Data Mining and Knowledge Discovery
https://doi.org/10.1007/s10618-022-00820-9

Weighted sparse simplex representation: a unified framework for subspace clustering, constrained clustering, and active learning

Hankui Peng[1] · Nicos G. Pavlidis[2]

Received: 7 March 2021 / Accepted: 13 January 2022

Abstract

Spectral-based subspace clustering methods have proved successful in many challenging applications such as gene sequencing, image recognition, and motion segmentation. In this work, we first propose a novel spectral-based subspace clustering algorithm that seeks to represent each point as a sparse convex combination of a few nearby points. We then extend the algorithm to a constrained clustering and active learning framework. Our motivation for developing such a framework stems from the fact that typically either a small amount of labelled data are available in advance; or it is possible to label some points at a cost. The latter scenario is typically encountered in the process of validating a cluster assignment. Extensive experiments on simulated and real datasets show that the proposed approach is effective and competitive with state-of-the-art methods.

Keywords Subspace clustering · Constrained clustering · Active learning

1 Introduction

In many challenging real-world applications involving the grouping of high-dimensional data, points from each group (cluster) can be well approximated by a distinct lower dimensional linear subspace. This is the case in gene sequenc-

Responsible editors: Annalisa Appice, Sergio Escalera, Jose A. Gamez and Heike Trautman.

✉ Hankui Peng
hp467@cam.ac.uk

Nicos G. Pavlidis
n.pavlidis@lancaster.ac.uk

[1] Department of Applied Mathematics and Theoretical Physics, University of Cambridge, Cambridge, UK

[2] Department of Management Science, Lancaster University, Lancaster, UK

Published online: 11 February 2022

ing (McWilliams and Montana 2014), cancer genomics (Yeoh et al. 2002), face clustering (Elhamifar and Vidal 2013), motion segmentation (Rao et al. 2010), and text mining (Peng et al. 2018). The problem of simultaneously estimating the linear subspace corresponding to each cluster, and assigning each point to the closest subspace is known as *subspace clustering* (Vidal 2011). In the data mining literature, this problem has also been referred to as correlation clustering (Kriegel et al. 2009), but we refrain from using this terminology here. It is important to note that in data mining the term "subspace clustering" has been used to refer to a number of distinct high-dimensional clustering problems (Kriegel et al. 2009).[1] A formal definition of the problem we consider as well as a brief overview of existing methods is provided in Sect. 2.

Spectral methods for subspace clustering have demonstrated excellent performance in numerous real-world applications (Liu et al. 2012; Elhamifar and Vidal 2013; Huang et al. 2015; Li et al. 2017). These methods construct an affinity matrix for spectral clustering by solving an optimisation problem that aims to approximate each point through a linear combination of other points from the same subspace. In this paper, we first propose a method called *Weighted Sparse Simplex Representation* (WSSR). Our method is based on the Sparse Simplex Representation (SSR) of Huang et al. (2013), in which each point is approximated through a convex combination of other points. This method was not proposed as a subspace clustering method, but rather as a method for modelling the brain anatomical and genetic networks. We modify SSR to ensure that each point is approximated through a sparse convex combination of nearby neighbours, and thus obtain an algorithm that is effective for the subspace clustering problem.

Due to the complete lack of labelled data, clustering methods rarely achieve perfect performance. For this reason, in real-world applications, clustering models are rarely immediately accepted and acted upon. Instead, they undergo one or more rounds of "validation", which commonly involves domain experts assessing whether the model is sensible. A generic description of the information generated during validation is to assume that domain experts assess whether the assignment of a (small) subset of the originally unlabelled data is sensible. To be accepted the clustering model needs to be consistent with this information. In this paper, we assume that this external (side) information can be translated to labels. To accommodate the existence of label information, we consider constrained subspace clustering (Basu et al. 2008). An even more interesting problem arises if the learning algorithm can select the points that experts consider during validation. An effective active learning strategy is highly beneficial not only because it minimises the cost of obtaining labels, but also because producing a valid clustering in as few iterations of the validation process as possible improves user confidence in the model. We therefore discuss an active learning strategy that is designed to query the labels of points so as to maximise the overall quality of the subspace clustering model. In particular, we draw on the work of Peng and Pavlidis (2019) to select informative points to label for subspace clustering. The label information

[1] In particular, the term subspace clustering is frequently used to refer to the problem of identifying clusters that are defined in potentially different lower dimensional subspaces (Kriegel et al. 2009). (This problem and the algorithms to solve it are related to biclustering, coclustering and multi-type clustering.) This is problem differs from the one we consider here.

is subsequently incorporated in a constrained clustering formulation that combines WSSR and constrained K-subspace clustering. The resulting cluster assignment is guaranteed to satisfy all the constraints arising from the set of labelled data.

The rest of this paper is organised as follows. In Sect. 2, we discuss some relevant existing literature in the areas of subspace clustering, constrained clustering, and active learning. In Sect. 3, we propose the problem formulation of WSSR, discuss its properties, and present an approach for solving the problem. We develop an integrated active learning and constrained clustering framework in Sect. 4. The experimental results are organised in three sections. First, in Sect. 5, we use synthetic data to evaluate the performance of WSSR under various configurations of the subspace clustering problem. Next, we consider real datasets to assess the comparative performance of our proposed method in the completely unsupervised setting (Sect. 6), as well as in the constrained clustering with active learning settings (Sect. 7). The paper ends with concluding remarks and future research directions in Sect. 8.

2 Related work

The linear subspace clustering problem can be defined as follows. A collection of N data points $\mathcal{X} = \{\boldsymbol{x}_i\}_{i=1}^N \subset \mathbb{R}^P$ is drawn from a union of K linear subspaces $\{\mathcal{S}_k\}_{k=1}^K$ with added noise. Each subspace can be defined as,

$$\mathcal{S}_k = \left\{\boldsymbol{x} \in \mathbb{R}^P \mid \boldsymbol{x} = V_k \boldsymbol{y}\right\}, \quad \text{for} \;\; k = 1, \ldots, K, \tag{1}$$

where $V_k \in \mathbb{R}^{P \times d_k}$, with $1 \leq d_k < P$, is a matrix whose columns constitute a basis for $\mathcal{S}_k$, and $\boldsymbol{y} \in \mathbb{R}^{d_k}$ is the representation of $\boldsymbol{x}$ in terms of the columns of V_k. The goal of subspace clustering is to find the number of subspaces K; the subspace dimensions $\{d_k\}_{k=1}^K$; a basis $\{V_k\}_{k=1}^K$ for each subspace; and finally the assignments of the points in $\mathcal{X}$ to clusters. A natural formulation of this problem is as the minimisation of the *reconstruction error*,

$$\sum_{i=1}^N \min_{V_1,\ldots,V_K} \left\{ \min_{k=1,\ldots,K} \|\boldsymbol{x}_i - V_k V_k^\top \boldsymbol{x}_i\|_2^2 \right\}. \tag{2}$$

K-subspace clustering (KSC) (Bradley and Mangasarian 2000) is an iterative algorithm to solve the problem in (2). Like the classical K-means clustering, KSC alternates between estimating the subspace bases (for a fixed cluster assignment), and assigning points to clusters (for a fixed set of bases). However, iterative algorithms are very sensitive to initialisation and most often converge to poor local minima (Lipor and Balzano 2017).

Currently the most effective approach to subspace clustering is through spectral-based methods (Lu et al. 2012; Elhamifar and Vidal 2013; Hu et al. 2014; You et al. 2016). Spectral-based methods consist of two steps: first an affinity matrix is estimated, and then normalised spectral clustering (Ng et al. 2002) is applied to this affinity matrix. The affinity matrix is constructed by exploiting the *self-expressive* property:

any $\boldsymbol{x}_i \in \mathcal{S}_k$ can be expressed as a linear combination of d_k other points from $\mathcal{S}_k$. Thus, for each $\boldsymbol{x}_i \in \mathcal{X}$ they first solve a convex optimisation problem of the form,

$$\boldsymbol{\beta}_i^\star = \min_{\boldsymbol{\beta}_i \in \mathbb{R}^{N-1}} \left\| \boldsymbol{x}_i - X_{-i}\boldsymbol{\beta}_i \right\|_p + \rho \left\| \boldsymbol{\beta}_i \right\|_q , \tag{3}$$

where $X_{-i} = \left[\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{i-1}, \boldsymbol{x}_{i+1} \ldots, \boldsymbol{x}_N\right] \in \mathbb{R}^{P \times (N-1)}$ is a matrix whose columns correspond to the points in $\mathcal{X} \backslash \{\boldsymbol{x}_i\}$; and $\rho > 0$ is a penalty parameter. The first term in the objective function quantifies the error of approximating $\boldsymbol{x}_i$ through $X_{-i}\boldsymbol{\beta}_i$. The penalty (regularisation) term is included to promote solutions in which $\beta_{ij}^\star$ is small (and ideally zero) if $\boldsymbol{x}_j$ belongs to a different subspace than $\boldsymbol{x}_i$. After solving the problem in (3) for each $\boldsymbol{x}_i \in \mathcal{X}$, the affinity matrix is typically defined as $A = \left(|B| + |B|^\top\right)/2$, where $B = [\boldsymbol{\beta}_1^\star, \ldots, \boldsymbol{\beta}_N^\star]$.

Least Squares Regression (LSR) (Lu et al. 2012) uses the L_2-norm for both the approximation error, and the regularisation term ($p = q = 2$). Smooth Representation Clustering (SMR) (Hu et al. 2014) also uses the L_2-norm on the approximation error, while the penalty term is given by $\|L^{1/2}\boldsymbol{\beta}_i\|_2^2$ in which L a positive definite Laplacian matrix constructed from pairwise similarities. The main advantage of using the L_2-norm is that the optimisation problem has a closed-form solution. However the resulting coefficient vectors are dense and hence the affinity matrix contains connections between points from different subspaces. The most prominent spectral-based subspace clustering algorithm is Sparse Subspace Clustering (SSC) (Elhamifar and Vidal 2013). In its most general formulation, SSC accommodates the possibility that points from each subspace are contaminated by both noise and sparse outlying entries. In SSC, $\boldsymbol{\beta}_i^\star$ is the solution to the following problem,

$$\begin{aligned} \min_{\boldsymbol{\beta}_i \in \mathbb{R}^{N-1}} \; & \|\boldsymbol{\beta}_i\|_1 + \rho_\eta \|\boldsymbol{\eta}_i\|_1 + \frac{\rho_z}{2}\|\boldsymbol{z}_i\|_2^2, \\ \text{s.t. } & \boldsymbol{x}_i = X_{-i}\boldsymbol{\beta}_i + \boldsymbol{\eta}_i + \boldsymbol{z}_i. \end{aligned} \tag{4}$$

SSC therefore decomposes the approximation error into two components ($\boldsymbol{\eta}_i$ and $\boldsymbol{z}_i$), which are measured with different norms. Following the success of SSC, several variants have been proposed, including SSC with Orthogonal Matching Pursuit (SSC-OMP) (You et al. 2016), Structured Sparse Subspace Clustering (S3C) (Li et al. 2017), and Affine Sparse Subspace Clustering (ASSC) (Li et al. 2018a).

The method most closely connected to our approach is SSR, proposed by Huang et al. (2013) for the modelling of brain networks. SSR solves the problem in (3) using $p = 2$ and $q = 1$ with the additional constraint that the coefficient vector has to lie in the $(N-1)$-dimensional unit simplex $\boldsymbol{\beta}_i \in \Delta^{N-1} = \{\boldsymbol{\beta} \in \mathbb{R}^{N-1} \,|\, \boldsymbol{\beta} \geq 0, \boldsymbol{\beta}^\top \mathbf{1} = 1\}$. Since SSR approximates $\boldsymbol{x}_i$ through a convex combination of other points, the coefficients have a probabilistic interpretation. However, SSR induces no regularisation since $\|\boldsymbol{\beta}_i\|_1 = 1$ for all $\boldsymbol{\beta}_i \in \Delta^{N-1}$, hence coefficient vectors are dense.

We next provide a short overview of clustering with external information, called *constrained clustering* (Basu et al. 2008), and *active learning*. Due to space limitations, we only mention the work that is most closely related to our problem. In constrained clustering, the external information can be either in the form of class labels or as pairwise “must-link” and “cannot-link” constraints. Spectral methods for constrained clustering incorporate this information by modifying the affinity matrix. Constrained Spectral Partitioning (CSP) (Wang and Davidson 2010) introduces a pairwise constraint matrix and solves a modified normalised cut spectral clustering problem. Partition Level Constrained Clustering (PLCC) (Liu et al. 2018) forms a pairwise constraint matrix through a side information matrix which is included as a penalty term into the normalised cut objective. Constrained Structured Sparse Subspace Clustering (CS3C) (Li et al. 2017) is specifically designed for subspace clustering. CS3C incorporates a side information matrix that encodes the pairwise constraints into the formulation of S3C. The algorithm alternates between solving for the coefficient matrix and solving for the cluster labels. Constrained clustering algorithms that rely exclusively on modifying the affinity matrix cannot guarantee that all the constraints will be satisfied. CS3C+ (Li et al. 2018b) is an extension of CS3C that applies constrained K-means algorithm (Wagstaff et al. 2001) within the spectral clustering stage, to ensure constraints are satisfied.

In active learning the algorithm controls the choice of points for which external information is obtained. The majority of active learning techniques are designed for supervised methods, and little research has considered the problem of active learning for subspace clustering (Lipor and Balzano 2015, 2017; Peng and Pavlidis 2019). Lipor and Balzano (2015) propose two active strategies. The first queries the point(s) with the largest reconstruction error to its allocated subspace. The second queries the point(s) that is maximally equidistant to its two closest subspaces. Lipor and Balzano (2017) extend the second strategy for spectral clustering by setting the affinity of “must-link” and “cannot-link” pairs of points to one and zero, respectively. Both strategies by Lipor and Balzano (2015) are effective in identifying mislabelled points. However, correctly assigning these points is not guaranteed to maximally improve the accuracy of the estimated subspaces, hence the overall quality of the clustering. Peng and Pavlidis (2019) propose an active learning strategy for sequentially querying point(s) to maximise the decrease of the overall reconstruction error in (2).

3 Weighted sparse simplex representation

In this section, we describe the proposed spectral-based subspace clustering method, called Weighted Sparse Simplex Representation (WSSR). In this description, we assume that no labelled data is available. The constrained clustering version described in the next section accommodates the case of having a subset of labelled observations at the start of the learning process.

Let $d_{ij} \geq 0$ denote a measure of dissimilarity between $\boldsymbol{x}_i, \boldsymbol{x}_j \in \mathcal{X}$, and $\mathcal{I}$ the set of indices of all points in $\mathcal{X}\{x_i\}$ with finite dissimilarity to $\boldsymbol{x}_i$, that is $\mathcal{I} = \{1 \leq j \leq N \mid d_{ij} < \infty,\ \ j \neq i\}$. In WSSR, the coefficient vector for each $\boldsymbol{x}_i \in \mathcal{X}$ is the solution to the following convex optimisation problem,

$$\begin{aligned}\boldsymbol{\beta}_i^\star = \arg\min_{\boldsymbol{\beta}_i} \frac{1}{2}\left\|\boldsymbol{x}_i - \hat{X}_{\mathcal{I}}\boldsymbol{\beta}_i\right\|_2^2 + \rho\left\|D_{\mathcal{I}}\boldsymbol{\beta}_i\right\|_1 + \frac{\xi}{2}\|D_{\mathcal{I}}\boldsymbol{\beta}_i\|_2^2 \\ \text{s.t.} \quad \boldsymbol{\beta}_i^\top\mathbf{1} = 1, \quad \boldsymbol{\beta}_i \geq \mathbf{0},\end{aligned} \tag{5}$$

where $\rho, \xi > 0$ are penalty parameters, $\hat{X}_{\mathcal{I}} \in \mathbb{R}^{P\times|I|}$ is a matrix whose columns are the scaled versions of the points in $\mathcal{X}_{\mathcal{I}}$, and $D_{\mathcal{I}} = \text{diag}(\boldsymbol{d}_{\mathcal{I}})$ is a diagonal matrix of finite pairwise dissimilarities between $\boldsymbol{x}_i$ and the points in $\mathcal{X}_{\mathcal{I}}$. We first outline our motivation for the choice of penalty function, and then discuss the definition of $\hat{X}_{\mathcal{I}}$ and the choice of d_{ij} in the next paragraph. The use of both an L_1 and an L_2-norm in (5) is motivated by the elastic net formulation (Zou and Hastie 2005). The L_1-norm penalty promotes solutions in which coefficients of dissimilar points are zero. The L_2-norm penalty encourages what is known as the *grouping effect*: if a group of points induces a similar approximation error, then either all points in the group are represented in $\boldsymbol{\beta}_i^\star$, or none is. This is desirable for subspace clustering, because the points in such a group should belong to the same subspace. In this case, if this subspace is different from the one $\boldsymbol{x}_i$ belongs to, then all points should be assigned a coefficient of zero. If instead the group of points are from the same subspace as $\boldsymbol{x}_i$, then it is beneficial to connect $\boldsymbol{x}_i$ to all of them, because this increases the probability that points from each subspace will belong to a single connected component of the graph defined by the affinity matrix A.

Spectral subspace clustering algorithms commonly normalise the points in $\mathcal{X}$ to have unit L_2-norm prior to estimating the coefficient vectors (Elhamifar and Vidal 2013; You et al. 2016). The simple example in Fig. 1 illustrates that this normalisation tends to increase cluster separability. In the following, we denote $\bar{\boldsymbol{x}}_i = \boldsymbol{x}_i/\|\boldsymbol{x}_i\|_2$. However, projecting the data onto the unit sphere has important implications for the WSSR problem. In (5) we want the two conflicting objectives of minimising the approximation error and selecting a few nearby points to be separate. Figure 2 contains an example that shows that this is not true after projecting onto the unit sphere. In Fig. 2, the point closest to $\bar{\boldsymbol{x}}_i$ on the unit sphere is $\bar{\boldsymbol{x}}_1$. The direction of $\bar{\boldsymbol{x}}_i$ can be perfectly approximated by a convex combination of $\bar{\boldsymbol{x}}_1$ and $\bar{\boldsymbol{x}}_2$, but all convex combinations $\alpha\bar{\boldsymbol{x}}_1 + (1-\alpha)\bar{\boldsymbol{x}}_2$ with $\alpha \in (0, 1)$ have an L_2-norm less than one. Since the cardinality of $\boldsymbol{\beta}_i$ affects the length of the approximation, it affects both the penalty term and the approximation error. A simple solution to address this problem is to scale every point $\boldsymbol{x}_j$ with $j \in \mathcal{I}$ such that $\hat{\boldsymbol{x}}_j^i = t_j^i\boldsymbol{x}_j$ lies on the hyperplane perpendicular to the unit sphere at $\bar{\boldsymbol{x}}_i$, $\hat{\boldsymbol{x}}_j^i \in \{\hat{\boldsymbol{x}} \in \mathbb{R}^P \mid \hat{\boldsymbol{x}}^\top\bar{\boldsymbol{x}}_i = 1\}$. Note that this implies that if $\boldsymbol{x}_j^\top\boldsymbol{x}_i < 0$, then t_j^i is negative. An inspection of Fig. 1 suggests that this is sensible. An appropriate measure of pairwise dissimilarity given the aforementioned preprocessing steps is the inverse absolute cosine similarity,

$$d_{ij} = \|\boldsymbol{x}_i\|_2\|\boldsymbol{x}_j\|_2/(\boldsymbol{x}_i^\top\boldsymbol{x}_j) = |\bar{\boldsymbol{x}}_i^\top\bar{\boldsymbol{x}}_j|^{-1}. \tag{6}$$

Since d_{ij} is infinite when $\boldsymbol{x}_j^\top\boldsymbol{x}_i = 0$, such points could never be assigned a non-zero coefficient, therefore they are excluded from $\hat{X}_{\mathcal{I}}$.

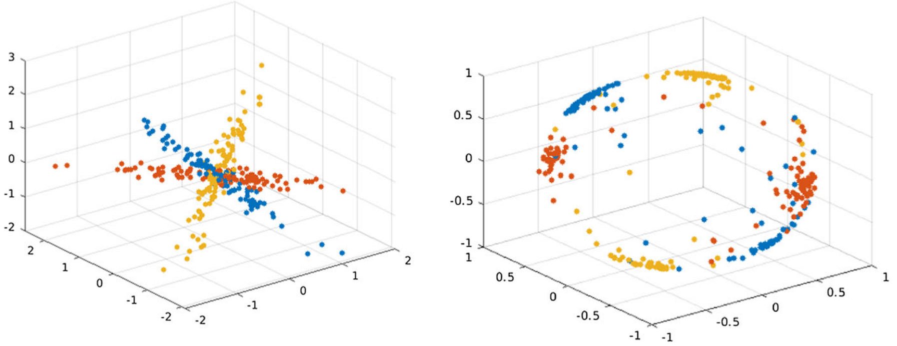
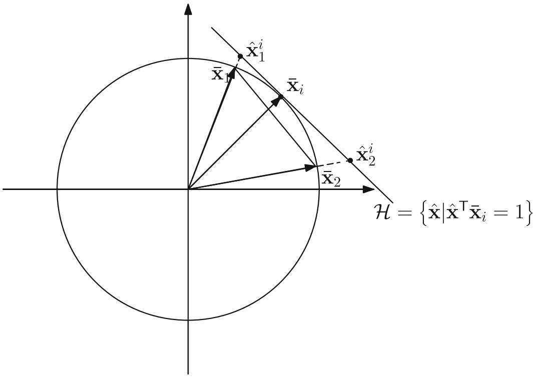


Fig. 1 An illustration of the data normalisation step, and the rationale for using the inverse cosine similarity as the dissimilarity measure. *Left*: The original data points. *Right*: The data points that have been normalised to lie on the unit sphere

Fig. 2 A geometric illustration of the necessity for stretching points in X



We now return to the optimisation problem in (5). Due to the constraint $\boldsymbol{\beta}_i \in \Delta^{|\mathcal{I}|}$ and the fact that $\boldsymbol{d}_{\mathcal{I}} > 0$, $\|D_{\mathcal{I}}\boldsymbol{\beta}_i\|_1 = \boldsymbol{d}_{\mathcal{I}}^\top \boldsymbol{\beta}_i$. This implies that the objective function is a quadratic, and the minimisation problem in (5) is equivalent to the one below,

$$\min_{\boldsymbol{\beta}_i} \frac{1}{2} \boldsymbol{\beta}_i^\top (\hat{X}_{\mathcal{I}}^\top \hat{X}_{\mathcal{I}} + \xi D_{\mathcal{I}}^2) \boldsymbol{\beta}_i + (\rho \boldsymbol{d}_{\mathcal{I}} - \hat{X}_{\mathcal{I}}^\top \hat{\boldsymbol{x}}_i)^\top \boldsymbol{\beta}_i, \quad \text{s.t.} \quad \boldsymbol{\beta}_i \geq 0, \quad \boldsymbol{\beta}_i^\top \mathbf{1} = 1. \tag{7}$$

For any $\xi > 0$, the above objective function is guaranteed to be strictly convex, therefore WSSR corresponds to a Quadratic Programme (QP).

The choice of ρ in (7) is critical to obtain an affinity matrix that accurately captures the cluster structure. The ridge penalty parameter, ξ, is typically assigned a small value, e.g. 10^{-4} (Gaines et al. 2018). For "large" ρ, the optimal solution assigns a coefficient of one to the nearest neighbour of $\bar{\boldsymbol{x}}_i$ and all other coefficients are zero. This is clearly undesirable. Setting this parameter is complicated by the fact that an appropriate choice of ρ differs for each $\bar{\boldsymbol{x}}_i$. Lemma 1 is a result which can be used to obtain a lower bound on ρ such that the solution of (7) is the "nearest neighbour" approximation. The proof of this lemma, as well as a second lemma which establishes its geometric interpretation can be found in Appendix B.

Lemma 1 *Let (j) denote the index of the j-th nearest neighbour of $\bar{\boldsymbol{x}}_i$, and $\hat{\boldsymbol{x}}_i^{(j)}$ denote the j-th nearest neighbour of $\bar{\boldsymbol{x}}_i$. Assume that $\hat{\boldsymbol{x}}_i^{(1)}$ is unique and that $\hat{\boldsymbol{x}}_i^{(1)} \neq \bar{\boldsymbol{x}}_i$. We also assume that the pairwise dissimilarities satisfy:*

$$\|\bar{\boldsymbol{x}}_i - \hat{\boldsymbol{x}}_i^{(j)}\|_2 > \|\bar{\boldsymbol{x}}_i - \hat{\boldsymbol{x}}_i^{(k)}\|_2 \Rightarrow d_{ij} > d_{ik},$$
$$\|\bar{\boldsymbol{x}}_i - \hat{\boldsymbol{x}}_i^{(j)}\|_2 = \|\bar{\boldsymbol{x}}_i - \hat{\boldsymbol{x}}_i^{(k)}\|_2 \Rightarrow d_{ij} = d_{ik}.$$

If

$$\boldsymbol{e}_1 = [1, 0, \dots, 0]^T = \arg\min_{\boldsymbol{\beta}_i \in \Delta^{|\mathcal{I}|}} \frac{1}{2}\boldsymbol{\beta}_i^T(\hat{X}_{\mathcal{I}}^T\hat{X}_{\mathcal{I}} + \xi D_{\mathcal{I}}^T D_{\mathcal{I}})\boldsymbol{\beta}_i + (\rho - \hat{X}_{\mathcal{I}}^T\bar{\boldsymbol{x}}_i)^T\boldsymbol{\beta}_i,$$

then

$$\rho > \max\left\{0, \max_{j\in\{2,\dots,|\mathcal{I}|\}} \frac{(\hat{\boldsymbol{x}}_i^{(1)} - \hat{\boldsymbol{x}}_i^{(j)})^T(\hat{\boldsymbol{x}}_i^{(1)} - \bar{\boldsymbol{x}}_i) + \xi(d_i^{(1)})^2}{d_i^{(j)} - d_i^{(1)}}\right\}. \tag{8}$$

Note that our definition of pairwise dissimilarities satisfies the requirements of the lemma. The proof uses directional derivatives and the convexity of the objective function. In effect, Lemma 1 states that there are cases in which the nearest-neighbour approximation is optimal for all $\rho > 0$. This occurs when it is possible to define a hyperplane that contains $\hat{\boldsymbol{x}}_i^{(1)}$, the nearest neighbour of $\bar{\boldsymbol{x}}_i$, and separates the column vectors in $\hat{X}_{\mathcal{I}}$ from $\bar{\boldsymbol{x}}_i$. In all other cases, there exists a positive value of ρ such that $\boldsymbol{\beta}_i^\star$ has cardinality greater than one.

We close this section by outlining a simple proximal gradient descent algorithm (Parikh and Boyd 2014) that is faster for large instances of the problem than standard QP solvers. To this end, we first express (7) as an unconstrained minimisation problem through the use of an indicator function,

$$\min_{\boldsymbol{\beta}_i} f(\boldsymbol{\beta}_i) + \mathbb{1}_{\Delta^{|\mathcal{I}|}}(\boldsymbol{\beta}_i),$$

where $f(\boldsymbol{\beta}_i)$ is the quadratic function in Eq. (7), and $\mathbb{1}_{\Delta^{|\mathcal{I}|}}(\boldsymbol{\beta}_i)$ is zero for $\boldsymbol{\beta}_i \in \Delta^{|\mathcal{I}|}$ and infinity otherwise. At each iteration, proximal gradient descent updates $\boldsymbol{\beta}_i^t$ by projecting onto the unit simplex a step of the standard gradient descent,

$$\boldsymbol{\beta}_i^{t+1} = \arg\min_{\boldsymbol{\beta}\in\Delta^{|\mathcal{I}|}} \frac{1}{2}\|\boldsymbol{\beta} - \boldsymbol{\beta}_i^t + \eta^t\nabla f(\boldsymbol{\beta}_i^t)\|_2^2,$$

where η^t is the step size at iteration t. Projecting onto $\Delta^{|\mathcal{I}|}$ can be achieved via a simple algorithm with complexity $\mathcal{O}(|\mathcal{I}|\log(|\mathcal{I}|))$ (Wang and Carreira-Perpinán 2013).

4 Active learning and constrained clustering

In this section, we describe the process of identifying informative points to query (active learning), and then updating the clustering model to accommodate the most recent labels (constrained clustering). The constrained clustering algorithm described in this section can also be used if the labels for a subset of points were available at the start of the learning process.

We adopt the active learning strategy of Peng and Pavlidis (2019), that queries the points whose labelling information is expected to induce the largest decrease in the reconstruction error function (defined in (2)). Let $\mathcal{U} \subseteq \{1, \ldots, N\}$ denote the set of indices of the unlabelled points, and $\mathcal{L} \subseteq \{1, \ldots, N\}$ denote the set of indices of labelled points. Furthermore, let $\{c_i\}_{i=1}^N$ denote the cluster assignment of each point, and $\{l_i\}_{i \in \mathcal{L}}$ the class labels for the labelled points. To quantify the expected decrease in the reconstruction error after obtaining the label of $\boldsymbol{x}_i$ with $i \in \mathcal{U}$, we estimate two quantities. The first is the decrease in the reconstruction error that can be achieved if $\boldsymbol{x}_i$ is removed from its currently assigned cluster c_i. This is measured by the function $U_1(\boldsymbol{x}_i, V_{c_i})$, where V_{c_i} is a matrix containing a basis for cluster c_i. The second is the increase in the reconstruction error due to the addition of $\boldsymbol{x}_i$ to a different cluster c_i'. This is measured by the function $U_2(\boldsymbol{x}_i, V_{c_i'})$. To estimate U_2 we assume that c_i' is the cluster that is the second nearest to $\boldsymbol{x}_i$ in terms of reconstruction error. This assumption is not guaranteed to hold but it is valid in the vast majority of cases. According to Peng and Pavlidis (2019), the most informative point to query is defined as,

$$\boldsymbol{x}_i^\star = \arg\max_{i \in \mathcal{U}} \left\{ U_1(\boldsymbol{x}_i, V_{c_i}) - U_2(\boldsymbol{x}_i, V_{c_i'}) \right\}. \tag{9}$$

The difficulty in calculating $U_1(\boldsymbol{x}_i, V_{c_i})$ and $U_2(\boldsymbol{x}_i, V_{c_i'})$ is that one needs to account for the fact that a change in the cluster assignment of $\boldsymbol{x}_i$ affects both V_{c_i} and $V_{c_i'}$. Recall that (irrespective of the choice of the clustering algorithm), the basis V_k for each cluster (subspace) k is computed by performing Principal Component Analysis (PCA) on the set of points assigned to this cluster. The advantage of the approach proposed by Peng and Pavlidis (2019) is that this is recognised, and a computationally efficient method to approximate U_1 and U_2 is proposed. In particular, using perturbation results for PCA (Critchley 1985), a first-order approximation of the changes in V_{c_i} and $V_{c_i'}$ are computed at a cost of $\mathcal{O}(P)$ (compared to the cost of PCA which is $\mathcal{O}(\min\{N_k P^2, N_k^2 P\})$, where N_k is the number of points in cluster k).

Once the labels of the queried points are obtained we proceed to the constrained clustering stage in which we update the cluster assignment to accommodate the new information. The first step in our approach modifies pairwise dissimilarities between points in a manner similar to the work of Li et al. (2017). Specifically, for each $\boldsymbol{x}_i \in \mathcal{X}$ we update all pairwise dissimilarities $d_{ij} \in \boldsymbol{d}_{\mathcal{I}}$ according to,

$$d_{ij} = \begin{cases} \frac{\|\boldsymbol{x}_i\|_2 \|\boldsymbol{x}_j\|_2}{\boldsymbol{x}_i^\top \boldsymbol{x}_j} e^{1-2\cdot\mathbb{1}(l_i=l_j)} + \alpha \mathbb{1}(l_i \neq l_j), & \text{if } i, j \in \mathcal{L}, \\ \frac{\|\boldsymbol{x}_i\|_2 \|\boldsymbol{x}_j\|_2}{\boldsymbol{x}_i^\top \boldsymbol{x}_j} + \alpha \mathbb{1}(c_i \neq c_j), & \text{otherwise.} \end{cases} \tag{10}$$

The first fraction is the dissimilarity measure in the absence of any label information as defined in (6). If the labels of both $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are known and they are different then the dissimilarity is first scaled by e and a constant $\alpha \in [0, 1]$ is added. If $l_i = l_j$, then the original dissimilarity is scaled by e^{-1}. If the label of either $\boldsymbol{x}_i$ or $\boldsymbol{x}_j$ is unknown then no scaling is applied, but if in the previous step the two points were assigned to different clusters then their dissimilarity is increased by α. The term α quantifies the confidence of the algorithm in the previous cluster assignment. A simple and effective heuristic is to assign α equal to the proportion of labelled data.

After updating pairwise dissimilarities through (10) we update the coefficient vectors for each point by solving the problem in (7). The resulting affinity matrix is the input to the normalised spectral clustering of Ng et al. (2002). This cluster assignment is not guaranteed to satisfy all the constraints. To ensure constraint satisfaction, we use this cluster assignment as the initialisation for the K-Subspace Clustering with Constraints (KSCC) algorithm (Peng and Pavlidis 2019). KSCC is an iterative algorithm to solve the following optimisation problem,

$$\min_{V_1,\ldots,V_K} \left\{ \sum_{i \in \mathcal{U}} \min_{k=1,\ldots,K} \|\boldsymbol{x}_i - V_k V_k^{\top} \boldsymbol{x}_i\|_2^2 + \min_{P \in \mathcal{P}(K)} \sum_{k=1}^{K} \sum_{\substack{j \in \mathcal{L}: \\ l_j = k}} \|\boldsymbol{x}_j - V_{P_k} V_{P_k}^{\top} \boldsymbol{x}_j\|_2^2 \right\}, \tag{11}$$

where $\mathcal{P}(K)$ is the set of all permutations of the cluster indices $\{1, \ldots, K\}$; while each permutation $P \in \mathcal{P}(K)$ is a vector $P = (P_1, \ldots, P_K)$ whose k-th element, P_k, indicates the cluster label associated with class k. The first term in (11) is the reconstruction error for the unlabelled points. The second term quantifies the reconstruction error for the labelled / queried points. To estimate this, we first need to match each cluster label to a unique class label. Each permutation $P \in \mathcal{P}(K)$ of the set $\{1, \ldots, K\}$ represents a unique matching of cluster to class labels. According to (11), for every class $k = 1, \ldots, K$, the reconstruction error of the labelled points of this class, $\{x_j : j \in \mathcal{L}, l_j = k\}$, is computed by projecting them onto the linear subspace (cluster) defined by the basis V_{P_k}. By minimising over all possible permutations $P \in \mathcal{P}(K)$, we identify the matching of cluster to class labels that achieves the smallest overall reconstruction error for the labelled points, subject to the constraint that labelled points from each class are assigned to a unique cluster. This is an instance of the *minimum weight perfect matching problem*, and can be solved by the Hungarian algorithm (Kuhn 1955) in $\mathcal{O}(K^3)$. Note that by design the minimisation problem is always feasible irrespective of the composition of the set $\mathcal{L}$.

KSCC is an iterative algorithm that monotonically reduces the value of the objective function. It therefore converges to the local minimum of (11) whose region of attraction contains the initial cluster assignment. The computational complexity of KSCC is the same as that of KSC, namely $\mathcal{O}(\min\{N_k P^2, N_k^2 P\})$. We summarise the active learning and constrained clustering framework in procedural form in Algorithm 1. We refer to this extended version of WSSR as **WSSR+**.

Algorithm 1: Active Learning and Constrained Clustering with WSSR (WSSR+)

```
Input  : WSSR-related parameters; Sets of ‘must-link’ and ‘cannot-link’ constraints: $\mathcal{S}_M, \mathcal{S}_C$;
         Penalty parameter: $\alpha$; Number of points to query in each iteration: $b$
% Active learning
- Query the $b$ most informative point according to (9)
% Constraint incorporation
For $\boldsymbol{x} \in \mathcal{X}$:
    1. Compute the updated weight vector $\boldsymbol{d}^\star$ according to (10)
    2. Normalise and stretch each column vector in $X$
    3. Solve the WSSR problem in (5)
    to obtain the coefficient vector $\boldsymbol{\beta}$
End
- Combine all $\boldsymbol{\beta}$s to obtain the coefficient matrix $B \in \mathbb{R}^{N \times N}$
- Apply normalised cut spectral clustering (Ng et al. 2002) to the data affinity matrix
        $A = \frac{1}{2}\left(|B| + |B|^\mathsf{T}\right)$        (12)
% Constraint satisfaction
- Enforce the constraint information using KSCC (Peng and Pavlidis 2019) and obtain the updated
  cluster labels
```

5 Experiments on synthetic data

In this section, we conduct experiments on synthetic data to evaluate the performance of WSSR under various settings. We compare to the following state-of-the-art spectral-based subspace clustering methods: SSC (Elhamifar and Vidal 2013), S3C (Li et al. 2017), ASSC (Li et al. 2018a), SSC-OMP (You et al. 2016), LSR (Lu et al. 2012), and SMR (Hu et al. 2014). In the set of competing algorithms, we also include SSR (Huang et al. 2013) to allow us to assess the extent to which WSSR improves performance over this algorithm. The purpose of using synthetic data is to evaluate the impact on clustering accuracy of (a) the angles between the linear subspaces (clusters); (b) the noise level; and (c) the subspace dimensionality.[2]

5.1 Parameter settings

We first report the parameter settings for all the algorithms considered. These settings are used in all the experiments reported in this paper. In WSSR, to estimate the coefficient vector for each point in the optimisation problem in (7) we consider only 10-nearest neighbourhoods (rather than all the $(N-1)$ points in the dataset). The L_1 and L_2 penalty parameters are set to be $\rho = 10^{-2}$ and $\xi = 10^{-4}$, respectively.

Both SSC and S3C have the settings of linear subspace and no outliers. For SSC, the penalty parameters ρ_η and ρ_z in (4) are computed as $\rho_\eta = \alpha_\eta/\mu_\eta$ and $\rho_z = \alpha_z/\mu_z$, where the αs are user-specified and the μs are data dependent as computed according to (14) in Elhamifar and Vidal (2013). We use the default parameter values of $\alpha_\eta = \alpha_z = 20$. There is a *hard* and a *soft* version of S3C. We use the soft version with all its

[2] Code for our proposed method is available at: https://github.com/hankuipeng/WSSR.

default parameter settings, as Li et al. (2017) report that this yields better clustering results. ASSC shares the same algorithmic implementation as SSC, except that the affine subspace setting is used. We set the maximum number of non-zero coefficients in SSC-OMP to ten to be consistent with the corresponding setting in WSSR. The OMP algorithm involves a termination threshold, which is set to 10^{-6}.

Unlike the previously discussed methods, LSR and SMR yield dense coefficient vectors. There are two versions of LSR, called LSR1 and LSR2. The first solves the LSR problem without allowing self-representation, that is all diagonal entries in the coefficient matrix are forced to be zero. In LSR2, this constraint is absent. We use LSR1 as Lu et al. (2012) report that it performs better in practice. Furthermore, by not allowing self-representation, LSR1 is directly comparable with all the other methods. There are also two different versions of SMR, called SMR-J1 and SMR-J2. These differ in how the affinity matrix, A, is constructed from the coefficient matrix, B. SMR-J1 defines A as in (12). We therefore use SMR-J1 as this is how all other methods define the affinity matrix. Finally, we set the SMR-J1 nearest neighbour parameter to ten to be consistent with the corresponding settings in WSSR and SSC-OMP, while all other parameter values are set to their default settings (Hu et al. 2014).

5.2 Varying angles between subspaces

In this set of experiments, we generate data from two 5-dimensional subspaces embedded in a 50-dimensional space. Each cluster contains 200 data points drawn from one of the subspaces. In addition, additive Gaussian noise with standard deviation $\sigma = 0.001$ is added to the data uniformly. We vary the angles between the two subspaces θ between 10 and 60 degrees. The smaller the angle between two subspaces, the more difficult the clustering problem is. We evaluate the performance of various algorithms under each setting using clustering accuracy, and the performance results are reported in Table 1. The best performance results are highlighted in bold, and the second best performance results are underlined.

A general trend across almost all methods is that accuracy increases as the angle between the two subspaces increases. WSSR achieves the best performance in all scenarios. Even when the angle between the two subspaces is only 10 degrees it achieves an accuracy close to 0.95, while for angles greater or equal to 30 degrees clustering accuracy is perfect. This can be attributed to the small standard deviation of the noise term, and as we discuss in the next subsection, performance deteriorates when σ increases. However the accuracy of most other methods, including SSC and several of its variants, is much lower, despite very little noise. S3C is the strongest competitor and achieves the second best performance in effectively all scenarios.

5.3 Varying noise levels

Next, we explore the effect of various noise levels on cluster performance. Data are generated from five linear subspaces (clusters). Each subspace contains 200 data points that lie in a 5-dimensional subspace within an ambient space of dimension 50. Gaus-

Table 1 Accuracy of various subspace clustering algorithms on synthetic data with varying angles between subspaces

	$\theta = 10$	$\theta = 20$	$\theta = 30$	$\theta = 40$	$\theta = 50$	$\theta = 60$
WSSR	**0.948**	**0.998**	**1.000**	**1.000**	**1.000**	**1.000**
SSR	0.517	0.550	0.532	0.510	0.547	0.647
SSC	0.537	0.667	0.825	0.868	0.782	0.892
S3C	0.892	0.885	0.873	0.890	0.785	0.907
ASSC	0.580	0.530	0.547	0.517	0.522	0.605
SSC-OMP	0.535	0.520	0.525	0.500	0.535	0.517
LSR1	0.767	0.790	0.797	0.845	0.787	0.845
SMR-J1	0.522	0.537	0.502	0.512	0.520	0.568

Table 2 Accuracy of various subspace clustering algorithms on synthetic data with varying noise levels. The first column is not highlighted as all methods have perfect cluster performance in the noise-free scenario

	$\sigma = 0.0$	$\sigma = 0.1$	$\sigma = 0.2$	$\sigma = 0.3$	$\sigma = 0.4$	$\sigma = 0.5$
WSSR	1.000	**1.000**	**0.996**	**0.980**	**0.942**	**0.817**
SSR	1.000	0.999	0.990	0.938	0.818	0.340
SSC	1.000	0.994	0.954	0.845	0.678	0.322
S3C	1.000	0.999	0.961	0.852	0.706	0.329
ASSC	1.000	0.990	0.941	0.830	0.616	0.351
SSC-OMP	1.000	0.992	0.710	0.420	0.253	0.268
LSR1	1.000	0.995	0.985	0.932	0.843	0.669
SMR-J1	1.000	**1.000**	0.990	0.953	0.877	0.765

sian noise, $\varepsilon \sim N(0, \sigma^2 I_P)$, is added to the full-dimensional data. Table 2 presents clustering accuracy of the different algorithms for $\sigma \in [0, 0.5]$.

It is worth noting that all methods enjoy perfect clustering accuracy in the noise-free scenario. The accuracy scores for all methods decrease as the noise variance increases, although the speed and magnitude of the performance degradation differs markedly. WSSR enjoys the best performance among all competing methods for all values of σ, and furthermore its performance degrades much more gradually as σ increases. It is also the only method that maintains an accuracy of over 0.9 for σ up to 0.4. It is worth noting that the performance of SSC-based methods (SSC, S3C, ASSC, SSC-OMP) degrades rapidly with increasing levels of noise. Indeed, all these methods have accuracy scores substantially less than 0.5 for $\sigma = 0.5$. In comparison, WSSR, LSR, and SMR, which use the Frobenius norm on the error matrix, or the L_2-norm on the reconstruction error, exhibit better performance.

Table 3 Accuracy of various subspace clustering algorithms on synthetic data with varying subspace dimensions

	$d = 10$	$d = 15$	$d = 20$	$d = 25$	$d = 30$	$d = 35$	$d = 40$	$d = 45$
WSSR	**0.931**	**0.964**	**0.961**	**0.970**	0.960	0.955	0.925	0.904
SSR	0.498	0.703	0.567	0.514	0.474	0.338	0.305	0.350
SSC	0.495	0.743	0.863	0.893	0.919	0.926	0.902	0.897
S3C	0.573	0.785	0.856	0.909	0.919	0.920	0.926	0.902
ASSC	0.449	0.730	0.847	0.902	0.914	0.927	0.905	0.883
SSC-OMP	0.305	0.398	0.532	0.706	0.659	0.617	0.507	0.483
LSR1	0.733	0.905	0.939	0.953	**0.966**	**0.960**	**0.952**	**0.937**
SMR-J1	0.905	0.925	0.939	0.951	0.941	0.921	0.878	0.842

5.4 Varying subspace dimensions

In this subsection we investigate the impact of the subspace dimensionality, d, on performance, under a fixed ambient space dimensionality, P. In this set of experiments, we fix the ambient space dimension to $P = 100$, and allow d to increase from ten to 45. All experiments involve data from five d-dimensional linear subspaces. As before Gaussian noise with $\sigma = 0.5$ is added to the data.

Table 3 reports the performance of the considered methods. The table shows that for the lowest values of d all considered methods perform relatively poorly. As d increases performance initially improves, but as we approach the maximum value of this parameter accuracy deteriorates. This happens because the combination of a relatively high variance for the noise term, and increasing d cause higher overlap among clusters. WSSR achieves very good relative performance especially for the lower values of d. For $d \geq 30$ LSR1 is the best performing method. Even for the higher values of d, however, WSSR achieves the best accuracy among the methods that estimate a sparse coefficient vector (with one exception for $d = 40$, where WSSR achieves an accuracy of 0.925, while S3C has an accuracy of 0.926). WSSR is affected by the higher cluster overlap, because it becomes increasingly likely that some of the 10-NNs of each point belong to different clusters. This can be overcome by increasing the number of NNs, but for consistency we use the same value, $k = 10$, throughout all the experiments in this paper.

6 Experiments on real data

In this section, we use real datasets to assess the comparative performance of WSSR on a range of real-world datasets including the MNIST database (LeCun et al. 1998), three Cancer gene datasets, and the Hopkins155 motion segmentation database.

Table 4 Median and standard deviation of clustering accuracy on the MNIST handwritten digits dataset over 20 replications with varying number of (randomly selected) clusters, K

	$K = 2$		$K = 3$		$K = 5$		$K = 8$		$K = 10$	
	Med	Std	Med	Std	Med	Std	Med	Std	Med	Std
WSSR	**1.000**	0.004	**0.993**	0.005	**0.992**	0.004	**0.979**	0.003	**0.980**	0.003
SSR	0.953	0.086	0.887	0.112	0.676	0.113	0.641	0.060	0.603	0.046
SSC	0.993	0.029	0.980	0.047	0.856	0.084	0.823	0.033	0.810	0.022
S3C	0.985	0.027	0.957	0.099	0.791	0.083	0.808	0.041	0.805	0.029
ASSC	0.988	0.014	0.973	0.023	0.948	0.051	0.870	0.055	0.844	0.037
SSC-OMP	0.983	0.014	0.972	0.023	0.949	0.039	0.899	0.040	0.863	0.031
LSR	0.980	0.108	0.680	0.106	0.858	0.071	0.816	0.043	0.785	0.020
SMR	0.993	0.036	0.982	0.024	0.946	0.060	0.864	0.052	0.828	0.032

6.1 MNIST data

The MNIST handwritten digits database contains greyscale images of handwritten digits from 0 to 9 (LeCun et al. 1998), and has been widely used in the machine learning literature to benchmark the performance of supervised and unsupervised learning methods. In the context of subspace clustering, You et al. (2016) used this dataset to demonstrate the effectiveness of SSC-OMP.

The original MNIST database contains 60,000 points in $P = 3472$ dimensions. The data is organised in ten clusters, with each cluster correspoding to a different digit. We adopt the experimental design proposed by You et al. (2016) and conduct two sets of experiments. The first aims to investigate the effect of the number of clusters, K, on performance. To this end, we randomly select $K \in \{2, 3, 5, 8, 10\}$ clusters out of the ten digits, and from each chosen cluster (digit) we sample uniformly at random 100 points. The data is then projected onto the first 200 principal components. The rationale for projecting onto 200 dimensions is that when $K = 2$ we obtain a sample of size 200, and the maximum dimensionality of the subspace spanned by these vectors is 200. The second set of experiments proposed by You et al. (2016), is designed to investigate the effect of the number of data points per cluster, N_k, on performance. In these experiments $K = 10$, while $N_k \in \{50, 100, 500, 1000, 2000\}$ points are selected uniformly at random from each cluster (digit). As before, the data is first projected onto the first 200 principal components, and then clustering is applied. For every choice of K (first setup) and N_k (second setup) we sample 20 datasets and apply the considered clustering algorithms on each of them.

Table 4 reports the median and standard deviation of clustering accuracy for the first set of experiments in which K varies. WSSR always achieves the highest median performance and lowest standard deviation, while the improvement it achieves over competing methods increases as the clustering problem becomes more difficult (K increases). Note that WSSR achieves a substantial improvement over SSR, which is the worst performing method for $K \geq 5$. Most of the competing algorithms achieve excellent accuracy for the smaller values of K, but their performance deteriorates

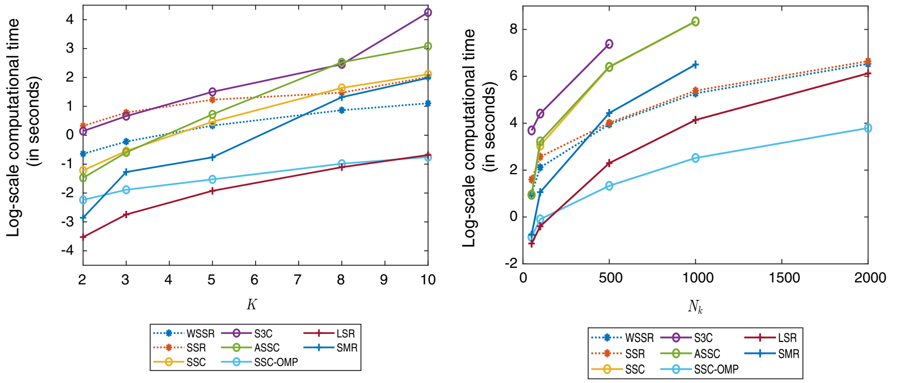
Table 5 Median clustering accuracy along with the standard deviations on the MNIST handwritten digits data across 20 replications with varying number of points per cluster

	$N_k = 50$		$N_k = 100$		$N_k = 500$		$N_k = 1000$		$N_k = 2000$	
	Med	Std	Med	Std	Med	Std	Med	Std	Med	Std
WSSR	**0.963**	0.033	**0.980**	0.003	**0.990**	0.001	**0.992**	0.000	**0.994**	0.000
SSR	0.551	0.038	0.610	0.046	0.855	0.020	0.865	0.035	0.871	0.046
SSC	0.779	0.042	0.812	0.030	0.837	0.008	0.851	0.009	–	–
S3C	0.786	0.044	0.822	0.042	0.829	0.012	–	–	–	–
ASSC	0.822	0.037	0.846	0.038	0.824	0.011	0.829	0.009	–	–
SSC-OMP	0.834	0.043	0.877	0.028	0.845	0.044	0.851	0.049	0.854	0.047
LSR	0.661	0.040	0.736	0.032	0.797	0.010	0.802	0.006	0.810	0.003
SMR	0.764	0.033	0.817	0.048	0.890	0.034	0.889	0.029	–	–

considerably as K increases. SSC and S3C exhibit very similar performance, with SSC achieving slightly higher median accuracy. SSC-OMP appears to have a clear advantage over the other SSC-based methods for larger K. ASSC performs similarly to SSC-OMP in this dataset. LSR compares unfavourably with most other methods, both in terms of median performance, and in terms of performance variability especially for small K.

Table 5 reports the results for the second set of experiments where K is always ten, while N_k, the number of points per cluster, varies. Note that for the largest N_k values the experiments with SSC, S3C, and SMR did not finish their 20 replications within 24 hours. We report those as dashed lines in Table 5. For all considered methods, as N_k increases median accuracy improves, while performance variability decreases (with the exception of SSC-OMP). WSSR achieves the highest median accuracy across all settings, and exhibits the least performance variability. SSC-OMP is the algorithm with the second highest median performance when $N_k = 50, 100$, while for larger N_k values SMR is second best.

All our experiments on the MNIST database were performed on a cloud computing machine with four CPU cores and 8 GB of RAM. All the algorithms were implemented in MATLAB. Figure 3 illustrates computational times in log-scale for the considered algorithms for each of the two sets of experiments on the MNIST dataset. Both figures indicate that S3C is the most computationally intensive method. SSC and ASSC have similar computational times because they are based on the same optimisation framework. The most computationally efficient methods are SSC-OMP and LSR. SSC-OMP is the SSC variant that is most suitable for large-scale problems, while LSR admits a closed-form solution. For smaller problem sizes, the computational time for WSSR is comparable to that of SSC and ASSC, but the comparison becomes more favourable to WSSR for larger values of N_k and K.



(a) Varying number of clusters (K).

(b) Varying number of points per cluster (N_k).

Fig. 3 Median running times (in log-scale of seconds) of different algorithms on the MNIST handwritten digits data

Table 6 Summary information on the gene expression datasets

Datasets	No. of points (N)	No. of features (P)	No. of clusters (K)
St. Jude Leukemia	248	985	6
Lung Cancer	197	1000	4
Novartis BPLC	103	1000	4

6.2 Gene expression datasets

Gene expression data have been shown to exhibit a grouping structure, in which each subtype of gene expression forms a different linear subspace (McWilliams and Montana 2014). The following three datasets have been previously adopted to demonstrate the effectiveness of subspace clustering: *St. Jude Leukemia*, *Lung Cancer*, and *Novartis BPLC* (Li et al. 2018b). A summary of their basic characteristics is provided in Table 6.

Table 7 reports the performance of all clustering algorithms on these datasets. WSSR achieves the best performance on the *St. Jude Leukemia* and *Novartis BPLS* datasets, while on the *Lung Cancer* dataset it achieves an accuracy in excess of 0.9. Its performance on the last dataset is 5.7% lower compared to the best performing algorithm (S3C) and 0.5% lower compared to the second best (ASSC). In these gene expression datasets SSC-based methods performed well, while the performance of LSR and SMR was considerably lower.

6.3 Hopkins155 motion segmentation data

A fundamental problem in computer vision is to infer structures and movements of three-dimensional objects from a video sequence. Video sequences often contain multiple objects moving independently in a scene, captured from a potentially moving

Table 7 Clustering accuracy of subspace clustering algorithms on three cancer gene expression datasets

	WSSR	SSR	SSC	S3C	ASSC	SSC-OMP	LSR	SMR
St. Jude Leukemia	**0.960**	<u>0.935</u>	0.879	0.931	0.895	0.762	0.448	0.891
Lung Cancer	0.909	0.863	0.914	**0.964**	<u>0.914</u>	0.843	0.706	0.756
Novartis BPLS	**0.990**	0.922	0.913	0.932	<u>0.951</u>	0.709	0.291	0.777

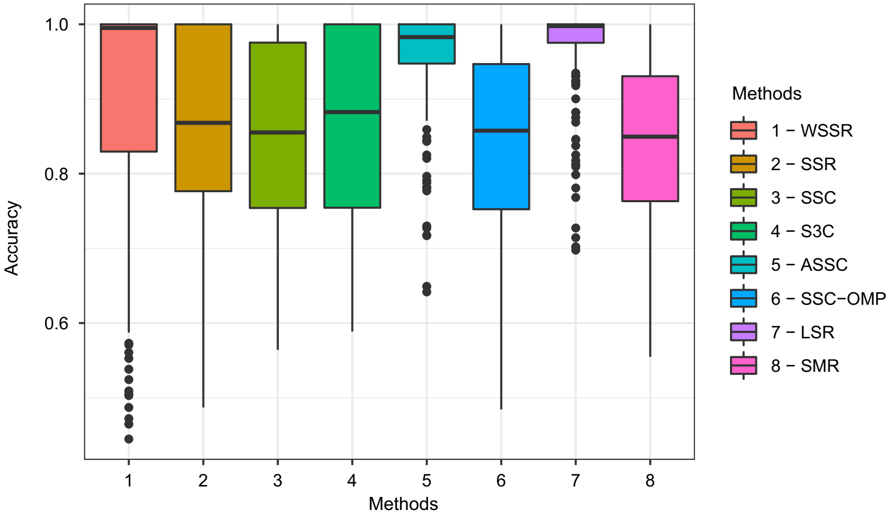


Fig. 4 Performance of subspace clustering methods on Hopkins155 database

camera. Thus, an important initial step in the analysis of video sequences is the motion segmentation problem: Given a set of feature points that are tracked through a sequence of video frames, cluster the trajectories of those points according to different motions (Rao et al. 2010). Using the affine camera model, this problem can be cast as the problem of segmenting samples drawn from multiple linear subspaces.

The Hopkins155 motion segmentation database (Tron and Vidal 2007), which contains 155 video sequences, has been widely used in computer vision to benchmark subspace clustering algorithms. Figure 4 presents boxplots of accuracy across the 155 datasets, for all the considered algorithms. LSR achieves the highest median accuracy and overall the most stable performance. ASSC also exhibits a very high median accuracy and stable performance. WSSR achieves the second highest median accuracy (higher than ASSC) but its performance is more variable compared to LSR and ASSC. Moreover, it performs relatively poorly on specific datasets. With the exception of ASSC other SSC variants achieve a median performance considerably lower than WSSR, and exhibit similar variability. The same is true of SMR and SSR.

7 Constrained clustering with active learning

In this section, we assess the performance of the constrained clustering with active learning framework, WSSR+, introduced in Sect. 4. In the previous subsection, we saw that WSSR achieved low accuracy on specific video sequences of the Hopkins155 motion segmentation dataset. In this section, we select the sequences on which WSSR did not produce perfect performance, to assess the benefits of incorporating label information through WSSR+.

We compare WSSR+ to three state-of-the-art constrained clustering methods: PLCC (Liu et al. 2018), CSP (Wang et al. 2014), and LCVQE (Pelleg and Baras 2007). PLCC has one tuning parameter λ that controls the weight assigned to the constrained information. We use the default setting of $\lambda = 10^4$, as it is the recommended default setting by the authors of Liu et al. (2018). CSP and LCVQE do not involve any user-specified parameter. Note that all three algorithms describe how to incorporate label information into a clustering model. Thus they can be used in conjunction with any clustering algorithm. To ensure a fair comparison, we always use WSSR as the underlying clustering algorithm and select the queried points through the active learning strategy of Peng and Pavlidis (2019). The latter choice is made because Hopkins155 datasets are known to exhibit subspace clustering structure.

Figure 5 presents boxplots of the clustering accuracy with respect to the proportion of labelled data, which ranges between zero and 0.5. When this proportion is zero, there are no constraints due to labelled data, and hence the corresponding four boxplots illustrate the clustering accuracy of WSSR on the selected datasets. (This is the reason that these four boxplots are identical.) We include the case of no labels because a natural benchmark for any constrained clustering algorithm is its performance relative to the fully unsupervised problem. As the first four boxplots in Fig. 5 show, WSSR achieves a median accuracy close to 0.85, but on some datasets accuracy is lower than 0.5.

Figure 5 shows that on these datasets incorporating label information through CSP and PLCC initially degrades performance. In fact it is not until the proportion of labelled data reaches 0.4 and 0.5 that the performance of CSP and PLCC (respectively) is not worse than in the fully unsupervised case. WSSR+ and LCVQE improve performance, both in terms of the median and the interquartile range, as the proportion of labelled data increases. When the proportion of labelled data is higher or equal to 0.3, WSSR+ clearly outperforms LCVQE, especially in terms of performance variability. Note that the high variability in performance across all methods when the proportion of labelled data is small is due to the diverse range of datasets considered, rather than an inherent performance variability of the considered constrained clustering algorithms.

8 Conclusions and future work

In this work, we proposed a subspace clustering method called Weighted Sparse Simplex Representation (WSSR), which relies on estimating an affinity matrix by approximating each data point as a sparse convex combination of nearby points. We derived a lemma that provides a lower bound that can be used to select the critical

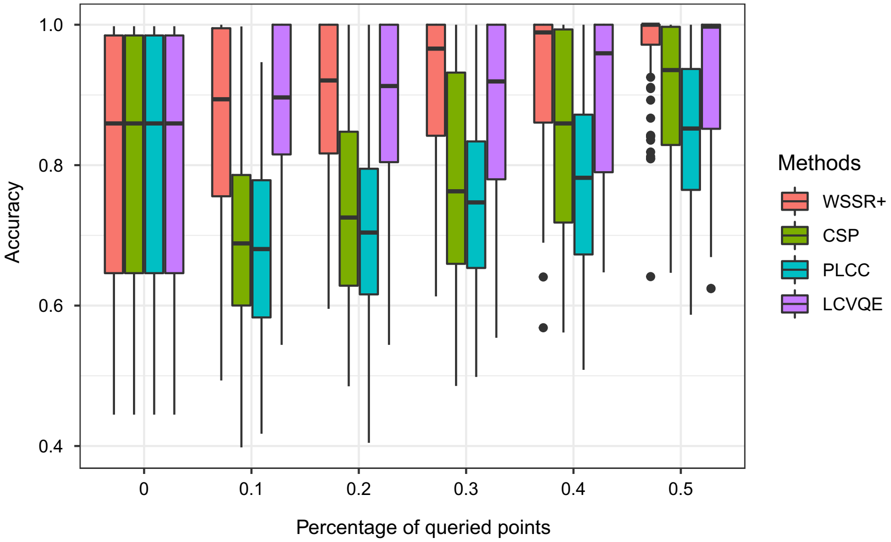


Fig. 5 Clustering accuracy of constrained clustering methods on the selected Hopkins155 datasets with respect to varying proportions of labelled data. Queried points are selected through the active learning strategy of Peng and Pavlidis (2019)

penalty parameter, that controls the degree of sparsity. Extensive experimental results show that WSSR is competitive with state-of-the-art subspace clustering methods. We extended WSSR to the problem of constrained clustering, to accommodate cases where external information about the actual assignment of some points is available. Our constrained clustering approach combines the strengths of spectral-based methods, and constrained K-subspace clustering to ensure that the resulting clustering is accurate, and consistent with the label information. We also discussed an appropriate active learning strategy that aims to query points whose label information can maximally improve the quality of the overall subspace clustering model. Experiments on motion segmentation datasets, in which the unsupervised WSSR algorithm does not perform well, document the effectiveness of the proposed approach in incorporating label information.

In this work we focused on subspace clustering. For this problem, the inverse cosine similarity is a natural proximity measure after projecting the data onto the unit sphere. It would be interesting to explore other affinity measures that can potentially be used to capture data from manifolds. The proposed approach was developed under the assumption that the label information is always correct. This assumption can be violated in certain applications. A challenging future research direction is to design methods that can accommodate uncertainty in the validity of the observed labels, especially when the proportion of labelled data is small.

Acknowledgements This paper is based on work completed while Hankui Peng was a PhD student in the EPSRC funded (grant number EP/L015692/1) STOR-i Centre for Doctoral Training at Lancaster University. Hankui Peng acknowledges the Office for National Statistics (ONS) for financial support. The authors would also like to thank the associate editor and two anonymous reviewers for comments that helped us improve the paper substantially.

A KKT conditions for optimality

In this section, we derive the Karush-Kuhn-Tucker (KKT) conditions for our proposed WSSR problem formulation. For any optimisation problem with differentiable objective and constraint functions for which strong duality holds, the KKT conditions are necessary and sufficient conditions for obtaining the optimal solution (Boyd and Vandenberghe 2004).

Firstly, the stationarity condition in the KKT conditions states that when optimality is achieved, the derivative of the Lagrangian with respect to $\boldsymbol{\beta}_i$ is zero. The Lagrangian $L\left(\boldsymbol{\beta}_i; \lambda_i, \boldsymbol{\mu}_i\right)$ associated with the WSSR problem in (5) can be expressed as

$$\begin{aligned} L(\boldsymbol{\beta}_i; \lambda_i, \boldsymbol{\mu}_i) = \frac{1}{2}\boldsymbol{\beta}_i^{\mathsf{T}} \left(\hat{X}_{\mathcal{I}}^{\mathsf{T}}\hat{X}_{\mathcal{I}} + \xi D_{\mathcal{I}}^{\mathsf{T}} D_{\mathcal{I}}\right)\boldsymbol{\beta}_i + \left(\rho \boldsymbol{d}_{\mathcal{I}} - \hat{X}_{\mathcal{I}}^{\mathsf{T}}\hat{\boldsymbol{x}}_i\right)^{\mathsf{T}}\boldsymbol{\beta}_i \\ -\boldsymbol{\mu}_i^{\mathsf{T}}\boldsymbol{\beta}_i + \lambda_i\left(\boldsymbol{\beta}_i^{\mathsf{T}}\mathbf{1} - 1\right), \end{aligned} \tag{13}$$

in which λ_i is a scalar and $\boldsymbol{\mu}_i$ is a vector of non-negative Lagrange multipliers. Thus, the stationarity condition gives the following

$$\nabla L(\boldsymbol{\beta}_i; \lambda_i, \boldsymbol{\mu}_i) = \left(\hat{X}_{\mathcal{I}}^{\mathsf{T}}\hat{X}_{\mathcal{I}} + \xi D_{\mathcal{I}}^{\mathsf{T}} D_{\mathcal{I}}\right)\boldsymbol{\beta}_i - \hat{X}_{\mathcal{I}}^{\mathsf{T}}\boldsymbol{x}_i + \rho\boldsymbol{d}_{\mathcal{I}} + \lambda_i\mathbf{1} - \boldsymbol{\mu}_i = \mathbf{0}, \tag{14}$$

which can be simplified to

$$\boldsymbol{\beta}_i = \left(\hat{X}_{\mathcal{I}}^{\mathsf{T}}\hat{X}_{\mathcal{I}} + \xi D_{\mathcal{I}}^{\mathsf{T}} D_{\mathcal{I}}\right)^{-1}\left(\hat{X}_{\mathcal{I}}^{\mathsf{T}}\hat{\boldsymbol{x}}_i + \boldsymbol{\mu}_i - \rho\boldsymbol{d}_{\mathcal{I}} + \lambda_i\mathbf{1}\right). \tag{15}$$

Since all diagonal entries in $D_{\mathcal{I}}$ are positive, the matrix $\left(\hat{X}_{\mathcal{I}}^{\mathsf{T}}\hat{X}_{\mathcal{I}} + \xi D_{\mathcal{I}}^{\mathsf{T}} D_{\mathcal{I}}\right)$ is full rank thus invertible.

Secondly, the KKT conditions state that any primal optimal $\boldsymbol{\beta}_i$ must satisfy both the equality and inequality constraints in (5). In addition, any dual optimal λ_i and $\boldsymbol{\mu}_i$ must satisfy the dual feasibility constraint $\boldsymbol{\mu}_i \geq \mathbf{0}$. Thirdly, the KKT conditions state that $\mu_{ij}\beta_{ij} = 0$ for all $j \in \mathcal{I}$ for any primal optimal $\boldsymbol{\beta}_i$ and dual optimal $\boldsymbol{\mu}_i$ when strong duality holds. This is called the complementary slackness condition. To put everything together, when strong duality holds, any primal optimal $\boldsymbol{\beta}_i$ and any dual optimal λ_i and $\boldsymbol{\mu}_i$ must satisfy the following KKT conditions:

$$\begin{aligned}
&\text{Stationarity: } \boldsymbol{\beta}_i = \left(\hat{X}_{\mathcal{I}}^{\mathsf{T}}\hat{X}_{\mathcal{I}} + \xi D_{\mathcal{I}}^{\mathsf{T}} D_{\mathcal{I}}\right)^{-1}\left(\hat{X}_{\mathcal{I}}^{\mathsf{T}}\hat{\boldsymbol{x}}_i + \boldsymbol{\mu}_i - \rho \boldsymbol{d}_{\mathcal{I}} + \lambda_i \mathbf{1}\right),\\
&\text{Equality constraint: } \boldsymbol{\beta}_i^{\mathsf{T}}\mathbf{1} = 1,\\
&\text{Inequality constraint: } \boldsymbol{\beta}_i \geq \mathbf{0},\\
&\text{Dual feasibility: } \boldsymbol{\mu}_i \geq \mathbf{0},\\
&\text{Complementary slackness: } \mu_{ij}\beta_{ij} = 0, \quad \forall\ j \in \mathcal{I}.
\end{aligned}$$

B Necessary and sufficient conditions for the trivial solution

In Sects. B.1 and B.2, we investigate the necessary and sufficient conditions under which the trivial solution is obtained. That is, only the most similar point is chosen and has coefficient one.

B.1 Necessary condition for the trivial solution

Consider the WSSR problem formulation in (5) for a given $\boldsymbol{x} \in \mathcal{X}$, which we restate below:

$$\begin{aligned}
\min_{\boldsymbol{\beta}_i} \quad & \frac{1}{2}\boldsymbol{\beta}_i^{\mathsf{T}}\left(\hat{X}_{\mathcal{I}}^{\mathsf{T}}\hat{X}_{\mathcal{I}} + \xi D_{\mathcal{I}}^{\mathsf{T}} D_{\mathcal{I}}\right)\boldsymbol{\beta}_i + \left(\rho \boldsymbol{d}_{\mathcal{I}} - \hat{X}_{\mathcal{I}}^{\mathsf{T}}\hat{\boldsymbol{x}}_i\right)^{\mathsf{T}}\boldsymbol{\beta}_i \\
\text{s.t.} \quad & \boldsymbol{\beta}_i^{\mathsf{T}}\mathbf{1} = 1, \quad \boldsymbol{\beta}_i \geq \mathbf{0}.
\end{aligned} \tag{16}$$

Without loss of generality, we assume that $\hat{X}_{\mathcal{I}} = \left[\hat{\boldsymbol{x}}_{(1)}, \hat{\boldsymbol{x}}_{(2)}, \ldots, \hat{\boldsymbol{x}}_{(|\mathcal{I}|)}\right]$ where $\hat{\boldsymbol{x}}_{(k)}$ ($k \in \{1, 2, \ldots, |\mathcal{I}|\}$) is the k-th nearest neighbour of $\bar{\boldsymbol{x}}_i$ that lies on the perpendicular hyperplane of $\bar{\boldsymbol{x}}_i$. Similarly $\boldsymbol{d}_{\mathcal{I}} = \text{diag}(D_{\mathcal{I}}) = \left[d_{(1)}, d_{(2)}, \ldots, d_{(|\mathcal{I}|)}\right]^{\mathsf{T}}$. Let $\boldsymbol{\beta}_i^{\star}$ denote the optimal solution to (16), we establish the necessary condition for the trivial solution that $\|\boldsymbol{\beta}_i^{\star}\|_{\infty} = 1$ in Proposition 1.

Proposition 1 *Assume the nearest neighbour of $\hat{\boldsymbol{x}}_i$ ($\hat{\boldsymbol{x}}_i = \bar{\boldsymbol{x}}_i$) is unique, i.e. $\hat{\boldsymbol{x}}_i^{(1)} \neq \hat{\boldsymbol{x}}_i^{(j)}$ for $(j) \neq (1)$. If the solution of the WSSR problem in (16) is given by $\boldsymbol{\beta}_i^{\star} = \boldsymbol{e}_1 = [1, 0, \ldots, 0]^T \in \mathbb{R}^{|\mathcal{I}|}$, then the following holds*

$$\rho > \max\left\{0, \max_{j \in \{2, \ldots, |\mathcal{I}|\}} \frac{(\hat{\boldsymbol{x}}_i^{(1)} - \hat{\boldsymbol{x}}_i^{(j)})^T(\hat{\boldsymbol{x}}_i^{(1)} - \bar{\boldsymbol{x}}_i) + \xi(d_i^{(1)})^2}{d_i^{(j)} - d_i^{(1)}}\right\}. \tag{17}$$

Proof To establish the above claim, it suffices to show that the directional derivative of the objective function at $\boldsymbol{e}_1$ is positive for all feasible directions in the unit simplex $\Delta^{|\mathcal{I}|}$. Without causing confusion, we drop the subscript i in the following proof for ease of notation. Let us denote the objective function value in (16) as $f(\boldsymbol{\beta})$, then the derivative of the objective function is

$$\nabla f(\boldsymbol{\beta}) = \left(\hat{X}_{\mathcal{I}}^{\mathsf{T}}\hat{X}_{\mathcal{I}} + \xi D_{\mathcal{I}}^{\mathsf{T}}D_{\mathcal{I}}\right)\boldsymbol{\beta} + \rho \boldsymbol{d}_{\mathcal{I}} - \hat{X}_{\mathcal{I}}^{\mathsf{T}}\hat{\boldsymbol{x}} = H\boldsymbol{\beta} + \rho \begin{bmatrix} d^{(1)} \\ d^{(2)} \\ \vdots \\ d^{|\mathcal{I}|} \end{bmatrix} - \begin{bmatrix} (\hat{\boldsymbol{x}}^{(1)})^{\mathsf{T}}\hat{\boldsymbol{x}} \\ (\hat{\boldsymbol{x}}^{(2)})^{\mathsf{T}}\hat{\boldsymbol{x}} \\ \vdots \\ (\hat{\boldsymbol{x}}^{|\mathcal{I}|})^{\mathsf{T}}\hat{\boldsymbol{x}} \end{bmatrix},$$

where

$$H = \begin{bmatrix} (\hat{\boldsymbol{x}}^{(1)})^{\mathsf{T}}\hat{\boldsymbol{x}}^{(1)} + \xi(d^{(1)})^2 & (\hat{\boldsymbol{x}}^{(1)})^{\mathsf{T}}\hat{\boldsymbol{x}}^{(2)} & \dots & (\hat{\boldsymbol{x}}^{(1)})^{\mathsf{T}}\hat{\boldsymbol{x}}^{|\mathcal{I}|} \\ (\hat{\boldsymbol{x}}^{(2)})^{\mathsf{T}}\hat{\boldsymbol{x}}^{(1)} & (\hat{\boldsymbol{x}}^{(1)})^{\mathsf{T}}\hat{\boldsymbol{x}}^{(2)} + \xi(d^{(2)})^2 & \dots & (\hat{\boldsymbol{x}}^{(2)})^{\mathsf{T}}\hat{\boldsymbol{x}}^{|\mathcal{I}|} \\ \vdots & \vdots & \ddots & \vdots \\ (\hat{\boldsymbol{x}}^{|\mathcal{I}|})^{\mathsf{T}}\boldsymbol{x}^{(1)} & (\hat{\boldsymbol{x}}^{|\mathcal{I}|})^{\mathsf{T}}\hat{\boldsymbol{x}}^{(2)} & \dots & (\hat{\boldsymbol{x}}^{|\mathcal{I}|})^{\mathsf{T}}\hat{\boldsymbol{x}}^{|\mathcal{I}|} + \xi(d^{|\mathcal{I}|})^2 \end{bmatrix}.$$

Therefore $\nabla f(\boldsymbol{e}_1)$ is equal to

$$\nabla f(\boldsymbol{e}_1) = \begin{bmatrix} (\hat{\boldsymbol{x}}^{(1)})^{\mathsf{T}}\hat{\boldsymbol{x}}^{(1)} + \xi(d^{(1)})^2 \\ (\hat{\boldsymbol{x}}^{(2)})^{\mathsf{T}}\hat{\boldsymbol{x}}^{(1)} \\ \vdots \\ (\hat{\boldsymbol{x}}^{|\mathcal{I}|})^{\mathsf{T}}\hat{\boldsymbol{x}}^{(1)} \end{bmatrix} + \rho \begin{bmatrix} d^{(1)} \\ d^{(2)} \\ \vdots \\ d^{|\mathcal{I}|} \end{bmatrix} - \begin{bmatrix} (\hat{\boldsymbol{x}}^{(1)})^{\mathsf{T}}\hat{\boldsymbol{x}} \\ (\hat{\boldsymbol{x}}^{(2)})^{\mathsf{T}}\hat{\boldsymbol{x}} \\ \vdots \\ (\hat{\boldsymbol{x}}^{|\mathcal{I}|})^{\mathsf{T}}\hat{\boldsymbol{x}} \end{bmatrix}.$$

The directional derivative of f at point $\boldsymbol{\beta}$ in the direction $\boldsymbol{e}_j$ is given by $\nabla f(\boldsymbol{\beta})^{\mathsf{T}}\boldsymbol{e}_j$ for $j \in \{1, 2, \dots, |\mathcal{I}|\}$. To ensure that the directional derivative of f at $\boldsymbol{e}_1$ towards any feasible direction (that is any direction that retains $\boldsymbol{\beta}$ within the unit simplex $\Delta^{|\mathcal{I}|}$) is positive, it suffices to ensure that

$$\nabla f(\boldsymbol{e}_1)^{\mathsf{T}}(\boldsymbol{e}_j - \boldsymbol{e}_1) > 0, \quad \forall\, j \in \{2, \dots, |\mathcal{I}|\}\,. \tag{18}$$

The above condition holds if the following holds

$$\rho > \max_{j \in \{2,\dots,|\mathcal{I}|\}} \frac{(\hat{\boldsymbol{x}}^{(1)} - \hat{\boldsymbol{x}}^{(j)})^{\mathsf{T}}(\hat{\boldsymbol{x}}^{(1)} - \hat{\boldsymbol{x}}) + \xi(d^{(1)})^2}{d^{(j)} - d^{(1)}}. \tag{19}$$

(17) is obtained by combining the above inequality with the requirement that $\rho \geq 0$. □

B.2 Sufficient condition for the trivial solution

Next, we show that (19) is a sufficient condition for the trivial solution $\boldsymbol{\beta}_i^{\star} = \boldsymbol{e}_1$.

Proposition 2 *Assume the nearest neighbour of* $\hat{\boldsymbol{x}}_i$ *(*$\hat{\boldsymbol{x}}_i = \bar{\boldsymbol{x}}_i$*) is unique, i.e.* $\hat{\boldsymbol{x}}_i^{(1)} \neq \hat{\boldsymbol{x}}_i^{(j)}$ *for* $(j) \neq (1)$*. If the following holds*

$$\rho > \max_{j \in \{2,\ldots,|\mathcal{I}|\}} \frac{(\hat{\boldsymbol{x}}_i^{(1)} - \hat{\boldsymbol{x}}_i^{(j)})^T (\hat{\boldsymbol{x}}_i^{(1)} - \hat{\boldsymbol{x}}_i) + \xi (d_i^{(1)})^2}{d_i^{(j)} - d_i^{(1)}}, \tag{20}$$

then the solution to (16) is given by $\boldsymbol{\beta}_i^\star = \boldsymbol{e}_1$*. In addition, if for all* $j \in \{2, \ldots, |\mathcal{I}|\}$ *we have*

$$\left(\bar{\boldsymbol{x}}_i^{(j)} - \bar{\boldsymbol{x}}_i^{(1)}\right)^T \left(\bar{\boldsymbol{x}}_i - \bar{\boldsymbol{x}}_i^{(1)}\right) \leq 0,$$

then the solution to (16) is given by $\boldsymbol{\beta}_i^\star = \boldsymbol{e}_1$ *for all* $\rho > 0$*.*

Proof For the first part of the proposition, if (20) holds, then for all j ($j \in \{2, \ldots, |\mathcal{I}|\}$) we have

$$\begin{aligned}
&\rho > \frac{(\hat{\boldsymbol{x}}_i^{(1)} - \hat{\boldsymbol{x}}_i^{(j)})^\mathsf{T} (\hat{\boldsymbol{x}}_i^{(1)} - \hat{\boldsymbol{x}}_i) + \xi (d_i^{(1)})^2}{d_i^{(j)} - d_i^{(1)}} \\
&\quad \Leftrightarrow (\hat{\boldsymbol{x}}_i^{(j)})^\mathsf{T} \left(\hat{\boldsymbol{x}}_i^{(1)} - \hat{\boldsymbol{x}}_i\right) + \rho d_i^{(j)} > (\hat{\boldsymbol{x}}_i^{(1)})^\mathsf{T} \left(\hat{\boldsymbol{x}}_i^{(1)} - \hat{\boldsymbol{x}}_i\right) + \xi (d_i^{(1)})^2 + \rho d_i^{(1)} \\
&\quad \Leftrightarrow \nabla f(\boldsymbol{e}_1)^\mathsf{T} \boldsymbol{e}_j > \nabla f(\boldsymbol{e}_1)^\mathsf{T} \boldsymbol{e}_1 \\
&\quad \Leftrightarrow \nabla f(\boldsymbol{e}_1)^\mathsf{T} \left(\boldsymbol{e}_j - \boldsymbol{e}_1\right) > 0.
\end{aligned}$$

The last line from above means that the directional derivative at $\boldsymbol{e}_1$ towards any other feasible direction within the unit simplex $\Delta^{(N-1)}$ is positive. Thus the solution to (16) is given by $\boldsymbol{\beta}^\star = \boldsymbol{e}_1$.

For the second part of the proposition, we first provide a geometric interpretation in Fig. 6 for the meaning of the statement.

In Fig. 6, $\bar{\boldsymbol{x}}_i$ is the point to be approximated and $\bar{\boldsymbol{x}}_i^{(1)}$ is its nearest neighbour on the unit sphere. The bold black line is the perpendicular hyperplane of $\left(\bar{\boldsymbol{x}}_i - \bar{\boldsymbol{x}}_i^{(1)}\right)$, which is denoted by $\mathcal{H} = \left\{\boldsymbol{y} | \boldsymbol{y}^\mathsf{T} \left(\bar{\boldsymbol{x}}_i - \bar{\boldsymbol{x}}_i^{(1)}\right) = 0\right\}$.

Assume that all points apart from $\bar{\boldsymbol{x}}_i$ and $\bar{\boldsymbol{x}}_i^{(1)}$ lie on one side of the hyperplane $\mathcal{H}$, opposite the side to which $\bar{\boldsymbol{x}}_i$ resides in. That is, $\left(\bar{\boldsymbol{x}}_i^{(j)} - \bar{\boldsymbol{x}}_i^{(1)}\right)^\mathsf{T} \left(\bar{\boldsymbol{x}}_i - \bar{\boldsymbol{x}}_i^{(1)}\right) \leq 0$ for all $j \in \{2, \ldots, |\mathcal{I}|\}$. We can see that $\mathcal{H}$ is a supporting hyperplane for $\text{conv}(\bar{\mathcal{X}} \setminus \{\bar{\boldsymbol{x}}_i\})$. Consider for an arbitrary point $\boldsymbol{y} = Y\boldsymbol{\beta} \in \text{conv}(\bar{\mathcal{X}} \setminus \{\bar{\boldsymbol{x}}_i\})$, we have

$$\begin{aligned}
&\left(\bar{\boldsymbol{x}}_i - \bar{\boldsymbol{x}}_i^{(1)}\right)^\mathsf{T} \left(Y\boldsymbol{\beta} - \bar{\boldsymbol{x}}_i^{(1)}\right) \\
&\quad = \left(\bar{\boldsymbol{x}}_i - \bar{\boldsymbol{x}}_i^{(1)}\right)^\mathsf{T} \left[\sum_{j=1}^{|\mathcal{I}|} \beta_j \left(\bar{\boldsymbol{x}}_i^{(j)} - \bar{\boldsymbol{x}}_i^{(1)}\right)\right]
\end{aligned}$$

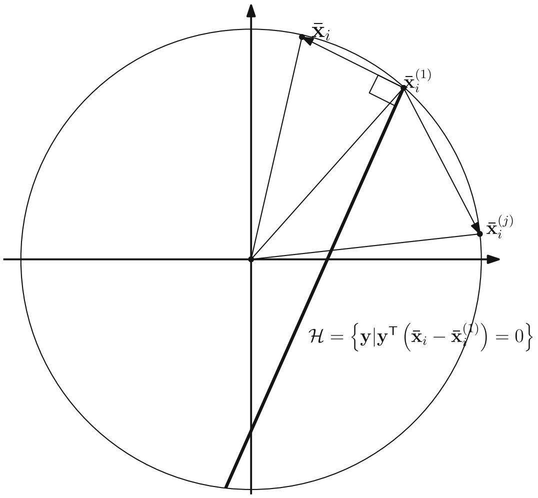

$\bar{\mathbf{x}}_i$
$\bar{\mathbf{x}}_i^{(1)}$
$\bar{\mathbf{x}}_i^{(j)}$
$\mathcal{H} = \left\{\mathbf{y}|\mathbf{y}^\top\left(\bar{\mathbf{x}}_i - \bar{\mathbf{x}}_i^{(1)}\right) = 0\right\}$

Fig. 6 A geometric interpretation for when the trivial solution is obtained

$$
\begin{aligned}
&= \sum_{j=1}^{|\mathcal{I}|} \beta_j \left(\bar{\boldsymbol{x}}_i - \bar{\boldsymbol{x}}_i^{(1)}\right)^\top \left(\bar{\boldsymbol{x}}_i^{(j)} - \bar{\boldsymbol{x}}_i^{(1)}\right) \\
&\leq 0.
\end{aligned}
$$

That is, all points apart from $\bar{\boldsymbol{x}}_i$ and $\bar{\boldsymbol{x}}_i^{(1)}$ lie on one side of the supporting hyperplane $\mathcal{H} = \left\{\boldsymbol{y}|\boldsymbol{y}^\top\left(\bar{\boldsymbol{x}}_i - \bar{\boldsymbol{x}}_i^{(1)}\right) = 0\right\}$. That is, $\left(\bar{\boldsymbol{x}}_i^{(j)} - \bar{\boldsymbol{x}}_i^{(1)}\right)^\top \left(\bar{\boldsymbol{x}}_i - \bar{\boldsymbol{x}}_i^{(1)}\right) \leq 0$. In this case, any linear combination of the column vectors in Y would be further away from $\bar{\boldsymbol{x}}_i$ than using $\bar{\boldsymbol{x}}_i^{(1)}$ itself as the approximation.

Therefore, the proposition says if $\left(\bar{\boldsymbol{x}}_i^{(j)} - \bar{\boldsymbol{x}}_i^{(1)}\right)^\top \left(\bar{\boldsymbol{x}}_i - \bar{\boldsymbol{x}}_i^{(1)}\right) \leq 0$ is satisfied for all $j \in \{2, \ldots, |\mathcal{I}|\}$, then the trivial solution can be obtained for any $\rho > 0$. □

C Experiments on USPS

In this subsection, we evaluate the performance of different subspace clustering methods on the USPS digits data (Hull 1994). USPS is another widely used benchmark dataset, which has been used to demonstrate the effectiveness of subspace clustering methods (Hu et al. 2014; Yang et al. 2019). USPS consists of 9298 images of handwritten digits that range from 0 to 9, and each image contains 16×16 pixels. We follow the exact same experimental settings as in Hu et al. (2014), which uses the first 100 images from each digit.

We investigate the performance of various algorithms under varying number of clusters K. All experiments are conducted for 20 replications, and we report both the median and standard deviation of the clustering accuracy in Table 8. For K from 2 to 8, we randomly sample data from K digits. Therefore the variability in the cluster performance comes from both the variability in the subset of the data, and the variability

Table 8 Median clustering accuracy along with the standard deviations on the USPS data across 20 replications

	$K = 2$		$K = 3$		$K = 5$		$K = 8$		$K = 10$	
	Med	Std	Med	Std	Med	Std	Med	Std	Med	Std
WSSR	**1.00**	0.01	**0.99**	0.01	**0.98**	0.01	**0.97**	0.00	**0.97**	0.00
SSR	0.96	0.10	0.95	0.08	0.80	0.09	0.76	0.06	0.69	0.01
SSC	0.96	0.03	0.85	0.12	0.63	0.14	0.60	0.05	0.51	0.00
S3C	0.97	0.02	0.83	0.13	0.69	0.09	0.62	0.05	0.51	0.01
ASSC	0.97	0.02	0.83	0.12	0.61	0.14	0.62	0.06	0.58	0.00
SSC-OMP	0.95	0.04	0.74	0.11	0.42	0.09	0.41	0.09	0.35	0.01
LSR	0.71	0.14	0.63	0.13	0.65	0.05	0.55	0.04	0.53	0.01
SMR	0.99	0.04	0.98	0.02	0.95	0.06	0.86	0.05	0.83	0.03
FGNSC	0.99	0.02	0.98	0.05	0.95	0.06	0.83	0.05	0.85	0.02

of the corresponding algorithm. For $K = 10$, we use the same dataset with 1000 images across all replications. In this case, the standard deviation reflects only the variability of the algorithms.

SSC, S3C, and ASSC exhibit similar performance for all values of K. On the USPS dataset, the performance of these methods degrades much more as K increases compared to the MNIST dataset. Performance variability is also higher for $K = 2, 3$ as evinced by the higher values of the reported standard deviations. SSC-OMP performs worse than the previous three SSC variants in every case. LSR fails to achieve high accuracy across all settings, and is it also characterised by a much higher performance variability when K is small. SSR, SMR and FGNSC have excellent performance when K is small. However, their performance decreases with the increase of K, though the performance degrade more gradually for SMR and FGNSC than the previously discussed methods. WSSR is the best performing method on this dataset. It manages to achieve median accuracy that is close to perfect, and very small performance variability for all values of K.

D WSSR+ experiments on MNIST data

In this section, we assess the performance of the proposed framework for constrained clustering in two cases. First, when a random subset of labelled points is available at the outset; and second, when active learning is used to select which points to label. We use the MNIST dataset from Sect. 6.1, and consider different values of K. For the constrained clustering problem, for each dataset and for different values of K, we obtain the labels of a proportion $p \in \{0.1, 0.2, 0.3\}$ of randomly selected points. We compare the performance of WSSR+ to that of Partition Level Constrained Clustering (PLCC) (Liu et al. 2018), Constrained Spectral Partitioning (CSP) (Wang et al. 2014), and LCVQE (Pelleg and Baras 2007).

Table 9 Clustering accuracy of various constrained clustering methods on the MNIST dataset

K	Pct	WSSR	WSSR+		PLCC		CSP		LCVQE	
	(p%)		AL	RS	AL	RS	AL	RS	AL	RS
2	10%	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
	20%		1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
	30%		1.00	1.00	0.94	1.00	1.00	1.00	1.00	1.00
3	10%	1.00	1.00	1.00	1.00	1.00	0.68	0.99	1.00	0.98
	20%		1.00	1.00	1.00	1.00	0.68	1.00	1.00	0.98
	30%		1.00	1.00	1.00	0.52	0.68	1.00	1.00	0.98
5	10%	1.00	1.00	1.00	1.00	0.99	0.79	0.44	0.79	0.79
	20%		1.00	1.00	1.00	0.99	0.80	0.64	0.85	0.87
	30%		1.00	1.00	0.98	0.45	0.80	0.99	0.94	0.92
8	10%	0.98	0.99	0.98	0.86	0.62	0.97	0.44	0.78	0.79
	20%		0.99	0.98	0.86	0.55	0.99	0.65	0.81	0.81
	30%		0.99	0.99	0.85	0.80	0.99	0.89	0.94	0.88
10	10%	0.98	0.98	0.98	0.79	0.78	0.97	0.38	0.71	0.69
	20%		0.99	0.99	0.77	0.82	0.98	0.52	0.81	0.76
	30%		0.99	0.99	0.80	0.46	0.88	0.78	0.86	0.70

The initial affinity matrix for all methods is produced by WSSR. Various proportions of side information are obtained via active learning (AL) and random sampling (RS)

To ensure a fair comparison, the same initial affinity matrix is used in all three constrained clustering algorithms. In particular the initial affinity matrix is the one produced by WSSR. PLCC involves one tuning parameter, λ, which controls the weight assigned on the side information. Although in Liu et al. (2018) it is recommended to set λ to be above 10,000 for stable performance, we found that in our experiments this is a poor choice. Instead we sampled 20 random values for λ in the range (0,1) and chose the one that produced the highest clustering accuracy. CSP involves no tuning parameters.

The experimental results of varying levels of side information are reported in Table 9. For these cases, we would like to inspect whether various constrained clustering algorithms retain this performance after additional labelling information becomes available. The columns titled "AL" and "RS" correspond to the scenarios where the labelled points are obtained through the active learning (AL) strategy in Sect. 4 and random sampling (RS) respectively. For the random sampling scenario, we replicate the experiment 20 times for varying proportions of labelled points and report the median performance. A comparison of this active learning strategy to those proposed by Lipor and Balzano (2015, 2017) is provided in Peng and Pavlidis (2019).

When K is in the range $[2, 5]$, WSSR (without any side information) produces perfect clustering results. In these scenarios, WSSR+ accommodates the label information for all values of $p\%$ without degrading accuracy on the rest of the data. This is not always the case for the competing methods, especially as K increases. The performance of the competing methods are considerably improved if the labelled points are chosen through the active learning strategy of Peng and Pavlidis (2019) compared to random sampling. The performance difference between the constrained version of WSSR+ and the other constrained clustering algorithms becomes more pronounced when $K = 8, 10$. Furthermore, the performance of the competing methods are not monotonically increasing as the proportion of labelled points increases.

References

Basu S, Davidson I, Wagstaff K (2008) Constrained clustering: advances in algorithms, theory, and applications. CRC Press

Boyd S, Vandenberghe L (2004) Convex optimization. Cambridge University Press, Cambridge

Bradley PS, Mangasarian OL (2000) *k*-plane clustering. J Global Optim 16(1):23–32

Critchley F (1985) Influence in principal components analysis. Biometrika 72(3):627–636

Elhamifar E, Vidal R (2013) Sparse subspace clustering: Algorithm, theory, and applications. IEEE Trans Pattern Anal Mach Intell 35(11):2765–2781

Gaines BR, Kim J, Zhou H (2018) Algorithms for fitting the constrained lasso. J Comput Graph Stat 27(4):861–871

Hu H, Lin Z, Feng J, Zhou J (2014) Smooth representation clustering. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 3834–3841

Huang H, Yan J, Nie F, Huang J, Cai W, Saykin AJ, Shen L (2013) A new sparse simplex model for brain anatomical and genetic network analysis. In: International conference on medical image computing and computer-assisted intervention. Springer, pp 625–632

Huang J, Nie F, Huang H (2015) A new simplex sparse learning model to measure data similarity for clustering. In: 24th international joint conference on artificial intelligence

Hull JJ (1994) A database for handwritten text recognition research. IEEE Trans Pattern Anal Mach Intell 16(5):550–554

Kriegel HP, Kröger P, Zimek A (2009) Clustering high-dimensional data: A survey on subspace clustering, pattern-based clustering, and correlation clustering. ACM Trans Knowl Discov Data 3(1):1–58

Kuhn HW (1955) The hungarian method for the assignment problem. Naval Research Logistics Quarterly 2(1–2):83–97

LeCun Y, Bottou L, Bengio Y, Haffner P (1998) Gradient-based learning applied to document recognition. Proc IEEE 86(11):2278–2324

Li C, You C, Vidal R (2017) Structured sparse subspace clustering: a joint affinity learning and subspace clustering framework. IEEE Trans Image Process 26(6):2988–3001

Li C, You C, Vidal R (2018a) On geometric analysis of affine sparse subspace clustering. IEEE J Selected Topics Sig Process 12(6):1520–1533

Li C, Zhang J, Guo J (2018b) Constrained sparse subspace clustering with side-information. In: 2018 24th international conference on pattern recognition. IEEE, pp 2093–2099

Lipor J, Balzano L (2015) Margin-based active subspace clustering. In: 2015 IEEE 6th international workshop on computational advances in multi-sensor adaptive processing. IEEE, pp 377–380

Lipor J, Balzano L (2017) Leveraging union of subspace structure to improve constrained clustering. In: Proceedings of the 34th international conference on machine learning, JMLR, vol 70, pp 2130–2139

Liu G, Lin Z, Yan S, Sun J, Yu Y, Ma Y (2012) Robust recovery of subspace structures by low-rank representation. IEEE Trans Pattern Anal Mach Intell 35(1):171–184

Liu H, Tao Z, Fu Y (2018) Partition level constrained clustering. IEEE Trans Pattern Anal Mach Intell 40(10):2469–2483

Lu C, Min H, Zhao Z, Zhu L, Huang D, Yan S (2012) Robust and efficient subspace segmentation via least squares regression. In: European conference on computer vision. Springer, pp 347–360

McWilliams B, Montana G (2014) Subspace clustering of high-dimensional data: A predictive approach. Data Min Knowl Disc 28(3):736–772

Ng AY, Jordan MI, Weiss Y (2002) On spectral clustering: Analysis and an algorithm. In: Advances in neural information processing systems, pp 849–856

Parikh N, Boyd S (2014) Proximal algorithms. Found Trends Optim 1(3):127–239

Pelleg D, Baras D (2007) K-means with large and noisy constraint sets. In: European conference on machine learning. Springer, pp 674–682

Peng H, Pavlidis NG (2019) Subspace clustering with active learning. In: IEEE international conference on big data (big data). IEEE, pp 135–144

Peng H, Pavlidis NG, Eckley IA, Tsalamanis I (2018) Subspace clustering of very sparse high-dimensional data. In: IEEE international conference on big data (big data). IEEE, pp 3780–3783

Rao S, Tron R, Vidal R, Ma Y (2010) Motion segmentation in the presence of outlying, incomplete, or corrupted trajectories. IEEE Trans Pattern Anal Mach Intell 32(10):1832–1845

Tron R, Vidal R (2007) A benchmark for the comparison of 3-D motion segmentation algorithms. In: Proceedings of the IEEE conference on computer vision and pattern recognition. IEEE, pp 1–8

Vidal R (2011) Subspace clustering. IEEE Sig Process Mag 28(2):52–68

Wagstaff K, Cardie C, Rogers S, Schrödl S (2001) Constrained k-means clustering with background knowledge. In: Proceedings of the 18th international conference on machine learning, vol 1, pp 577–584

Wang W, Carreira-Perpinán MA (2013) Projection onto the probability simplex: an efficient algorithm with a simple proof, and an application.

Wang X, Davidson I (2010) Flexible constrained spectral clustering. In: Proceedings of the 16th ACM SIGKDD international conference on knowledge discovery and data mining. ACM, pp 563–572

Wang X, Qian B, Davidson I (2014) On constrained spectral clustering and its applications. Data Min Knowl Disc 28(1):1–30

Yang J, Liang J, Wang K, Rosin P, Yang MH (2019) Subspace clustering via good neighbors. IEEE Trans Pattern Anal Mach Intell

Yeoh EJ, Ross ME, Shurtleff SA, Williams WK, Patel D, Mahfouz R, Behm FG, Raimondi SC, Relling MV, Patel A, Cheng C, Campana D, Wilkins D, Zhou X, Li J, HL H, Pui CH, Evans WE, Naeve C, Wong L, Downing JR (2002) Classification, subtype discovery, and prediction of outcome in pediatric acute lymphoblastic leukemia by gene expression profiling. Cancer Cell 1:133–143

You C, Robinson D, Vidal R (2016) Scalable sparse subspace clustering by orthogonal matching pursuit. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 3918–3927

Zou H, Hastie T (2005) Regularization and variable selection via the elastic net. J Roy Stat Soc Ser B 67(2):301–320