Clustering on Sparse Data in Non-Overlapping Feature Space with Applications to Cancer Subtyping

Tianyu Kang*, Kourosh Zarringhalam†, Marieke Kuijjer‡, Ping Chen*, John Quackenbush‡ and Wei Ding*
*Department of Computer Science, University of Massachusetts Boston, Boston, USA
Email: {tianyu.kang001, ping.chen, wei.ding}@umb.edu
†Department of Mathematics, University of Massachusetts Boston, Boston, USA
Email: kourosh.zarringhalam@umb.edu
‡Dana-Farber Cancer Institute, Boston, USA
Email: {mkuijjer, johnq}@jimmy.harvard.edu

***Abstract*—This paper presents a new algorithm, Reinforced and Informed Network-based Clustering (RINC), for finding unknown groups of similar data objects in sparse and largely non-overlapping feature space where a network structure among features can be observed. Sparse and non-overlapping unlabeled data become increasingly common and available especially in text mining and biomedical data mining. RINC inserts a domain informed model into a modelless neural network. In particular, our approach integrates physically meaningful feature dependencies into the neural network architecture and soft computational constraint. Our learning algorithm efficiently clusters sparse data through integrated smoothing and sparse auto-encoder learning. The informed design requires fewer samples for training and at least part of the model becomes explainable. The architecture of the reinforced network layers smooths sparse data over the network dependency in the feature space. Most importantly, through back-propagation, the weights of the reinforced smoothing layers are simultaneously constrained by the remaining sparse auto-encoder layers that set the target values to be equal to the raw inputs. Empirical results demonstrate that RINC achieves improved accuracy and renders physically meaningful clustering results.**

***Keywords*-Unsupervised Learning, Clustering, Artificial Neural Networks**

I. Introduction

The exploratory and unsupervised nature of a clustering task inherently determines that it is an ill-posed problem in contrast to supervised learning in machine learning [1]. The challenges of clustering include 1) A clustering algorithm may produce solutions seem equally plausible without prior adequate knowledge about the data domain, or may produce meaningless results if it has erroneous assumptions about the underlying data distribution. Therefore integration of any possible prior information about the data domain is desirable for a proper clustering solution [2]. 2) Every clustering algorithm seeks a way to find groups of similar objects. It is crucial in clustering analysis to calculate similarity between two vectors of data objects. Curse of dimensionality in high dimensional space makes similarity measurement for sparse data extremely challenging [3]. 3) Data collected in text mining and biomedical data mining, especially in cancer research, are highly variable in nature. For example, there are not clearly defined, well-accepted definition of molecular subtypes in most cancers and in the precise identification of molecular subtypes in disease remains an open problem [4]. Complex diseases such as cancer call for data-driven machine learning algorithms that can identify disease subtypes, differing in response to therapy, recurrence risk, and overall survival.

In this paper, we present a new algorithm, called Reinforced and Informed Network-based Clustering (RINC), for finding clusters in sparse and largely non-overlapping feature space where a network structure among features can be observed from domain. Our approach integrates physically meaningful feature dependencies into neural network architecture and soft computational constraint design and efficiently clusters sparse data through integrated smoothing and sparse auto-encoder learning , which will increase the information entropy in the model, decrease the uncertainty of the results, without adding more variables to the model. We leave the things we are uncertain about flexible to change, and only cut out the edges less possible. The informed design requires less samples for training and at least part of the model becomes explainable.

The use of neural networks allows the implementation with multi-layered, arbitrarily non-linear structures, which is essential for addressing the complexities of highly nonlinear real datasets [5]. However, a standard neural network, given its data hungry nature, cannot achieve its full potential when data is sparse and samples sizes are hundreds of orders of magnitude smaller than the dimension of the feature space [6]. Our model consists of integrated layers of informed and reinforced network smoothing and sparse auto-encoder. The architect of hidden layers incorporates existing network dependency in feature space. The reinforced network layers smooth sparse data over the network structure. Most

* This work is supported by National Science Foundation grant 1743010, Oracle Doctoral Research Fellowship and Sanofi Doctoral Research Fellowship.

importantly, through back-propagation, the weights of the reinforced smoothing layers are simultaneously constrained by the remaining sparse auto-encoder layers that set the target values to be equal to the raw inputs.

Moreover, empirical results demonstrate that RINC outperforms competitors and achieves improved accuracy and render physically meaningful clustering results. Thus, our main contributions are as follows.

1) Robustness on sparse and non-overlapping data. RINC integrates prior domain knowledge into the learning model in forms of architecture, network smoothing, and regularization. It achieves good average performance for finding clusters in sparse and non-overlapping feature space.
2) Information Integration. RINC optimally puts domain knowledge into a new design of reinforced smoothing structure and auto-encoder.
3) Physically Meaningful Clustering Results. In our empirical studies, successful clustering is obtained with clinically relevant outcomes.

II. RINC: Reinforced and Informed Network-based Clustering

A. Problem Statement and Notations

Our RINC neural network model is designed to deal with non-overlapping sparse data. The cost function of the neural network essentially helps us to dynamically decide clusters during the training process.

The key points in our overall cost function are: 1) *sparsity and non-overlap*: integrating smoothing operations in a reinforced and informed neural network structure; and 2) *non-overlap*: enforcing the inter-feature graph structure in a "guided auto-encoder."

Our proposed cost function measures how well a neural network does with respect to its given training samples and the expected outputs:

$$Cost = Loss_\alpha + Reg_\alpha, \tag{1}$$

where $Loss_\alpha$ is the inner loss that oversees both data smoothing and auto-encoder learning, and Reg_α is the regularization that guides the auto-encoder. In particular,$Loss_\alpha = ||X - W_\alpha H||_2^2$, $Reg_\alpha = \lambda Trace(HL_\alpha H^T)$. Let $X \in \mathbb{R}^{m\times n}$ denote the raw input data matrix with n features and m samples, $H \in (\mathbb{R}^+)^{k\times n}$ represent the decoder weight matrix that has k suggested clusters , λ be the regularization hyper-parameter, and $W_\alpha \in (\mathbb{R}^+)^{m\times k}$ represent the vector of the hidden neurons of the auto-encoder. H contains the edges that connect the 2^{nd} layer neurons to W hidden neurons. The 2^{nd} layer neurons are iteratively smoothed by defusing the feature values through the neighboring features as determined by the inter-feature relation network. Here we choose the activation functions of the neurons that can produce non-negative values, such as a rectified linear unit (ReLU), to produce more interpretable clustering result. The input of auto-encoder is the smoothed data. The subscript α of W is the smoothing hyper-parameter used by the first two layers smoothing unit. Finally, $L_\alpha \in \mathbb{R}^{n\times n}$ is a supporting matrix, obtained by the Laplacian of the inter-feature relation network. Figure 1 represents an schematic overview of the model.

Notation	Description
n	# of features
m	# of samples
k	# of clusters
p	# of smoothing operation in each iteration
X	Input sample matrix
x_i	i^{th} sample vector
$x_{\alpha i}$	Smoothed i^{th} sample vector
W_α	Encoded sample matrix
$w_{\alpha i}$	Encoded vector corresponds to the i^{th} sample
S,L_α	Supporting matrices
H,H^*	Decoder and Encoder matrices
α	Smoothing parameter
η	Learning rate
λ	Regularization hyper-parameter

Table I: Notation Table

To help understand the algorithms and formulas in this paper, We list the notations used in the RINC model in Table I.

B. Informed Design: Integrate Inter-feature Relation Information into the Design of Neural Network Architecture

Non-overlapping and sparse datasets are naturally hard to cluster, but we can use smoothing methods to eliminate the non-overlapping property and make the datasets less sparse. Our approach to do the smoothing process is integrate inter-feature relation information in the clustering problem. One of the key contributions of our model is that unlike other existing methods, where smoothing and subsequent clustering are performed independently, we integrate these operations into a unified neural network framework.

Let $G = (V, E)$ denote a inter-feature relation network, with V representing vertices (nodes) and E representing edges. Nodes of the inter-feature relation network are partitioned by $V = V_{inf} \cup V_{aff}$, where V_{inf} denotes the set of influencer features and V_{aff} denotes the set of affected features. Existence of a inter-feature relation between a influencer feature $v_{inf} \in V_{inf}$ and an affected features $v_{aff} \in V_{aff}$, implies an edge $v_{inf} \rightarrow v_{aff}$ in the network. We denote this edge by $E_{inf,aff} = (v_{inf}, v_{aff})$. Let $\sigma(v_{aff}) = \{v_{inf}|(v_{inf}, v_{aff}) \in E\}$ and define the smoothing matrix

$$S = D^{-1/2} A^g D^{-1/2}. \tag{2}$$

Here A^g is the $n \times n$ adjacency matrix of the inter-feature

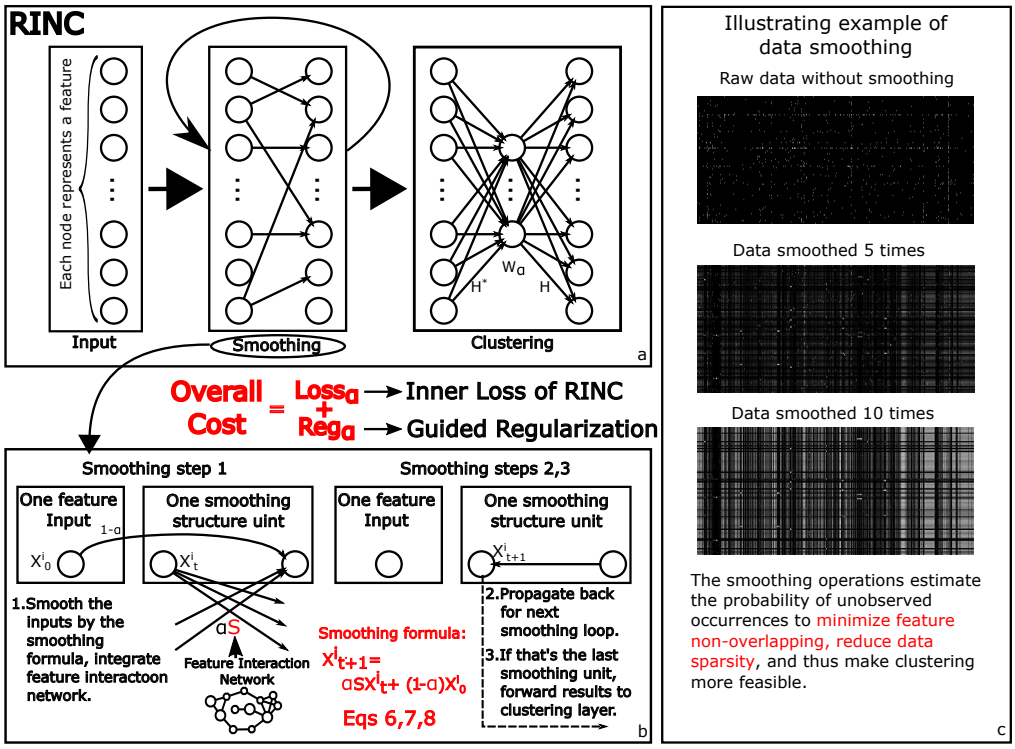

Figure 1: a) Overview of RINC model. The input, smoothing, clustering input and output node numbers are equal to the number of features. The intermediate layer in clustering structure has the number of nodes equal to the number of clusters we want to find in the datasets. b) The smoothing structure uses a recursively reinforced structure to do multiple smoothing operations described in Equation 4. S in the formula is the inter-feature network connection between these two layers described by A^g in Equation 3. c) Illustrating examples of the raw data and smoothed data. We show about 1000 genes over the 92 Adenoid Cystic Carcinoma cancer patients. Black means the patient's gene related to that position is not mutated, and white means mutated. After smoothing by gene-interaction, sparse raw data start to present meaningful patterns.

graphic network, where

$$A^g_{ij} = \begin{cases} 1, & i \neq j \text{ and } \sigma(v_{aff_i}) \cap \sigma(v_{aff_j}) \neq \emptyset \\ 0, & \text{otherwise} \end{cases} \quad (3)$$

and D is a diagonal matrix with D_{ii} equal to the sum of the i^{th} row of A^g. Let A_0 denote a normalized input vector with $A \in \{0,1\}^n$ and let $\alpha \in (0,1)$. Smoothing of A_0 over the network is then obtained by [7]:

$$A_{t+1} = \alpha S A_t + (1-\alpha)A_0. \quad (4)$$

Note that as $t \to \infty$, A_t will converge to a stable solution A^*, which is the smoothed A_0 over the network. We reformulate Equation 4 as follows to facilitate neural network implementation:

$$x_{\alpha i} \leftarrow \alpha S c + (1-\alpha)x_i, \qquad c \leftarrow x_{\alpha i} \quad (5)$$

here x_i is initialized as the input vector of the i^{th} sample, and $x_{\alpha i}$ is the smoothed input of the i^{th} sample. The vector c conceptually corresponds to a layer of the neural network that stores and transfers information among different samples. We refer to this layer as the "context" layer (the first layer in Figure 1).

C. Informed Design: Integrate Inter-feature Relation Information into Regularization

We design a sparse auto-encoder under a soft constraint of inter-feature network structure. We also take an additional step to apply activation function on the encoded layer to ensure the encoded results are non-negatives. Non-negativity on the encoded layer produces interpretable clustering results that are also desirable in scientific domains [8]. We connect this auto-encoder to the output from previous smoothing units and build everything into one neural network. We impose a regularization based on inter-feature relation information to guide the sparsity and increase interpretability of the results in a physically meaningful way. As before, let $x_{\alpha i}$ be the smoothed input, x_i the i^{th} original input in the dataset X, H^* the encoder matrix, H the decoder matrix, and w the encoded vector. The auto-encoder performs the following operations:

$$ReLu(x_{\alpha i}H^*) = w_{\alpha i}, \qquad w_{\alpha i}H \approx x_i \quad (6)$$

Here $ReLu(x) = max(0,x)$. Most importantly we enforce a similarity constraint based on the network structure among features, i.e., we would like to bring data samples in close

proximity in the inter-feature network closer to each other. This can be achieved by enforcing a network-based ℓ_2 penalty [9] $Reg_\alpha = trace(HL_\alpha H^T)$ where

$$L_\alpha = G - A^{smp} \tag{7}$$

is the graph Laplacian and A^{smp} is the feature adjacency matrix based on samples, defined by

$$A^{smp}_{ij} = \begin{cases} 1, & \text{if } x_i \in NN_q(x_j) \text{ or } x_j \in NN_q(x_i) \\ 0, & \text{otherwise} \end{cases} \tag{8}$$

Here G is a diagonal matrix with G_{ii} equal to the sum of the i^{th} row of A^{smp} and $NN_q(x_i)$ denote the q nearest neighbors of sample x_i. The distance we used is a modified Euclidean distance between the samples which each dimension can be replaced by any neighbour according to the inter-feature graph structure of the features. i.e., consider two samples a and b that have different values on feature i. If $a_i = 1$ but $b_i = 0$, where a_i and b_i are normalized between 0 and 1. Samples a and b are faraway in the i^{th} dimension in the original feature space due to their completely different values. But if a and b are related via the inter-feature relation network, $\exists j, v_{aff_j} = 1 \in \sigma(v_{aff_i})$, RINC will bring a and b closer.

D. Combine Computational Problems Into One Objective Function

Figure 1 shows the overall design of RINC, which is guided by the principle of inter-feature relation information. The inputs of RINC model are the raw non-overlapping and sparse data where a network structure among features can be observed. The output of RINC is the clustering assignment of the objects. The number of clusters is a user defined parameter k. The clustering assignment of the i^{th} sample is calculated by

$$y_i = argmax(w_{\alpha i}), \quad Y = argmax(W_\alpha) \tag{9}$$

Same as in a regular neural network, the RINC algorithm proceeds by a forward information propagation for each sample, followed by an error back propagation and the subsequent weight update. These are straight forward, so we do not discuss update formulas in this paper.

III. Empirical Study

We test RINC model with respect to accuracy, robustness, and clinical relevance of the clustering solutions in cancer subtyping. We evaluate RINC using carefully designed simulation data and real-world cancer datasets. In particular, we design experiments to evaluate the following properties of the model RINC:

- *Evaluation on synthetic data*: Can the reinforced smoothing structure in RINC accurately and automatically learn the value of smoothing factor α from data? How does RINC improve the performance of clustering using gene to gene interaction information in neural network structure and regularization?
- *Effectiveness in real cancer data including two solid tumors and a liquid cancer*: Can RINC identify clinical relevant cancer subtypes, in comparison with its competing methods in real gene mutation cancer data, with higher stability?

It is now widely accepted that mutation in gene sets, if they are part of important pathways such as apoptosis and cell proliferations, is a more significant contributor to cancer than single gene mutations [10]. So we design simulated gene mutation data sets to "mimic" this biological property, that aggregation of sporadic mutations along biological pathways can be a better predictor of tumor biology and cancer subtypes than single gene mutations. We construct gene relation networks using real gene regulatory networks. For the choice of the gene regulatory network, we use real causal/non-causal protein-protein and protein-gene interactions in the STRING DB database [11]. This network consists of approximately $40,000$ nodes and $400,000$ edges.

Automatically Learning the Smoothing Factor α From Data. To verify whether our reinforced smoothing layers can accurately find the appropriate α, we first disconnect the smoothing layers from the autoencoder part and separately test this unit. We use the real uterine endometrial carcinoma somatic mutation datasets, obtained from the TCGA (the Cancer Genome Atlas) data portal [12]. Only mutation data generated using the Illumina GAIIx platform were retained, and patients with fewer than 10 mutations gene were discarded. The final dataset includes 248 patients with mutations in $17,968$ genes. We filter out those genes not in the gene regulatory network, and get a binary matrix X of 0 and 1 values in dimensions 248 by $6,324$. We then use Equation 4 to compute the converged value Y with a target α using the ground truth value α_0. According to the proof done in [7], we can always get a unique converged value Y. In our experiments, our model can find the accurate α with $|\alpha - \alpha_0| \leq 1e^{-3}$ within 30 iterations. We let the sub-model start with a random $\alpha \in (0, 1)$, and the value of the cost function is the root mean square error between the sub-model output and converged value Y. The sub-model updates α during each iteration. We test the smoothing unit 1000 times using $1,000$ random values as the initial values for α.

Assessing the significance of integration of biological interaction to clustering. The gene regulatory network have clear influencer features, which are the regulators. And it also has clear affected feature, which are the affected genes by those regulators. Following Equation 3, we can build the adjacency matrix S.

We randomly select two non-overlapping paths of fixed length in the real gene relation network calculated from the STRING DB database. Mutations along each path are assumed to associate with a subtype, resulting in a total

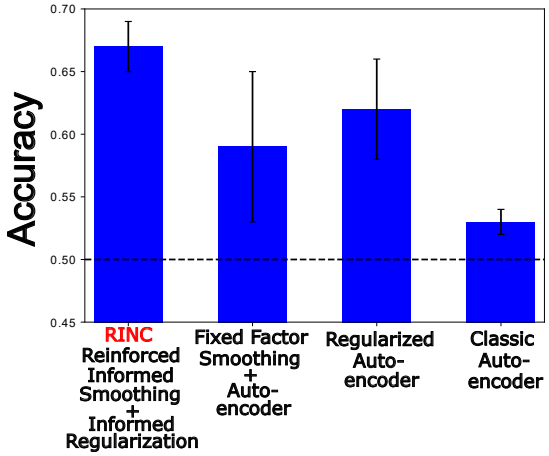

Figure 2: Models performance on simulated data using real gene relation networks.

of two subtypes. Secondly, we simulate mutational profiles by mutating genes with a background probability 0.001 independent of the paths, and a higher probability with hundreds of times of background probability if genes fall on one of the paths.

Notice that, when we simulate the samples, we are using pathways which consider really functional in domain.

In our experiments, each sub-type of the simulated data has 200 samples, 6,000 features, and uses one pathway of the gene relation networks. The features of the samples are the genes appear that in the gene relation networks. Notice that we integrate the complete gene regulatory network to the RINC model, thus our model has the capacity to simulate any possible pathways. We use the simulated data with pathway length equal to 15 and mutation rate equal to 0.5 to perform clustering with the following different design of neural networks:

1) A classic auto-encoder without any modifications,
2) A regularized auto-encoder without pre-processed smoothing unit,
3) A classic auto-encoder with fixed factor smoothed inputs but without regularization,
4) RINC (Model with informed regularization and integrated reinforcement informed smoothing layers).

Figure 2 summarizes the clustering results. We can observe that either the integration of gene-network based regularization or the smoothing operation can improve the performance of the plain auto-encoder. The RINC model has the best clustering accuracy with relatively smaller variance, compared to the individual use of smoothing unit and regulated auto-encoder. Clustering accuracy is calculated as follows. We enumerate all object sample pairs. For each simulated sample pair that belongs to the same sub-type in the ground truth, if they are still reported in the same clusters, then it is counted as a true positive. Otherwise it is considered incorrect.

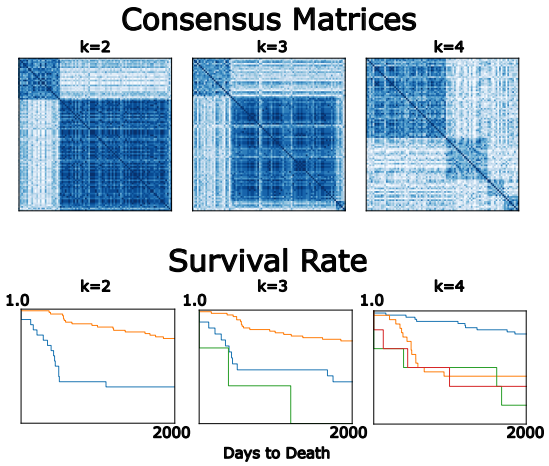

Figure 3: Models performance evaluations using ACC cancer consensus matrices and survival rate graphs and calculated using numbers of clusters k=2,3,4. The consensus matrices are drawing upon 20 runs, the graph shows RINC is a method which have high consistency. These graphs shows RINC successfully find the subtypes that consist of cancer patients with different survival rates.

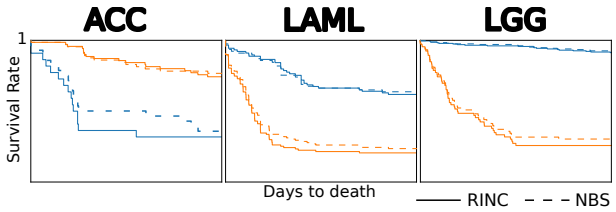

Figure 4: Benchmark between NBS and RINC on ACC, LAML, LGG datasets. We expect the line of two clusters(blue and orange) as spread as possible. The results show RINC slightly better spread the group of patients, which means RINC can do a relatively better work on cancer subtyping with respect to clinical observation of survival rates.

The state-of-art models such as LDA, LSA and NMF cannot deal with such sparse, high-dimensional, non-overlapping samples. They are far below our RINC model and classic auto-encoder. We use Python machine learning library from PyPI for the LDA, LSA, and NMF implementation [13]–[15]. The NBS [16] model is the state-of-art NMF based model published in the journal of Nature Methods for cancer patient clustering with gene mutation data problem. It is our closest rival method, we thus perform a comprehensive comparative study in the following section using three different real cancer datasets in solid tumors and a liquid cancer.

A. *Cancer Subtyping evaluation using Real Cancer Data*

We use 3 datasets which are the gene mutation data of patients with Adenoid cystic carcinoma(ACC), Acute Myeloid Leukemia(LAML) and Brain Lower Grade Glioma(LGG). ACC and LGG are solid tumor cancer type and LAML is

liquid cancer type. The ACC dataset has 10,213 features, 92 samples, and 0.12 overlapping ratio. The LAML has 8,175 features, 196 samples, and 0.02 overlapping ratio. The LGG has 13,229 features, 296 samples, and 0.29 overlapping ratio. We evaluate the performance of our model with consensus matrices and survival rate graphs, to show the robustness of our proposed method RINC. The experimental results on the ACC dataset are available in Fig 3. The 20-run consensus matrices show RINC consistently provides stable results.

An inconsistent clustering algorithm will produce consensus matrices that are blurry without clearly identifiable clusters. But in the consensus matrices of RINC, we can clearly see several blocks in the graph for clustering assignment 2, 3, and 4, which means RINC will converge to the similar clustering assignments in the most of cases.

In clinical observation, two groups of patients with different cancer subtypes but under the same treatments should have different survival rate. If the clustering result is not associated with the true cancer sub-type, the survival rate graph of these two groups of patients may be overlapped and unseparated.

The survival rate graph based on RINC results shows this point well, especially for the group which have higher overall survival(Fig 3).

We also benchmark the performance of our model with the competing method NBS, which is considered as the state-of-art cancer subtyping method using gene mutation data. Because there are few models that deal with the non-overlapping feature space sparse datasets, LDA, LSA, and NMF barely can produce any meaningful results. NBS [16]is a model published in Nature Methods and is a more advanced implementation of plain NMF, so we do not present comparisons on plain NMF. NBS has a strong hyper-parameter tuning approach and also takes advantage from gene relations.

Because there is no ground truth for comparing clustering results on the three unlabeled cancer datasets, we use the survival rate graph to illustrate the difference between RINC and NBS. The results in Fig 4 are the median over 20 runs, which show that RINC has better spreads the group of patients than NBS.

IV. Conclusion

In summary, we present a new learning algorithm to address the challenges of sparse and non-overlapping data. Our RINC model incorporates a network smoothing procedure through a reinforced module in a neural network, coupled with an auto-encoder module, designed to perform advanced clustering through one overall objective function. The auto-encoder module incorporates inter-feature relation information through network-based regularization based on the graph Laplacian, resulting in optimal model sparsity and higher interpretability. Importantly our smoothing procedure is integrated into the cost function, eliminating the need for manually adjustment of smoothing parameters. We simulate the data using biologically motivated hypothesis on tumor biology and benchmark our method with stat-of-the-art models using simulation data and real cancer data. Our model is implemented in Python and code is available upon request.

References

[1] J. M. Kleinberg, "An impossibility theorem for clustering," in *Advances in neural information processing systems*, 2003, pp. 463–470.

[2] Topchy *et al.*, "Clustering ensembles: Models of consensus and weak partitions," *IEEE transactions on pattern analysis and machine intelligence*, vol. 27, no. 12, pp. 1866–1881, 2005.

[3] A. Bellet, A. Habrard, and M. Sebban, "A survey on metric learning for feature vectors and structured data," *arXiv preprint arXiv:1306.6709*, 2013.

[4] S. Gourgou-Bourgade *et al.*, "Guidelines for time-to-event end point definitions in breast cancer trials," *Annals of Oncology*, vol. 26, no. 5, pp. 873–879, 2015.

[5] P. Smaragdis and S. Venkataramani, "A neural network alternative to non-negative audio models," in *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*. IEEE, 2017, pp. 86–90.

[6] C. M. Bishop, *Neural networks for pattern recognition*. Oxford university press, 1995.

[7] D. Zhou, O. Bousquet, T. N. Lal, J. Weston, and B. Schölkopf, "Learning with local and global consistency," in *Advances in neural information processing systems*, 2004, pp. 321–328.

[8] J.-P. Brunet, P. Tamayo, T. R. Golub, and J. P. Mesirov, "Metagenes and molecular pattern discovery using matrix factorization," *Proceedings of the National Academy of Sciences*, vol. 101, no. 12, p. 41644169, Nov 2004.

[9] D. Cai, X. He, X. Wu, and J. Han, "Non-negative matrix factorization on manifold," in *Data Mining, 2008. ICDM'08. Eighth IEEE International Conference on*. IEEE, 2008, pp. 63–72.

[10] P. K. Kreeger and D. A. Lauffenburger, "Cancer systems biology: a network modeling perspective," *Carcinogenesis*, vol. 31, no. 1, pp. 2–8, 2009.

[11] Szklarczyk *et al.*, "String v10: protein–protein interaction networks, integrated over the tree of life," *Nucleic acids research*, vol. 43, no. D1, pp. D447–D452, 2014.

[12] "The cancer genome atlas home page," 2017. [Online]. Available: https://cancergenome.nih.gov/

[13] A. Riddell, T. Hopper, and A. Grivas, "lda: 1.0.4," Jul 2016.

[14] L. C. Xia *et al.*, "Efficient statistical significance approximation for local similarity analysis of high-throughput time series data," *Bioinformatics*, vol. 29, no. 2, p. 230237, 2012.

[15] L. C. X *et al.*, "Extended local similarity analysis (elsa) of microbial community and other time series data with replicates," *BMC Systems Biology*, vol. 5, no. 2, p. S15, Dec 2011. [Online]. Available: https://doi.org/10.1186/1752-0509-5-S2-S15

[16] M. Hofree, J. P. Shen, H. Carter, A. Gross, and T. Ideker, "Network-based stratification of tumor mutations," *Nature methods*, vol. 10, no. 11, pp. 1108–1115, 2013.