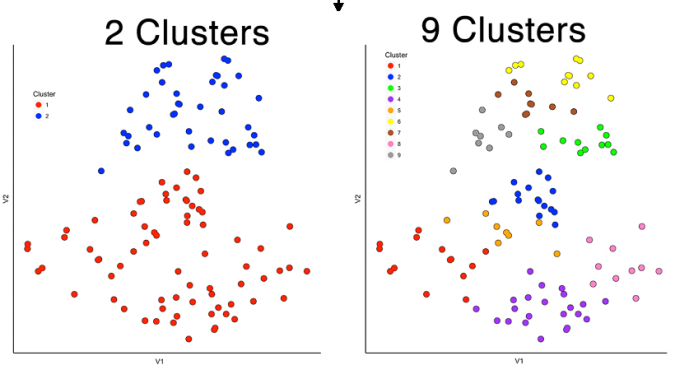
1.) Process the data



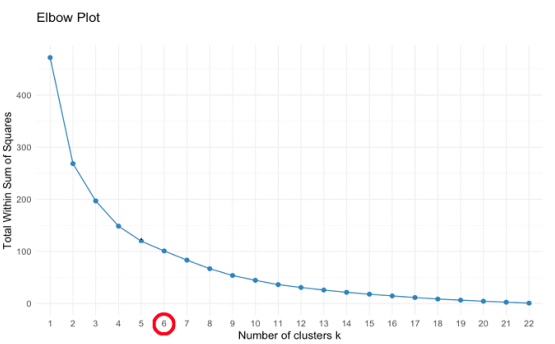
2.) Select the clustering algorithm

CLIQUE, PROCLUS, SPARCL, RSKC

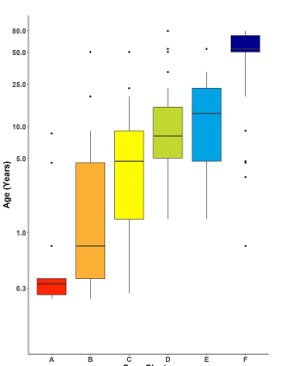
3.) Test a range of *k*-values

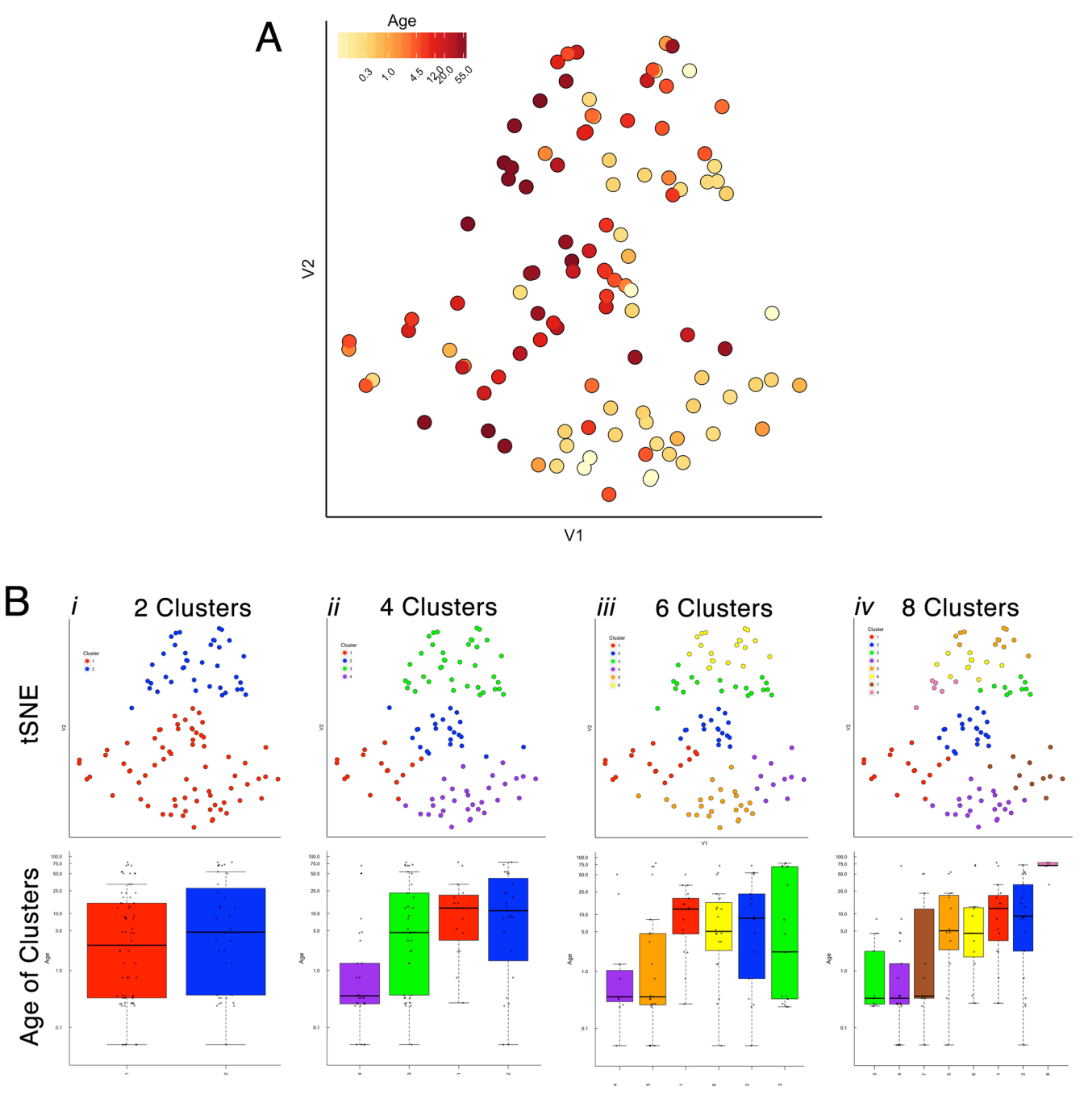


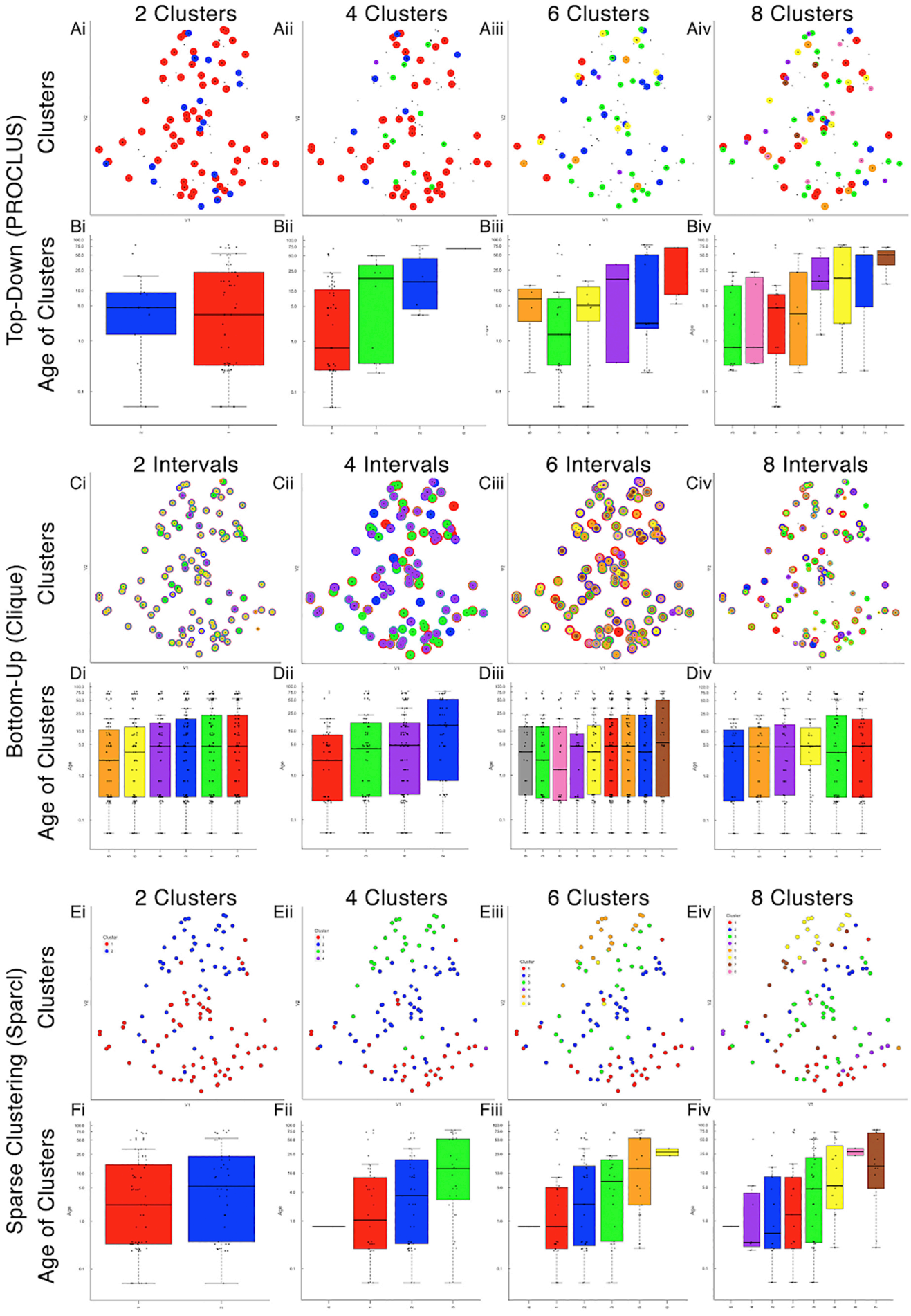
4.) Use the elbow method to determine the optimal *k*-value

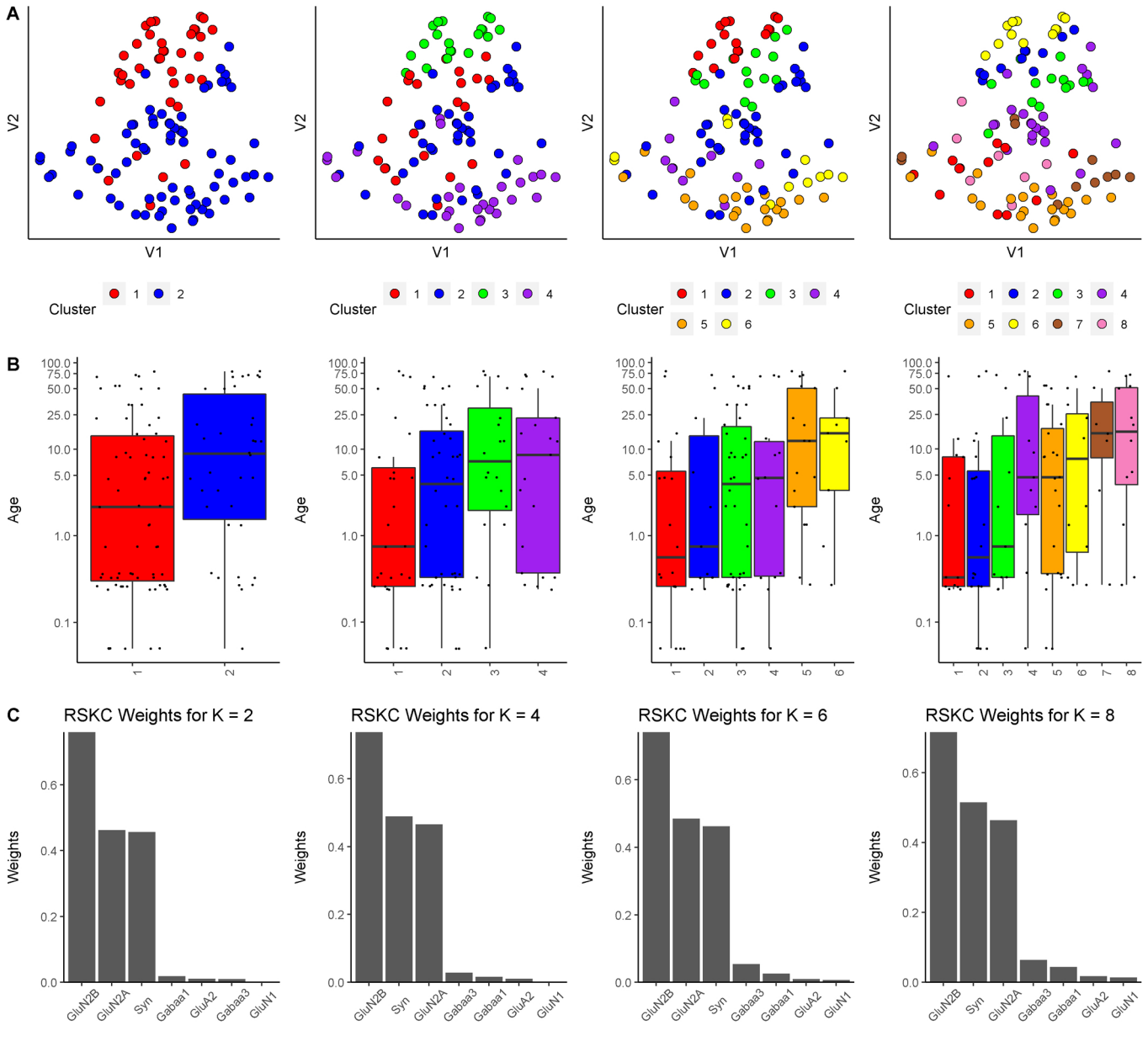


5.) Examine the median age of clusters

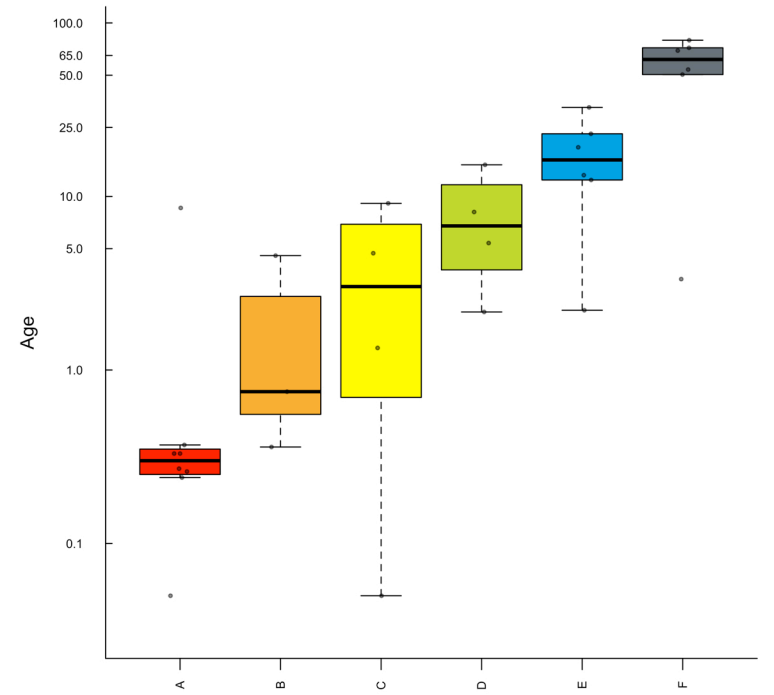



A
Age
0.3
1.0
4.5
12.0
20.0
55.0
V2
V1
B
i
2 Clusters
ii
4 Clusters
iii
6 Clusters
iv
8 Clusters
tSNE
Age of Clusters
Cluster
Age

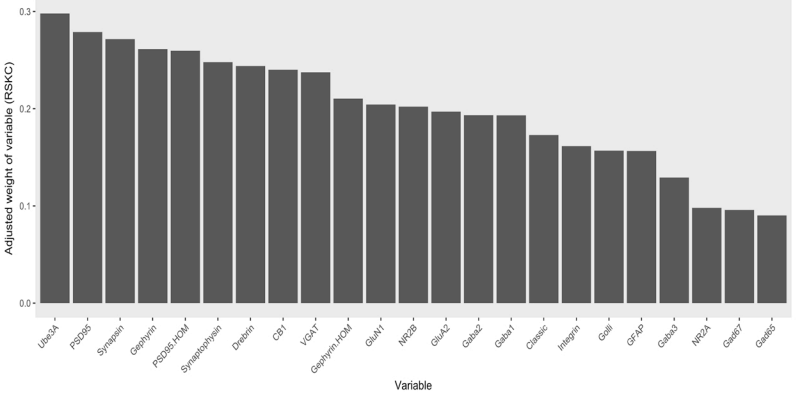

Top-Down (PROCLUS)
Clusters
Age of Clusters
2 Clusters
4 Clusters
6 Clusters
8 Clusters
Ai
Aii
Aiii
Aiv
Bi
Bii
Biii
Biv
Bottom-Up (Clique)
2 Intervals
4 Intervals
6 Intervals
8 Intervals
Ci
Cii
Ciii
Civ
Di
Dii
Diii
Div
Sparse Clustering (Sparcl)
Ei
Eii
Eiii
Eiv
Fi
Fii
Fiii
Fiv


A
V2
V1
Cluster
1
2
3
4
5
6
7
8
B
Age
100.0
75.0
50.0
25.0
10.0
5.0
1.0
0.1
C
RSKC Weights for K = 2
RSKC Weights for K = 4
RSKC Weights for K = 6
RSKC Weights for K = 8
Weights
0.0
0.2
0.4
0.6
GluN2B
GluN2A
Syn
Gabaa1
GluA2
Gabaa3
GluN1

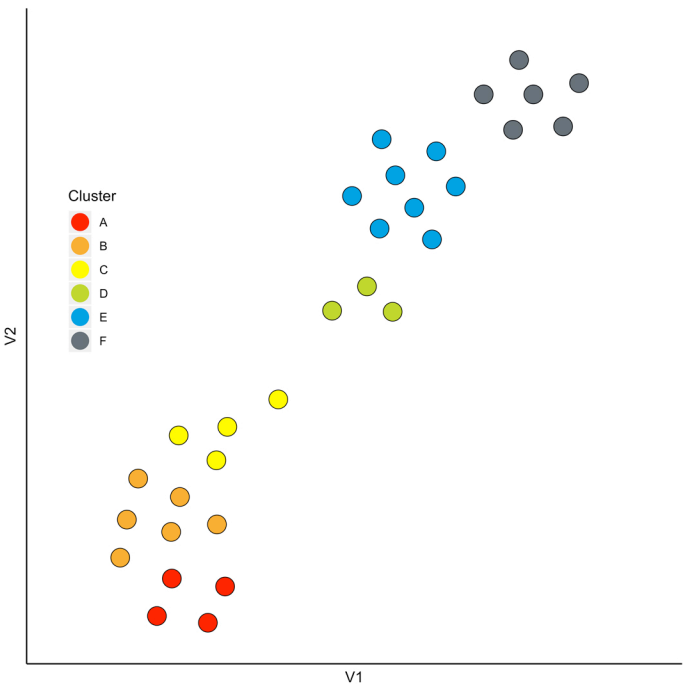
A

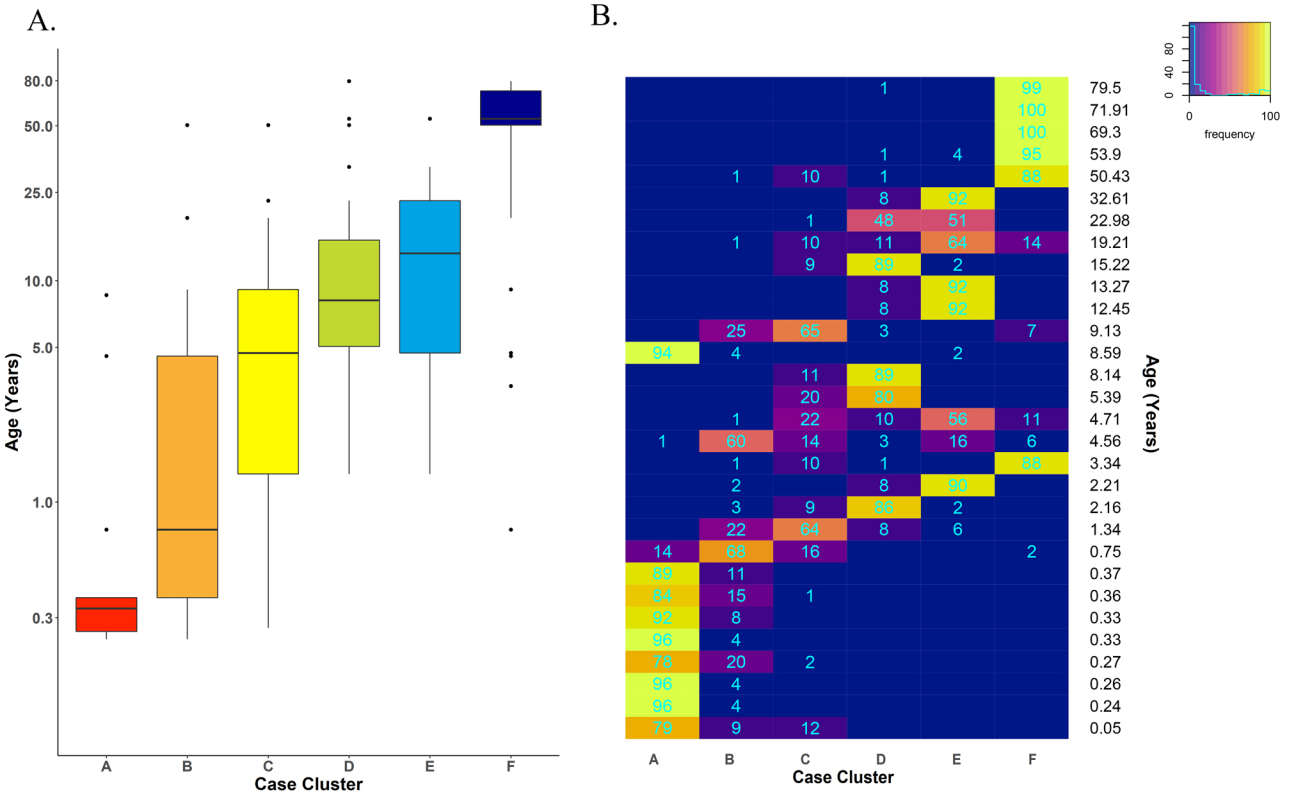

Age
100.0
65.0
50.0
25.0
10.0
5.0
1.0
0.1
A
B
C
D
E
F

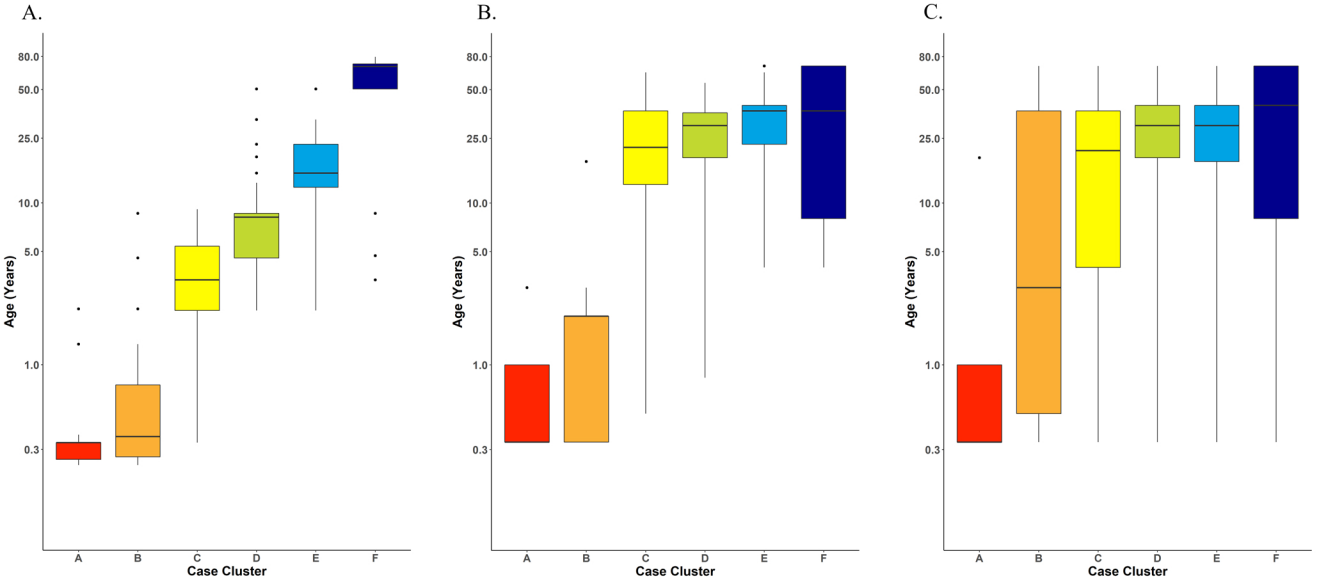
B


Adjusted weight of variable (RSKC)
0.3
0.2
0.1
0.0
Ube3A
PSD95
Synapsin
Gephyrin
PSD95-HOM
Synaptophysin
Drebrin
CB1
VGAT
Gephyrin-HOM
GluN1
NR2B
GluA2
Gaba2
Gaba1
Classic
Integrin
Golli
GFAP
Gaba3
NR2A
Gadd67
Gad65
Variable

C


Cluster
A
B
C
D
E
F
V2
V1


A.
B.
Age (Years)
80.0
50.0
25.0
10.0
5.0
1.0
0.3
A
B
C
D
E
F
Case Cluster
frequency
0
40
80
100
Age (Years)
79.5
71.91
69.3
53.9
50.43
32.61
22.98
19.21
15.22
13.27
12.45
9.13
8.59
8.14
5.39
4.71
4.56
3.34
2.21
2.16
1.34
0.75
0.37
0.36
0.33
0.33
0.27
0.26
0.24
0.05


A.
B.
C.
80.0
50.0
25.0
10.0
5.0
1.0
0.3
Age (Years)
A
B
C
D
E
F
Case Cluster