Submitted to the Annals of Applied Statistics
arXiv: arXiv:0000.0000

INTEGRATIVE SPARSE *K*-MEANS FOR DISEASE SUBTYPE DISCOVERY USING MULTI-LEVEL OMICS DATA

By Zhiguang Huo and George Tseng*

University of Pittsburgh

Cancer subtypes discovery is the first step to deliver personalized medicine to cancer patients. With the accumulation of massive multi-omics datasets and established biological knowledge databases, omics data integration with incorporation of rich existing biological knowledge is essential for deciphering biological mechanism behind the complex diseases. In this manuscript, we proposed an integrative sparse K-means (IS-Kmeans) approach to combine multi-omics datasets to discover disease subtypes with the guidance of prior biological knowledge via sparse overlapping group lasso. An algorithm using alternating direction method of multiplier (ADMM) will be applied for fast optimization. Simulation and three real applications in leukemia and breast cancer will be used to compare IS-Kmeans with existing methods and demonstrate its superior clustering accuracy, computing efficiency and functional annotation of detected molecular features.

1. Introduction. While cancer has been thought to be a single type of disease, increasing evidence from modern transcriptomic studies have suggested that each specific cancer may consist of multiple subtypes, with different disease mechanisms, survival rates and treatment responses. Cancer subtypes have been intensively studied, including in leukemia (Golub et al., 1999), lymphoma (Rosenwald et al., 2002), glioblastoma (Parsons et al., 2008; Verhaak et al., 2010), breast cancer (Lehmann et al., 2011; Parker et al., 2009), colorectal cancer (Sadanandam et al., 2013) and ovarian cancer (Tothill et al., 2008). These subtypes usually have strong clinical relevance since they show different outcome, and might be responsive to different treatments (Abramson et al., 2015). However, single cohort/single omics (e.g. transcriptome) analysis suffers from sample size limitation and reproducibility issues (Simon et al., 2003; Simon, 2005; Domany, 2014). Over the years large amount of omics data are accumulated in public databases and depositories; for example, The Cancer Genome Atlas (TCGA)

*To whom correspondence should be addressed.

Keywords and phrases: cancer subtype, omics integrative analysis, overlapping group lasso, ADMM

http://cancergenome.nih.gov, Gene Expression Omnibus (GEO) http://www.ncbi.nlm.nih.gov/geo/, Sequence Read Archive (SRA) http://www.ncbi.nlm.nih.gov/sra, just to name a few. These datasets provided unprecedented opportunities to reveal cancer mechanisms via combining multiple cohorts or multiple-level omics data types (a.k.a. horizontal omics meta-analysis and vertical omics integrative analysis; see below) (Tseng, Ghosh and Feingold, 2012). Omics integrative analysis has been found successful in many applications: (e.g. breast cancer (Network et al., 2012), stomach cancer (Network et al., 2014)). On the other hand, tremendous amount of biological information has been accumulated in public databases. Proper usage of these prior information (e.g. pathway information, miRNA targeting gene database) can greatly guide the modeling of omics integrative analysis.

In the literature, researchers have applied various types of clustering methods for high-throughput experimental data (e.g. microarray) to identify novel disease subtypes. Popular methods include hierarchical clustering (Eisen et al., 1998), K-means (Dudoit and Fridlyand, 2002), mixture model-based approaches (Xie, Pan and Shen, 2008; McLachlan, Bean and Peel, 2002) and non-parametric approaches (Qin, 2006), for analysis of single transcriptomic study. Resampling and ensemble methods have been used to improve stability of the clustering analysis (Kim et al., 2009; Swift et al., 2004) or to pursue tight clusters by leaving scattered samples that are different from major clusters (Tseng, 2007; Tseng and Wong, 2005; Maitra and Ramler, 2009). Witten and Tibshirani (2010) proposed a sparse K-means algorithm that can effectively select gene features and perform sample clustering simultaneously. To extend these techniques towards integration of multiple omics data sets, Tseng, Ghosh and Feingold (2012) categorized omics data integration into two major types: (A) Horizontal omics meta-analysis and (B) Vertical omics integrative analysis. For horizontal meta-analysis, multiple studies of the same omics data type (e.g. transcriptome) from different cohorts are combined to increase sample size and statistical power, a strategy often used in differential expression analysis (Ramasamy et al., 2008), pathway analysis (Shen and Tseng, 2010) or subtype discovery (Huo et al., 2015). In contrast, vertical integrative analysis aims to integrate multi-level omics data from the same patient cohort (e.g. gene expression data, genome-wide profiling of somatic mutation, DNA copy number, DNA methylation, or microRNA expression from the same set of biological samples (Richardson, Tseng and Sun, 2016)). In this paper, we focus on vertical omics integrative analysis for disease subtype discovery. Several methods for this purpose have been proposed in the literature. Lock and Dunson

(2013) fitted a finite Dirichlet mixture model to perform Bayesian consensus clustering that allows common clustering across omics types as well as omics-type-specific clustering. The model, however, does not perform proper feature selection and thus is not suitable for high-dimensional omics data. Shen, Olshen and Ladanyi (2009) proposed a latent variable factor model (namely iCluster) to cluster cancer samples by integrating multi-omics data. The method does not incorporate prior biological knowledge and requires extensive computing due to EM algorithm with large matrix operation. We will use the popular iCluster method as the baseline method to compare in this paper.

The central question we ask in this paper is: "Can we identify cancer subtypes by simultaneously integrating multi-level omics datasets and utilizing existing biological knowledge to increase accuracy and interpretation?" Several statistical challenges will arise when we attempt to achieve this goal: (1) If multi-level omics data are available for a given patient cohort, what kind of method is effective to achieve robust and accurate disease subtype detection via integrating multi-omics data? (2) Since only a small subset of intrinsic omics features are relevant to the disease subtype characterization, how can we perform effective feature selection in the high-dimensional integrative analysis? (3) With the rich biological information (e.g. targeted genes of each miRNA or potential cis-acting regulatory mechanism between copy number variation, methylation and gene expression), how can we fully utilize the prior information to guide feature selection and clustering? In this paper, we propose an integrative sparse K-means (IS-Kmeans) approach by extending the sparse K-means algorithm with overlapping group lasso technique to accommodate the three goals described above. Note that our method is also capable for an easier problem: clustering single omics dataset with prior knowledge. The lasso penalty in the sparse K-means method allows effective feature selection for clustering. In the literature, (non-overlapping) group lasso (Yuan and Lin, 2006) has been developed to encourage features of the same group to be selected or excluded together. Since such grouping information from prior biological knowledge often generates overlapping groups (e.g. the targeted genes of two pathways may have overlap), overlapping group lasso (Jacob, Obozinski and Vert, 2009) has been discussed to accommodate such scenario. However, our utilization of group lasso or overlapping group lasso is different from their original perspective. We want prior biological knowledge to guide feature selection, rather than selecting/excluding features of the same group simultaneously. Therefore we will utilize sparse overlapping group lasso, similar to a sparse group lasso (Simon et al., 2013). This will brings in optimization challenge since this is a sparse overlapping

group lasso problem in clustering setting. The original duplication technique (Jacob, Obozinski and Vert, 2009) and the sparse group lasso optimization procedure (Simon et al., 2013) won't directly apply. Latter we will transform our objective function and adopt fast optimization techniques using alternating direction method of multiplier (ADMM) (Boyd et al., 2011) into our IS-Kmeans framework.

The rest of the paper is structured as following. Section 2 gives a motivating example. Section 3 establishes the method and optimization procedure. Section 4 comprehensively compares the proposed method with the popular iCluster method using simulation and three real datasets. Section 5 provides final conclusion and discussion.

2. Motivating example. Figure 1A shows a clustering result using single study sparse K-means (detailed algorithm see Section 3.1) on the mRNA, methylation and copy number variation (CNV) datasets separately from 770 samples in TCGA. As expected, they generate very different disease subtyping without regulatory inference across mRNA, methylation and CNV. In this example of mRNA expression, copy number variation (CNV) and methylation, single study sparse K-means fails to consider that different omics features belonging to the of the same genes are likely to contain cis-acting regulatory mechanisms related to the disease subtypes. Figure 1B combined the three datasets to perform IS-Kmeans. The IS-Kmeans generates a single disease subtyping and takes into account of the prior knowledge of mutual regulation information between mRNA, methylation and CNV. The prior knowledge can also be pathway database (e.g. KEGG, BIOCARTA, REACTOME) or knowledge of miRNA target prediction databases (e.g. PicTar, TargetScan, DIANA-microT, miRanda, rna22 and PITA) (Witkos, Koscianska and Krzyzosiak, 2011; Fan and Kurgan, 2014). Incorporating such prior information of feature grouping increases statistical power and interpretation. Figure 1C shows a simple example of such group prior knowledge. Pathway1 targets mRNA1, mRNA2 , mRNA3 and mRNA6 so they form a group $\mathcal{J}_1$; Pathway2 targets mRNA3, mRNA4, mRNA5 and mRNA7 so they form a group $\mathcal{J}_2$. Note that mRNA3 can appear in both group $\mathcal{J}_1$ and $\mathcal{J}_2$, which requires our algorithm to allow overlapping groups. Our goal is to develop a sparse clustering algorithm integrating multi-level omics datasets and the aforementioned prior knowledge by overlapping group lasso. The algorithm is also suitable for single omics dataset with incorporating prior overlapping group information.

3. Method.

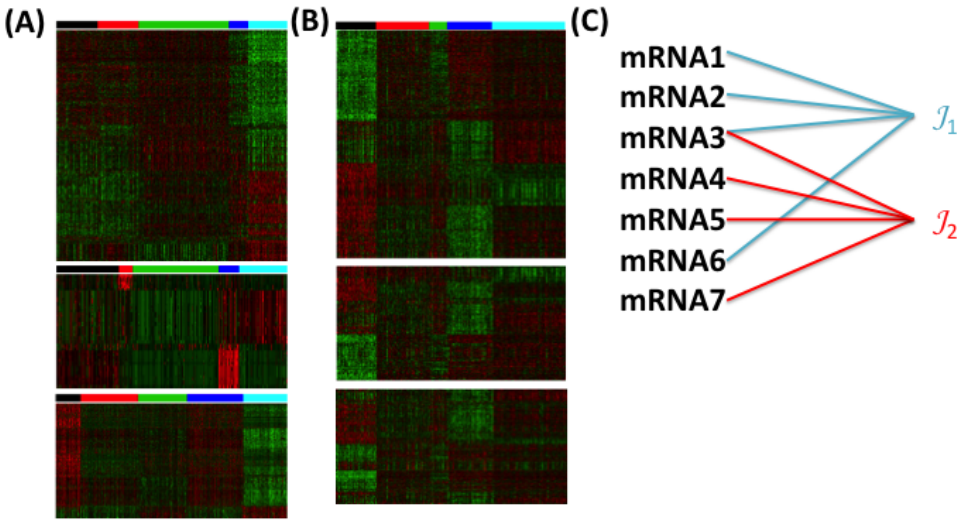

FIG 1. *(A) Clustering of mRNA (upper heatmap) CNV (middle heatmap) and methylation (lower heatmap) profiles separately results in very different five clusters of breast cancer subtypes (represented by color bars of five colors). (B) IS-K means merging mRNA (upper heatmap) CNV (middle heatmap) and methylation (lower heatmap) and perform sample clustering. Inter-omics biological knowledge is also taken into account and the clustering result looks promising. (C) An illustrating example of design of overlapping group lasso penalty term* $\Omega(w)$ *to incorporate prior knowledge of miRNA target prediction. Here* $\Omega(\mathbf{z}) = \sqrt{1+1+1/2+1}\sqrt{z_1^2+z_2^2+1/2\times z_3^2+z_6^2}+\sqrt{1/2+1+1+1}\sqrt{1/2\times z_3^2+z_4^2+z_5^2+z_7^2}$.

3.1. *K-means and sparse K-means.* Consider X_{jq} the gene expression intensity of gene j and sample q. The K-means method targets to minimize the within-cluster sum of squares (WCSS):

$$\min_C \sum_{j=1}^{J} WCSS_j(C) = \min_C \sum_{j=1}^{J} \sum_{k=1}^{K} \frac{1}{n_k} \sum_{p,q\in C_k} d_{pq,j} \tag{3.1}$$

where K is the number of clusters, J is the number of genes (features), $C = (C_1, C_2, \ldots, C_K)$ denotes the clustering result containing partitions of all samples into K clusters, n_k is the number of samples in cluster k and $d_{pq,j} = (X_{jp} - X_{jq})^2$ denotes the squared Euclidean distance of gene j between sample p and q. One drawback of K-means is that it assumes all J features with equal weights in the distance calculation. In genomic applications, J is usually large but biologically only a small subset of genes may contribute to the sample clustering. Witten and Tibshirani (2010) tackled this problem by proposing a sparse K-means approach with lasso regularization on gene-

specific weights. They found that direct application of lasso regularization to Equation 3.1 will result in a meaningless null solution. Instead, they utilized the fact that minimizing $WCSS$ is equivalent to maximizing between-cluster sum of squares ($BCSS$) since $WCSS$ and $BCSS$ add up to a constant value of total sum of squares ($TSS_j = BCSS_j(C) + WCSS_j(C)$). The optimization in Equation 3.1 is equivalent to

$$\max_C \sum_{j=1}^{J} BCSS_j(C) = \max_C \sum_{j=1}^{J} \left[\frac{1}{n} \sum_{p,q} d_{pq,j} - \sum_{k=1}^{K} \frac{1}{n_k} \sum_{p,q \in C_k} d_{pq,j} \right] \tag{3.2}$$

The lasso regularization on gene-specific weights in Equation 3.2 gives the following sparse K-means objective function:

$$\begin{aligned} &\max_{C,\mathbf{z}} \sum_{j=1}^{J} z_j BCSS_j(C) \\ &\text{subject to } \|\mathbf{z}\|_2 \le 1, \|\mathbf{z}\|_1 \le \mu, z_j \ge 0, \forall j, \end{aligned} \tag{3.3}$$

where z_j denotes weight for gene j, $C = (C_1, \ldots, C_K)$ is the clustering result, K is the pre-estimated number of clusters and $\|\mathbf{z}\|_1$ and $\|\mathbf{z}\|_2$ are the l_1 and l_2 norm of the weight vector $\mathbf{z} = (z_1, \ldots, z_J)$. The regularization shrinks most gene weights to zero and μ is a tuning parameter to control the number of non-zero weights (i.e. the number of intrinsic genes for subtype characterization). This objective function can be re-written in its Lagrangian form:

$$\begin{aligned} &\min_{C,\mathbf{z}} - \sum_{j=1}^{J} z_j BCSS_j(C) + \gamma \|\mathbf{z}\|_1 \\ &\text{subject to } \|\mathbf{z}\|_2 \le 1, z_j \ge 0, \forall j, \end{aligned}$$

3.2. *Integrative Sparse K-means (IS-Kmeans).* We extended the sparse K-means objective function to group structured sparse K-means. Here we consider J to be the total number of features combing all levels of omics datasets. Due to different types of omics datasets may have their own value range, (e.g. continuous for microarray gene expression, count for RNAseq, binary for mutation), the Euclidean distance is not adequate to accommodate all types of data. This issue can be accommodated by replacing Euclidean distance to the most appropriate distance measurement (e.g. Gower's distance for binary categorical and ordinal, Bray-Curtis dissimilarity for count data). Now we assume that the most appropriate distance measurement has been applied for the objective function. In order to make features of different

omics data types on the same scale and comparable, we normalized $BCSS_j$ by TSS_j and denote $R_j(C) = \frac{BCSS_j(C)}{TSS_j}$. We put the overlapping group lasso penalty term $\Omega(\mathbf{z})$ in the objective function.

$$\min_{C,\mathbf{z}} -\sum_{j=1}^{J} z_j R_j(C) + \gamma\alpha\|\mathbf{z}\|_1 + \gamma(1-\alpha)\Omega(\mathbf{z}) \quad \text{subject to } \|\mathbf{z}\|_2 \leq 1, z_j \geq 0, \tag{3.4}$$

where γ is the penalty tuning parameter controlling the numbers of non-zero features, $\alpha \in [0,1]$ is a term controlling the balance between individual feature penalty and group feature penalty. If $\alpha = 1$, there is no group feature penalty term and the objective function is equivalent to sparse K-means objective function after standardizing each feature. If $\alpha = 0$, there is no individual feature penalty and only group feature penalty exist. Here $\Omega(\mathbf{z}) = \sum_{0\leq g\leq \mathcal{G}_0} w_g\|\mathbf{m}_g \circ \mathbf{z}\|_2$ is the overlapping group lasso penalty term, $\mathcal{G}_0$ is the number of (possibly overlapping) feature groups from prior biological knowledge, $w_g \in \mathbb{R}$ is the group weight coefficient for group g, $\mathbf{m}_g = (\mathbf{m}_{g1}, \ldots, \mathbf{m}_{gJ})$ is the design vector of the g^{th} feature group to be discussed below and $\circ$ represents Hadamard product. Note that features with no group information are also treated as a group by itself (a group only contains a feature); such a design is to avoid bias towards a feature with no group information by receiving no penalization. We need to carefully design w_g and $\mathbf{m}_g$ and details will be discussed in Section 3.3. The feature groups can either come from external information (e.g. pathway information), or from basic biological knowledge (CNV and methylation features in the neighborhood of a nearby gene region). The first term in Equation 3.4 encourages large weights of features with strong clustering separability. The second term is an l_1 norm lasso penalty to encourage sparsity. Finally, $\Omega(\mathbf{z})$ serves as overlapping group lasso to encourage features in the prior knowledge groups to be selected simultaneously (or discarded together). The intuition of group lasso is that if we transform the Lagrange form of $\Omega(\mathbf{z})$ to its constraint form, it becomes an ellipse constraint. Features of the same group are preferred to be selected together (Yuan and Lin, 2006; Jacob, Obozinski and Vert, 2009). The combination of l_1 norm lasso penalty and overlapping group lasso penalty $\Omega(\mathbf{z})$ serves to achieve a sparse feature selection and also encourage features of the same group come out together (but not definitely).

3.3. *Design of overlapping group lasso penalty.* In this section, we discuss the design of overlapping group lasso penalty such that feature selection is not biased. We first define unbiased feature selection to be: given equal

separation ability R_j for each feature and proposed overlapping group lasso penalty, the optimum solution of $\mathbf{z}$ of Equation 3.4 will end up $z_1 = z_2 = \ldots = z_J$. We denote $\mathcal{J}_g$ as the collection of features in group g ($0 \leq g \leq \mathcal{G}_0$), $\mathcal{J}_g = \{j : j \in \mathcal{J}_g\}$. Define frequency of feature j: $h(j) = \sum_{0 \leq g \leq \mathcal{G}_0} \mathbb{I}\{j \in \mathcal{J}_g\}$.

THEOREM 3.1. $\Omega(\mathbf{z}) = \sum_{0 \leq g \leq \mathcal{G}_0} w_g \|\mathbf{m}_g \circ \mathbf{z}\|_2$, $\mathbf{m}_g = (\mathbf{m}_{g1}, \ldots, \mathbf{m}_{gj}, \ldots, \mathbf{m}_{gJ})$. *if we define* $\mathbf{m}_{gj} = \mathbb{I}\{j \in \mathcal{J}_g\}/\sqrt{h(j)}$, $w_g = \sqrt{\sum_{j \in \mathcal{J}_g} 1/h(j)}$. *Given equal separation ability* $R_1 = \ldots = R_j = \ldots = R_J = R$ *for each feature and proposed overlapping group lasso penalty, and further assume* $R > \gamma$, *the optimum solution of* $\boldsymbol{z}$ *of Equation 3.4 will end up* $z_j = 1/\sqrt{J}$, $\forall j$.

Theorem 3.1 gives a design of overlapping group lasso penalty such that given equal separation ability for all features, the feature selection is unbiased. When all the groups are non-overlapping, $h(j) = 1, \forall j$, then

$$\Omega(\mathbf{z}) = \sum_{0 \leq g \leq \mathcal{G}_0} \left(\sqrt{|\mathcal{J}_g|} \sqrt{\sum_{j \in \mathcal{J}_g} z_j^2} \right),$$

where $|\mathcal{J}_g|$ is number of features in group $\mathcal{J}_g$, which is the non-overlapping group lasso penalty (Yuan and Lin, 2006). However this proposal (e.g. $w_g = \sqrt{\sum_{j \in \mathcal{J}_g} 1/h(j)}$) won't fit our purpose since the underlying intrinsic features are sparse. If inside group g, there are many non-intrinsic features, the intrinsic features in group g is over penalized since w_g is inflated by the contribution of non-intrinsic features. Therefore we proposed $w_g = \sqrt{\sum_{j \in (\mathcal{J}_g \cap \mathcal{I})} 1/h(j)}$ in Theorem 3.2. One thing to note is that both Theorem 3.1 and Theorem 3.2 work for the ideal case while all features have either no separating ability or equal separating ability. But it is reasonable to apply the idea from ideal case to real complicated case where all features may have different separation ability.

THEOREM 3.2. *Suppose the intrinsic feature set* $\mathcal{I} = \{j : R_j = R > 0\}$ *and the non-intrinsic feature set* $\bar{\mathcal{I}} = \{j : R_j = 0\}$. *If we define* $\mathbf{m}_{gj} = \mathbb{I}\{j \in \mathcal{J}_g\}/\sqrt{h(j)}$, $w_g = \sqrt{\sum_{j \in (\mathcal{J}_g \cap \mathcal{I})} 1/h(j)}$. *Further assume* $R > \gamma$, *the optimum solution of* $\boldsymbol{z}$ *of Equation 3.4 ends up* $z_j = 1/\sqrt{|\mathcal{I}|}$ *for* $j \in \mathcal{I}$ *and* $z_j = 0$ *for* $j \in \bar{\mathcal{I}}$.

Under this scenario (Theorem 3.2), we take into account of both the non-intrinsic features and the intrinsic features. Only intrinsic features contribute to the group weight coefficient w_g. The design vector $\mathbf{m}_g$ remains

the same. We need to estimate the intrinsic feature set first. We utilize an adaptive lasso/group lasso coefficients which has been discussed in the literature and they maintain consistency property under certain condition (Zou, 2006; Huang, Horowitz and Wei, 2010). The overlapping group lasso penalty is designed in the following way. First, in Equation 3.4, we set $\alpha = 1$ where only individual feature penalty is considered and the solution is $\hat{\mathbf{z}}$. We define intrinsic feature set $\mathcal{I} = \{j : \hat{\mathbf{z}}_j > 0\}$ and non-intrinsic feature set $\bar{\mathcal{I}} = \{j : \hat{\mathbf{z}}_j = 0\}$. Define $\mathbf{m}_{gj} = \mathbb{I}\{j \in \mathcal{J}_g\}/\sqrt{h(j)}$ and $w_g = \sqrt{\sum_{j\in(\mathcal{J}_g\cap\mathcal{I})} 1/h(j)}$. Finally, we obtain the overlapping group penalty term: $\Omega(\mathbf{z}) = \sum_{0\le g\le\mathcal{G}_0} w_g\|\mathbf{m}_g \circ \mathbf{z}\|_2$. In the example of Figure 1C, suppose all 7 features are intrinsic genes. pathway database $\mathcal{J}_1$ contains mRNA1, mRNA2, mRNA3 and mRNA6, reflecting prior knowledge from pathway databases. Similarly, Group $\mathcal{J}_2$ contains mRNA3, mRNA4, mRNA5 and mRNA7. As a result, $\mathbf{m}_1 = (1, 1, 1/2, 0, 0, 1, 0)$ and $\mathbf{m}_2 = (0, 0, 1/2, 1, 1, 0, 1)$ and

$$\begin{aligned}\Omega(\mathbf{z}) =& \sqrt{1+1+1/2+1}\sqrt{z_1^2+z_2^2+1/2\times z_3^2+z_6^2}+\\ & \sqrt{1/2+1+1+1}\sqrt{1/2\times z_3^2+z_4^2+z_5^2+z_7^2}.\end{aligned}$$

Note that in our example mRNA3 is shared by pathway database $\mathcal{J}_1$ and $\mathcal{J}_2$, representing potential overlapping group lasso.

3.4. *Optimization.* In this section, we discuss major issues for optimization of Equation 3.4. Firstly we introduce transformation of Equation 3.4 such that l_1 norm penalty can be absorbed in l_2 norm group penalty. Secondly we introduce the optimization procedure for the proposed objective function. Thirdly, we discuss how to use ADMM to optimize the weight term, which is critical and a difficult problem since it involves both the l_1 norm penalty and overlapping group lasso penalty. Lastly, we discuss the stopping rule for the optimization.

3.4.1. *Reformulation and iterative optimization.* We use the fact that $\gamma\alpha\|\mathbf{z}\|_1$ can be re-written as $\gamma\alpha\|\mathbf{z}\|_1 = \gamma\alpha\sum_{j=1}^{J}\|\mathbf{z_j}\|_2$ and $\mathbf{z_j} = (0, \ldots, z_j, \ldots, 0)^\top$ with only the j^{th} element non-zero. In other words, the l_1 norm penalty of a single feature can be deemed as group penalty with only one feature within a group. Therefore we can rewrite objective function Equation 3.4 as

$$\min_{C,\mathbf{z}} -\sum_{j=1}^{J} z_j R_j(C) + \sum_{j=1}^{J}\|\gamma\alpha\boldsymbol{\phi_j}\circ\mathbf{z}\|_2 + \sum_{0\le g\le\mathcal{G}_0}\|\gamma(1-\alpha)\mathbf{m}_g\circ\mathbf{z}\|_2 \tag{3.5}$$

s.t. $\|\mathbf{z}\|_2 \leq 1$, $z_j \geq 0$, where $\boldsymbol{\phi_j} = (\boldsymbol{\phi_j}_1, \ldots, \boldsymbol{\phi_j}_J)$, $\boldsymbol{\phi_j}_i = 1$ if $j = i$ and $\boldsymbol{\phi_j}_i = 0$ if $j \neq i$. We combine J and $\mathcal{G}_0$ groups and the combined groups are of size $\mathcal{G} = J + \mathcal{G}_0$. Define

$$\boldsymbol{\beta}_g = \begin{cases} \gamma\alpha\boldsymbol{\phi_j}, & \text{if } 1 \leq g \leq J, \\ \gamma(1-\alpha)\mathbf{m}_g, & \text{if } J+1 \leq g \leq \mathcal{G}. \end{cases}$$

Therefore we can rewrite objective function Equation 3.5 as

$$\begin{aligned} &\min -\mathbf{R}(C)^\top \mathbf{z} + \sum_{1\leq g\leq \mathcal{G}} \|\boldsymbol{\beta}_g \circ \mathbf{z}\|_2 \\ &\text{subject to } \|\mathbf{z}\|_2 \leq 1, z_j \geq 0, \end{aligned} \tag{3.6}$$

where $\mathbf{R}(C) = (R_1(C), \ldots, R_J(C))^\top$. The optimization procedure are outlined below:

1. Initialize weight $\mathbf{z}$ using the original sparse K-means method without the group lasso term.
2. Given weight $\mathbf{z}$, use weighted K-means to update cluster labels C ($\mathbf{R}$ is the normalized WCSS so minimizing $-\mathbf{R}(C)^\top \mathbf{z}$ is essentially weighed K-means); or partition around medoids (PAM). This is an non-convex problem so multiple random starts are recommended to alleviate local minimum problem.
3. Given the cluster label C, $\mathbf{R}$ is fixed so optimizing the objective function is a convex problem with respect to solving weight $\mathbf{z}$. We use ADMM in the next subsection to update weight $\mathbf{z}$.
4. Iterate 2 and 3 until converge.

The detailed algorithm for Step 3 is outlined in Section 3.4.2 and the stopping rules of Step 3 and Step 4 are described in Section 3.4.3.

3.4.2. *Update weight using ADMM.* Alternating direction method of multiplier (ADMM) (Boyd et al., 2011) is ideal for solving the optimization in Equation 3.6. We introduce an auxiliary variable $\mathbf{x}_g$ and write down the augmented Lagrange.

$$\min -\mathbf{R}(C)^\top \mathbf{z} + \sum_{1\leq g\leq \mathcal{G}} \|\mathbf{x}_g\|_2 + \sum_{1\leq g\leq \mathcal{G}} \{\mathbf{y}_g^\top (\mathbf{x}_g - \boldsymbol{\beta}_g \circ \mathbf{z}) + \frac{\rho}{2}\|\mathbf{x}_g - \boldsymbol{\beta}_g \circ \mathbf{z}\|_2^2\} \tag{3.7}$$

s.t. $\|\mathbf{z}\|_2 \leq 1$, $z_j \geq 0$, and $\mathbf{x}_g = \boldsymbol{\beta}_g \circ \mathbf{z}$. This problem (Equation 3.7) is clearly equivalent to the original objective function (Equation 3.6), since for any feasible $\mathbf{z}$ the terms added to the objective is zero. ρ is the augmented Lagrange parameter which will be discussed in more detail in Section 3.4.4.

Here the augmented Lagrange is minimized jointly with respect to the two primal variables $\mathbf{x}_g$, $\mathbf{z}$ and the dual variable $\mathbf{y}_g$. In ADMM, $\mathbf{x}_g$, $\mathbf{z}$ and $\mathbf{y}_g$ are updated in an alternating or sequential fashion (Boyd et al., 2011) and thus the optimization problem can be decomposed into three parts. Given ($\mathbf{x}_g$, $\mathbf{z}$ and $\mathbf{y}_g$), the new iteration of ($\mathbf{x}_g^+$, $\mathbf{z}^+$ and $\mathbf{y}_g^+$) in Equation 3.7 is updated as following.

$$\begin{cases} \mathbf{x}_g^+ = \arg\min_{\mathbf{x}_g} \|\mathbf{x}_g\|_2 + \mathbf{y}_g^\top \mathbf{x}_g + \frac{\rho}{2}\|\mathbf{x}_g - \boldsymbol{\beta}_g \circ \mathbf{z}\|_2^2 \\ \mathbf{z}^+ = \arg\min_{\mathbf{z}} -\sum z_j R_j - \sum_{1\le g\le \mathcal{G}} \mathbf{y}_g^\top (\boldsymbol{\beta}_g \circ \mathbf{z}) + \frac{\rho}{2}\|\mathbf{x}_g^+ - \boldsymbol{\beta}_g \circ \mathbf{z}\|_2^2 \\ \quad \text{subject to } \|\mathbf{z}\|_2 \le 1, z_j \ge 0. \\ \mathbf{y}_g^+ = \mathbf{y}_g + \rho(\mathbf{x}_g^+ - \boldsymbol{\beta}_g \circ \mathbf{z}^+) \end{cases}$$

Where the updating equation of $\mathbf{x}_g^+$ and $\mathbf{z}^+$ are derived from Equation 3.7 and the the updating equation of $\mathbf{y}_g^+$ is imbedded in ADMM procedure (Boyd et al., 2011). We can derive close form solution for $\mathbf{x}_g$ part and $\mathbf{z}$ part by Karush-Kuhn-Tucker (KKT) condition. Details are given in the Appendix.

1. Define $\mathbf{a}_g = \boldsymbol{\beta}_g \circ \mathbf{z} - \frac{\mathbf{y}_g}{\rho}$, we have $\mathbf{x}_g^+ = (1 - \frac{1}{\rho\|\mathbf{a}_g\|_2})_+ \mathbf{a}_g$, where $(\cdot)_+ = \max(0, \cdot)$.
2. Define $b_j = \sum_{1\le g\le \mathcal{G}} \rho\boldsymbol{\beta}_{gj}^2$ and $c_j = \sum_{1\le g\le \mathcal{G}} \left(\rho\mathbf{x}_{gj}^+ + \mathbf{y}_{gj}\right) \circ \mathbf{m}_{gj}$, where $\boldsymbol{\beta}_g = (\boldsymbol{\beta}_{g1}, \boldsymbol{\beta}_{g2}, \ldots, \boldsymbol{\beta}_{gJ})^\top$, $\mathbf{x}_g = (\mathbf{x}_{g1}, \mathbf{x}_{g2}, \ldots, \mathbf{x}_{gJ})^\top$ and $\mathbf{y}_g = (\mathbf{y}_{g1}, \mathbf{y}_{g2}, \ldots, \mathbf{y}_{gJ})^\top$. The solution is given as following: we define $f_j(u) = (\frac{R_j + c_j}{b_j + 2u})_+$. If $\sum_j f_j(u)^2 < 1$, $z_j^+ = f_j(0)$ $\forall j$. Otherwise $z_j^+ = f_j(u)$ $\forall j$ and u is selected s.t. $\|\mathbf{z}^+\|_2 = 1$.

3.4.3. *Stopping rules.* We have two algorithms which require stopping rules. For ADMM in the optimization of Step 3, the primal residual of group g in ADMM iteration t is: $\mathbf{r}_g^t = \mathbf{x}_g^t - \boldsymbol{\beta}_g \circ \mathbf{z}^t$, and the l_2 norm of primal residual is $r^t = \sqrt{\sum_g \|\mathbf{r}_g^t\|_2^2}$. The l_2 norm of dual residual is: $v^t = \sqrt{\sum_g \|\boldsymbol{\beta}_g \circ (\mathbf{z}^t - \mathbf{z}^{t-1})\|_2^2}$. We set our ADMM stopping criteria such that simultaneously $r^t < 10^{-10}$ and $v^t < 10^{-10}$. For convergence of IS-Kmeans, we iterate weighted K-means (Step 2) and updating weight by ADMM (Step 3) until converge. (i.e. $\frac{\sum_{j=1}^J |z_j^{(c)} - z_j^{(c-1)}|}{\sum_{j=1}^J |z_j^{(c-1)}|} < 10^{-4}$), where $z_j^{(c)}$ represents the z_j estimate in the c^{th} iteration of the IS-Kmeans algorithm.

3.4.4. *augmented Lagrangian parameter ρ.* Augmented Lagrangian parameter ρ controls the convergence of ADMM. In fact, large value of ρ will lead to small primal residual by placing a large penalty on violations of primal feasibility. And conversely, small value of ρ tend to produce small dual

residual, but it will result in a large primal residual by reducing the penalty on primal feasibility (Boyd et al., 2011). An adaptive scheme of varying ρ to balance the primal and dual residual has been proposed (He, Yang and Wang, 2000; Wang and Liao, 2001) which greatly accelerates ADMM convergence in practice.

$$\rho^{t+1} = \begin{cases} \tau^{\text{incr}}\rho^t, & \text{if } \|r^t\|_2 > \eta\|v^t\|_2, \\ \rho^t/\tau^{\text{decr}}, & \text{if } \|v^t\|_2 > \eta\|r^t\|_2, \\ \rho^t, & \text{otherwise.} \end{cases}$$

We set $\eta = 10$ and $\tau^{\text{incr}} = \tau^{\text{decr}} = 2$. The intuition behind this scheme is to control both primal and dual residuals for converging to zero simultaneously.

3.5. *Select tuning parameters.* In the objective function of IS-Kmeans, the number of clusters K is pre-specified. The issue of estimating K has been widely discussed in the literature and has been well-recognized as a difficult and data-dependent problem. (Milligan and Cooper, 1985; Kaufman and Rousseeuw, 2009). Here, we suggest the number of clusters to be estimated in each study separately using conventional methods such as prediction strength (Tibshirani and Walther, 2005) or gap statistics (Tibshirani, Walther and Hastie, 2001) and jointly compared across studies (such that the numbers of clusters are roughly the same for all studies) for a final decision before applying integrative sparse K-means. Below we assume that a common K is pre-estimated for all omics datasets.

Another important parameter to be estimated is α, which controls the balance between individual feature penalty and overlapping group penalty. According to the Equation 3.5, $\alpha = 1$ means we only emphasize on individual feature penalty and ignore overlapping group penalty. In this case the IS-Kmeans is equivalent to sparse K-means. $\alpha = 0$ means we only emphasize overlapping group penalty and ignore individual feature penalty. This is a similar issue discussed in "A sparse group lasso" (Simon et al., 2013). There is no theoretically optimal value for α, because the optimal value is a function of the number of features, group sizes and other things. In practice when we expect strong overall sparsity and would like to encourage grouping we have used $\alpha = 0.05$ (See the example of $\theta = 1$ when the grouping information is not correct in Table 1). In contrast, if we expect strong group-wise sparsity, but only mild sparsity within group we have used $\alpha = 0.95$ (See $\theta = 0.2$ when grouping information is correct in Table 1). This indicated different problems will possibly be better served by different values of α and in practice some exploration may be needed. It is noticed that when $\alpha = 0.5$ (See the example in Table 1 and supplementary Table), the performance is robust again the

correctness of grouping information, therefore our final suggestion is $\alpha = 0.5$, unless the user has a strong expectation about grouping information, unless users have particular reasons to change (e.g. the users are very confident that group information is accurate and informative).

The last tuning parameter is γ, which is the penalty coefficient. When γ is large, we place large penalty on the objective function and end up with less selected features. When γ is small, we put small penalty and will include more features. We follow and extend the gap statistic procedure (Tibshirani, Walther and Hastie, 2001) to estimate γ:

1. For each feature in each omics type, randomly permute the gene expression (permute samples). This creates a permuted data set $X^{(1)}$. Repeat for B times to generate $X^{(1)}, X^{(2)}, \ldots, X^{(B)}$.
2. For each potential tuning parameter γ, compute the gap statistics as below.

$$\text{Gap}(\gamma) = O(\gamma) - \frac{1}{B}\sum_{b=1}^{B} O_b(\gamma), \tag{3.8}$$

   where $O(\gamma) = -\sum_{j=1}^{J} z_j^* R_j(C^*)$ is from observed data, where $\mathbf{z}^*, C^*$ are the minimizer of the objective function given γ. $O_b(\gamma)$ is similar to $O(\gamma)$ but it is from permuted data $X^{(b)}$
3. For a range of selections of γ, select γ^* such that the gap statistics in Equation 3.8 is minimized.

Figure 2 shows an example of a simulated dataset that will be discussed in Section 4.1. In this example, we chose $\alpha = 0.5$ for IS-Kmeans and the minimum gap statistics corresponded to 1778 genes, which is very close to the underlying truth 1800. The gap statistics figure with $\alpha = 0.05, 0.95, 1$ are in Supplementary Figures. In practice, calculating gap statistics from a chain of γ can be done quickly, since we can adopt warm start for adjacent γ's. For example, after calculating $O(\gamma_1)$, the resulting weights can be used as an initial value for the next nearby $\gamma_2 = \gamma_1 + \Delta$ to calculate $O(\gamma_2)$ in the optimization iteration for fast convergence.

4. Result. We evaluated integrative sparse K-means (IS-Kmeans) on simulation datasets, three leukemia expression datasets and two multiple-level omics types breast cancer examples. In the simulation, we demonstrated that integrative sparse K-means outperforms original sparse K-means and iCluster. in terms of cluster accuracy, feature selection. In the three leukemia example, we utilized external pathway databases. In the TCGA breast cancer example, we combined gene expression, DNA methylation and copy

number variation (CNV). In the METABRIC breast cancer example, we combined mRNA gene expression and CNV.

4.1. *Simulation.*

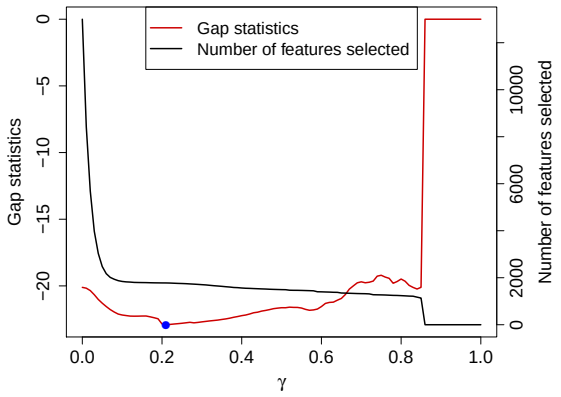

FIG 2. *Selection of tuning parameter γ. This figure was from the simulated dataset in Section 4.1 with $\alpha = 0.5$. X-axis is tuning parameter γ, red curve and left y-axis denote the corresponding gap statistics, black curve and right y-axis denote the corresponding number of selected features. The blue dot ($\gamma = 0.21$) represents where the gap statistics is minimized, and the corresponding number of selected feature is 1778.*

4.1.1. *Simulation setting.* To assess the performance of integrative sparse K-means with different choices of α and compare to the original sparse K-means and iCluster, we simulated $K = 3$ subtypes characterized by several groups of subtype predictive genes in each of $S = 2$ omics datasets with $1 \leq s \leq S$ as the omics dataset index (e.g. $s = 1$ represents gene expression and $s = 2$ represents DNA methylation). The prior group information was imposed between groups of subtype predictive genes across omics datasets. These prior group information represent the possibility that a group of genes and DNA methylations might be co-regulated. To best preserve the data nature of genomic studies, we also simulated confounding variables, correlated gene structure and non-informative genes. Below is the generative process:

(a) Subtype predictive genes.

1. Denote by N_k is the number of subjects in subtype $k (1 \leq k \leq 3)$. We simulate $N_1 \sim \text{POI}(40)$, $N_2 \sim \text{POI}(40)$, $N_3 \sim \text{POI}(30)$ and the number of subjects is $N = \sum_k N_k$. Simulate $S = 2$ omics datasets, which share the samples and subtypes. Specifically, we denote $s = 1$ to be

the gene expression dataset and $s = 2$ to be the DNA methylation dataset.

2. Simulate $M = 30$ feature modules ($1 \leq m \leq M$) for each omics dataset. Denote n_{sm} to be the number of features in omics dataset s and module m. For each module in $s = 1$, sample $n_{1m} = 30$ genes. For each module in $s = 2$, sample $n_{2m} = 30$ methylations. Therefore, there will be of 1800 subtype predictive features among two omics datasets.
3. Denote by μ_{skm} is the template gene expression (on log scale) of omics dataset $s(1 \leq s \leq S)$, subtype $k(1 \leq k \leq 3)$ and module $m(1 \leq m \leq M)$. Simulate the template gene expression $\mu_{skm} \sim \mathrm{N}(9, 2^2)$ with constrain $\max_{p,q} |\mu_{spm} - \mu_{sqm}| \geq 1$, where p, q denote two subtypes. This part defines the subtype mean intensity for each module in all omics datasets.
4. In order to tune the signal of the template gene expression, we introduce a parameter $f > 0$, such that $\mu^{'}_{skm} = (\mu_{skm} - \min_k \mu_{skm}) \times f + \min_k \mu_{skm}$. If $f = 1$, we didn't tune the signal. If $f < 1$, we decrease the signal and if $f > 1$, we amplify the signal.
5. Add biological variation $\sigma_1^2 = 1$ to the template gene expression and simulate $X^{'}_{skmi} \sim \mathrm{N}(\mu^{'}_{skm}, \sigma_1^2)$ for each module m, subject $i(1 \leq i \leq N_k)$ of subtype k and omics dataset s.
6. Simulate the covariance matrix Σ_{mks} for genes in module m, subtype k and omics dataset s, where $1 \leq m \leq M$, $1 \leq k \leq 3$ and $1 \leq s \leq S$. First simulate $\Sigma^{'}_{mks} \sim \mathrm{W}^{-1}(\Phi, 100)$, where $\Phi = 0.5 I_{n_{sm} \times n_{sm}} + 0.5 J_{n_{sm} \times n_{sm}}$, W^{-1} denotes the inverse Wishart distribution, I is the identity matrix and J is the matrix with all elements equal 1. Then Σ_{mks} is calculated by standardizing $\Sigma^{'}_{mks}$ such that the diagonal elements are all 1's.
7. Simulate gene expression levels of genes in cluster m as $(X_{1skmi}, \ldots, X_{n_{sm}skmi})^\top \sim \mathrm{MVN}(X^{'}_{skmi}, \Sigma_{mks})$, where $1 \leq i \leq N_{ks}$, $1 \leq m \leq M$, $1 \leq k \leq 3$ and $1 \leq s \leq S$.

(b) Non-informative genes.

1. Simulate 5000 non-informative genes denoted by $g(1 \leq g \leq 5000)$ in each omics dataset. For omics dataset, we generate the mean template gene expression $\mu_{sg} \sim \mathrm{N}(9, 2^2)$. Then we add biological variance $\sigma_2^2 = 1$ to generate $X_{sgi} \sim N(\mu_{sg}, \sigma_2^2), 1 \leq i \leq N_s$.

(c) Confounder impacted genes.

1. Simulate $C = 2$ confounding variables. In practice, confounding variables can be gender, race, other demographic factors or disease stage etc. They will add heterogeneity to each study to complicate disease subtype discovery. For each confounding variable $c(1 \leq c \leq C)$, we simulate $R = 10$ modules in each omics dataset. For each of these modules $r_c(1 \leq r_c \leq R)$, sample number of genes $n_{r_c} = 30$. Therefore, totally 600 confounder impacted genes are generated in each omics dataset. This procedure is repeated in all S omics datasets.
2. For each omics dataset $s(1 \leq s \leq S)$ and each confounding variable c, sample the number of confounder subclass $h_{sc} = k$. The N samples in omics dataset s will be randomly divided into h_{sc} subclasses.
3. Simulate confounding template gene expression $\mu_{slrc} \sim \mathrm{N}(9, 2^2)$ for confounder c, gene module r, subclass $l(1 \leq l \leq h_{sc})$ and omics dataset s. Similar to Step a5, we add biological variation σ_1^2 to the confounding template gene expression $X'_{scrli} \sim \mathrm{N}(\mu_{slrc}, \sigma_1^2)$. Similar to Step a6 and a7, we simulate gene correlation structure within modules of confounder impacted genes.

(d) Gene grouping information.

1. We assume omics dataset $s = 1$ and $s = 2$ have prior group information on subtype predictive gene modules. There are $M = 30$ modules in each omics dataset.
2. Suppose subtype predictive genes in the m^{th} module of the first omics dataset are grouped with methylation features in the second omics dataset (totally $n_{1m} + n_{2m} = 30 + 30 = 60$ features are in the same group). With probability $1-\theta$ $(0 \leq \theta \leq 1)$, each feature out of the 60 features will be randomly replaced by a confounder impacted gene or Non-informative gene. Note that the same replaced feature can appear in multiple subtype predictive gene groups. We set $\theta = 1$ and 0.2 to reflect 100%, 20% accuracy of prior group information.

4.1.2. *Simulation result.* For IS-Kmeans, the tuning parameter γ was selected by gap statistics introduced in Section 3.5. Table 1 shows a simulation result of gap statistics to select the best γ in the simulation of $\alpha = 0.5$, $\theta = 1$. The smallest gap statistics was selected at $\gamma = 0.21$ that correspond to selecting 1778 features, which was close to the underlying truth. Similarly, gap statistics result for $\alpha = 1, 0.95, 0.05$ are in the Supplementary Figure S1. For simulation, we generate two scenario with $f = 0.6$ and $f = 0.8$. The complete simulation result of $f = 0.6$ is shown in Table 1 and the result

TABLE 1

Comparison table of simulation. We simulated $B = 100$ times and calculated mean and standard error (se) of each quantity. θ denotes the probability grouping information is correct for each feature inside groups. α is the tuning parameter balancing the emphasis between individual penalty and group penalty. For each method, we allow its own tuning parameter selection method to optimize their performance.

θ	method	α	ARI	Jaccard index	AUC	# features	time [mins]
1	IS-Kmeans	1	0.940 (0.239)	0.781 (0.202)	0.943 (0.138)	1465	0.44
		0.95	0.940 (0.239)	0.791 (0.204)	0.945 (0.136)	1483	0.52
		0.5	0.940 (0.239)	0.779 (0.202)	0.971 (0.084)	1420	0.56
		0.05	0.940 (0.239)	0.946 (0.214)	0.997 (0.012)	1723	0.67
0.2	IS-Kmeans	1	0.940 (0.239)	0.781 (0.202)	0.943 (0.138)	1465	0.44
		0.95	0.940 (0.239)	0.783 (0.202)	0.943 (0.138)	1469	0.57
		0.5	0.940 (0.239)	0.602 (0.159)	0.943 (0.134)	1105	0.57
		0.05	0.940 (0.239)	0.467 (0.096)	0.888 (0.111)	2824	1.2
	iCluster		0.374 (0.323)	0.383 (0.274)		1239	26
	sparse Kmeans 1		0.312 (0.370)	0.105 (0.101)		896	0.12
	sparse Kmeans 2		0.361 (0.424)	0.204 (0.124)		2137	0.13

for $f = 0.8$ is in the Supplementary Table S1. For iCluster and sparse K-means, we allowed them to choose their own optimum tuning parameters. Note that sparse K-means was adopted to each individual omics datatype. We used ARI (Hubert and Arabie, 1985) and Jaccard index (Jaccard, 1901) to evaluate the clustering and feature selection performance. ARI calculated similarity of the clustering result with the underlying true clustering in simulation (range from -1 to 1 and 1 represents exact same partition compared to the underlying truth). Jaccard index compared the similarity and diversity of two feature sets. It is the size of the intersection of two feature sets divided by the size of the union of two feature sets (range from 0 to 1 and 1 represent identical feature sets compared to the underlying truth). Clearly, IS-Kmeans outperformed iCluster and individual study sparse K-means in terms of ARI, Jaccard index. IS-Kmeans and sparse K-means outperformed iCluster in terms of computing time. Within IS-Kmeans, we compared feature selection in terms of area under the curve (AUC) of ROC curve, which would avoid the issue of tuning parameter selection. When $\theta = 1$ (representing the grouping information is correct), smaller α (representing larger emphasize on grouping information) yielded better performance in terms of AUC than larger α. However when $\theta = 0.2$ (representing many errors in the grouping information), smaller α yielded worse performance in terms of AUC than larger α. This is reasonable since our prior group information links the subtype predictive features from multi-level omics dataset and incorporating these information will improve accuracy of clustering result and feature selection.

4.2. *Three leukemia datasets using pathway database as prior knowledge.* We studied three leukemia gene expression dataset with prior pathway information. Supplementary Table S2 shows a summary description of three Leukemia transcriptomic studies: Verhaak (Verhaak et al., 2009), Balgobind (Balgobind et al., 2010), Kohlmann (Kohlmann et al., 2008). We only considered samples from acute myeloid leukemia (AML) with subtype inv(16)(inversions in chromosome 16), t(15;17)(translocations between chromosome 15 and 17), t(8;21)(translocations between chromosome 8 and 21). These three gene-translocation AML subtypes have been well-studied with different survival, treatment response and prognosis outcomes. We treated these class labels as the underlying truth to evaluate the clustering performance. The expression data for Verhaak, Balgobind ranged from around $[3.169, 15.132]$ while Kohlmann ranged in $[0, 1]$. All the datasets were downloaded directly from NCBI GEO website. Originally there were 54,613 probe sets in each study. For each study, we removed genes with any missing value in it. If multiple microarray probes matched to the same gene symbol, we selected the probe with the largest inner quantile range (IQR) to represent the gene. We ended up with 20,154 unique genes in Verhaak, 20,155 unique genes in Balgobind and 20,155 unique genes in Kohlmann. We further filtered out 30% low expression genes in each study, which were defined as 30% of genes with lowest mean expression. We ended up with 14,108 unique genes in each study.

We considered pathway databases (BioCarta, KEGG and Reactome) obtained from MSigDB (`http://www.broadinstitute.org/gsea/msigdb/collections.jsp#C2`) as the prior group information to guide feature selection in IS-Kmeans. The original pathway sizes were 217, 186 and 674 for BioCarta, KEGG and Reactome. We only kept pathways with size (number of genes inside pathway) greater or equal to 15 and less or equal than 200 after intersecting with 14,108 unique genes. After gene size restriction, we ended up with 114, 160 and 428 pathways for BioCarta, KEGG and Reactome. Note that these pathway databases could potentially overlap (e.g. same gene appears in multiple pathways).

For each of the three studies, we applied IS-Kmeans (with BioCarta, KEGG and Reactome as prior group information respectively), sparse K-means and iCluster. Note that in this example, IS-Kmeans dealt with single omics dataset with prior knowledge. For a fair comparison, we enumerated a chain of all possible tuning parameter for each method and selection the result with number of selected feature most close to 1,000. The result is shown in Table 2. The result shows for Verhaak and Kohlmann, IS-Kmeans and sparse K-means almost recover the underlying true clustering labels, while iCluster has relatively smaller ARI. The heatmaps of the clustering

TABLE 2
Comparison of different methods by ARI

method	pathway	Verhaak		Kohlmann		Balgobind	
		# features	ARI	# features	ARI	# features	ARI
IS-Kmeans	biocarta	1009	0.932	1000	0.948	999	0.792
	kegg	1002	0.901	1013	0.948	990	0.792
	reactome	993	0.932	994	0.948	1008	0.792
iCluster		982	0.733	1233	0.504	1020	0.214
sparse K-means		992	0.932	998	0.948	1014	0.792

result of Verhaak is shown in Supplementary Figure S2. As a result, all of IS-Kmeans, sparse K-means and iCluster converge to a stable clustering configuration, representing different local optimum of the clustering problem. And the clustering configurations of IS-Kmeans and sparse K-means are closer to the underlying truth.

To further evaluate biological meaning of the selected genes via each method, we explored pathway enrichment analysis (Figure 3) using Biocarta as testing pathway via Fisher exact test. Five methods (iCluster, IS-Kmeans (Biocarta), IS-Kmeans (Kegg), IS-Kmeans (Reactome), sparse K-means) were compared. Jittered plot of $-\log_{10}$ p-values are shown in Figure 3. IS-Kmeans (Biocarta) shows the most significant pathways consistently across three studies, this is what we expected since we used Biocarta pathway as prior knowledge to guide our feature selection. IS-Kmeans (Kegg) and IS-Kmeans (Reactome) also showed more significant pathways than sparse K-means and iCluster, indicating incorporating prior knowledge indeed improved feature selection (in the sense that the selected feature are more biological meaningful). Note that IS-Kmeans (Kegg) and IS-Kmeans (Reactome) didn't have overfitting issue since the test pathway databases were different from the prior knowledge (biocarta) we utilized. Similarly, the results using Kegg and Biocarta as testing pathway are in Supplementary Figure 3.

4.3. *Integrating TCGA Breast cancer mRNA, CNV and methylation.* We downloaded TCGA breast cancer (BRCA) multi-level omics datasets from TCGA NIH official website. TCGA BRCA gene expression (IlluminaHiSeq RNAseqV2) was downloaded on 04/03/2015 with 20,531 genes and 1,095 subjects. TCGA BRCA DNA methylation (Methylation450) was downloaded on 09/12/2015 with 485,577 probes and 894 subjects. TCGA BRCA copy number variation (BI gistic2) was downloaded on 09/12/2015 with 24,776 genes and 1,079 subjects. There were 770 subjects with all these three omics data types. Features (probes/genes) with any missing value were removed.

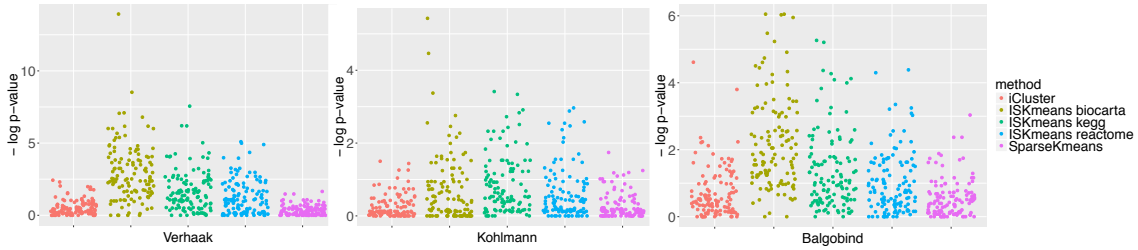

FIG 3. *Pathway enrichment analysis result for Leukemia Biocarta*

For gene expression, we transformed the FPKM value by $\log_2(\cdot + 1)$, where 1 is a forge parameter to avoid blow up at $\log_2(0)$, such that the transformed value was on continuous scale. For methylation, the Methylation450 platform provided beta value with range $0 < \beta < 1$, where 0 represents unmethylated to 1 represents methylated. We transformed the beta value to M value, which is defined by a logit transformation ($M = \log_2[\frac{\beta}{1-\beta}]$). Therefore methylation characterized by M value is on continuous scale, similar to mRNA and CNV. If multiple methylation probes matched to the same gene symbol, we selected one methylation probe as representative, which had the largest average correlation with other methylation probes of the same gene symbol. We ended up with 20,147 methylation probes with unique gene symbols.

We filtered out 50% low expression genes (unexpressed genes) and then 50% low variance genes (non-informative genes). 50% low expression genes are genes with lowest 50% mean of gene expression across samples and 10,250 genes remained after this step. 50% low variance genes are genes with lowest 50% variance of gene expression across samples and 5,125 genes remained after this step. We obtained 4,815 CNV features and 5,035 methylation features by matching the 5,125 gene symbols. The features from three different omics datasets that shared the same cis-regulatory annotation (same gene symbol) were grouped together to form 5,125 feature groups. In this case, each group had one mRNA gene expression, one CNV gene and/or one methylation probe. Each group contained potential multi-omics regulatory information because CNV and methylation could regulate mRNA expression. We applied IS-Kmeans with $\alpha = 0.5$, sparse K-means by directly merging three omics dataset together as well as iCluster. Number of clusters K was set to be 5 since it was well established that breast cancer has 5 subtypes PAM50 (Parker et al., 2009). For a fair comparison, we selected

TABLE 3
Comparison of different methods using TCGA breast cancer (K=5). G3 represents feature groups (gene symbol) where all three types of features are selected. Similarly, G2 represents feature groups (gene symbol) where only two types of features are selected; G1 represents feature groups (gene symbol) where only one type of feature is selected; We also compared the clustering result with PAM50 subtype definition in terms of ARI.

method	ARI	nfeature	G1	G2	G3	time
ISKmeans	0.379	2066	843	538	49	12.1 mins
SparseKmeans	0.332	2034	1466	284	0	6.85 mins
iCluster	0.272	2475	1725	375	0	3.91 hours

the tuning parameter for each method such that number of selected features are closest to 2,000.

For evaluation purpose, we investigated three categories of groups among selected features: G1, G2 and G3. G3 represents feature groups (gene symbol) where all three types (mRNA, CNV and methylation) of features are selected. Similarly, G2 represents feature groups (gene symbol) where only two types of features are selected; G1 represents feature groups (gene symbol) where only one type of feature is selected; We also compared the clustering result with PAM50 subtype definition in terms of ARI. The result is shown in Table 4.3. Clearly, IS-Kmeans obtained more G2 and G3 features than sparse K-means and iCluster. This is not surprising since IS-Kmeans incorporate the multi-omics regulatory information and we expected feature of the same group were encouraged to come out together. Besides, IS-Kmeans has higher ARI compared to sparse K-means and iCluster, indicating the clustering result of IS-Kmeans is closer to PAM50 definition than sparse K-means and iCluster. The 5 by 5 confusion table of IS-Kmeans clustering result and PAM50 subtypes is in Supplementary Table S3. One should note the the ARI for all these three methods are not very high, this could be because PAM50 was defined by gene expression only and in our scenario we integrate multi-omics information. The heatmaps of IS-Kmeans result is shown in Figure 1B. In terms of computing time, IS-Kmeans is nearly 20 times faster than iCluster.

4.4. *Integrating METABRIC Breast cancer mRNA and CNV.* We tested the performance of IS-Kmeans in another large breast cancer transcriptomic (sample size n=1,981) dataset METABRIC (Curtis et al., 2012) with mRNA expression (llumina HumanHT12v3) and CNV (Affymetrix SNP 6.0 chip) and survival information. The datasets are available at `https://www.synapse.org/#!Synapse:syn1688369/wiki/27311`. There were originally 49,576 probes in gene expression. If multiple probes matched to the same

TABLE 4

Comparison of different methods using metabric breast cancer (K=5). G2 represents feature groups (gene symbol) where all two types of features are selected; G1 represents feature groups (gene symbol) where only one type of feature is selected; Clustering result is compared with PAM50 subtype definition in terms of ARI. Survival p-value obtained from log rank test are given for clustering assignment for each method.

method	ARI	nfeature	G1	G2	p value	time
ISKmeans	0.233	1882	1494	194	8.29×10^{-17}	38.4 mins
SparseKmeans	0.22	2004	2004	0	3.04×10^{-13}	34.3 mins
iCluster	0.0572	2471	2471	0	0.143	11.8 hours

gene symbol, we selected the probe with the largest IQR (inner quantile range) to represent the gene. After mapping the probes to gene symbols, we obtained 19,489 mRNA expression features and 18,538 CNV features, which shared 1981 samples. After filtering out 30% low expression mRNA based on mean gene expression across samples and then 30% low variance mRNA based on variance of gene expression across samples, which were not informative and might contribute to false positives, we ended up with 9,504 mRNA features. We obtained 8,696 CNV feature symbols by matching with mRNA feature symbols. Therefore, we had totally 18,200 features and 9,504 feature groups (share the same gene symbol) among 1,981 samples.

We applied IS-Kmeans with $\alpha = 0.5$, sparse K-means by directly merging three omics dataset together as well as iCluster. Number of clusters K was set to be 5 (same reason as choose $K = 5$ for TCGA breast cancer dataset). For a fair comparison, we selected the tuning parameter for each method such that number of selected features are closest to 2,000. For evaluation purpose, we investigated two categories of groups among selected features: G1, G2. Similarly, G2 represents feature groups (gene symbol) where all two types of features are selected; G1 represents feature groups (gene symbol) where only one type of feature is selected; We also compared the clustering result with PAM50 subtype definition in terms of ARI. The result is shown in Table 4. Similar to the TCGA example in Section 4.3, IS-Kmeans obtained more $G2$ features than sparse K-means and iCluster. This is not surprising since IS-Kmeans incorporate the multi-omics regulatory information and we expected feature of the same group were encouraged to come out together. The log-rank test of clustering result defined by IS-Kmeans is more significant than sparse K-means and iCluster. Furthermore, IS-Kmeans has higher ARI compared to sparse K-means and iCluster, indicating the clustering result of IS-Kmeans is closer to PAM50 definition than sparse K-means and iCluster. The 5 by 5 confusion table of IS-Kmeans clustering result and PAM50 subtypes is in Supplementary Table S4. One should note the the

ARI for all these three methods are not very high, this could be because PAM50 was defined by gene expression only and in our scenario we integrate multi-omics information. In terms of computing time, IS-Kmeans and sparse K-means are much faster than iCluster.

5. Conclusion and discussion. Cancer subtype discovery is a critical step for the personalized treatment of the disease. In the era of massive omics datasets and biological knowledge, how to effectively integrate omics datasets and/or incorporate existing biological evidence brings new statistical and computational challenges. In this paper, we proposed an integrative sparse K-means (IS-Kmeans) approach for this purpose. The existing biological information is incorporated in the model and the resulting sparse features can be further used to characterize the cancer subtype properties in clinical application.

Our proposed IS-Kmeans has the following advantages. Firstly, integrative analysis increases clustering accuracy, statistical power and explainable regulatory flow between different omics types of data. The existing biological information is taken into account by using overlapping group lasso. Fully utilizing the inter-omics regulatory information and external biological information will increase the accuracy and interpretation of the cancer subtype findings. Secondly, we reformulated the complex objective function into a simplified form where weighted K-means and ADMM can be iteratively applied to optimize the convex sub-problems with closed form solutions. Due to the nature of classification EM algorithm in K-means and close form iteration updates of ADMM, implementation of the IS-Kmeans framework is computationally efficient. IS-Kmeans only takes 10-15 minutes for 15,000 omics features and more than 700 subjects on a standard desktop with single computing thread while iCluster takes almost 4 hours. Thirdly, the resulting sparse features from IS-Kmeans have better interpretation than features selected from iCluster.

IS-Kmeans potentially has the following limitations. The existing biological information is prone to errors and can be updated frequently. Incorporating false biological information may dilute information contained in the data and even lead to biased finding. Therefore, we suggest not to over-weigh the overlapping group lasso term and choose $\alpha = 0.5$ to adjust for the balance between information from existing biological knowledge and information from the omics datasets. The users can, however, tune this parameter depending on the strength of their prior belief of the biological knowledge. Another limitation is that IS-Kmeans can only deal with one cohort with multiple types of omics data. How to effectively combine multiple

cohorts with multi-level omics data is an appealing future work. R package "IS-Kmeans" incorporates c++ for fast computing and it is publicly available on GitHub https://github.com/Caleb-Huo/IS-Kmeans as well as authors' websites. All the data and code are also available on authors' websites.

APPENDIX A

A.1. Proof for Theorem 3.1 and Theorem 3.2.

PROOF OF THEOREM 3.1. Given equal separation ability for each feature $R_1 = \ldots = R_j = \ldots = R_J = R$ and the proposed design of overlapping group lasso penalty, Equation 3.4 becomes

$$\min_{C,\mathbf{z}} -\sum_{j=1}^{J} z_j R + \gamma\alpha\|\mathbf{z}\|_1 + \gamma(1-\alpha)\sum_{0\le g\le \mathcal{G}_0}\left(\sqrt{\sum_{j\in\mathcal{J}_g} 1/h(j)}\sqrt{\sum_{j\in\mathcal{J}_g} 1/h(j)\times z_j^2}\right),$$
$$\text{subject to } \|\mathbf{z}\|_2 \le 1, z_j \ge 0, \forall j.$$

First we can take away the constraint $z_j \ge 0, \forall j$. It is easy to see that if any $z_j < 0$, we can always use $-z_j$ to replace the solution and the objective function will decrease. We can write down the Lagrange function of Equation 3.4 after dropping the constraint $z_j \ge 0, \forall j$:

$$L(\mathbf{z},\lambda) = -\sum_{j=1}^{J} z_j R + \gamma\alpha\|\mathbf{z}\|_1 + \gamma(1-\alpha)\sum_{0\le g\le \mathcal{G}_0}\left(\sqrt{\sum_{j\in\mathcal{J}_g} 1/h(j)}\sqrt{\sum_{j\in\mathcal{J}_g} 1/h(j)\times z_j^2}\right) + \lambda(\|\mathbf{z}\|_2^2 - 1)$$

Partial derivative of the Lagrange is:

$$\frac{\partial L(\mathbf{z})}{\partial z_j} = -R + \gamma\alpha\frac{\partial|z_j|}{\partial z_j} + \gamma(1-\alpha)\sum_{\mathcal{J}_g\in\mathcal{G}}\left(\sqrt{\sum_{j'\in\mathcal{J}_g} 1/h(j')}\frac{\mathbb{I}\{j\in\mathcal{J}_g\}\times 1/h(j)\times z_j}{\sqrt{\sum_{j'\in\mathcal{J}_g} 1/h(j')\times z_{j'}^2}}\right) + 2\lambda z_j$$

It is easy to verify that $z_1 = z_2 = \ldots = z_J = 1/\sqrt{J}$, $\lambda = \frac{\sqrt{J}(R-\gamma)}{2}$ will make $\frac{\partial L(\mathbf{z})}{\partial z_j} = 0, \forall j$. Since the object function is a convex function, according to sufficiency of KKT condition, the proposed penalty design will lead to unbiased solution. □

PROOF OF THEOREM 3.2. For intrinsic gene set $\mathcal{I}$, we have $R_j = R > 0$ for $j \in \mathcal{I}$. For non-intrinsic gene set $\bar{\mathcal{I}}$, we have $R_j = 0$ for $j \in \bar{\mathcal{I}}$. Given the

proposed design of overlapping group lasso penalty, Equation 3.4 becomes

$$\min_{C,\mathbf{z}} -\sum_{j=1}^{J} z_j R\mathbb{I}(j \in \mathcal{I}) + \gamma\alpha\|\mathbf{z}\|_1 + \gamma(1-\alpha)\sum_{0\le g\le \mathcal{G}_0}\left(\sqrt{\sum_{j\in(\mathcal{J}_g\cap\mathcal{I})} 1/h(j)}\sqrt{\sum_{j\in\mathcal{J}_g} 1/h(j)\times z_j^2}\right),$$
$$\text{subject to } \|\mathbf{z}\|_2 \le 1, z_j \ge 0, \forall j.$$

First we can take away the constraint $z_j \ge 0, \forall j$. It is easy to see that if any $z_j < 0$, we can always use $-z_j$ to replace the solution and the objective function will decrease. We can write down the Lagrange function of Equation 3.4 after dropping the constraint $z_j \ge 0, \forall j$:

$$L(\mathbf{z},\lambda) = -\sum_{j=1}^{J} z_j R\mathbb{I}(j \in \mathcal{I}) + \gamma\alpha\|\mathbf{z}\|_1 + \gamma(1-\alpha)\sum_{0\le g\le \mathcal{G}_0}\left(\sqrt{\sum_{j\in(\mathcal{J}_g\cap\mathcal{I})} 1/h(j)}\sqrt{\sum_{j\in\mathcal{J}_g} 1/h(j)\times z_j^2}\right) + \lambda(\|\mathbf{z}\|_2^2 - 1)$$

Partial derivative of the Lagrange is:

$$\frac{\partial L(\mathbf{z})}{\partial z_j} = -R\mathbb{I}(j \in \mathcal{I}) + \gamma\alpha\frac{\partial|z_j|}{\partial z_j} + \gamma(1-\alpha)\sum_{\mathcal{J}_g\in\mathcal{G}}\left(\sqrt{\sum_{j'\in(\mathcal{J}_g\cap\mathcal{I})} 1/h(j')}\frac{\mathbb{I}\{j\in\mathcal{J}_g\}\times 1/h(j)\times z_j}{\sqrt{\sum_{j'\in\mathcal{J}_g} 1/h(j')\times z_{j'}^2}}\right) + 2\lambda z_j$$

It is easy to verify that if for $j \in \mathcal{I}$, $z_j = 1/\sqrt{J}$, $j \in \bar{\mathcal{I}}$, $z_j = 0$ and $\lambda = \frac{\sqrt{J}(R-\gamma)}{2}$ is a zero solution to the partial derivative of the Lagrange function. Note here we set the subgradient $\frac{\partial|z_j|}{\partial z_j} = 0$ at $z_j = 0$. Since the object function is a convex function, according to sufficiency of KKT condition, the proposed penalty design will lead to unbiased solution.

□

A.2. Optimization by KKT condition. There are two optimization problems.

$$\begin{cases} \mathbf{x}_g^+ = \arg\min_{\mathbf{x}_g} \|\mathbf{x}_g\|_2 + \mathbf{y}_g^\top\mathbf{x}_g + \frac{\rho}{2}\|\mathbf{x}_g - \boldsymbol{\beta}_g\circ\mathbf{z}\|_2^2 \\ \mathbf{z}^+ = \arg\min_{\mathbf{z}} -\sum z_j R_j - \sum_{1\le g\le\mathcal{G}}\mathbf{y}_g^\top(\boldsymbol{\beta}_g\circ\mathbf{z}) + \frac{\rho}{2}\|\mathbf{x}_g^+ - \boldsymbol{\beta}_g\circ\mathbf{z}\|_2^2 \\ \quad\text{subject to } \|\mathbf{z}\|_2 \le 1, z_j \ge 0. \end{cases}$$

It is a convex optimization problem for $\mathbf{x}_g^+$ with no constraint. The stationarity condition states that the sub-gradient of the objective function will be

0 at the optimum solution. Therefore we have:

$$S(\mathbf{x}_g^+) + \mathbf{y}_g + \rho(\mathbf{x}_g^+ - \boldsymbol{\beta}_g \circ \mathbf{z}) = 0,$$

where $S(\mathbf{v})$ is the sub-gradient of $\|\mathbf{v}\|_2$ and

$$S(\mathbf{v}) \in \begin{cases} \frac{\mathbf{v}}{\|\mathbf{v}\|_2}, \text{if } \|\mathbf{v}\|_2 \geq 1 \\ \mathbf{0}, \text{otherwise} \end{cases}$$

If we define $\mathbf{a}_g = \boldsymbol{\beta}_g \circ \mathbf{z} - \frac{\mathbf{y}_g}{\rho}$, it can be derived that $\mathbf{x}_g^+ = (1 - \frac{1}{\rho\|\mathbf{a}_g\|_2})_+ \mathbf{a}_g$, where $(\cdot)_+ = \max(0, \cdot)$.

The optimization problem for $\mathbf{z}^+$ is a convex optimization problem with two constraints. We first write down the Lagrange function and convert the constrained optimization problem into an un-constrained optimization problem:

$$\arg\min_{\mathbf{z}} -\sum_j z_j R_j - \sum_{1\leq g\leq \mathcal{G}} \mathbf{y}_g^\top (\boldsymbol{\beta}_g \circ \mathbf{z}) + \frac{\rho}{2}\|\mathbf{x}_g^+ - \boldsymbol{\beta}_g \circ \mathbf{z}\|_2^2 + u(\|\mathbf{z}\|_2 - 1) - \sum_j v_j z_j$$

such that $u \in \mathbb{R}$, $u \geq 0$, $v_j \in \mathbb{R}$ and $v_j \geq 0\ \forall j$. Taking gradient of the Lagrange function with respect to $\mathbf{z}$ and use the constraints, we can derive the solution to this problem. Define $b_j = \sum_{1\leq g\leq \mathcal{G}} \rho\boldsymbol{\beta}_{gj}^2$ and $c_j = \sum_{1\leq g\leq \mathcal{G}} \left(\rho\mathbf{x}_{gj}^+ + \mathbf{y}_{gj}\right) \circ \mathbf{m}_{gj}$, where $\boldsymbol{\beta}_g = (\boldsymbol{\beta}_{g1}, \boldsymbol{\beta}_{g2}, \ldots, \boldsymbol{\beta}_{gJ})^\top$, $\mathbf{x}_g = (\mathbf{x}_{g1}, \mathbf{x}_{g2}, \ldots, \mathbf{x}_{gJ})^\top$, $\mathbf{y}_g = (\mathbf{y}_{g1}, \mathbf{y}_{g2}, \ldots, \mathbf{y}_{gJ})^\top$, and $\mathbf{m}_g = (\mathbf{m}_{g1}, \mathbf{m}_{g2}, \ldots, \mathbf{m}_{gJ})^\top$. The solution is given as following: we define $f_j(u) = (\frac{R_j + c_j}{b_j + 2u})_+$. If $\sum_j f_j(u)^2 < 1$, $z_j^+ = f_j(0)$. Otherwise $z_j^+ = f_j(u)$ and u is selected s.t. $\|\mathbf{z}^+\|_2 = 1$.

ACKNOWLEDGEMENTS

The authors are supported by the National Institutes of Health (NIH [RO1CA190766]). The authors appreciated for reviews' instructive comments on the first submission.

REFERENCES

Abramson, V. G., Lehmann, B. D., Ballinger, T. J. and Pietenpol, J. A. (2015). Subtyping of triple-negative breast cancer: Implications for therapy. *Cancer* **121** 8–16.

Balgobind, B. V., den Heuvel-Eibrink, M. M. V., Menezes, R. X. D., Reinhardt, D., Hollink, I. H. I. M., Arentsen-Peters, S. T. J. C. M., van Wering, E. R., Kaspers, G. J. L., Cloos, J., de Bont, E. S. J. M., Cayuela, J. M., Baruchel, A., Meyer, C., Marschalek, R., Trka, J., Stary, J., Beverloo, H. B., Pieters, R., Zwaan, C. M. and den Boer, M. L. (2010). Evaluation of gene expression signatures predictive of cytogenetic and molecular subtypes of pediatric acute myeloid leukemia. *Haematologica* **96** 221–230.

Boyd, S., Parikh, N., Chu, E., Peleato, B. and Eckstein, J. (2011). Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning* **3** 1–122.

Curtis, C., Shah, S. P., Chin, S.-F., Turashvili, G., Rueda, O. M., Dunning, M. J., Speed, D., Lynch, A. G., Samarajiwa, S., Yuan, Y. et al. (2012). The genomic and transcriptomic architecture of 2,000 breast tumours reveals novel subgroups. *Nature* **486** 346–352.

Domany, E. (2014). Using high-throughput transcriptomic data for prognosis: a critical overview and perspectives. *Cancer research* **74** 4612–4621.

Dudoit, S. and Fridlyand, J. (2002). A prediction-based resampling method for estimating the number of clusters in a dataset. *Genome biology* **3** research0036.

Eisen, M. B., Spellman, P. T., Brown, P. O. and Botstein, D. (1998). Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Academy of Sciences* **95** 14863–14868.

Fan, X. and Kurgan, L. (2014). Comprehensive overview and assessment of computational prediction of microRNA targets in animals. *Briefings in bioinformatics* bbu044.

Golub, T. R., Slonim, D. K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J. P., Coller, H., Loh, M. L., Downing, J. R., Caligiuri, M. A. et al. (1999). Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *science* **286** 531–537.

He, B., Yang, H. and Wang, S. (2000). Alternating direction method with self-adaptive penalty parameters for monotone variational inequalities. *Journal of Optimization Theory and applications* **106** 337–356.

Huang, J., Horowitz, J. L. and Wei, F. (2010). Variable selection in nonparametric additive models. *Annals of statistics* **38** 2282.

Hubert, L. and Arabie, P. (1985). Comparing partitions. *Journal of classification* **2** 193–218.

Huo, Z., Ding, Y., Liu, S., Oesterreich, S. and Tseng, G. (2015). Meta-analytic framework for sparse K-means to identify disease subtypes in multiple transcriptomic studies. *Journal of the American Statistical Association* **just-accepted**.

Jaccard, P. (1901). *Etude comparative de la distribution florale dans une portion des Alpes et du Jura*. Impr. Corbaz.

Jacob, L., Obozinski, G. and Vert, J.-P. (2009). Group lasso with overlap and graph lasso. In *Proceedings of the 26th annual international conference on machine learning* 433–440. ACM.

Kaufman, L. and Rousseeuw, P. J. (2009). *Finding groups in data: an introduction to cluster analysis* **344**. Wiley. com.

Kim, E.-Y., Kim, S.-Y., Ashlock, D. and Nam, D. (2009). MULTI-K: accurate classification of microarray subtypes using ensemble k-means clustering. *BMC bioinformatics* **10** 260.

Kohlmann, A., Kipps, T. J., Rassenti, L. Z., Downing, J. R., Shurtleff, S. A., Mills, K. I., Gilkes, A. F., Hofmann, W.-K., Basso, G., Dell'Orto, M. C., Foà, R., Chiaretti, S., Vos, J. D., Rauhut, S., Papenhausen, P. R., Hernández, J. M., Lumbreras, E., Yeoh, A. E., Koay, E. S., Li, R., min Liu, W., Williams, P. M., Wieczorek, L. and Haferlach, T. (2008). An international standardization programme towards the application of gene expression profiling in routine leukaemia diagnostics: the Microarray Innovations in LEukemia study prephase. *British Journal of Haematology* **142** 802–807.

Lehmann, B. D., Bauer, J. A., Chen, X., Sanders, M. E., Chakravarthy, A. B., Shyr, Y. and Pietenpol, J. A. (2011). Identification of human triple-negative breast

cancer subtypes and preclinical models for selection of targeted therapies. *The Journal of clinical investigation* **121** 2750.

Lock, E. F. and Dunson, D. B. (2013). Bayesian consensus clustering. *Bioinformatics* btt425.

Maitra, R. and Ramler, I. P. (2009). Clustering in the Presence of Scatter. *Biometrics* **65** 341–352.

McLachlan, G. J., Bean, R. and Peel, D. (2002). A mixture model-based approach to the clustering of microarray expression data. *Bioinformatics* **18** 413–422.

Milligan, G. W. and Cooper, M. C. (1985). An examination of procedures for determining the number of clusters in a data set. *Psychometrika* **50** 159–179.

Network, C. G. A. et al. (2012). Comprehensive molecular portraits of human breast tumours. *Nature* **490** 61–70.

Network, C. G. A. R. et al. (2014). Comprehensive molecular characterization of gastric adenocarcinoma. *Nature*.

Parker, J. S., Mullins, M., Cheang, M. C., Leung, S., Voduc, D., Vickery, T., Davies, S., Fauron, C., He, X., Hu, Z. et al. (2009). Supervised risk predictor of breast cancer based on intrinsic subtypes. *Journal of clinical oncology* **27** 1160–1167.

Parsons, D. W., Jones, S., Zhang, X., Lin, J. C.-H., Leary, R. J., Angenendt, P., Mankoo, P., Carter, H., Siu, I.-M., Gallia, G. L. et al. (2008). An integrated genomic analysis of human glioblastoma multiforme. *Science* **321** 1807–1812.

Qin, Z. S. (2006). Clustering microarray gene expression data using weighted Chinese restaurant process. *Bioinformatics* **22** 1988–1997.

Ramasamy, A., Mondry, A., Holmes, C. C. and Altman, D. G. (2008). Key issues in conducting a meta-analysis of gene expression microarray datasets. *PLoS Med* **5** e184.

Richardson, S., Tseng, G. C. and Sun, W. (2016). Statistical Methods in Integrative Genomics. *Annual Review of Statistics and Its Application* **3**.

Rosenwald, A., Wright, G., Chan, W. C., Connors, J. M., Campo, E., Fisher, R. I., Gascoyne, R. D., Muller-Hermelink, H. K., Smeland, E. B., Giltnane, J. M. et al. (2002). The use of molecular profiling to predict survival after chemotherapy for diffuse large-B-cell lymphoma. *New England Journal of Medicine* **346** 1937–1947.

Sadanandam, A., Lyssiotis, C. A., Homicsko, K., Collisson, E. A., Gibb, W. J., Wullschleger, S., Ostos, L. C. G., Lannon, W. A., Grotzinger, C., Del Rio, M. et al. (2013). A colorectal cancer classification system that associates cellular phenotype and responses to therapy. *Nature medicine* **19** 619–625.

Shen, R., Olshen, A. B. and Ladanyi, M. (2009). Integrative clustering of multiple genomic data types using a joint latent variable model with application to breast and lung cancer subtype analysis. *Bioinformatics* **25** 2906–2912.

Shen, K. and Tseng, G. C. (2010). Meta-analysis for pathway enrichment analysis when combining multiple genomic studies. *Bioinformatics* **26** 1316–1323.

Simon, R. (2005). Development and validation of therapeutically relevant multi-gene biomarker classifiers. *Journal of the National Cancer Institute* **97** 866–867.

Simon, R., Radmacher, M. D., Dobbin, K. and McShane, L. M. (2003). Pitfalls in the use of DNA microarray data for diagnostic and prognostic classification. *Journal of the National Cancer Institute* **95** 14–18.

Simon, N., Friedman, J., Hastie, T. and Tibshirani, R. (2013). A sparse-group lasso. *Journal of Computational and Graphical Statistics* **22** 231–245.

Swift, S., Tucker, A., Vinciotti, V., Martin, N., Orengo, C., Liu, X. and Kellam, P. (2004). Consensus clustering and functional interpretation of gene-expression data. *Genome biology* **5** R94.

Tibshirani, R., Walther, G. and Hastie, T. (2001). Estimating the number of clusters

in a data set via the gap statistic. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **63** 411–423.

Tibshirani, R. and Walther, G. (2005). Cluster validation by prediction strength. *Journal of Computational and Graphical Statistics* **14** 511–528.

Tothill, R. W., Tinker, A. V., George, J., Brown, R., Fox, S. B., Lade, S., Johnson, D. S., Trivett, M. K., Etemadmoghadam, D., Locandro, B. et al. (2008). Novel molecular subtypes of serous and endometrioid ovarian cancer linked to clinical outcome. *Clinical Cancer Research* **14** 5198–5208.

Tseng, G. C. (2007). Penalized and weighted K-means for clustering with scattered objects and prior information in high-throughput biological data. *Bioinformatics* **23** 2247–2255.

Tseng, G. C., Ghosh, D. and Feingold, E. (2012). Comprehensive literature review and statistical considerations for microarray meta-analysis. *Nucleic Acids Research*.

Tseng, G. C. and Wong, W. H. (2005). Tight Clustering: A Resampling-Based Approach for Identifying Stable and Tight Patterns in Data. *Biometrics* **61** 10–16.

Verhaak, R. G., Wouters, B. J., Erpelinck, C. A., Abbas, S., Beverloo, H. B., Lugthart, S., Löwenberg, B., Delwel, R. and Valk, P. J. (2009). Prediction of molecular subtypes in acute myeloid leukemia based on gene expression profiling. *haematologica* **94** 131–134.

Verhaak, R. G., Hoadley, K. A., Purdom, E., Wang, V., Qi, Y., Wilkerson, M. D., Miller, C. R., Ding, L., Golub, T., Mesirov, J. P. et al. (2010). Integrated Genomic Analysis Identifies Clinically Relevant Subtypes of Glioblastoma Characterized by Abnormalities in *PDGFRA*, *IDH1*, *EGFR*, and *NF1*. *Cancer cell* **17** 98–110.

Wang, S. and Liao, L. (2001). Decomposition method with a variable parameter for a class of monotone variational inequality problems. *Journal of optimization theory and applications* **109** 415–429.

Witkos, T., Koscianska, E. and Krzyzosiak, W. (2011). Practical aspects of microRNA target prediction. *Current molecular medicine* **11** 93.

Witten, D. M. and Tibshirani, R. (2010). A framework for feature selection in clustering. *Journal of the American Statistical Association* **105**.

Xie, B., Pan, W. and Shen, X. (2008). Penalized model-based clustering with cluster-specific diagonal covariance matrices and grouped variables. *Electronic journal of statistics* **2** 168.

Yuan, M. and Lin, Y. (2006). Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **68** 49–67.

Zou, H. (2006). The adaptive lasso and its oracle properties. *Journal of the American statistical association* **101** 1418–1429.

Zhiguang Huo
Department of Biostatistics
University of Pittsburgh
Pittsburgh, PA 15261
E-mail: zhh18@pitt.edu

George Tseng
Department of Biostatistics, Human Genetics and Computational Biology
University of Pittsburgh
Pittsburgh, PA 15261
E-mail: ctseng@pitt.edu