Appendix for Hierarchically Clustered PCA and CCA via a Convex Clustering Penalty

Anonymous Author(s)
Affiliation
Address
email

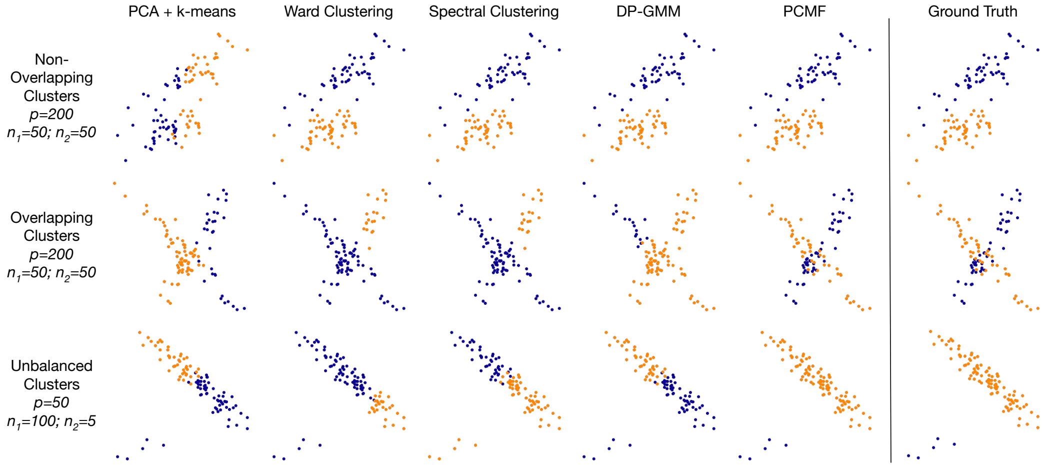

Figure 1: Examples of the three types of synthetic datasets considered in Supplementary Table 1 for comparing other standard clustering methods and PCMF. Note: Sample sizes and means are slightly different than in Supplementary Table 1, but the distributions are representative (plots show the 1st and 2nd variables plotted as a scatter plot and colored by class for two-class synthetic data).

1 PCMF empirical validation on synthetic data

1.1 PCMF comparisons to standard methods on synthetic data

We measure the clustering performance of the PCMF method against classical (PCA+KMeans, Ward, spectral, and DP-GMM) and state-of-the-art (Elastic Subspace Clustering [26] and gMADD [20, 18]) clustering methods (Supplementary Table 1). We test performance of PCMF solved with ADMM (PCMF-ADMM) and with PALS (PCMF-PALS) using either no nearest neighbors (N.N. denotes number of nearest neighbors) or $N.N. = 25$ We report the mean and standard deviation of the adjusted rand index (ARI) and normalized mutual information (NMI) over 10 randomly generated replicates of two-cluster data (examples of three types of synthetic data sets are given in Supplementary Figure 1), where a score of 1.0 indicates perfect recovery of cluster structure. We find that PCMF-ADMM and PCMF-PALS with $N.N. = 25$ outperforms the other clustering methods in almost all cases (Supplementary Table 1; bold indicates best cluster recovery), and is the only clustering method that performs well in all synthetic datasets across 10 runs). Further details of synthetic dataset generation are given in Appendix §10 below.

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

Table 1: Clustering accuracy of PCMF on 2-class data. We generate X from 2 distributions that differ in centroid and slope: non-overlapping clusters (centroids $\in \{-0.2, 0.2\}$), overlapping clusters (centroids $\in \{-0.05, 0.05\}$), or non-overlapping but unbalanced cluster size (centroids $\in \{-0.2, 0.2\}$). δ indicates the fraction of variables redundantly containing signal. $N.N.$ denotes number of nearest neighbors used in the convex clustering penalty.

	Non-overlapping $\delta = 0.5; N_1, N_2 = 50$		Non-overlapping $\delta = 0.2; n_1, n_2 = 50$		Overlapping $\delta = 0.5; n_1, n_2 = 50$		Overlapping $\delta = 0.2; N_1, N_2 = 50$		Unbalanced $\delta = 0.5; n_1 = 80, n_2 = 20$		Unbalanced $\delta = 0.2; N_1 = 80, N_2 = 20$	
	ARI	NMI	ARI	NMI	ARI	NMI	ARI	NMI	ARI	NMI	ARI	NMI
					$p = 200$							
PCA+KMeans	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	0.06 ± 0.07	0.10 ± 0.11	**1.0 ± 0.0**	**1.0 ± 0.0**	0.155 ± 0.0	0.275 ± 0.0	0.155 ± 0.0	0.275 ± 0.0
Ward	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	0.31 ± 0.36	0.40 ± 0.32	**1.0 ± 0.0**	**1.0 ± 0.0**	0.16 ± 0.0	0.28 ± 0.0	0.16 ± 0.0	0.28 ± 0.0
Spectral	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	0.16 ± 0.0	0.28 ± 0.0	0.16 ± 0.0	0.28 ± 0.0
DP-GMM	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	0.25 ± 0.10	0.33 ± 0.11	**1.0 ± 0.0**	**1.0 ± 0.0**	0.16 ± 0.0	0.28 ± 0.0	0.16 ± 0.0	0.28 ± 0.0
Elastic Subspace	0.162 ± 0.09	0.25 ± 0.13	0.06 ± 0.10	0.09 ± 0.14	0.80 ± 0.09	0.75 ± 0.09	0.09 ± 0.08	0.14 ± 0.11	0.05 ± 0.04	0.05 ± 0.03	0.03 ± 0.03	0.04 ± 0.03
gMADD	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	0.0 ± 0.01	0.02 ± 0.02	0.61 ± 0.48	0.61 ± 0.48	0.16 ± 0.01	0.27 ± 0.02	0.16 ± 0.0	0.28 ± 0.0
PCMF-ADMM; No N.N.	0.01 ± 0.03	0.03 ± 0.04	0.0 ± 0.01	0.01 ± 0.01	0.11 ± 0.30	0.12 ± 0.30	0.0 ± 0.01	0.01 ± 0.01	0.01 ± 0.025	0.03 ± 0.04	0.0 ± 0.01	0.01 ± 0.01
PCMF-PALS; No N.N.	0.54 ± 0.46	0.54 ± 0.45	0.60 ± 0.49	0.60 ± 0.49	0.09 ± 0.18	0.11 ± 0.18	0.14 ± 0.26	0.13 ± 0.22	0.54 ± 0.46	0.54 ± 0.45	0.60 ± 0.49	0.60 ± 0.49
PCMF-ADMM; N.N.=25	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**
PCMF-PALS; N.N.=25	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	0.62 ± 0.45	0.64 ± 0.42	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**
					$p = 2000$							
PCA+KMeans	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	0.05 ± 0.08	0.09 ± 0.12	**1.0 ± 0.0**	**1.0 ± 0.0**	0.16 ± 0.0	0.28 ± 0.0	0.16 ± 0.0	0.28 ± 0.0
Ward	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	0.24 ± 0.26	0.34 ± 0.22	**1.0 ± 0.0**	**1.0 ± 0.0**	0.16 ± 0.0	0.28 ± 0.0	0.16 ± 0.0	0.28 ± 0.0
Spectral	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	0.13 ± 0.05	0.23 ± 0.09	0.16 ± 0.0	0.28 ± 0.0
DP-GMM	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	0.35 ± 0.32	0.41 ± 0.30	**1.0 ± 0.0**	**1.0 ± 0.0**	0.16 ± 0.0	0.28 ± 0.0	0.16 ± 0.0	0.28 ± 0.0
Elastic Subspace	0.08 ± 0.11	0.13 ± 0.16	0.09 ± 0.12	0.14 ± 0.16	0.39 ± 0.08	0.32 ± 0.06	0.11 ± 0.06	0.20 ± 0.10	0.06 ± 0.04	0.05 ± 0.03	0.04 ± 0.043	0.04 ± 0.04
gMADD	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	0.0 ± 0.01	0.01 ± 0.02	0.80 ± 0.39	0.81 ± 0.38	0.16 ± 0.01	0.27 ± 0.02	0.16 ± 0.0	0.28 ± 0.0
PCMF-ADMM; No N.N.	0.41 ± 0.49	0.43 ± 0.47	0.0 ± 0.0	0.0 ± 0.01	0.0 ± 0.0	0.02 ± 0.02	0.40 ± 0.49	0.40 ± 0.49	0.40 ± 0.49	0.43 ± 0.47	0.0 ± 0.01	0.01 ± 0.02
PCMF-PALS; No N.N.	0.43 ± 0.45	0.42 ± 0.45	0.44 ± 0.46	0.44 ± 0.45	0.06 ± 0.11	0.08 ± 0.15	0.14 ± 0.28	0.14 ± 0.27	0.43 ± 0.45	0.42 ± 0.45	0.44 ± 0.46	0.44 ± 0.45
PCMF-ADMM; N.N.=25	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**
PCMF-PALS; N.N.=25	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**

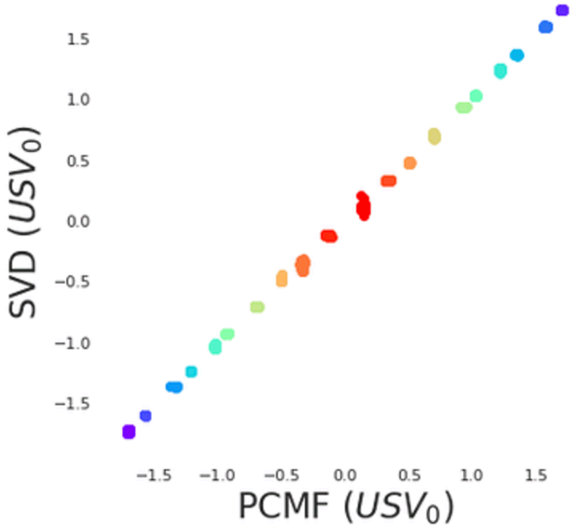

Figure 2: Comparison of PCMF embeddings (PCMF USV_0) to per cluster tSVD (SVD USV_0). We calculated tSVD within each cluster using the ground truth clustering. We find that PCMF rank-1 matrices per cluster are highly similar to those found using tSVD within each ground truth cluster (here the two estimates are shown scatter plotted against one another with colors indicating the 20 clusters, as in Figure 1 of the main text).

Table 2: Comparison of PCMF embeddings to per cluster tSVD. Using the ground truth clustering, we calculated the tSVD within each cluster to find the within cluster singular value S and singular vectors U and V. We compared these to the estimates from PCMF. S, U, and V for the 20 clusters estimated by PCMF are highly similar to the S, U, and V calculated by SVD within ground truth clusters, indicating that PCMF is successfully approximating within-cluster tSVD estimates (uncentered PCA).

Cluster	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
S (PCMF)	122.55	122.50	113.04	112.99	96.58	96.53	87.29	87.01	73.04	73.09	66.00	65.99	49.55	49.35	35.39	35.37	23.76	23.52	9.37	9.456
S (SVD)	122.55	122.51	113.05	113.00	96.59	96.54	87.23	87.03	73.04	73.09	66.01	66.00	49.59	49.38	35.41	35.38	23.87	23.56	9.42	9.54
$U_{0:20}(PCMF)$	0.14	0.14	0.14	0.14	0.14	0.14	0.14	0.14	0.14	0.14	0.14	0.14	0.14	0.14	0.14	0.14	0.14	0.14	0.14	0.14
$U_{0:20}(SVD)$	0.14	0.18	0.14	0.14	0.14	0.14	0.14	0.14	0.14	0.14	0.14	0.14	0.14	0.14	0.13	0.14	0.14	0.14	0.14	0.14
$V_{0,0:20}(PCMF)$	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10
$V_{0,0:20}(SVD)$	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10

1.2 Comparison of PCMF embeddings and tSVD on synthetic data with ground-truth clusters

We note that although the full data matrix X is centered in the work presented here, by construction in most cases the clusters that make up the rows of X will be uncentered (except in the "overlapping" cluster case with all clusters having mean zero). This results in the cluster tSVD estimates representing uncentered PCA estimates in general, and thus the left singular vectors for tSVD/PCA fit to the true classes will represent a mix of the within-cluster direction of maximum variance and the cluster mean (to the extent that they both contribute in the total sum-of-squares). Still, our PCMF-ADMM algorithm results in estimates very similar to those resulting from tSVD/uncentered-PCA on the known ground-truth clusters (Supplementary Figure 2 and Supplementary Table 2). Interestingly, our relaxed PCMF-PALS formulation qualitatively seems to capture both the means of the uncentered clusters and, at values of λ that perform well for correct cluster recovery, spread along the direction of (centered) within-cluster variance (see Supplementary Figure 3). Further investigation of this phenomenon is left to future work.

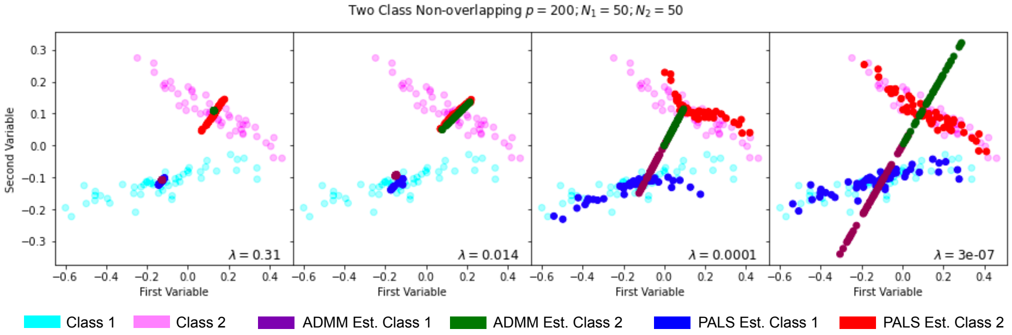

Figure 3: Example of PCMF-ADMM vs. PCMF-PALS estimates (both fit with $N.N. = 25; \rho = 1.5; \gamma = 2.0; K = 5$ ADMM iterations) evolving as λ decreases (4 solutions along the path shown for a two-class, balanced problem with non-overlapping classes with $N_1 = 50; N_2 = 50$ and $p = 200$). Note: The PCMF-PALS relaxation is more flexible, as it allows different estimates for the PCA loadings (rows of V) to emerge per-cluster due to problem formulation and relaxation (see $\S 4 - 7$ for more details).

2 Societal Impact

We introduce a new interpretable method to decompose latent structure in clustered data, which has many potential applications including as a tool to decode multiomic and neurobiological heterogeneity underlying clinical diseases such as cancer, depression, or Autism Spectrum Disorder (ASD). Our method is also rather general, in that it can be applied to different datasets with a single or multi-view to learn latent covariance structure that exists within clusters of data. As with any clustering algorithm, the potential for negative societal impact depends on the dataset and intentions of the analysis; while we do not believe PCMF poses a clear potential negative social impact, it will be critical for the use of such clustering algorithms to be on datasets that following ethical guidelines in terms of data collection and data use. The robustness and explainability provided by PCMF for unbalanced, small clusters could enable positive or negative impact on underrepresented classes depending on its application.

3 Derivation of Algorithm 1: PCMF-ADMM

Denote the PCMF optimization problem for truncated SVD of rank r:

$$
\begin{aligned}
\underset{\widehat{X},G,U_r,S_r,V_r}{\text{minimize}}\ & \frac{1}{2}\|X-\widehat{X}\|_F^2 + \lambda P_{\mathbf{w},q}(G) \\
\text{subject to}\ & \widehat{X} - U_r S_r V_r^T = 0,\ U_r^T U_r = V_r^T V_r = I_r, \\
& S_r = \text{diag}(s_1,\ldots,s_r),\ s_1 \geq \ldots \geq s_r > 0, \\
& G - D\widehat{X} = 0.
\end{aligned}
$$

For simplicity we suppress the rank subscripts letting $U = U_r,\ S = S_r,\ V = V_r$ in the following. We use the multi-convex scaled form of ADMM (see [1] and [11]), yielding ADMM updates from the augmented Lagrangian with penalty parameter $\rho > 0$:

$$
\begin{array}{rcl}
\widehat{X}^{k+1} & \leftarrow & \underset{\widehat{X}\in\mathbb{R}^{N\times p}}{\text{argmin}}\ \frac{1}{2}\|X-\widehat{X}\|_F^2 + \frac{\rho}{2}\|U^kS^kV^k - Z_2^k\|_F^2 + \frac{\rho}{2}\|D\widehat{X} - G^k - Z_1^k\|_F^2 \\
G^{k+1} & \leftarrow & \underset{G\in\mathbb{R}^{|\mathcal{E}|\times p}}{\text{argmin}}\ \lambda P_{\mathbf{w},q}(G) + \frac{\rho}{2}\|D\widehat{X}^{k+1} - G + Z_1^k\|_F^2 \\
(U^{k+1},S^{k+1},V^{k+1}) & \leftarrow & \underset{\substack{U\in\mathbb{R}^{N\times r},\ V\in\mathbb{R}^{p\times r}\ ,U^TU=V^TV=I_r \\ S=\text{diag}(s_1,\ldots,s_r),\ s_1\geq\cdots\geq s_r>0}}{\text{argmin}}\ \|\widehat{X}^{k+1} - USV^T + Z_2^k\|_F^2 \\
Z_1^{k+1} & \leftarrow & Z_1^k + D\widehat{X}^{k+1} - G^{k+1} \\
Z_2^{k+1} & \leftarrow & Z_2^k + \widehat{X}^{k+1} - U^{k+1}S^{k+1}V^{k+1,T}.
\end{array}
$$

Note that the G^{k+1} update can be re-expressed using the proximal operator for $P_{\mathbf{w},q}(G)$, since

$$
\begin{aligned}
\underset{G\in\mathbb{R}^{|\mathcal{E}|\times p}}{\text{argmin}}\ \lambda P_{\mathbf{w},q}(G) + \frac{\rho}{2}\|D\widehat{X}^{k+1} - G + Z_1^k\|_F^2 &= \underset{G\in\mathbb{R}^{|\mathcal{E}|\times p}}{\text{argmin}}\ \frac{\lambda}{\rho}P_{\mathbf{w},q}(G) + \frac{\rho}{2}\left\|G - \left(D\widehat{X}^{k+1} + Z_1^k\right)\right\|_F^2 \\
&= \text{prox}_{\frac{\lambda}{\rho}P_{\mathbf{w},q}(G)}\left(D\widehat{X}^{k+1} + Z_1^k\right).
\end{aligned}
$$

Further, due to the Eckhart-Young Theorem [5], the solution to the $(U^{k+1},S^{k+1},V^{k+1})$ update can be expressed as simply the SVD of $\widehat{X}^{k+1} + Z_2^k$. Further, writing out the the primal update, taking the gradient with respect to $\widehat{X}$ and setting the gradient equal to zero yields updates:

$$
\widehat{X}^{k+1} \leftarrow (I + \rho I + D^TD)^{-1}(X + \rho D^T(G^k - Z_1^k) + \rho(U^kS^kV^k - Z_2^k)).
$$

Finally, letting $LL^T = I + \rho I + D^TD$, we obtain the updates in Algorithm 1. See [11] for proof of convergence of a constrained NMF problem (Algorithm 2) equivalent to our problem in structure that shows that for such problems, the algorithm converges globally to the KKT conditions for $\rho > 1$.

4 Derivation of Algorithm 2: PCMF-PALS relaxation

Consider the PCMF optimization problem:

$$
\begin{aligned}
\underset{\widehat{X},U_r,S_r,V_r}{\text{minimize}}\ & \frac{1}{2}\|X-\widehat{X}\|_F^2 + \lambda\sum_{i<j} w_{ij}\|\widehat{X}_{i\cdot} - \widehat{X}_{j\cdot}\|_q \\
\text{subject to}\ & \widehat{X} - U_rS_rV_r^T = 0, \\
& U_r^TU_r = V_r^TV_r = I_r,\ S_r = \text{diag}(s_1,\ldots,s_r), \\
& s_1 \geq s_2 \geq \cdots \geq s_r > 0,
\end{aligned}
$$

Algorithm 1 PCMF-ADMM

```
Input: data $X$, decreasing path $\{\lambda\}$, weights $\mathbf{w}$
Notation: rank $r$, iteration $k$, norm $q \in \{1, 2\}$, $\rho \geq 1$
$G^0 \leftarrow Z_1^0 \leftarrow DX$; $\widehat{X} \leftarrow Z_2^0 \leftarrow \bar{X}$; $(U_r^0, S_r^0, V_r^0) \leftarrow \mathrm{SVD}_r(\widehat{X})$; $L = \mathrm{chol}(I + \rho I + \rho D^T D)$
for $\lambda \in \{\lambda\}$ do
    for $k = 1, \ldots, K$ do
        $\widehat{X}^{k+1} \leftarrow L^{-T} L^{-1}\big(X + \rho D^T(G^k - Z_1^k) + \rho(U_r^k S_r^k V_r^{kT} - Z_2^k)\big)$
        $G^{k+1} \leftarrow \mathrm{prox}_{\lambda/\rho P_{\mathbf{w},q}(\cdot,\cdot)}(D\widehat{X}^{k+1} + Z_1^k)$
        $(U_r^{k+1}, S_r^{k+1}, V_r^{k+1}) \leftarrow \mathrm{SVD}_r(\widehat{X}^{k+1} + Z_2^k)$
        $Z_1^{k+1} \leftarrow Z_1^k + D^T \widehat{X}^{k+1} - G^{k+1}$
        $Z_2^{k+1} \leftarrow Z_2^k + \widehat{X}^{k+1} - U_r^{k+1}, S_r^{k+1}, V_r^{k+1}$
    end for
    Save current path solutions: $\widehat{X}_\lambda \leftarrow \widehat{X}^K$, $G_\lambda \leftarrow G^K$; $(U_{r,\lambda}, S_{r,\lambda}, V_{r,\lambda}) \leftarrow (U_r^K, S_r^K, V_r^K)$
    Initialize for next path solution: $\widehat{X}^0 \leftarrow \widehat{X}^K$, $G^0 \leftarrow G^K$; $(U_r^0, S_r^0, V_r^0) \leftarrow (U_r^K, S_r^K, V_r^K)$
end for
return pathwise solutions $\{\widehat{X}_\lambda\}, \{G_\lambda\}, \{U_{r,\lambda}\}, \{S_{r,\lambda}\}, \{V_{r,\lambda}\}$
```

we note that the first part of the objective can be expanded as:

$$\begin{aligned}
\frac{1}{2}\|X - \widehat{X}\|_F^2 &= \mathrm{tr}\left((X - \widehat{X})^T(X - \widehat{X})\right) \\
&= \|X\|_F^2 - \mathrm{tr}\left(\widehat{X}^T X\right) + \|\widehat{X}\|_F^2 \\
&= \|X\|_F^2 - \mathrm{tr}\left(V_r S_r U_r^T X\right) + \|V_r S_r U_r^T U_r S_r V_r\|_F^2 \\
&= \|X\|_F^2 - \mathrm{tr}\left(S_r U_r^T X V_r\right) + \sum_{m=1}^{r} s_m^2 \\
&= \|X\|_F^2 - \sum_{m=1}^{r} s_m \mathbf{u}_m^T X \mathbf{v}_m + \sum_{m=1}^{r} s_m^2,
\end{aligned}$$

where we note that $\mathbf{v}_m$ and $\mathbf{u}_m$ denote column vectors here, and that the final two equivalences use the fact that $U^T U = I_r,\ V^T V = I_r$. In the rank-1 case, and letting $u = \mathbf{u}_1,\ \mathbf{v} = \mathbf{v}_1$ now be row vectors (as in the main text) and denoting the first singular value $d = s_1$**,** this can be rewritten

$$\begin{aligned}
&\underset{\widehat{X}, d, \mathbf{u}, \mathbf{v}}{\text{minimize}}\ \|X\|_F^2 - d\mathbf{u}X\mathbf{v}^T + d^2 + \lambda \sum_{i<j} w_{ij} \|\widehat{X}_{i\cdot} - \widehat{X}_{j\cdot}\|_q \\
&\text{subject to } \widehat{X}_{i\cdot} = d u_i \mathbf{v},\ \|\mathbf{u}\|_2^2 = 1,\ \|\mathbf{v}\|_2^2 = 1,\ d > 0.
\end{aligned}$$

By considering at the gradient of the objective, we can see this problem has the same solution as the following problem [25]:

$$\begin{aligned}
&\underset{\widehat{X}, d, \mathbf{u}, V}{\text{minimize}}\ - d\mathbf{u}X\mathbf{v}^T + \lambda \sum_{i<j} w_{ij} \|\widehat{X}_{i\cdot} - \widehat{X}_{j\cdot}\|_q \\
&\text{subject to } \widehat{X}_{i\cdot} = d u_i \mathbf{v},\ \|\mathbf{u}\|_2^2 = 1,\ \|\mathbf{v}\|_2^2 = 1,\ d > 0.
\end{aligned}$$

This formulation does not allow our rank-1 approximation to approximate every element of X with a corresponding element of $\widehat{X}$ as required by the convex clustering formulation, however. To remedy this, we introduce the overparameterization (now letting $\mathbf{v}_i = V_{i\cdot}$ and without loss of generality setting $d = 1$):

$$\begin{aligned}
&\underset{\widehat{X}, \mathbf{u}, V}{\text{minimize}}\ \sum_i u_i X_{i\cdot} \mathbf{v}_i^T + \lambda \sum_{i<j} w_{ij} \|\widehat{X}_{i\cdot} - \widehat{X}_{j\cdot}\|_q \\
&\text{subject to } \widehat{X}_{i\cdot} = u_i \mathbf{v}_{i\cdot},\ \|\mathbf{u}\|_2^2 = 1,\ \|\mathbf{v}_i\|_2^2 = 1,\ , i = 1, \ldots, N,
\end{aligned}$$

where each row of $\widehat{X}_i$ is now approximated by a potentially unique value $\widehat{X}_{i\cdot} = u_i \mathbf{v}_{i\cdot}$, just as in the convex clustering problem.

Next, we note that due to the quadratic equality constraints $\|\mathbf{u}\|_2^2 = 1,\ \|\mathbf{v}_i\|_2^2 = 1,\ , i = 1, \ldots, N$, the following relationships hold:

$$\frac{1}{2}\|\mathbf{x}_i \mathbf{v}_i^T - u_i\| = \frac{1}{2}\mathbf{v}_i \mathbf{x}_i^T \mathbf{x}_i \mathbf{v}_i^T - u_i \mathbf{x}_i \mathbf{v}_i^T + u_i^2 = -u_i \mathbf{x}_i \mathbf{v}_i^T + {\frac{1}{2}}^2 \mathbf{x}_i^T \mathbf{x}_i + 1,$$

and

$$\frac{1}{2}\|u_i \mathbf{x}_i - \mathbf{v}_i\| = \frac{1}{2}u_i^2 \mathbf{x}_i^T \mathbf{x}_i - u_i \mathbf{x}_i \mathbf{v}_i^T + \mathbf{v}_i^T \mathbf{v}_i = -u_i \mathbf{x}_i \mathbf{v}_i^T + \frac{1}{2}\mathbf{x}_i^T \mathbf{x}_i + 1,$$

and therefore it follows that (letting $y_i^{uk} = \mathbf{x}_i \mathbf{v}_i^{kT}$ and $\mathbf{y}_i^{vk} = u_i^k \mathbf{x}_i$) the updates:

$$\begin{aligned}
\mathbf{u}^{k+1} &\leftarrow \underset{\mathbf{u}}{\text{argmin}} \sum_{i=1}^{N} -u_i \mathbf{x}_i \mathbf{v}_i^{kT} + \lambda P_{\mathbf{w},q}(\mathbf{u}, V^k) \text{ subject to } \|\mathbf{u}\|_2^2 = 1, \\
\{\mathbf{v}_i\}^{k+1} &\leftarrow \underset{\{\mathbf{v}_i\}}{\text{argmin}} \sum_{i=1}^{N} -u_i^k \mathbf{x}_i \mathbf{v}_i^T + \lambda P_{\mathbf{w},q}(\mathbf{u}^{k+1}, V),\ \text{subject to } \|\mathbf{v}_i\|_2^2 = 1,
\end{aligned}$$

for $i = 1, \ldots, N$, have the same solutions as the updates:

$$\begin{aligned}
\mathbf{u}^{k+1} &\leftarrow \underset{\mathbf{u}}{\text{argmin}} \sum_{i=1}^{N} \|y_i^{uk} - u_i\|_2^2 + \lambda P_{\mathbf{w},q}(\mathbf{u}, V^k) \text{ subject to } \|\mathbf{u}\|_2^2 = 1, \\
\{\mathbf{v}_i\}^{k+1} &\leftarrow \underset{\{\mathbf{v}_i\}}{\text{argmin}} \sum_{i=1}^{N} \|y_i^{vk} - \mathbf{v}_i\|_2^2 + \lambda P_{\mathbf{w},q}(\mathbf{u}^{k+1}, V) \text{ subject to } \|\mathbf{v}_i\|_2^2 = 1,
\end{aligned}$$

for $i = 1, \ldots, N$. To enforce the quadratic equality constraints, we use proximal projection [17] updates following K iterations of the $\mathbf{u}$ and $\mathbf{v}$ updates, projecting onto the squared L2 unit ball associated with the equality constraints as intermediate steps in the algorithm. Thus if $K = 1$ we would have:

$$\begin{aligned}
\mathbf{u}^{k+\frac{1}{2}} &\leftarrow \underset{\mathbf{u}}{\text{argmin}} \sum_{i=1}^{N} \|y_i^{uk} - u_i\|_2^2 + \lambda P_{\mathbf{w},q}(\mathbf{d}^k, \mathbf{u}, V^k), \\
\mathbf{u}^{k+1} &\leftarrow \text{prox}_{\|\cdot\|_2^2}(\mathbf{u}^{k+\frac{1}{2}}) \\
\{\mathbf{v}_i\}^{k+\frac{1}{2}} &\leftarrow \underset{\{\mathbf{v}_i\}}{\text{argmin}} \sum_{i=1}^{N} \|y_i^{vk} - \mathbf{v}_i\|_2^2 + \lambda P_{\mathbf{w},q}(\mathbf{d}^k, \mathbf{u}^{k+1}, V), \\
\{\mathbf{v}_i\}^{k+1} &\leftarrow \text{prox}_{\|\cdot\|_2^2}(\mathbf{v}_i^{k+\frac{1}{2}}), i = 1, \ldots, N
\end{aligned}$$

where

$$\text{prox}_{\|\cdot\|_2^2}(\mathbf{a}) = \begin{cases} \frac{\mathbf{a}}{\|\mathbf{a}\|_2^2} & \text{if } \|\mathbf{a}\|_2^2 > 1 \\ \mathbf{a} & \text{if } \|\mathbf{a}\|_2^2 \leq 1. \end{cases}$$

We find that by relaxing $P_{\mathbf{w},q}(\mathbf{u}, V)$ to $Q_{\mathbf{w},q}^{\mathbf{u}}(\mathbf{u})$ and $Q_{\mathbf{w},q}^{V}(V)$ in or iterations, there is no need to recompute the difference matrix for each penalty, allowing the Cholesky factorization associated with that matrix to be cached to speed up computations (often significantly) and still yielding good clustering performance (see Experiments). We further note that the PCMF-PALS update allow both $\mathbf{u}$ and V to take unique values in the iterative updates, further explaining the additional fitting flexibility of this relaxed problem.

Both updates are standard convex clustering problems, solvable a number of ways (we use ADMM updates that are easily incorporated into an Algorithmic Regularization scheme; in particular we use the updates from [24]—see their Appendix Algorithms A1 and A2 and our supplementary Algorithms 2 and 3 below). Putting these together, we obtain main text Algorithm 2. Finally, for additional reasons why such multi-block algorithms may be advantageous, see [8].

5 Derivation of Algorithm 3: Pathwise Clustered Canonical Correlation Analysis (P3CA)

Letting $\mathbf{u} \in \mathbb{R}^{p_X}$ and $\mathbf{v} \in \mathbb{R}^{p_Y}$ be column vectors of coefficients we wish to estimate, then for two data matrices with observations in the rows $X \in \mathbb{R}^{N \times p_X}$, $Y \in \mathbb{R}^{N \times p_Y}$ we can write the rank-1 canonical correlation analysis (CCA) problem [12]:

$$\underset{\mathbf{u},\mathbf{v}}{\text{maximize}} \ \mathbf{u}^T X^T Y \mathbf{v} \text{ subject to } \mathbf{u}^T X^T X \mathbf{u} = 1, \ \mathbf{v}^T Y^T Y \mathbf{v} = 1.$$

Following previous work in $p > N$ problems, we treat the covariance matrices in this problem as diagonal [4, 23, 25] , and let $\Sigma = X^T Y$ resulting in the simplified problem:

$$\underset{\mathbf{u},\mathbf{v}}{\text{maximize}} \ \mathbf{u}^T \Sigma \mathbf{v} \text{ subject to } \|\mathbf{u}\|_2^2 = 1, \ \|\mathbf{v}\|_2^2 = 1.$$

In order to generalize this to convex clustering, we once again must introduce an overparameterization to allow (when $\lambda \to 0$) one unique parameter for each element of the matrix we are trying to approximate (in this case Σ). We do this by constructing the outer product matrices of the rows of X and Y as $\Sigma_i = X_{i\cdot}^T Y_{i\cdot} \in \mathbb{R}^{p_X \times p_Y}$, and then letting the row vectors $\mathbf{u}_i = U_{i\cdot}$ and $\mathbf{v}_i = V_{i\cdot}$ and penalty function $Q_{\mathbf{w},q}(A) = \sum_{(i,j)\in\mathcal{E}_A} w_{ij} \|\mathbf{a}_i - \mathbf{a}_j\|_q$ defining the P3CA problem as:

$$\begin{aligned} &\underset{U,V}{\text{maximize}} \ \sum_{i=1}^{N} \mathbf{u}_i \Sigma_i \mathbf{v}_i^T + \lambda Q_{\mathbf{w},q}(U) + \lambda Q_{\mathbf{w},q}(V) \\ &\text{subject to } \|\mathbf{u}_i\|_2^2 = 1, \ \|\mathbf{v}_i\|_2^2 = 1, \ i = 1, \ldots, N. \end{aligned}$$

Following similar logic as we did for Algorithm 2 above yields updates:

$$\begin{aligned}
\mathbf{z}_{\mathbf{u},i}^{k+1} &\leftarrow \Sigma_i \mathbf{v}_i^{kT} \ (\Sigma_i = X_{i\cdot}^T Y_{i\cdot} \in \mathbb{R}^{p_X \times p_Y}) \\
\{\mathbf{u}^{k+\frac{1}{2}}\} &\leftarrow \underset{\mathbf{u}}{\text{argmin}} \sum_{i=1}^{N} \|\mathbf{z}_{\mathbf{u},i}^{k+1} - \mathbf{u}_i\|_2^2 + \lambda Q_{\mathbf{w},q}(U) \\
\{\mathbf{u}^{k+1}\} &\leftarrow \text{prox}_{\|\cdot\|_2^2}(\mathbf{u}_i^{k+\frac{1}{2}}), i = 1, \ldots, N \\
\mathbf{z}_{\mathbf{v},i}^{k+1} &\leftarrow \Sigma_i^T \mathbf{u}_i^{k+1,T} \ (\Sigma_i^T = Y_{i\cdot} X_{i\cdot}^T \in \mathbb{R}^{p_Y \times p_X}) \\
\{\mathbf{v}_i\}^{k+\frac{1}{2}} &\leftarrow \underset{\{\mathbf{v}_i\}}{\text{argmin}} \sum_{i=1}^{N} \|\mathbf{z}_{\mathbf{v},i}^{k+1} - \mathbf{v}_i\|_2^2 + \lambda Q_{\mathbf{w},q}(V) \\
\{\mathbf{v}_i\}^{k+1} &\leftarrow \text{prox}_{\|\cdot\|_2^2}(\mathbf{v}_i^{k+\frac{1}{2}}), i = 1, \ldots, N
\end{aligned}$$

and once again noting that the $\{\mathbf{u}^{k+\frac{1}{2}}\}$ and $\{\mathbf{v}^{k+\frac{1}{2}}\}$ updates are convex clustering problems solvable using plugin algorithm CONVEXCLUSTER, we arrive at the updates in Algorithm 3 in the main text.

6 CONVEXCLUSTER Algorithm

We remark in the manuscript that to solve the convex clustering problem we use the ADMM approach of [24] — see their Appendix Algorithm A1 and A2 for details and Supplementary Algorithm 2 and 3 below. We choose this ADMM approach in particular as it is efficient and amenable to running for just a few ADMM iterations at each step inside our own iterative algorithms, allowing us to apply Algorithmic Regularization along the path of solutions.

Algorithm 2 CONVEXCLUSTER$(\mathbf{y}^k_{\mathbf{u},\lambda}, \mathbf{u}^k_\lambda)$ u Update

```
Input: Auxiliary variable $\mathbf{y}^k_{\mathbf{u}}$, u update $\mathbf{u}^k$
Notation: iteration $\mathbf{k}$, $\rho > 0$
Initialize (if k = 0):
  $L = \text{chol}(I + \rho I + \rho D^T D)$
  $W_2 = Z_2 = D\mathbf{y_u}^0$
ADMM:
  $\mathbf{u}^{k+\frac{1}{2}} = L^{-T} L^{-1}\big(\mathbf{y_u}^k + \rho D^T ({W_2}^k - {Z_2}^k)\big)$
  ${W_2}^{k+1} = \text{prox}_{\lambda/\rho, \|\cdot\|_2^2}(D\mathbf{u}^{k+\frac{1}{2}} + {Z_2}^k)$
  ${Z_2}^{k+1} = {Z_2}^k + D\mathbf{u}^{k+\frac{1}{2}} - {W_2}^{k+1}$
  return $\mathbf{u}_\lambda^{k+\frac{1}{2}}$
```

Algorithm 3 CONVEXCLUSTER$(Y^k_{\mathbf{v},\lambda}, \mathbf{V}^k_\lambda, \lambda, \mathbf{w}, q)$ V Update

```
Input: Auxiliary variable $\mathbf{y}^k_{\mathbf{v}}$, V update $\mathbf{V}^k$, penalty $\lambda$, weights $\mathbf{w}$
Notation: iteration $\mathbf{k}$, $\rho > 0$
Initialize (if k = 0):
  $L = \text{chol}(I + \rho I + \rho D^T D)$
  $W_1 = Z_1 = DX$
ADMM:
  $\mathbf{V}^{k+\frac{1}{2}} = L^{-T} L^{-1}\big(\mathbf{Y_v}^k + \rho D^T ({W_1}^k - {Z_1}^k)\big)$
  ${W_1}^{k+1} = \text{prox}_{\lambda/\rho, \|\cdot\|_2^2}(D\mathbf{V}^{k+\frac{1}{2}} + {Z_1}^k; \mathbf{w})$
  ${Z_1}^{k+1} = {Z_1}^k + D\mathbf{V}^{k+\frac{1}{2}} - {W_1}^{k+1}$
  return $\mathbf{V}_\lambda^{k+\frac{1}{2}}$
```

7 Algorithmic Regularization approaches

Note that in Algorithms 1-3, it is possible to obtain Algorithmic Regularization as presented previously [24] by setting $K = 1$, or to approach convergence at each value of λ by setting K large. We prefer to run just a few $K = 5$ iterations at each vale of λ, allowing some convergence at each value along the path. Supplementary Figure 3 compares AR (K=5) paths to ADMM ($K = 100$) paths for Algorithm 1 (PCMF-ADMM) for a four cluster problem, showing just the first two variables out of $p = 200$ for data with 25 observations per class and means $\in \{-1.0, -0.4, 0.4, 1.0\}$.

8 Consensus PCMF-ADMM Algorithm

8.1 Correcting for batch effects in Consensus PCMF-ADMM solution paths

Differences in batch means result in very similar but slightly shifted paths for the different batches, an effect that results in problems for our dedrogram fitting approach. We therefore correct for the batch effects using the following procedure: (1) cluster on the last solution in the solution path (at $\lambda = 0$), (2) for each of these clusters, calculate it's centroid at each time point and trace this path back to $= \infty$, (3) do this for all terminal clusters to yield a corrected set of paths that can then be input to or dendrogram fitting procedure.

In Algorithm 4, we split X into B batches. For each batch we store the batch mean $\mu_\mathbf{b}$ and demean X_b. The $\widehat{X_b}^{k+1}$ and G^{k+1} updates are run on each batch X_b and then the $\widehat{X_b}^{k+1}$ across B batches are stacked prior to calculating the SVD. This yields an almost-centered $U^{k+1}, S^{k+1}, V^{k+1}$ that needs to be batch corrected by accounting for the individual batch means. We recalculate $T^{k+1} = U^{k+1}S^{k+1}$ by the dot product of the update $(\widehat{X_b}^{k+1} + {Z_2}^k)$ + the X_b mean $\mu_\mathbf{b}$ with the non-centered V^{k+1}. Finally, we re-calculate the U^{k+1} by dividing the centered T^{k+1} by S^{k+1} and unit normalizing the U^{k+1}. This yields a consensus U^{k+1} update that is equivalent to the centered SVD U^{k+1} update. We use this U^{k+1} for the ${Z_1}_b^{k+1}$ and ${Z_1}_b^{k+1}$ updates that we calculate in each batch. Finally, we stack the output $\widehat{X_b}^{k+1}$ across batches to yield $\widehat{X}^{k+1}$.

Algorithm 4 Consensus PCMF-ADMM

```
**Input:** data $X$, $\downarrow$ path $\{\lambda\}$, weights $\mathbf{w}$
**Notation:** batch $b$, batch mean $\mu_{\mathbf{b}}$, rank $r$, iteration $k$, norm $q$, $\rho > 0$
Initialize:
Split $X$ into $B$ batches and demean
  $X =_{b=1}^{B} X_b \; = \overline{X_1 X_2} \cdots \overline{X_b}$
$_{b=1}^{B} \mu_b = X_b - \overline{X_b}$
$L = \text{chol}(I + \rho I + \rho D^T D)$
**for** $b = 1, \ldots, B$ **do**
    $G_b^0 = {Z_1}_b^0 = D\overline{X_b}$
    $\widehat{X_b} = {Z_2}_b^0 = \overline{X}$
**end for**
$({U_r}^0, {S_r}^0, {V_r}^0) \leftarrow \text{SVD}_r(\widehat{X})$
ADMM:
**for** $\lambda \in \{\lambda\}$ **do**
    **for** $k = 1, \ldots, K$ **do**
        **for** $b = 1, \ldots, B$ **do**
            $\widehat{X_b}^{k+1} \leftarrow L^{-T} L^{-1}\big(X_b + \rho D^T (G_b^k - {Z_1}_b^k) + \rho({U_r}_b^k {S_r}^k {{V_r}^k}^T - {Z_2}_b^k)\big)$
            $G_b^{k+1} \leftarrow \text{prox}_{\lambda/\rho, q(\cdot,\cdot)}(D\widehat{X_b}^{k+1} + {Z_1}_b^k; \mathbf{w}_b)$
        **end for**
        $\widehat{X}^{k+1} =_{b=1}^{B} \widehat{X_b}^{k+1}, \quad {Z_2}^k =_{b=1}^{B} {Z_2}_b^k$
        $({U_b}_r^{k+1}, S_r^{k+1}, {V_b}_r^{k+1}) \leftarrow \text{SVD}_r(\widehat{X}^{k+1} + Z_2^k)$
        $T_r^{k+1} \leftarrow (\widehat{X}^{k+1} + Z_2^k + \mu_b){V_b}_r^{k+1}$
        $(U_r^{k+1} \leftarrow T_r^{k+1} / S_r^{k+1}$
        ${\mathbf{U_r}}^{k+1} \; = \|{\mathbf{U_r}}^{k+1}\|_2^2$
        **for** $b = 1, \ldots, B$ **do**
            ${Z_1}_b^{k+1} \leftarrow {Z_1}_b^k + D^T \widehat{X_b}^{k+1} - G_b^{k+1}$
            ${Z_2}_b^{k+1} \leftarrow {Z_2}_b^k + \widehat{X_b}^{k+1} - {U_r}_b^{k+1}, {S_r}^{k+1}, {V_r}^{k+1}$
        **end for**
    **end for**
    $\widehat{X}_\lambda \leftarrow \widehat{X}^K =_{b=1}^{B} \widehat{X_b}^K, \quad G_\lambda \leftarrow G^K =_{b=1}^{B} G_b^K$
    $(U_{r,\lambda}, S_{r,\lambda}, V_{r,\lambda}) \leftarrow (U_r^K, S_r^K, V_r^K)$
    **for** $b = 1, \ldots, B$ **do**
        $\widehat{X_b}^0 \leftarrow \widehat{X_b}^K, G_b^0 \leftarrow G_b^K$
    **end for**
    $({U_r}^0, {S_r}^0, {V_r}^0) \leftarrow ({U_r}^K, {S_r}^K, {V_r}^K)$
**end for**

**return** **return**$\{\widehat{X}_\lambda\}, \{G_\lambda\}, \{U_{r,\lambda}\}, \{S_{r,\lambda}\}, \{V_{r,\lambda}\}$
```

9 Model selection

9.1 Pathwise dendrogram algorithm

Let $m = 1, \ldots, M$ index the decreasing path $\{\lambda\}$ such that $\{\lambda\} = \{\lambda_1 > \cdots > \lambda_m > \cdots > \lambda_M \geq 0\}$. Then to estimate a dendrogram from the smooth paths of the dual variables $\{G_{\lambda_m}\}$, we start at λ_1 (chosen large enough to yield only one cluster) and then proceed along the decreasing path of $\lambda_m > \lambda_{m+1}$. At each step, we make a binary choice between (a) keeping the same number of clusters $c_{m+1} \leftarrow c_m$, or (b) augmenting the number of clusters $c_{m+1} \leftarrow c_m + 1$. To make this choice, we find the partitions of the graph defined by $G_{\lambda_{m+1}}$ (see [2]) into c_m and c_{m+1} clusters, and then compare:

$$\text{loglik}_1(X, \widehat{X}(c_m), \lambda_m) = \frac{1}{2}\|X - \widehat{X}(c_m)\|_F^2 + \lambda_m \sum_{(i,j)\in\mathcal{E}} w_{ij}\|\widehat{X}_{i\cdot}(c_m) - \widehat{X}_{j\cdot}(c_m)\|_q,$$

and

$$\text{loglik}_2(X, \widehat{X}(c_m+1), \lambda_m) = \frac{1}{2}\|X - \widehat{X}(c_m+1)\|_F^2 + \lambda_m \sum_{(i,j)\in\mathcal{E}} w_{ij}\|\widehat{X}_{i\cdot}(c_m+1) - \widehat{X}_{j\cdot}(c_m+1)\|_q,$$

where $\widehat{X}(c_m)$ is $\widehat{X}$ clustered so that it's rows are replaced by c_m unique centroids (that is, $\widehat{X}$ clustered to have exactly c_m unique rows). If $\text{loglik}_1 > \text{loglik}_2$ then $c_{m+1} \leftarrow c_m + 1$, otherwise $c_{m+1} \leftarrow c_m$. This ensures a dendrogram fit along the paths with knots in the number of clusters appearing only when the improvement in model fit exceeds the additional cost of adding another cluster to the penalty. In our experiments, to approximate the portion of the graph defined by the $\{G_{\lambda_m}\}$, we use spectral clustering. As this algorithm uses k-means clustering on the eigenvectors of the affinity matrix estimated from the differences $G_{\lambda_m} = D\widehat{X}_{\lambda_m}$, it introduces random variation in the resulting dendrograms.

We therefore choose to take the median of several runs of the pathwise dendrogram algorithm as the final estimate of the dendrogram, which we refer to as $\textsc{Dendrogram}(\{G_\lambda\})$. Although this performs well in our experiments, other approaches to graph partitioning applied to the graph defined by $\{G_{\lambda_m}\}$ are worth exploring, as they may provide more stable or more efficient approaches. Finally, it is critical to note that the dendrogram denotes the evolution of the centroids, not the individual observations (although these do become the individual observations in the limit $\lambda \to 0$. It is possible, although rare, for observations to switch class membership as the centroid dendrogram is estimated, thus while the result will always be a tree structure, we take the end-leaf membership as final assignment in these cases and trace membership back up the estimated dendrogram post hoc for such cases to avoid ambiguity and satisfy the definition of a dendrogram.

9.2 A correlation-test-statistic-based heuristic for the number of clusters

Previous work has shown a close relationship between convex clustering and single-linkage hierarchical clustering by examining the dual problems of the convex clustering optimization problem and a related problem that has the same connected component structure as single-linkage hierarchical clustering (See Lemmas 2-4 in [22]). Other work developing correlation tests of significance for the number of connected components in the graphical lasso [7] fit along a path of penalty parameters that control model sparsity, have shown that this problem is also equivalent to thresholded single-linkage hierarchical clustering on correlations [10]. Taken together, these findings suggest extending the same correlation test statistic for the graphical lasso to the convex clustering problem in order to choose the best number of clusters for a give data set may be fruitful.

In particular, let $\{G_{\lambda_t}\}$ for $t = 1, \ldots, T$ be the values or "knots" at which the number of clusters chosen by the pathwise dendrogram algorithm change along path $\{G_{\lambda_m}\}$ (so $G_{\lambda_m} \in \{G_{\lambda_t}\}$ if and only if $G_{\lambda_m} \to G_{\lambda_{m+1}} \implies c_{m+1} = c_m + 1$), then we note that as in [10] the $\lambda_1 > \cdots > \lambda_t > \cdots > \lambda_T$ correspond to the subset of knots at which the connected components of the estimate change. This naturally leads to a set of hypotheses, $H_1, \ldots, H_T$, where the hypothesis H_t is that each connected component of the true dendrogram is contained within the connected component defined by the estimated $\textsc{Dendrogram}(\{G_\lambda\})\ \forall \lambda < \lambda_t$. To test these hypotheses, we note that the convex clustering problem is related to the group lasso estimator on the rows of G_λ, yielding the potential test statistic:

$$T_t = N\lambda_t(\lambda_t - \lambda_{t+1}),$$

an adaptation of the correlation test originally developed for the lasso in [14]. However, as our problem is nonconvex due to the SVD constraint on convex clustering, and as we note that unlike the graphical lasso problem our observations can switch components along the path (hypotheses are not strictly nested), here we note this approach as a useful heuristic rather than an asymptotic result.

9.3 Model selection for known number of clusters

Given a prespecified number of clusters c (if for example, we know the number of desired clusters beforehand), we may want to choose a best model conditional on c. As there may be more than one value of m for which $c = c_m$, we choose the m that solves

$$\underset{m}{\text{minimize}}\ \text{loglik}_1\left(X, \widehat{X}(c_m), \lambda_m\right)\ \text{subject to } c_m = c.$$

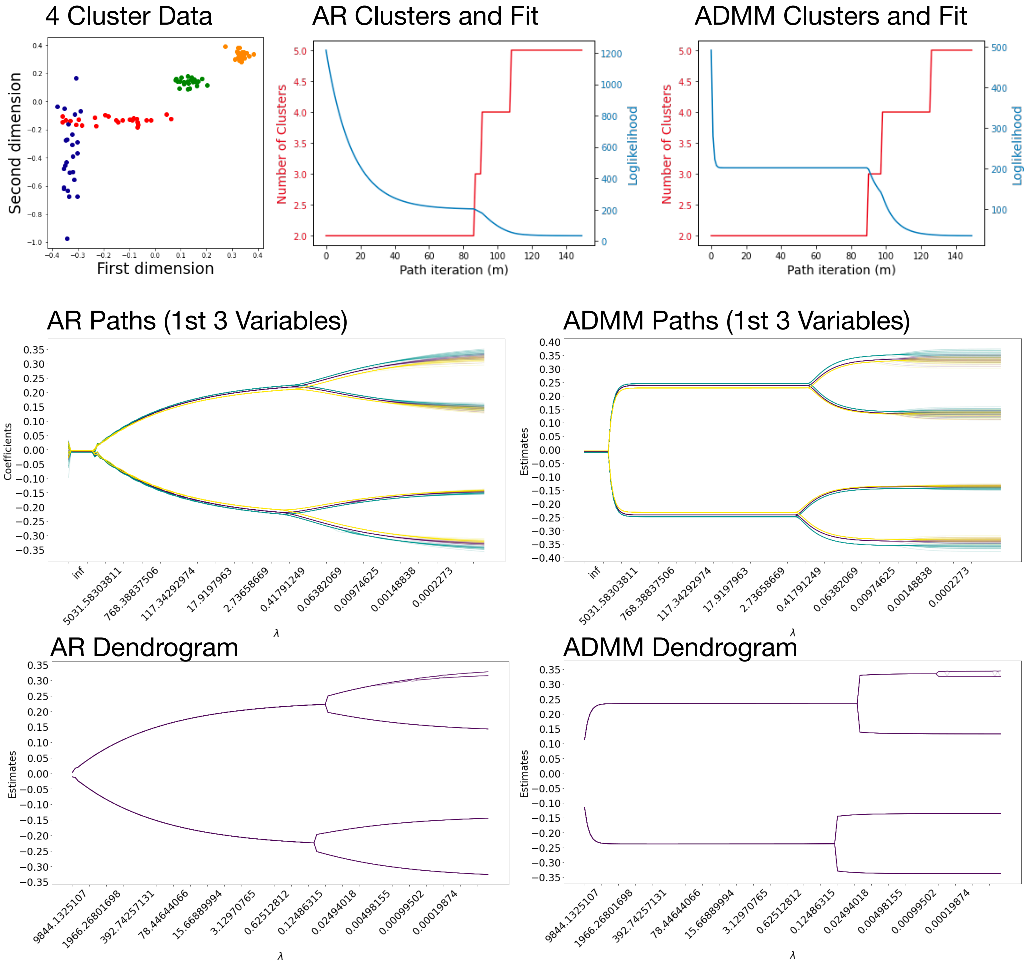

Figure 4: Comparison of AR and ADMM on a four cluster dataset, $p = 200$, 25 observations per class.

10 Additional Experimental Results

ADMM hyperparameter example. To demonstrate the effect of the ρ parameter, we ran PCMF on the tumors multiomics cancer dataset varying ρ and plotted the pathwise estimates Supplementary Figure 5. Paths are stable across values of $\rho = 1.0, 1.5, 2.0, 2.5$. For all subsequent experiments we do not tune ρ, and rather set $\rho = 1.0$.

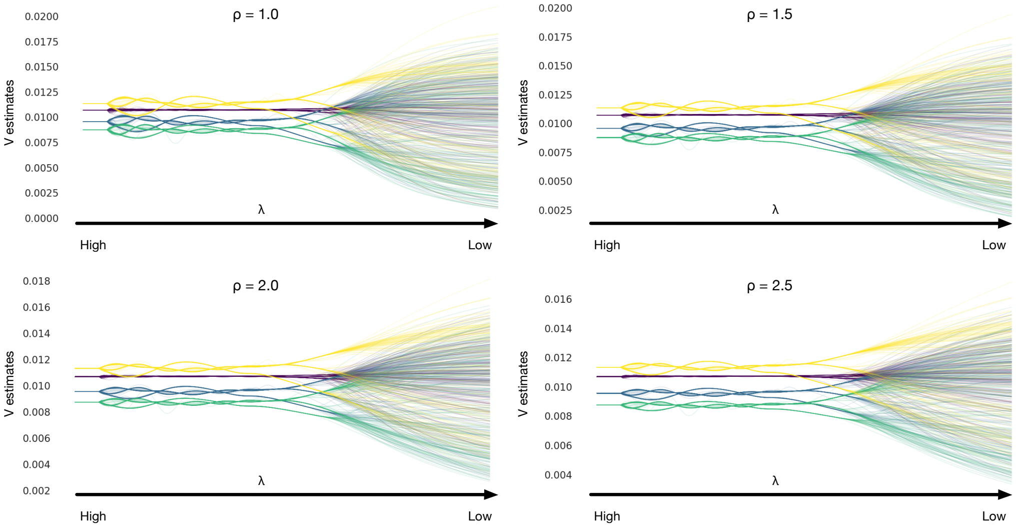

Figure 5: Pathwise estimates for PCMF on the Tumors Multiomics Cancer Dataset varying ρ. Threshold variable λ is decreasing from left to right, leading to a divisive-like clustering solution. As it decreases, observations split free from the initial cluster into the three cancer clusters. Paths are stable across values of $\rho = 1.0, 1.5, 2.0, 2.5$.

Autism Spectrum Disorder dataset. We find that the solution from sequential CCA+KMeans identifies a similar ASD-related brain-behavior embedding, but fails to fully separate the 2 clusters along this embedding (Figure 5 in Main Text, similar to prior sequential CCA+clustering approaches used in neuropsychiatry [3, 9]. We further point out that the resulting correlations between ASD clinical behaviors and the P3CA variate are significantly different across clusters (SupplementaryTable 3) as well as between ASD brain connections and the P3CA variate (Supplementary Table 4), consistent with known ASD subpopulation differences on RRBs and Verbal IQ and prefrontal cortex to somatosensory cortex, posterior parietal cortex, and middle temporal gyrus.

Table 3: Correlations Between ASD Behaviors and P3CA variate.

	Cluster 1	Cluster 2	Combined
RRB	-0.50	0.05	-0.45
Verbal IQ	0.25	0.63	0.63
Social Affect	-0.16	-0.35	-0.58

Table 4: Correlations between brain connections and P3CA variate.

	Cluster 1	Cluster 2	Combined
L MTG - L thalamus	-0.16	0.03	-0.23
L paracentral/S1 - R VMPFC/IFGorb	0.04	**0.20**	0.21
R SOG - L cerebellum	-0.11	-0.05	-0.19
R ITG/MTG - R PCC	**-0.21**	0.06	-0.27
L insula - L MOG	-0.13	-0.10	-0.05
R MTG/ITG - L MTG	0.09	0.14	0.21
L antPFC/SFG - L Nacc	-0.03	-0.01	-0.19
L precentral/M1 - R VMPFC/mOFC	-0.09	-0.14	-0.21
R ACC/MCC - R VLPFC/MFG	0.09	0.05	-0.13
R VMPFC/IFGorb - R PPC/SPL	0.18	**0.25**	0.31
R SMA/ACC - L MOG	-0.08	0.14	-0.01
R VMPFC/IFGorb - L PPC/IPL	0.11	**0.39**	0.24
R temp pole - L MTG	-0.04	0.06	0.15
R temp pole - L thalamus	**-0.28**	0.07	-0.21
R VLPFC/MFG - R MTG	**0.20**	0.05	0.20
R ACC/MCC - L vtrPut	-0.00	-0.03	-0.02
L lingual - L VLPFC/IFGorb	-0.10	-0.04	0.11
R VMPFC/IFGorb - L lingual	0.14	0.10	0.21
L cuneus - L fusiform	0.14	0.10	0.24
L lingual - L ACC	-0.03	-0.04	0.03

11 Additional experimental details

11.1 Hyperparameters

We only implemented limited hyperparameter tuning for the PCMF method on a very course grid, only varying nearest neighbors between three settings: None, 15, 25 and keeping rho, gamma, number of ADMM iterations and the lambda penalty range constant (see below). In comparison, for the deep clustering methods, the Louvain, and the Leidein method, we implemented a finer hyperparameter grid for tuning as detailed below. With additional hyperparameter tuning on a finer grid, we expect our PCMF and P3CA model may be further optimized. For all experiments, ρ was set to 1.0, γ was set to 2.0, and the number of ADMM iterations per λ penalty was set to 5. For the penalty path, we varied the value of λ along a path starting with 10 infinities followed by 150 evenly spaced points in the interval [10,-10], such that the penalty decreased along the path of embedding solutions. Although ADMM can in some cases be quite sensitive to choices of the augmented Lagrangian parameter ρ, we find our algorithm to be stable across a range of common ρ values (see Supplementary Figure 5).

For Deep Embedding Clustering we tuned the following hyperparameters: batch size (15, 30), finetune iterations (100, 1000), iterations for layerwise pretraining (100, 1000), and maximum iterations for clustering (100, 200). Layer sizes and other parameters were set to the defaults in the model code from https://github.com/fferroni/DEC-Keras.

For Improved Deep Embedding Clustering we tuned the following hyperparameters: batch size (15, 30), pretraining epochs (100, 1000) and training epochs (100, 1000). Layer sizes and other parameters were set to the defaults in the model code from https://github.com/dawnranger/IDEC-pytorch.

For CarDEC we tuned the following hyperparamters: number of neighbors (5, 10, 15, 20, 25) and number of top genes (100, 500).

For Louvain and Leidein we tuned number of neighbors (5, 10, 15, 20, 25).

The optimal hyperparameters for these deep clustering and Louvain and Leidein methods were as follows (N.N. indicates number of nearest neighbors):

1. For the NCI dataset: Leiden with 5 N.N.; Louvain with 15 N.N.; DEC with batch size 15, finetune iterations 100, layerwise iterations 100, cluster iterations 200; IDEC with batch size 30, pretrain iterations 100, and train iterations 1000; CarDEC with 5 N.N. and 100 top genes.
2. For the SRBCT dataset; Leiden tied with 10, 20, or 25 N.N.; Louvain with 15 N.N.; DEC with batch size 15, finetune iterations 1000, layerwise iterations 1000, cluster iterations 100; IDEC with batch size 30, pretrain iterations 100, and train iterations 100; CarDEC with 10 N.N. and tied with 100 or 200 top genes.
3. For the Mouse Organs dataset: Leiden tied with 20 or 25 N.N.; Louvain with 15 N.N.; DEC with batch size 15, finetune iterations 100, layerwise iterations 100, cluster iterations 100;

IDEC with batch size 30, pretrain iterations 100, and train iterations 100; CarDEC with 15 N.N. and 100 top genes.

4. For the Tumors dataset: Leiden tied with 15, 20, or 25 N.N.; Louvain tied with 20 or 25 N.N.; DEC tied with batch size 15 or 30, finetune iterations 100 or 1000, layerwise iterations 100 or 1000, cluster iterations 100 or 200; IDEC with batch size 15, pretrain iterations 100, and train iterations 100; CarDEC with 20 N.N. and 200 top genes.

5. For the COVID-19 dataset: Leiden tied with 20 or 25 N.N.; Louvain with 10 N.N.; DEC with batch size 15, finetune iterations 1000, layerwise iterations 100, cluster iterations 200; IDEC with batch size 30, pretrain iterations 100, and train iterations 100; CarDEC with 5 N.N. and 200 top genes.

11.2 Synthetic data generation

The **Synthetic Datasets** consisting of eighteen types of two clustered-data in a single view with $p = 20$, $p = 200$, or $p = 2000$ variables and variable redundancy set by $\delta = 0.5$ or $\delta = 0.2$, the fraction of variables containing signal. Varying p and δ, we generate data from two separate distributions with different slopes and cluster means in three variations: non-overlapping clusters (cluster centroids $\in \{-0.2, 0.2\}$; $N_1 = 50; N_2 = 50$), overlapping clusters (cluster centroids $\in \{-0.05, 0.05\}$; $N_1 = 50; N_2 = 50$), or non-overlapping but unbalanced cluster size (cluster centroids $\in \{-0.2, 0.2\}$; $N_1 = 80; N_2 = 20$).

For each dataset generated, the seed was set to ensure the two-cluster data was reproducible. Within a single dataset, for each cluster a random matrix of N x p with the specified mean and σ = 0.075 was generated. Next, **u** was generated as a random matrix of N x 1 and **v** was generated as a random matrix of p x 1. We selected $\delta * p$ features from a randomly permuted order of the total p features and added $\mathbf{v}[i] * \mathbf{u}$ to $\mathbf{X}[:, i]$ where i is the feature (column). Finally, we standardized features over samples (rows) by removing the mean and scaling to unit variance and added a column of ones as an intercept to the features. We compare results based on the adjusted rand index (ARI) and the normalized mutual information (NMI).

11.3 Mouse organ cancer genomics dataset

The **Mouse Organ Cancer Genomics Dataset** consists of single-cell RNA-sequencing measurements in $p = 16,944$ genes measured in $N = 125$ mouse organ samples from 7 different mouse organs collected in the Tabula Muris study. The 125 samples is a representative sample from the full dataset of 6,232 mouse organ samples. Data was scaled to be between 0 and 1 and genes were filtered to remove genes whose expression had low variance across rows (0.2 quantile) following previously published preprocessing steps for this dataset [13].

11.4 Multiomics cancer dataset

The **Multiomics Cancer Dataset** consists of $p = 11,931$ multiomics measurements (gene expression levels, DNA methylation, miRNA expression) in tumor samples from $N = 142$ patients for 3 cancer diagnoses (glioblastoma multiforme (GBM), breast invasive carcinoma (BIC), and lung adenocarcinoma) from The Cancer Genome Atlas Program (TCGA) Research Network [https://www.cancer.gov/tcga] and curated by [6]. The $N = 142$ samples is a representative sample from the full dataset of 424 patient samples. Gene expression levels, DNA methylation, miRNA expression measurements for each cancer type (GBM, BIC, and lung adenocarcinoma) were concatenated and then the three cancer types were merged such that all cancer types had the same features with no missing values. Data was scaled as following:

$$X_n = \frac{X_i - x_{min}}{x_{max} - x_{min}} \tag{1}$$

where X_i is the data for feature i while x_{max} and x_{min} are the minimum and maximum absolute value of the feature respectively. X_n is the normalized feature over rows. This followed previously published preprocessing steps for these datasets [6].

11.5 Autism Spectrum Disorder dataset

The **Autism Spectrum Disorder Dataset** consists of $p_X = 3$ clinical symptoms and $p_Y = 20$ resting state functional connectivity (RSFC) features measured from resting state functional MRI (rsfMRI) neuroimaging in $N = 299$ patients with autism spectrum disorder (ASD) from the ABIDE I and 2 datasets [16, 15]. The 3 clinical symptoms are verbal IQ (VIQ), ADOS-2 social affect CSS, and ADOS-2 repetitive, restricted behaviors and interests (RRB) CSS. The ADOS-2 is the Autism Diagnostic Observation Schedule-Second Edition CSS, a standardized observational scale for diagnosing ASD. CSS stands for the calibrated severity score.

Following standard preprocessing of the rsfMRI [3, 9, 21], we calculated the resting state functional connectivity (RSFC) matrices for each subject by the Pearson correlation between 247 regions of interest (ROIs) from the Power atlas [19]. We next performed feature selection based on previous methods [3, 9] by calculating the Spearman correlation between each RSFC feature ($p = 30,381$ unique RSFC) and each clinical symptom in 1,000 subsamples of 95% of the subjects ($n = 284$) and ranked RSFC features by the number of subsamples in which the RSFC had a significant correlation ($p-value < 0.05$) to one of the clinical symptoms. This rank list represented the relative importance of each RSFC feature to predicting clinical symptoms in ASD. We selected the top 20 RSFC from this rank list. For each view, X, we add a column of ones as a free variable in which the U coefficients can capture differences in cluster means. Thus the input into the P3CA analyses included the $p_X = 4$ (3 clinical symptoms + ones columns) and $p_Y = 21$ (20 most predictive RSFC features + ones columns) in $N = 299$ patients with ASD.

All human neuroimaging and behavioral data from the ABIDE I and II datasets is anonymized, there is no protected health information included, and the datasets are publicly available with approval. Protocols for human subject research in the ABIDE datasets are included in the study details for each of the 17 study sites for ABIDE I and 19 study sites for ABIDE II (see http://fcon_1000.projects.nitrc.org/indi/abide). Thus we did not collect any data using human subjects for this study, and all information about IRB approval and participant compensation can be found in the original datasets collected by the ABIDE I and II consortia. As we using resting state functional MRI data and the behavioral measures included are from standardized behavioral and clinical scales (intelligence quotient and ADOS-2), participants were not shown text instructions during MRI scanning.

References

[1] S. Boyd, N. Parikh, and E. Chu. *Distributed Optimization and Statistical Learning Via the Alternating Direction Method of Multipliers*. Now Publishers Inc, 2011.

[2] E. C. Chi and K. Lange. Splitting methods for convex clustering. *J. Comput. Graph. Stat.*, 24 (4):994–1013, Dec. 2015.

[3] A. T. Drysdale, L. Grosenick, J. Downar, K. Dunlop, F. Mansouri, Y. Meng, R. N. Fetcho, B. Zebley, D. J. Oathes, A. Etkin, A. F. Schatzberg, K. Sudheimer, J. Keller, H. S. Mayberg, F. M. Gunning, G. S. Alexopoulos, M. D. Fox, A. Pascual-Leone, H. U. Voss, B. J. Casey, M. J. Dubin, and C. Liston. Resting-state connectivity biomarkers define neurophysiological subtypes of depression. *Nat. Med.*, 23(1):28–38, Jan. 2017.

[4] S. Dudoit, J. Fridlyand, and T. P. Speed. Comparison of discrimination methods for the classification of tumors using gene expression data. *J. Am. Stat. Assoc.*, 97(457):77–87, Mar. 2002.

[5] C. Eckart and G. Young. The approximation of one matrix by another of lower rank. *Psychometrika*, 1(3):211–218, Sept. 1936.

[6] E. F. Franco, P. Rana, A. Cruz, V. V. Calderón, V. Azevedo, R. T. J. Ramos, and P. Ghosh. Performance comparison of deep learning autoencoders for cancer subtype detection using Multi-Omics data. *Cancers*, 13(9), Apr. 2021.

[7] J. Friedman, T. Hastie, H. Höfling, and R. Tibshirani. Pathwise coordinate optimization. *AOAS*, 1(2):302–332, Dec. 2007.

[8] A. Goncalves, X. Liu, and A. Banerjee. Two-block vs. multi-block ADMM: An empirical evaluation of convergence. July 2019.

[9] L. Grosenick, T. C. Shi, F. M. Gunning, M. J. Dubin, J. Downar, and C. Liston. Functional and optogenetic approaches to discovering stable Subtype-Specific circuit mechanisms in depression. *Biol Psychiatry Cogn Neurosci Neuroimaging*, 4(6):554–566, June 2019.

[10] M. G. G'Sell, J. Taylor, and R. Tibshirani. Adaptive testing for the graphical lasso. July 2013.

[11] D. Hajinezhad, T.-H. Chang, X. Wang, Q. Shi, and M. Hong. Nonnegative matrix factorization using ADMM: Algorithm and convergence analysis. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4742–4746, Mar. 2016.

[12] H. Hotelling. Relations between two sets of variates. *Biometrika*, 28(3-4):321–377, Dec. 1936.

[13] A. Kopf, V. Fortuin, V. R. Somnath, and M. Claassen. Mixture-of-Experts variational autoencoder for clustering and generating from similarity-based representations on single cell data. *PLoS Comput. Biol.*, 17(6):e1009086, June 2021.

[14] R. Lockhart, J. Taylor, R. J. Tibshirani, and R. Tibshirani. A significance test for the lasso. *Ann. Stat.*, 42(2):413–468, Apr. 2014.

[15] Martino. Enhancing studies of the connectome in autism using the autism brain imaging data exchange II. *Sci Data*, 4:170010, 2017.

[16] D. A. Martino. The autism brain imaging data exchange: towards a large-scale evaluation of the intrinsic brain architecture in autism. 19(6):659, 2014.

[17] N. Parikh and S. Boyd. Proximal algorithms. *Found. Trends Optim.*, 1(3):127–239, Jan. 2014.

[18] B. Paul, S. K. De, and A. K. Ghosh. Some clustering-based exact distribution-free k-sample tests applicable to high dimension, low sample size data. *J. Multivar. Anal.*, page 104897, Nov. 2021.

[19] J. D. Power, A. L. Cohen, S. M. Nelson, G. S. Wig, K. A. Barnes, J. A. Church, A. C. Vogel, T. O. Laumann, F. M. Miezin, B. L. Schlaggar, and S. E. Petersen. Functional network organization of the human brain. *Neuron*, 72(4):665–678, Nov. 2011.

[20] S. Sarkar and A. K. Ghosh. On perfect clustering of high dimension, low sample size data. *IEEE Trans. Pattern Anal. Mach. Intell.*, 42(9):2257–2272, Sept. 2020.

[21] T. D. Satterthwaite, D. H. Wolf, J. Loughead, K. Ruparel, M. A. Elliott, H. Hakonarson, R. C. Gur, and R. E. Gur. Impact of in-scanner head motion on multiple measures of functional connectivity: relevance for studies of neurodevelopment in youth. *Neuroimage*, 60(1):623–632, Mar. 2012.

[22] K. M. Tan and D. Witten. Statistical properties of convex clustering. *Electron J Stat*, 9(2): 2324–2347, Oct. 2015.

[23] R. Tibshirani, T. Hastie, B. Narasimhan, and G. Chu. Class prediction by nearest shrunken centroids, with applications to DNA microarrays. *Stat. Sci.*, 18(1):104–117, 2003.

[24] M. Weylandt, J. Nagorski, and G. I. Allen. Dynamic visualization and fast computation for convex clustering via algorithmic regularization. *J. Comput. Graph. Stat.*, 29(1):87–96, 2020.

[25] D. M. Witten, R. Tibshirani, and T. Hastie. A penalized matrix decomposition, with applications to sparse principal components and canonical correlation analysis. *Biostatistics*, 10(3):515–534, July 2009.

[26] C. You, C.-G. Li, D. P. Robinson, and R. Vidal. Oracle based active set algorithm for scalable elastic net subspace clustering. May 2016.