# Sparse Representation Learning Derives Biological Features with Explicit Gene Weights from the Allen Mouse Brain Atlas

Mohammad Abbasi[1], Connor R Sanderford[1], Narendiran Raghu[1], Mirjeta Pasha, Benjamin B Bartelle[1]

1:Arizona State University

**ABBREVIATIONS:**

| | |
|---|---|
| *AMBA* | Allen Mouse Brain Atlas |
| *AMI* | Adjusted Mutual Information |
| *ARA* | Allen Reference Atlas |
| *ARI* | Adjusted Rand Index |
| *CA* | Hippocampal Cornu Ammonis |
| *CP* | Caudate Putamen |
| *CUL4,5* | Lobules IV-V |
| *DG* | Dentate Gyrus |
| *DLSC* | Dictionary Learning and Sparse Components |
| *GO* | Gene Ontology |
| *ICA* | Independent Component Analysis |
| *ISH* | In Situ Hybridization |
| *KPCA* | Kernel Principal Component Analysis |
| *L-BFGS* | Limited-memory Broyden, Fletcher, Goldfarb, and Shanno |

| | |
|---|---|
| *MERFISH* | Multiplexed Error-Robust Fluorescence In Situ Hybridization |
| *MI* | Mutual Information |
| *MOB* | Main Olfactory Bulb |
| *PCA* | Principal Component Analysis |
| *ROC-AUC* | Receiver Operating Characteristic-Area Under the Curve |
| *SFt* | Sparse Filtering |
| *SPCA* | Sparse Principal Component Analysis |
| *SVD* | Singular Value Decomposition |
| *t-SNE* | t-Distributed Stochastic Neighborhood Embedding |
| *UMAP* | Uniform Manifold Approximation and Projection |
| *WT* | Wild Type |

**ABSTRACT:**

Unsupervised learning methods are commonly used to detect features within transcriptomic data and ultimately derive meaningful representations of biology. Contributions of individual genes to any feature however becomes convolved with each learning step, requiring followup analysis and validation to understand what biology might be represented by a cluster on a low dimensional plot. We sought learning methods that could preserve the gene information of detected features, using the spatial transcriptomic data and anatomical labels of the Allen Mouse Brain Atlas as a test dataset with verifiable ground truth. We established metrics for accurate representation of molecular anatomy to find sparse learning approaches were uniquely

capable of generating anatomical representations and gene weights in a single learning step. Fit to labeled anatomy was highly correlated with intrinsic properties of the data, offering a means to optimize parameters without established ground truth. Once representations were derived, complementary gene lists could be further compressed to generate a low complexity dataset, or to probe for individual features with >95% accuracy. We demonstrate the utility of sparse learning as a means to derive biologically meaningful representations from transcriptomic data and reduce the complexity of large datasets while preserving intelligible gene information throughout the analysis.

**INTRODUCTION:**

Dimensionality reduction, manifold learning, and clustering are essential methods for processing transcriptomic data into “representations” of biology interpretable to the human mind (Xiang et al., 2021). A typical workflow can include Principle Component Analysis (PCA) (Jiang et al., 2020), graph embedding with t-Distributed Stochastic Neighborhood Embedding (t-SNE) (Kobak and Berens, 2019) or Uniform Manifold Approximation and Projection (UMAP) (Ma and Lim, 2021), and separation by Leiden clustering (Ciortan and Defrance, 2021). This approach is robust, but presents several weaknesses: Initial dimensionality reduction filters for high variance global trends over localized features (Torgerson, 1952). Manifold Learning methods, UMAP and t-SNE, conversely, are more sensitive to local structures of the data (Liu, 2020). Clustering includes noise with informative groupings (Ben-David and

Haghtalab, 2014). Overall, each learning step further convolves input data such that the contribution of any one gene to a cluster is not explicit.

Well described markers solve most of these issues, offering the essential ground truth to connect points on a graph to biological states, providing a starting point for Gene Ontology (GO) analysis and experimental validation. Many functional states however can only be described by the relative expression levels of common genes, while data lacking cellular resolution cannot be fully described by a singular marker even if one were present. In these cases, each step of an analysis pipeline further obscures what a cluster might represent, requiring supervised learning approaches and careful validation to parse gene information.

Given the incredible advances in applied information theory, we asked if an unsupervised learning method could derive localized features within a dataset, while retaining actionable information about the inputs. Essentially, we asked if it was possible to generate representations of biology from a transcriptomic data, in a single step, without implicit priors. To eliminate the need for dimensionality reduction and clustering, we explored “sparse” learning methods, optimized for building localized features from a minimal number of input elements sparsely distributed within a large dataset (Kolali Khormuji and Bazrafkan, 2016).

To test if derived features were representations of biology and not just the isolation of a unique marker gene, we used a dataset with well described biological phenomena as a testable ground truth. The Allen Mouse Brain Atlas (AMBA) is the most comprehensive spatial transcriptomic dataset of the whole brain available during this

analysis. What makes AMBA the optimal test data for our methods is that every voxel of transcriptomic data has been labeled as an element of anatomy based on thousands of person hours worth of labor (Ng et al., 2009). The low resolution (200μm isotropic) of AMBA convolves multiple cell types, making single gene markers uncommon and an incomplete descriptor of any element of neuroanatomy (Bohland et al., 2010). While the developmental genes involved in forming different brain structures are mostly inactive in the adult mouse brain, gene expression can still functionally define anatomy (Bohland et al., 2010). These data therefore offer a means to test the ability of a learning method to describe established elements of anatomy with derived signatures of gene expression (Bohland et al., 2010; Ortiz et al., 2020).

We surveyed previously established methods and new advances to find sparse approaches offered the intriguing potential to return biologically meaningful representations without the need for secondary clustering steps. With some parameter tuning, all sparse representation learning methods tested could generate anatomical features with a corresponding ranked list of genes. Moreover, gene lists could be compressed to a minimal ensemble of markers for each element of anatomy while retaining high fidelity to ground truth offering a means to reduce the complexity of the whole dataset based on the informativeness of each element and a ready made probe list for detecting individual representations.

**RESULTS:**

To survey representation learning methods, we first established metrics for comparing representations derived from the AMBA dataset to ground truth anatomical information and to test the principles behind the algorithms that generated them. Because anatomy is presented as labels, we initially applied secondary clustering using K-means for all methods so that our results were directly comparable to previously published studies (Bohland et al., 2010; Li et al., 2017). AMBA labels 574 unique brain structures at greater than 200 μm resolution. Because many smaller brain regions are only a few voxels large, even if every anatomical region was perfectly defined by transcription, biological variance, resolution, and overfitting artifacts introduce significant error.

**ICA generates anatomical representations from the AMBA dataset with additional clustering by k-means.**

First postulated in information theory (Shannon, 1948), the maximum information principle presents a model of information transfer that has since been used to filter complex data (Bell and Sejnowski, 1995). Infomax based algorithms build representations by separating low Mutual Information (MI), or independent, components of a set, and grouping high MI, or the most mutually interdependent ones. InfoMax is central to independent component analysis (ICA) and a commonly-used objective in many other machine learning methods, such as variational autoencoders (Crescimanna and Graham, 2020; Hjelm et al., 2019). ICA was recently used on similarly low resolution brain-wide spatial transcriptomic data, so we asked if ICA

components would recapitulate anatomy in AMBA in a similar manner to sequencing based approaches (Ortiz et al., 2020).

ICA generated components that were broadly active across the brain, but with areas of high intensity that appeared anatomical (**Fig 1a**). To filter for highly active areas and parcelate the data into labels, we applied K-means clustering, resulting in representations that were more visually similar to anatomy, though less detailed (**Fig 1b**). To quantify how well clustered representations fit anatomy, we used Adjusted Mutual Information (AMI) and Adjusted Rand Index (ARI) to test if a region labeled by clustering was co-labeled by anatomy, then generated scores from 1 to 550 k-means clusters to look for an optimal fit to ground truth. Peak fitting occurred at 300 clusters by AMI (**Fig 1c**) and 150 by ARI (**Fig 1d**), well below the 574 labeled elements of anatomy.

Visual inspection of clustered representations showed the poorest fits to anatomy appearing at interfaces like olfactory bulb and nucleus, or as a single coronal slice, which was the orientation of the majority of In Situ Hybridization (ISH) data used to build AMBA (**Fig 1e**). Similar artifacts were previously reported with ICA of spatial transcriptomic data, requiring manual curation before subsequent clustering and parcellation by supervised learning methods (Ortiz et al., 2020). Without applying these additional steps, ICA/K-means representations offered the lowest similarity to AMBA labels of any method tested.

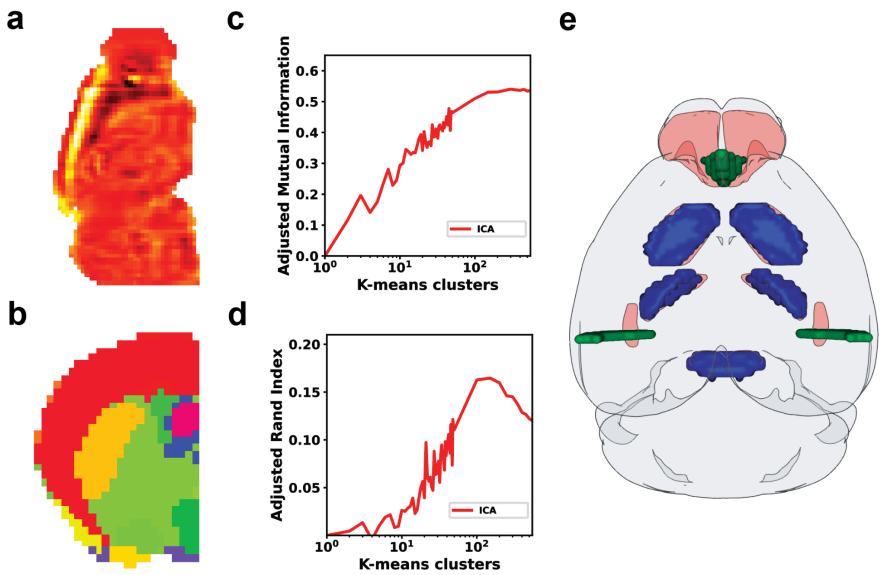


**Figure 1: Representation learning on AMBA with ICA.**

Figure of representations derived from ICA and sparse methods in the same sagittal slice, and the clustered features in the coronal view for better demonstration of cortical definitions.**(a)** Whole brain sagittal view of an ICA representation mostly active in the cortex. **(b)** coronal view of labels generated by K-means clustering of the ICA representations (K=200). **(e)** 3D representation of K-means clusters with selected top overlapping clusters (in blue) with brain regions (Olfactory Tubercle, Reticular nucleus of the Thalamus, and Pontine Gray) in red and low overlapping clusters (in green) with brain regions (Basolateral Amygdalar Nucleus, posterior part and Main Olfactory Bulb) in red.

**Variance based learning generates representations of ground truth anatomical information after clustering with no advantage to nonlinearity.**

The earliest analysis of the AMBA data used Singular Value Decomposition (SVD) and K-means clustering to generate representations that resembled anatomy (Ng et al., 2009). We applied PCA, a closely related method, using identical parameters for comparison (Ng et al., 2009). PCA of AMBA generated components with high intensity areas that visually recapitulate anatomy, but are active across most of the brain, similar to ICA (**Fig 2a**). K-means clustering on the full ensemble of PCA components parcellated brain voxels into finer and more anatomical looking representations of anatomy than ICA (**Fig 2b**).

Biological relationships are not always linear, so we asked if a nonlinear approach could more accurately reflect underlying biology. Unlike the eigendecomposition of PCA, Kernel PCA (KPCA) components are fit to a preselected kernel function. Parameter tuning across multiple nonlinear transformations however, did not produce visually different components from those generated by PCA (**Fig 2c**). Applying K-means to KPCA components produced similar parcellations to PCA/K-means (**Fig 2d**). Quantitative comparison of linear and nonlinear methods demonstrated no advantage to KPCA, with some kernels performing worse at every value of k, suggesting no advantage to non-linear methods.

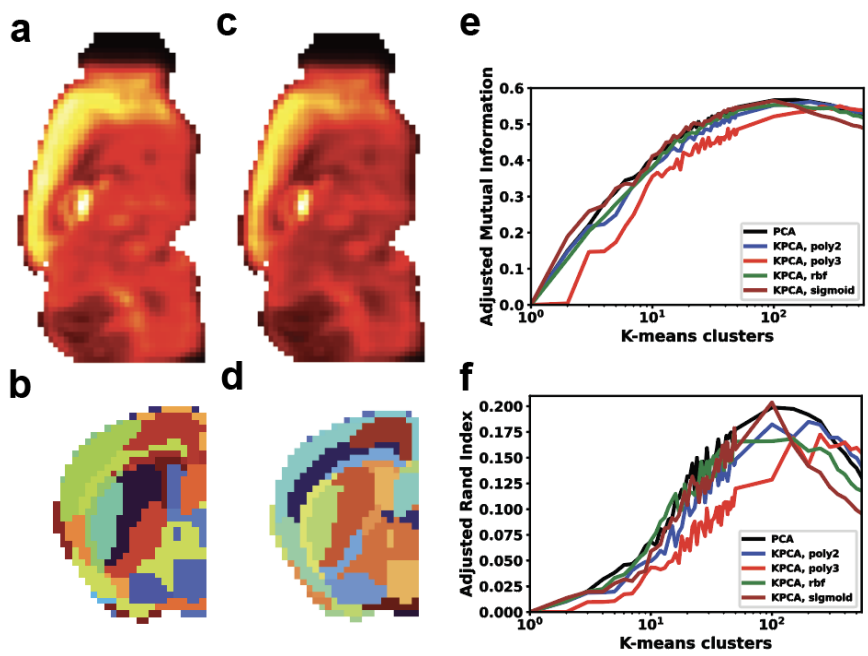


**Figure 2: Linear and nonlinear PCA generate equivalent representations of anatomy from AMBA.**

**(a)** First PCA component representation in a sagittal slice. **(b)** Representation of K-means clustering (K=200) of PCA components in a coronal slice of the left hemisphere. **(c)** First KPCA component with Quadratic kernel function in a sagittal slice. **(d)** Representation of K-means clustering (K=200) of KPCA components with Quadratic kernel function in a coronal slice of the left hemisphere.

**(e,f)** Adjusted Mutual Information and Adjusted Rand Index of clustered components using K-means clustering with K ranging from 1 to 50 and from 50 to 550 with a step of 50. The dimension reduction methods have been mentioned in the panel legends.

**Sparse methods generate the most accurate anatomical representations with clustering and in some cases in a single step.**

Sparse representation learning algorithms generally satisfy infomax principles but apply sparsity constraints to one or more aspects of their outputs: the number of representations that are informative of the input data (feature or population sparsity), the samples of data that comprise any given representation (sample or lifetime sparsity), and the distribution of informative samples across all representations (dispersal) (Ngiam et al., 2011). Ideally sparse representations would then be non-overlapping, with each one presenting similar levels of informativeness regarding the input data. Previous work has claimed AMBA representations generated with Dictionary Learning and Sparse Components (DLSC), outperformed PCA when combined with clustering (Li et al., 2017).

We generated representations starting with previously described optimal hyperparameters, including a low sparsity hyperparameter ($\alpha$ = 0.1) (Li et al., 2017). Sparsely derived features covered similar anatomy as other tested methods with similarly broad activity (**Fig 3a,c,e**). As previously reported, clustering of DLSC resulted in similar but visually finer representations than those generated by PCA (**Fig 3b**). Sparse PCA (SPCA) performed similarly with the same hyperparameters, generating larger representations with comparable anatomical features to DLSC (**Fig 3b,c**) We then tried Sparse Filtering (SFt), a method that derives its own sparsity levels. SFt generated uniquely compact features, with activation in single anatomical-like regions

(**Fig 3e**). SFt + K-means clustering yielded fine features, including the clearest depiction of cortical layers of any method tested (**Fig 3f**).

We quantified similarity to labeled anatomy across all sparse methods with fixed parameters across a range of clusters, showing greater accuracy for SFt than any other method, with a peak score of 0.61 at 200 clusters based on AMI scores (**Fig 3g**). ARI scores showed SFt performing substantially better, peaking at 0.28 with 50 clusters. In all methods, AMI scores peaked well below the 574 labeled brain regions, with most fitting optimally at ~200 clusters. The most accurate representations were predominantly subcortical, with resolution preventing an even more optimal fit to anatomy. Some visually obvious artifacts appeared as single slices or matched artifacts in the original histology data (**Fig 1g**). Visual inspection of each feature suggested improved fits were in part due to fewer coronal plane artifacts, but there is no way to determine if the remaining mismatch between SFt and AMBA labels are due to the limits of resolution.

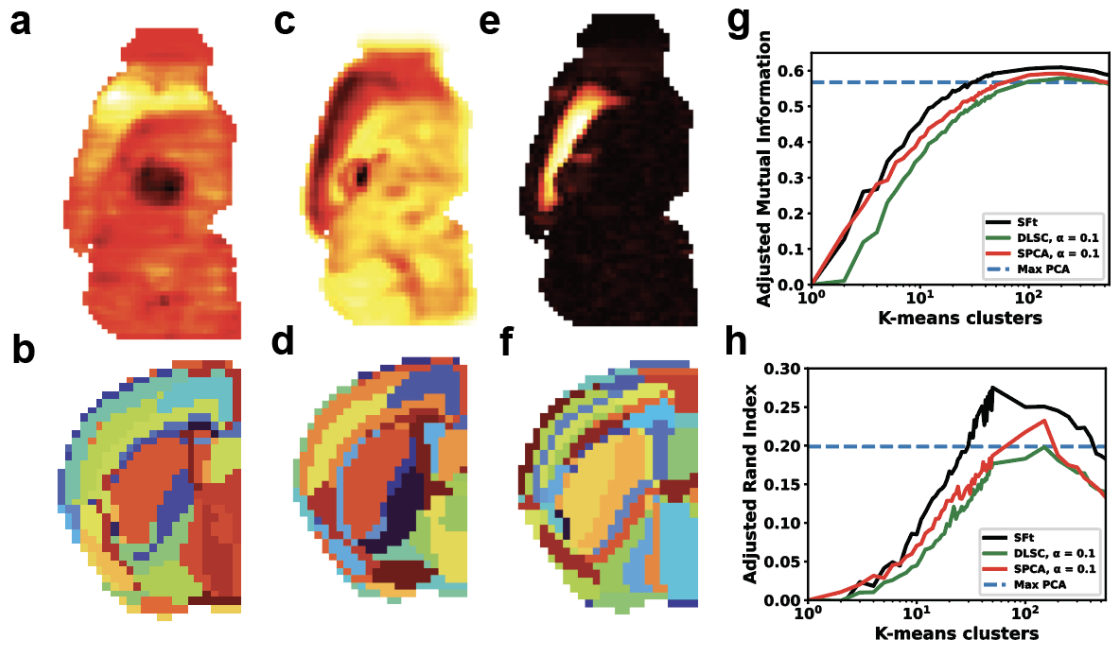


**Figure 3: Sparse representations of AMBA.**

(**a**) DLSC representation, active in the cortex. (**b**) K-means clustering (K=200) of DLSC representations in a coronal slice. (**c**) SPCA representation, chosen for similarity. (**d**) K-means clustering of SPCA representations. (**e**) SFt representation, active in the cortex . (**f**) K-means clustering of SFt components.

(**g,h**) Adjusted Mutual Information and Adjusted Rand Index of clustered components using K-means clustering with K ranging from 1 to 50 and from 50 to 550 with a step of 50.

**All sparse learning methods can be optimized to generate direct representations of anatomical features.**

The uniquely sparse and compact features from SFt suggested no secondary clustering was required to describe anatomy and we asked if this could be achieved with other sparse methods after parameter tuning. To optimize DLSC and SPCA for better representations, we established descriptive metrics of spatial properties, similarity to ground truth, and underlying gene information: Shannon entropy measures the total information contained in all features. Sørensen–Dice similarity coefficient is a measure of overlap between a representation and the best fitting anatomical ground truth. The number of connected components describes the contiguity of a representation. Feature sparsity measures the exclusivity of activation to a given representation, with a single activated point being maximally sparse. Spatial entropy measures the total homogeneity of representation space. Finally weight sparsity in the samples comprising each representation.

Across all conditions, Shannon entropy remained the same, indicating that no information was lost using a given method, only distributed differently depending on the method and parameters **(Fig 4a-c)**. As a baseline, metrics were applied to the expression of individual genes **(Fig 4a)**. Without a clustering step PCA made poor representations of anatomy. For individual PCA components, spatial entropy remained high, with a lower connected components score and poor fits to anatomy by Dice coefficient **(Fig 4b)**.

SFt offered the highest Dice coefficient of any method, demonstrating that individual representations fit directly with ground truth anatomy **(Fig 4b)**. This correlated with a high score for connected components, ideally one anatomical region

per representation. A high feature sparsity score indicated spatially regions of high activation, while spatial entropy score further described low activity across most of anatomical space. A similar weight sparsity score to PCA indicated that a similar number of genes were used to generate each representation, with minimal sparsity afforded by SFt.

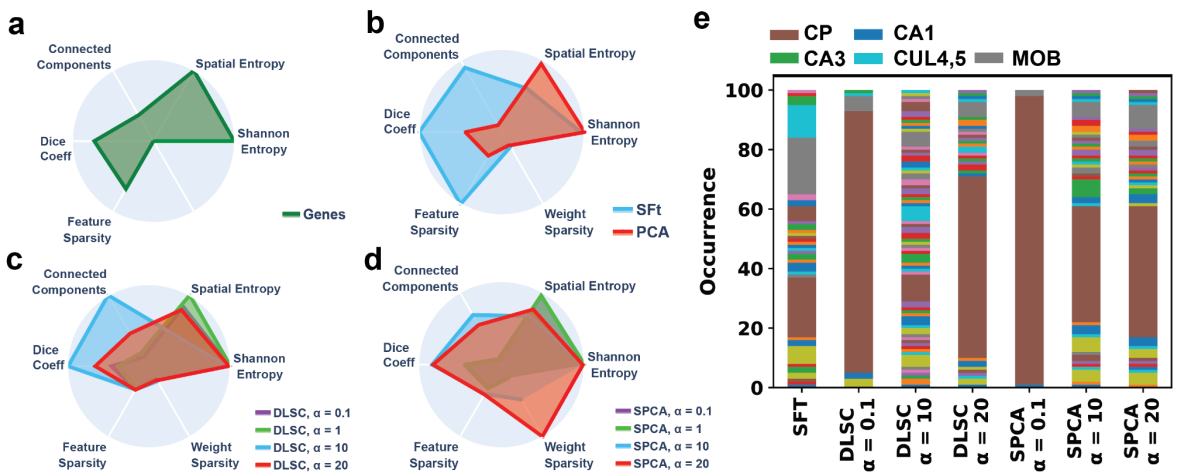


**Figure 4: Relative performance metrics of DLSC, SPCA, and SFt.** Representations derived from each method without additional clustering were evaluated along 5 metrics with increasing sparsity values ($\alpha$) when possible. Metrics, minus weight sparsity, applied to individual genes (**a**) PCA had the poorest metrics, while SFt had the best Dice coefficients of any method, correlating with feature sparsity and connected component score (**b**). With increasing $\alpha$, DLSC representations lose spatial entropy and gain feature sparsity, with fewer connected components and improvements in individual fit to ground truth as measured by

Dice coefficient (**c**). For SPCA, higher $\alpha$ increases weight sparsity while Dice and connected components peak at $\alpha$=10 (**d**). Labeling all representations by the anatomy they best represent showed some brain regions were redundantly represented by all methods, with the caudate putamen (CP) overrepresented across all methods and $\alpha$ values, with representation diversity peaking with Dice (**e**).

For DLSC, previous work reported a peak AMI score with clustering when the sparsity parameter $\alpha$ was low. Increasing $\alpha$ however improved Dice coefficient, peaking at 100X the optimal value for AMI **(Fig 4c)**. Peak Dice coefficient again correlated with peak connected component score, and comparable spatial entropy to SFt, indicating that information consolidated within features with accuracy until becoming incoherently sparse. Feature sparsity did not change with parameter tuning however, nor did weight sparsity, suggesting that the contents of the features did not change substantially with tuning.

For SPCA, the Dice coefficient did not match DLSC or SFt with optimal parameter tuning **(Fig 4d)**. While this did correlate with the number of connected components, it was not related to spatial entropy or feature sparsity. At high $\alpha$, optimizations were limited to weight sparsity, meaning the number of genes used to build each representation compressed without changing representation accuracy.

Comparing each feature across all methods and conditions, Dice coefficients were highly correlated with the connected components score ($r= 0.97$ $p<0.01$).

Individual measures of sparsity across features and weights did not reflect general properties of a biologically relevant feature.

**Sparse methods generate overlapping representations.**

Dice coefficient and connected component scores described how well a representation fit a single element of anatomy, however upon labeling which elements of anatomy returned an optimal Dice score for each of the 100 features generated, we found some brain regions were overrepresented. For SFt, half of the features represented just 3 regions, for a total of 33 unique representations.

In DLSC and SPCA overlap was more extreme at the lowest $\alpha$ tested, with ~95% of features being representations of the CP (**Fig 4e**). Secondary clustering with K-means consolidated this overwhelming association into one label, with the remaining information describing the rest of anatomy and surprisingly giving an optimal fit to ground truth by AMI score (Li et al., 2017). Increasing $\alpha$ improved distribution of informativeness, generating more diverse features while improving the Dice coefficient. This was optimal in DLSC, with 64 unique representations at the same $\alpha$ that returned optimal Dice coefficient. Overshooting optimal sparsity caused a collapse back to redundant representations with an overall poorer fit. For SPCA, just as increasing $\alpha$ beyond minimal levels did not improve the Dice coefficient, the number of unique representations did not increase. Across all methods and conditions, the number of unique features correlated with the Dice coefficient ($r= 0.91$ $p<0.01$).

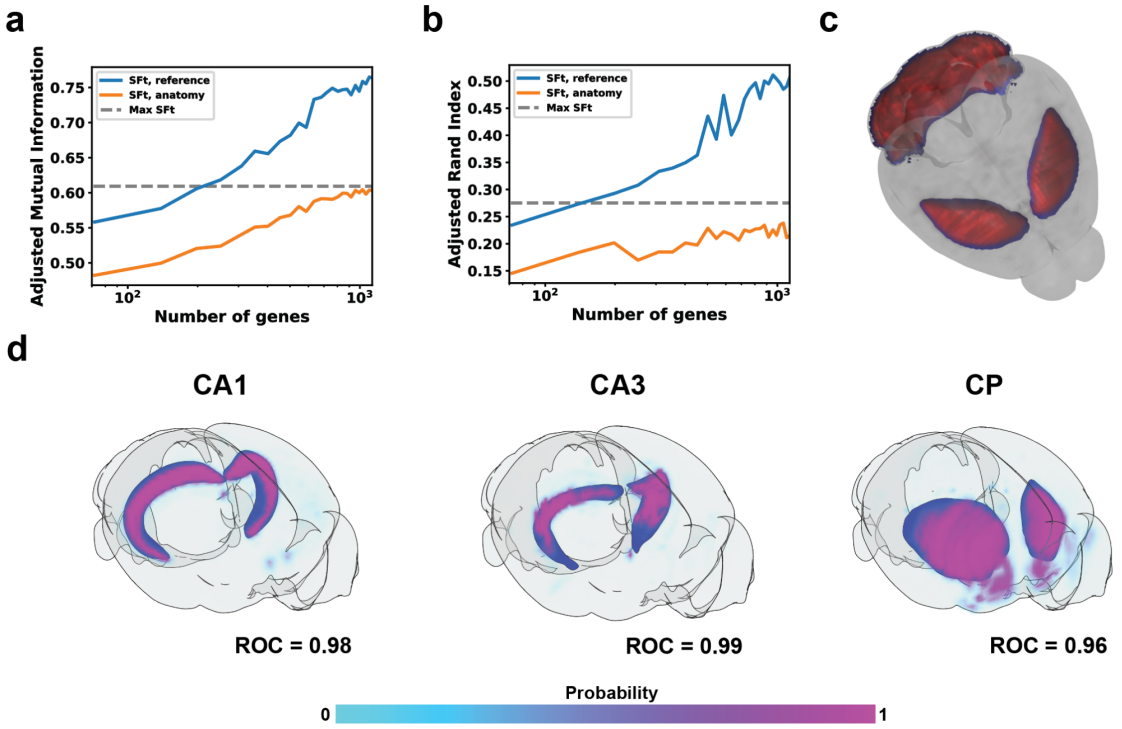


**Figure 5: Performance metrics of compressed gene sets.**

Lower ranked genes across all SFt representations were iteratively removed from AMBA data, followed by SFt/clustering on the compressed dataset. (**a**) Adjusted Mutual Information between the full dataset vs. compressed representations and anatomy vs compressed representations. (**b**) Adjusted Rand Index of compressed representations vs full representations and anatomy.

(**c**) Visualization of a selected feature derived from the SFt of the 2941-gene dataset (in red) with highest Pearson correlation to the 584-gene dataset (in blue), where 95% of the AMI score with reference is met. (**d**) Ten genes with the largest associated weights were used to train classifiers for their corresponding brain region. Probability

of positive classification is shown by the colorbar, and the corresponding anatomy shown in blue. ROC score for five-fold cross-validation is shown for each method.

**Compressed SFt features can be used to reduce data complexity or for robust classification of individual elements of anatomy.**

While information is distributed across all representations, the genes that comprise each one have explicit weightings. We asked if the gene weights from one step representation learning could be used as a basis for compressing data to the most relevant markers, while retaining representation fidelity. SPCA had the highest initial weight sparsity, but we chose SFt representations for their higher accuracy by Dice coefficient to see the effects of reducing the number of input genes (**Fig. 4b,d**).

We compressed the AMBA dataset by generating SFt representations, establishing a weighted gene list for the full representation ensemble, eliminating the lowest ranked genes from the remaining data, and repeating SFt until only one top rated gene for a given representation was left. Representation fidelity degraded linearly when compared against SFt labels from a full dataset (**Fig 5a**), however compared to anatomical ground truth, representations from the degraded data did not see a performance drop until the gene samples fell were compressed to under 1000 genes then fell with a shallower slope, retaining 95% fidelity to anatomy at over 80% compression, or 584 genes (**Fig 5b**). Inaccuracies at this stage were limited to the edges of anatomical regions, suggesting the initial loss of fidelity was masked by resolution and systemic variance (**Fig 5c**).

We then asked if the gene information from a single SFt representation could be compressed using a supervised approach. We took the highest fidelity SFt representations by Dice coefficient, hippocampal cornu ammonis (CA) 1,CA3, and the CP, as labels, then used the top 10 most weighted genes from each to train a logistic regression based classifier (**Fig 5d**). Of the most highly weighted genes, SFt picked several known markers for each region of the hippocampus CA1 (Hamilton et al., 2017), CA2 (Dudek et al., 2016), CA3 (Hamilton et al., 2017), and Dentate Gyrus (DG) (Radic et al., 2017) (Table 1). CP has high molecular diversity and uniquely expressed genes, several of which were selected by SFt. Highly weighted genes included known markers of each anatomical region, but also genes expressed across multiple elements of anatomy or even markers of adjacent anatomy that could define a border (**Table 1**). Data was randomized over 5 iterations, taking 80% for training and the remaining 20% and accuracy measured by receiver operating characteristic (ROC) curves. Classification was highly accurate, with scores above 95% for every permutation. This demonstrated that an additional supervised learning step on a representation can compensate for lower sample weight sparsity by reweighting gene lists for compression.

| Brain regions | CA1 | CA2 | CA3 | DG | CP |
|---|---|---|---|---|---|

| Genes | Pkp2 | Fibcd1 | Cyp7b1 | Crlf1 | Dock10 |
|---|---|---|---|---|---|
| | Tgfb2 | Arl15 | Bok | Prox1 | Serpina9 |
| | Tdo2 | C630041L24Rik | Cpne8 | Tdo2 | Prox1 |
| | Prox1 | Sstr4 | Tnfaip8l3 | Tnip2 | Rasd2 |
| | Camk1d | TC1412430 | Tanc1 | Dock10 | Tdo2 |
| | Crlf1 | Matn2 | Lats2 | C78409 | Cyp7b1 |
| | Il1rap | Cpne8 | 9930021J17Rik | Il1rap | C78409 |
| | Lpl | Arhgap12 | Fibcd1 | C1ql2 | Rasgrp2 |
| | Dock10 | Dock10 | Mycl1 | Cyp7b1 | Crlf1 |
| | Sms | Kcnd2 | Lhfpl2 | Pip5k1a | Pip5k1a |
| | Lct | Sh3rf1 | Iyd | Lct | C630035N08Rik |
| | C230081A13Rik | Prox1 | Camk1d | Syt6 | Dach1 |

**Table 1.** 12 genes with highest weights for the overlapping SFt representation with

the brain regions, with selectively expressed genes of a given brain region highlighted.

**DISCUSSION:**

In a broad comparison of representation learning methods, sparse algorithms generated more accurately described anatomical ground truth. SFt in particular generated direct anatomical representations without the need for clustering. More comprehensive metrics demonstrated that increasing the sparsity parameter for DLSC and SPCA could similarly produce direct representations of anatomy, with each method optimizing for different aspects of sparsity in the dataset. Sparse features did not represent every brain region, but each representation offered a weighted gene list that could be further used as a basis for compression to relevant genes for analysis, or as a minimal probe list for deriving specific elements of anatomy. SPCA was the best method for generating a minimal gene list, but we were able to further minimize SFt gene weights with a second learning step.

Previous use of DLSC reported low sparsity as optimal for anatomical representations by global analysis of fit to ground truth as measured by MI, but deeper analysis showed a different relationship with the sparsity parameter. Low sparsity derives redundant, broadly active, and overlapping features that only represent anatomy after secondary clustering. Increasing sparsity derives more accurate representations of specific elements of anatomy, directly resolving more unique regions

directly. DLSC was the least accurate method, but at optimal sparsity, generated nearly twice the number of unique representations.

The detailed extrinsic ground truth of anatomy made AMBA an ideal dataset for this comparison, but during analysis we found inherent properties of the data that could be used for optimization when anatomy is not available. Dice coefficient, or goodness of fit to ground truth correlated with connected components score, and the number of unique features. These metrics indicate that sparse learning methods are good at finding features that are spatially compact with unique molecular signatures. Conversely, contiguity and unique molecular function is a biological property of many elements of anatomy, and could themselves be a basis for learning in a similar unlabeled dataset.

By our metrics, SFt representations were more accurate than DLSC or SPCA even after parameter tuning, for the same number of features. This is likely due to how each method applies sparsity to feature detection. For DLSC and SPCA, the sparsity variable $\alpha$ is a regularization parameter, which imposes sparsity on components in DLSC and loadings in SPCA. SFt does not have a sparsity parameter, and instead uses a nonlinear transformation, $l_2$ normalization across features, $l_2$ normalization across samples and global $l_1$ minimization. The only parameter for SFt beyond the input data is the number of representations to optimize over and is common to all methods tested (Gultepe and Makrehchi, 2018). The simple implementation of SFt, makes it an attractive method for future applications, but raises the question if this means of applying sparsity is truly self optimizing or just particularly compatible with this dataset.

In future implementations SFt could be tuned for further sparsity by using a smoothed fractional norm in place of the $l_1$ norm (Buccini et al, 2020).

Sparse representation learning methods without secondary clustering offer the advantage of a ranked gene list of samples that comprise each feature. Representations were not simply correlatively expressed genes. Instead, genes that defined borders of anatomy from the inside or outside were both highly weighted. Within the hippocampus, Dock10, an established marker of CA2 was highly weighted in representations of every anatomical region that bordered its expression (CA1, CA3, DG, CP) (**Table 1**) (Kohara et al., 2014).

SPCA delivered the most sparse gene lists, but we found that SFt representations could be even further compressed without significant loss of fidelity. In this pilot study, after training the model on the 3k gene dataset we reduced the input data to 580 genes and still generated a better fit to anatomical ground truth than other methods using a full dataset. We could further minimize our gene lists for individual representations by applying supervised learning to the highest ranked elements to optimize their weighting. This two step approach allowed for high accuracy representations with a list of elements that can be curated and optimized for downstream applications, such as limiting the list to proteins with commercially available antibodies.

The anatomy of the wild type (WT) C57BL/6J mouse is well described, but much less so for knockout or transgenic mouse strains or for related species of interest. AMBA scale data acquisition is unlikely to be repeated, but sparse learning reduced the

genes needed to describe anatomy to a scale tractable by Multiplexed error-robust fluorescence in situ hybridization (MERFISH) or similar multiplexed spatial transcriptomic methods. What once took years of work could now be largely recapitulated in a single session (Lewis et al., 2021).

One obvious improvement that could be made to any sparse methods on spatial datasets would be consideration of neighboring voxels for unsupervised learning. In this study, spatial aspects of the dataset were implicit, with representations appearing compact and contiguous only by virtue of their fit to ground truth. Finer representations could potentially be achieved by adding a spatial convolution step to sparse learning, or using spatial metrics as the basis for a loss function. For nonspatial datasets, other inherent biology such as gene ontology could serve a similar role to leverage this versatile tool for representation learning with preserved gene information.

## METHODS:

### Allen Mouse Brain Atlas (AMBA)

The Allen brain Atlas project (http://mouse.brain-map.org) has provided a comprehensive set of ~20,000 gene expression profiles with cellular resolution in the male, 56-day old C57BL/6J mouse brain (Lein et al., 2007). Image data were collected using the in situ hybridization method in sagittally-oriented slices with 200 μm inter-slice resolution and 25 μm thickness. The expression patterns were replicated in the coronal sections for ~4,000 genes of high neurobiological interest (Ng et al., 2009). The expression patterns were reconstructed in 3D and registered to a Nissl stain-based

reference atlas (Allen Reference Atlas; ARA). The expression of a gene within 200 μm isotropic voxel is the average intensity of pixels in the pre-processed image called smoothed expression energy. Each 3D gene expression data consists of 67 × 41 × 58 (rostral-caudal, dorsal-ventral, left-right) spatially-matched volumes (Lein et al., 2007).

**Data Pre-Processing**

We followed the quality control measure implemented by (Bohland et al., 2010). The Pearson correlation coefficient was measured for each gene in coronal and sagittal experiments to find the higher-consistency dataset. Then 25% of genes with the lowest correlations were removed, resulting in a selection of 3,041 genes from which 2,941 still exist in the current version of the AMBA dataset, which we adopted for this study.

We extracted and concatenated the expression energy values in the brain for all dimensionality reduction methods to form a large 63,113 voxel × 2,941 gene matrix, $E_{(v,g)}$, where $v$ and $g$ denote voxels and genes. We then z-transformed the data for each gene so that the mean of each gene's expression across the voxels was zero, and the standard deviation was one.

**Representation Learning methods**

For PCA, KPCA, ICA, SPCA, and DLSC we used their implementation in the publicly available SciKit-Learn package in Python (Pedregosa et al., 2011)1). We selected 100 for the number of components in all methods. In the sparse methods, SPCA and DLSC, the amount of sparseness is adjustable by the coefficient of the $l_1$ penalty, given by the parameter $\alpha$, for which we tested different values of

$\alpha = \{0.1,\ 1,\ 10,\ 20\}$. The kernels used in KPCA were the quadratic and cubic polynomial, rbf, and sigmoid functions.

**Sparse Filtering (SFt)**

Sparse filtering is an unsupervised feature learning method that efficiently scales to handle large input dimensions. The single hyperparameter to tune is the number of features. SFt optimizes for population sparsity (a few non-zero features represent each sample), lifetime sparsity (each feature is active for a few samples), and high dispersal (features share similar statistics) (Zennaro and Chen, 2018).

The math proposed by (Ngiam et al., 2011) considers the feature distribution matrix indexed by j for features as rows and by i for samples as columns. In the corresponding feature distribution $f_j^{(i)}$ represents the $j^{th}$ feature value (rows) for the $i^{th}$ example (columns), resulting in $f_j^{(i)} = w_j^T x^{(i)}$. The first step is to normalize each feature by dividing it by its $l_2$norm across all examples $f\sim_j = f_j / \left\| f_j \right\|_2$. The features are then $l_2$ normalized across examples $f^{\wedge(i)} = f\sim^{(i)} / \left\| f\sim^{(i)} \right\|_2$. Then the $l_1$ penalty is used to optimize the features for sparsity so that in a dataset of M examples, the objective becomes: $minimize \sum_{i=1}^{M} \left\| f^{\wedge(i)} \right\|_1 = \sum_{i=1}^{M} \left\| \frac{f^{\sim(i)}}{\left\| f\sim^{(i)} \right\|_2} \right\|_1$ .

We implemented the open-source GitHub repository of Sparse Filtering in Python (https://github.com/jmetzen/sparse-filtering) in this study. This software package transposes the voxel × gene matrix, $E_{(v,\,g)}$, to be consistent with the Matlab code provided by (Ngiam et al., 2011). It also incorporated the soft-absolute activation

function and Limited-memory Broyden, Fletcher, Goldfarb, and Shanno (L-BFGS) minimizer. We chose 100 for the number of features to be consistent with the other learning methods.

**K-means Clustering**

After data representation using previously mentioned methods, we applied the unsupervised K-means clustering method. We used the publicly available K-means clustering implementation in SciKit-Learn (Pedregosa et al., 2011), and the number of clusters (K) ranged from 1 to 50 with the step of 1, and 50 to 550 with the step of 50.

**Similarity measures**

We chose Adjusted Mutual Information (AMI) and Adjusted Rand Index (ARI) in the SciKit-Learn package (Pedregosa et al., 2011) to measure the similarity of clustered representations with anatomy, where the adjusted versions of MI and RI account for the chance.

We developed a function to compute the DICE similarity coefficient to measure the spatial overlap of brain regions and representations (code example available on GitHub). It ranges from 0, for no overlap, to 1, showing a perfect overlap. The formula to calculate DSC for two regions (A and B) is:

$$DSC_{(A,B)} = 2\,(A \cap B) \,/\, (A + B)$$

where ∩ is the intersection (Zou et al., 2004).

**Logistic Regression**

For Sparse Filtering, we found the candidate pairs of brain regions and features by measuring the Dice coefficient of a given brain region against a thresholded feature.

We thresholded the features using K-means clustering, with the single feature of interest as input, and choosing the number of clusters at two. We compared both labels against the region of interest using the Dice coefficient, and kept the better of the two. We repeated this procedure for every brain region against all features, and kept the best region-feature combination.

We chose the brain regions CA1, CA3, and CP for logistic regressions, as they were among the top overlapping combinations for SFt and other methods. We determined the gene inputs for logistic regressions by identifying the largest ten weights by absolute value associated with a given feature produced by a transformation, and then finding the genes associated with these weights (**Table 1**). We then mean-centered and standard-scaled these ten genes, and prepared a shuffled five-fold cross-validation set. We trained a logistic regression from the normalized genes onto the region of interest, and reported the ROC-AUC score on the test set. Normalization, logistic regression, shuffling, cross-validation, and ROC-AUC score were all performed using their respective implementations in the scikit-learn library (Pedregosa et al., 2011).

**Sparsity Metrics**

Here we described the metrics used to measure sparsity and provided the code examples on the GitHub.

**Feature Sparsity**

To measure the feature sparsity, first, we found the number of active voxels in each feature by measuring the mean value of voxels in each feature and counting the

number of voxels with values above that. Then we used the ratio below to measure sparsity:

Feature Sparsity = Total number of voxels in the feature / Number of active voxels

**Weight Sparsity**

To measure the weight (gene) sparsity, first, we found the number of active genes in each feature transformation by measuring the mean value of genes in each feature and counting the number of genes with values above that. Then we used the ratio below to measure sparsity:

Weight (gene) Sparsity = Total number of genes / Number of active genes in the feature

**Connected Components**

We determined the number of connected components within a feature using the implementation of marching cubes in the scikit-image library (Pedregosa et al., 2011) to develop an adjacency matrix, and an algorithm that determines the number of connected components in a graph from its adjacency matrix available in the NetworkX library (Hagberg et al., 2008).

We reconstructed the representations generated by each method into the anatomical space, then applied the scikit-image implementation of marching cubes (Walt et al., 2014) to the resulting 3D array, using the mean value of the feature as the threshold level for the method. The output of this method is a set of vertices and faces that comprise a set of 3D surfaces. We constructed an adjacency matrix for the set of

vertices using the set of faces, and found the number of connected components by passing this adjacency matrix to the number_connected_components algorithm in the NetworkX library (Hagberg et al., 2008).

**Shannon Entropy**

We measured the Shannon entropy of a feature by binning out the continuous values of the feature into discrete levels, and treated each discrete value as though it were a unique symbol in the Shannon entropy formula:

$$H(x) = -\sum_{i=1}^{n} P(x_i)\, ln\, P(x_i)$$

where each $x_i$ in i=[1, n] is a discrete value of the binned feature.

**Spatial Entropy**

We developed a measurement of the spatial entropy of a feature using an extension of the grey level co-occurrence matrix method in use on 2D images. Pixel intensities present in an image are binned into discrete levels, and instances of co-occurrence of these levels is counted with specified spatial relationships as the elements of a matrix. We extended this method to a 3D image by including a three-dimensional relationship (Tsai et al., 2007) . Spatial entropy of an image can be measured by finding the Shannon entropy of this matrix, in which each (i,j) index of the matrix is as though it were a unique symbol in the Shannon entropy formula described previously.

**Data Availability**

The analysis pipeline is available for public use on GitHub:

https://github.com/BioProteanLabs/SFt_pipeline

**Supplementary material**

## AUTHOR CONTRIBUTION AND ACKNOWLEDGEMENTS:

MA: Set up the dataset, surveyed all algorithms presented, developed and ran performance metrics, generated figures and co-wrote the manuscript. CRS: Developed and ran performance metrics and logistic classifiers, generated figures and edited the manuscript. NR: Ran performance metrics generated figures and edited the manuscript. BBB: Managed the project and co-wrote the manuscript.

Thanks to Dr. Chris Plaisier for consultation on our metrics and Research Computing at Arizona State University for providing the high performance computing and data storage used to generate the analyses reported within this paper.

## REFERENCES:

Bell, A.J., Sejnowski, T.J., 1995. An Information-Maximization Approach to Blind Separation and Blind Deconvolution. Neural Computation 7, 1129–1159. https://doi.org/10.1162/neco.1995.7.6.1129

Ben-David, S., Haghtalab, N., 2014. Clustering in the Presence of Background Noise, in: Proceedings of the 31st International Conference on Machine Learning. Presented at the International Conference on Machine Learning, PMLR, pp. 280–288.

Bohland, J.W., Bokil, H., Pathak, S.D., Lee, C.-K., Ng, L., Lau, C., Kuan, C., Hawrylycz, M., Mitra, P.P., 2010. Clustering of spatial gene expression patterns in the mouse brain and

comparison with classical neuroanatomy. Methods, Spatial gene imaging in the brain 50, 105–112. https://doi.org/10.1016/j.ymeth.2009.09.001

Buccini, A., Pasha, M., Reichel, L., 2020. Modulus-based iterative methods for constrained ℓp-ℓq minimization. Inverse Problems 36, 084001. https://doi.org/10.1088/1361-6420/ab9f86

Ciortan, M., Defrance, M., 2021. Contrastive self-supervised clustering of scRNA-seq data. BMC Bioinformatics 22, 280. https://doi.org/10.1186/s12859-021-04210-8

Crescimanna, V., Graham, B., 2020. The Variational InfoMax AutoEncoder, in: 2020 International Joint Conference on Neural Networks (IJCNN). Presented at the 2020 International Joint Conference on Neural Networks (IJCNN), pp. 1–8. https://doi.org/10.1109/IJCNN48605.2020.9207048

Dudek, S.M., Alexander, G.M., Farris, S., 2016. Rediscovering area CA2: unique properties and functions. Nat Rev Neurosci 17, 89–102. https://doi.org/10.1038/nrn.2015.22

Gultepe, E., Makrehchi, M., 2018. Improving clustering performance using independent component analysis and unsupervised feature learning. Human-centric Computing and Information Sciences 8, 25. https://doi.org/10.1186/s13673-018-0148-3

Hagberg, A.A., Schult, D.A., Swart, P.J., 2008. Exploring Network Structure, Dynamics, and Function using NetworkX 5.

Hamilton, D., White, C., Rees, C., Wheeler, D., Ascoli, G., 2017. Molecular fingerprinting of principal neurons in the rodent hippocampus: a neuroinformatics approach. J Pharm Biomed Anal 144, 269–278. https://doi.org/10.1016/j.jpba.2017.03.062

Hjelm, R.D., Fedorov, A., Lavoie-Marchildon, S., Grewal, K., Bachman, P., Trischler, A., Bengio, Y., 2019. Learning deep representations by mutual information estimation and maximization. arXiv:1808.06670 [cs, stat].

Jiang, J., Wang, C., Qi, R., Fu, H., Ma, Q., 2020. scREAD: A Single-Cell RNA-Seq Database for Alzheimer's Disease. iScience 23, 101769. https://doi.org/10.1016/j.isci.2020.101769

Kobak, D., Berens, P., 2019. The art of using t-SNE for single-cell transcriptomics. Nat Commun

10, 5416. https://doi.org/10.1038/s41467-019-13056-x

Kohara, K., Pignatelli, M., Rivest, A.J., Jung, H.-Y., Kitamura, T., Suh, J., Frank, D., Kajikawa, K., Mise, N., Obata, Y., Wickersham, I.R., Tonegawa, S., 2014. Cell type–specific genetic and optogenetic tools reveal hippocampal CA2 circuits. Nat Neurosci 17, 269–279. https://doi.org/10.1038/nn.3614

Kolali Khormuji, M., Bazrafkan, M., 2016. A novel sparse coding algorithm for classification of tumors based on gene expression data. Med Biol Eng Comput 54, 869–876. https://doi.org/10.1007/s11517-015-1382-8

Lein, E.S., Hawrylycz, M.J., Ao, N., Ayres, M., Bensinger, A., Bernard, A., Boe, A.F., Boguski, M.S., Brockway, K.S., Byrnes, E.J., Chen, Lin, Chen, Li, Chen, T.-M., Chi Chin, M., Chong, J., Crook, B.E., Czaplinska, A., Dang, C.N., Datta, S., Dee, N.R., Desaki, A.L., Desta, T., Diep, E., Dolbeare, T.A., Donelan, M.J., Dong, H.-W., Dougherty, J.G., Duncan, B.J., Ebbert, A.J., Eichele, G., Estin, L.K., Faber, C., Facer, B.A., Fields, R., Fischer, S.R., Fliss, T.P., Frensley, C., Gates, S.N., Glattfelder, K.J., Halverson, K.R., Hart, M.R., Hohmann, J.G., Howell, M.P., Jeung, D.P., Johnson, R.A., Karr, P.T., Kawal, R., Kidney, J.M., Knapik, R.H., Kuan, C.L., Lake, J.H., Laramee, A.R., Larsen, K.D., Lau, C., Lemon, T.A., Liang, A.J., Liu, Y., Luong, L.T., Michaels, J., Morgan, J.J., Morgan, R.J., Mortrud, M.T., Mosqueda, N.F., Ng, L.L., Ng, R., Orta, G.J., Overly, C.C., Pak, T.H., Parry, S.E., Pathak, S.D., Pearson, O.C., Puchalski, R.B., Riley, Z.L., Rockett, H.R., Rowland, S.A., Royall, J.J., Ruiz, M.J., Sarno, N.R., Schaffnit, K., Shapovalova, N.V., Sivisay, T., Slaughterbeck, C.R., Smith, S.C., Smith, K.A., Smith, B.I., Sodt, A.J., Stewart, N.N., Stumpf, K.-R., Sunkin, S.M., Sutram, M., Tam, A., Teemer, C.D., Thaller, C., Thompson, C.L., Varnam, L.R., Visel, A., Whitlock, R.M., Wohnoutka, P.E., Wolkey, C.K., Wong, V.Y., Wood, M., Yaylaoglu, M.B., Young, R.C., Youngstrom, B.L., Feng Yuan, X., Zhang, B., Zwingman, T.A., Jones, A.R., 2007. Genome-wide atlas of gene expression in the adult mouse brain. Nature 445, 168–176.

https://doi.org/10.1038/nature05453

Lewis, S.M., Asselin-Labat, M.-L., Nguyen, Q., Berthelet, J., Tan, X., Wimmer, V.C., Merino, D., Rogers, K.L., Naik, S.H., 2021. Spatial omics and multiplexed imaging to explore cancer biology. Nat Methods 18, 997–1012. https://doi.org/10.1038/s41592-021-01203-6

Li, Y., Chen, H., Jiang, X., Li, X., Lv, J., Li, M., Peng, H., Tsien, J.Z., Liu, T., 2017. Transcriptome Architecture of Adult Mouse Brain Revealed by Sparse Coding of Genome-Wide In Situ Hybridization Images. Neuroinformatics 15, 285–295. https://doi.org/10.1007/s12021-017-9333-1

Liu, Z., 2020. Visualizing Single-Cell RNA-seq Data with Semisupervised Principal Component Analysis. Int J Mol Sci 21, E5797. https://doi.org/10.3390/ijms21165797

Ma, S.-X., Lim, S.B., 2021. Single-Cell RNA Sequencing in Parkinson's Disease. Biomedicines 9, 368. https://doi.org/10.3390/biomedicines9040368

Ng, L., Bernard, A., Lau, C., Overly, C.C., Dong, H.-W., Kuan, C., Pathak, S., Sunkin, S.M., Dang, C., Bohland, J.W., Bokil, H., Mitra, P.P., Puelles, L., Hohmann, J., Anderson, D.J., Lein, E.S., Jones, A.R., Hawrylycz, M., 2009. An anatomic gene expression atlas of the adult mouse brain. Nat Neurosci 12, 356–362. https://doi.org/10.1038/nn.2281

Ngiam, J., Chen, Z., Bhaskar, S., Koh, P., Ng, A., 2011. Sparse Filtering, in: Advances in Neural Information Processing Systems. Curran Associates, Inc.

Ortiz, C., Navarro, J.F., Jurek, A., Märtin, A., Lundeberg, J., Meletis, K., 2020. Molecular atlas of the adult mouse brain. Science Advances. https://doi.org/10.1126/sciadv.abb3446

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, É., 2011. Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research 12, 2825–2830.

Radic, T., Frieß, L., Vijikumar, A., Jungenitz, T., Deller, T., Schwarzacher, S.W., 2017. Differential Postnatal Expression of Neuronal Maturation Markers in the Dentate Gyrus of Mice and

Rats. Frontiers in Neuroanatomy 11.

Shannon, C.E., 1948. A mathematical theory of communication. The Bell System Technical Journal 27, 379–423. https://doi.org/10.1002/j.1538-7305.1948.tb01338.x

Torgerson, W.S., 1952. Multidimensional scaling: I. Theory and method. Psychometrika 17, 401–419. https://doi.org/10.1007/BF02288916

Tsai, F., Chang, C.-K., Rau, J.-Y., Lin, T.-H., Liu, G.-R., 2007. 3D Computation of Gray Level Co-occurrence in Hyperspectral Image Cubes, in: Yuille, A.L., Zhu, S.-C., Cremers, D., Wang, Y. (Eds.), Energy Minimization Methods in Computer Vision and Pattern Recognition, Lecture Notes in Computer Science. Springer, Berlin, Heidelberg, pp. 429–440. https://doi.org/10.1007/978-3-540-74198-5_33

Walt, S. van der, Schönberger, J.L., Nunez-Iglesias, J., Boulogne, F., Warner, J.D., Yager, N., Gouillart, E., Yu, T., 2014. scikit-image: image processing in Python. PeerJ 2, e453. https://doi.org/10.7717/peerj.453

Xiang, R., Wang, W., Yang, L., Wang, S., Xu, C., Chen, X., 2021. A Comparison for Dimensionality Reduction Methods of Single-Cell RNA-seq Data. Front Genet 12, 646936. https://doi.org/10.3389/fgene.2021.646936

Zennaro, F.M., Chen, K., 2018. Towards understanding sparse filtering: A theoretical perspective. Neural Netw 98, 154–177. https://doi.org/10.1016/j.neunet.2017.11.010

Zou, K.H., Warfield, S.K., Bharatha, A., Tempany, C.M.C., Kaus, M.R., Haker, S.J., Wells, W.M., Jolesz, F.A., Kikinis, R., 2004. Statistical Validation of Image Segmentation Quality Based on a Spatial Overlap Index. Acad Radiol 11, 178–189. https://doi.org/10.1016/S1076-6332(03)00671-8