GRAPH-BASED CLUSTERING ALGORITHMS FOR SINGLE-CELL RNA SEQUENCING DATA: METHODS AND THEORY

By

Andriana Manousidaki

A DISSERTATION

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

Statistics—Doctor of Philosophy

2023

ABSTRACT

The innovative technology of single-cell RNA sequencing (scRNAseq) allows us to extract gene expression information from each cell of a tissue, resulting in data sets of tens of thousands to millions of points (cells). Clustering of cells based on the similarity of their gene expression enables the understanding of their functions and hence the characterization of cell types in a tissue.

This dissertation focuses on the most widely used clustering methodology for scRNAseq data – clustering based on the graph representation of data points (cells as vertices on a graph). Firstly, we showcase how existing methods can effectively identify an important group of tumor growth related cells in the analysis of head and neck cancer scRNAseq data. The newly discovered marker genes can potentially facilitate new therapy approaches. Secondly, we introduce a novel clustering method that preserves both the global data geometry and cluster structure, via multidimensional scaling based on power-weighted path metrics. The new method outperforms prevailing scRNAseq clustering algorithms on a wide range of benchmarking data sets. Thirdly, we study spectral clustering on shared nearest neighbors (SNN) graphs. In contrast to current ad-hoc methods for number of neighbors selection, we develop a general cross-validation tuning algorithm to achieve effective clustering. Moreover, we provide a comprehensive theoretical analysis of SNN based spectral clustering in the nonparametric setting. Our theoretical results reveal an optimal range of the number of neighbors for cluster identification and characterize the impact of data density on spectral clustering.

This dissertation is dedicated to my family.

ACKNOWLEDGEMENTS

First and foremost, I would like to extend my gratitude to my advisors Dr. Haolei Weng and Dr. Yuying Xie, for their valuable mentoring and the precious research opportunities that offered me during my Ph.D. journey. Additionally, I would like to thank Dr. Anna Little and Dr. Yuehua Cui for their advice as members of my guidance committee. Furthermore, I would like to highlight my gratitude to Professor Camille Fairbourn for her continuous support and mentoring throughout my professional endeavours.

Moreover, I feel deeply thankful to Dr. Dimitra Papadovasilaki for advising me during my Ph.D life.

My life in East Lansing and my Ph.D studies wouldn't be so pleasant if I had not met my roommate and best friend Dr. Ana-Maria Raicu and the rest of my "Spartan" friends Dr. Ilias Magoulas, Eleni Lygda, Dr. Christos Gregoriadis, Dr. Ioannis Zaxos, Dr. George Psaromiligos, Dr. Michalis Paparizos and Dr. Dimitris Vardakis who have helped since before I arrived in United States and we are now family. Also I would like to thank my most recent friends Estefania Blancas Garcias, Manos Kokarakis and Mylena Ortiz.

I would like to highlight my gratitude to my husband, Dr. Marios Velivasakis, the most enthusiastic Mathematician I know for his constant support and mathematical discussion which played an important role for the completion of this thesis. I also want to thank my mother-in-law and father-in-law, Sofia Zervou and George Velivasakis for their love. I am proud to call them my parents.

I also want to thank my sisters Catherine Manousidaki and Dr. Maria Manousidaki, because they have shown to me what means to be a strong independent modern woman. With their studies, career and life experiences paved an easier way for me toward my Ph.D.

Finally, I would like to thank my father Ioannis Manousidakis and my mother Evaggelia Maslimopoulou for believing in me, encouraging me to pursue my dreams and teaching me how to continue fighting when things are hard.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

The pioneering technology of single-cell RNA sequencing (scRNAseq) enables the extraction of gene expression information from individual cells within a tissue, yielding datasets comprising tens of thousands to millions of cellular data points. Clustering cells based on the congruity of their gene expression profiles facilitates the comprehension of their functional attributes, thereby enabling the characterization of distinct cell types within a given tissue. Prevalent clustering methodologies developed for scRNAseq data rely on the representation of data points (cells) as vertices in a graph (Stuart et al., 2019; Wolf et al., 2018). The present dissertation primarily focuses on graph-based clustering methods tailored for scRNAseq data analysis. Firstly, we present the contribution of established approaches to the identification of Cancer Stem Cells (CSCs), a cellular cohort characterized by their resistance to therapeutic interventions and their pivotal role in tumor initiation and progression (Chen et al., 2021; Mroz et al., 2015). Secondly, we introduce a novel clustering methodology denoted as Single-Cell Path Metrics Profiling (scPMP), which concurrently upholds both local cluster structure and global data geometry. Thirdly, we undertake an exploration of the performance of Spectral Clustering on Shared Nearest Neighbors (SNN) graphs in relationship with the parameter of nearest neighbors used in the construction of the SNN grpah. We finally suggest a general cross-validation method for the tuning of this parameter.

In Chapter 2, an in-depth analysis of scRNAseq data originating from cell cultures of head and neck cancer lines, as well as 10 primary tumors, is conducted. The primary objective of this analysis revolves around the identification of the most homogeneous cluster of CSCs within each dataset, while simultaneously elucidating their dynamic states and plasticity via an extension of the repertoire of CSC marker genes.

Chapter 3 presents the introduction of the scPMP algorithm, a novel clustering methodology predicated upon path-metric distances among cells. Unlike conventional distance metrics, such as the Euclidean distance, path metrics possess the capacity to discern density variations and faithfully uphold the underlying data geometry. By integrating path metrics with multidimensional scaling

techniques, we obtain a low-dimensional representation of the data that faithfully encapsulates both the global data geometry and cluster structure. The efficacy of the scPMP algorithm is evaluated comprehensively in terms of clustering quality and geometric fidelity, ultimately establishing its superiority over current scRNAseq clustering algorithms across a diverse spectrum of benchmark datasets.

Chapter 4 delves into Spectral Clustering on SNN graphs. SNN graphs are constructed based on a k Nearest Neighbors (kNN) graph, thus rendering their properties contingent upon the choice of the parameter k. Our findings indicate that, in both the absence of noise and the presence of noise, it is imperative to select k of the magnitude cn in order to maximize the likelihood of cluster identification. This contrasts with the literature on random geometric graphs, which suggests an order of $\log n$ for k (Brito et al., 1997). Additionally, we propose a comprehensive cross-validation tuning approach for fine-tuning the parameters of clustering algorithms. We employ this approach to determine the optimal number of nearest neighbors, denoted as k, for the SNN spectral clustering algorithm using various types of simulated data.

BIBLIOGRAPHY

Brito, M. R., Chávez, E. L., Quiroz, A. J., and Yukich, J. E. (1997). Connectivity of the mutual k-nearest-neighbor graph in clustering and outlier detection. *Statistics & Probability Letters*, 35(1):33–42.

Chen, C.-Y., Ueha, S., Ishiwata, Y., Shichino, S., Yokochi, S., Yang, D., Oppenheim, J. J., Ogiwara, H., Deshimaru, S., Kanno, Y., et al. (2021). Combining an alarmin hmgn1 peptide with pd-l1 blockade results in robust antitumor effects with a concomitant increase of stem-like/progenitor exhausted cd8+ t cells. *Cancer immunology research*, 9(10):1214–1228.

Mroz, E. A., Tward, A. M., Hammon, R. J., Ren, Y., and Rocco, J. W. (2015). Intra-tumor genetic heterogeneity and mortality in head and neck cancer: analysis of data from the cancer genome atlas. *PLoS medicine*, 12(2):e1001786.

Stuart, T., Butler, A., Hoffman, P., Hafemeister, C., Papalexi, E., III, W. M. M., Hao, Y., Stoeckius, M., Smibert, P., and Satija, R. (2019). Comprehensive integration of single-cell data. *Cell*, 177:1888–1902.

Wolf, F. A., Angerer, P., and Theis, F. J. (2018). SCANPY: large-scale single-cell gene expression data analysis. *Genome Biology*, 19.

CHAPTER 2

SINGLE-CELL ANALYSIS OF CANCER STEM CELLS IN HEAD AND NECK CANCER

2.1 Introduction

Head and neck cancer (HNC) is a major global health problem, with an estimated 880,000 new cases and 445,000 deaths annually worldwide (Sung et al., 2021). While human papilloma (HPV)-associated HNC have improved outcomes, despite advances in comprehensive cancer care, HPV-negative HNC remains a highly morbid disease with stagnant survival rates hovering at 50%. This poor prognosis is due in part to the complex heterogeneity of HNC, which involves multiple cell types, genetic alterations and transitional states that lead to treatment resistance and poor outcomes (Chen et al., 2021; Puram et al., 2018; Mroz et al., 2015).

Tumoral heterogeneity is a well-established biomarker of poor prognosis, being associated with aggressive cancer behavior and treatment resistance in various cancer types, including HNC. Tumoral heterogeneity has been associated with worse outcomes, mediated by intrinsic and extrinsic factors related to subpopulations with distinct molecular profiles. Tumoral plasticity has been identified as a critical driver of tumoral heterogeneity, where clonal expansion and subclonal selection are based on evolutionary progression with each clone arising from cells with high propagation potential, plasticity, and self-renewal. Within the tumor microenvironment (TME), there is a subpopulation of tumor initiating cells, or cancer stem cells (CSC), that have the capacity to drive clonal and subclonal selection (O'Brien et al., 2007). Traditionally CSC were deemed fixed cells with limited to no plasticity based on their original definitions. However, as the field has advanced, the role of plasticity in CSC has expanded and the traditional view of CSCs has evolved. While the term CSC has persisted, despite much controversy on their existence, a more nuanced understanding of CSC is that their stem-like activity (CSC-state: self-renewal, tumorigenicity and asymmetric division) is not fixed but a transient state dictated by tumoral and environmental cues (Chaffer and Weinberg, 2011). When cells are in this CSC-state, they are associated with treatment resistance, metastasis, and tumor recurrence. However, CSCs in HNC remains controversial and the CSC-like state has been difficult to study as the mechanisms of CSC plasticity are poorly

understood.

Plasticity and heterogeneity are also critical components of epithelial to mesenchymal transition (EMT) programs. Weinberg and others have shown EMT represent transient cancer cell states with varying degrees of activities, strongly suggesting the EMT process enables cancer cells to acquire CSC-like properties and enhance their ability to initiate and sustain tumors (Mani et al., 2008; Tam and Weinberg, 2013). However, understanding the link between EMT and CSC remains elusive due to their rarity and potentially transiet states. Analyzing this interaction is critical to understanding the CSC-like state and defining potential mechanisms for plasticity and identify novel targets for therapy.

Recent advances in single-cell RNA sequencing (scRNAseq) technology have enabled the identification of distinct subpopulations of cells within tumors based on their gene expression profiles, providing a powerful tool to study the heterogeneity and plasticity of CSCs in HNC (Wang et al., 2019). Moreover, in vitro lineage tracing can be used to assess CSC's capacity for plasticity and evaluate their various states. In this study, we integrated scRNAseq and in vitro and in silico lineage tracing to analyze these rare CSC subpopulations in cell culture and primary HNC tumors to characterize their dynamic states and plasticity. Our study sheds new light on the dynamic nature and plasticity of CSCs in HNC, and their potential involvement in EMT programs. Our findings have important implications for the development of novel therapeutic strategies for HNC, as well as other cancers, and for the broader understanding of CSC-states and plasticity.

2.2 Methods

2.2.1 Cell lines analysis

To better evaluate transcriptional differences and controlling for tumoral heterogeneity, we subsequently performed single cell sequencing of two patient derived HNC cell lines (UMSCC-122 and UMSCC-103).Both cell lines were sorted to select for CSC (CD44high ALDHhigh) and non-CSC (CD44low/ ALDHlow) cells. After standard quality control filtering and integration of the two cell line expression data sets, we found 26 clusters using Seurat (Stuart et al., 2019). We observed that the clusters were not separated on the UMAP plot and suggesting that cells lie on a

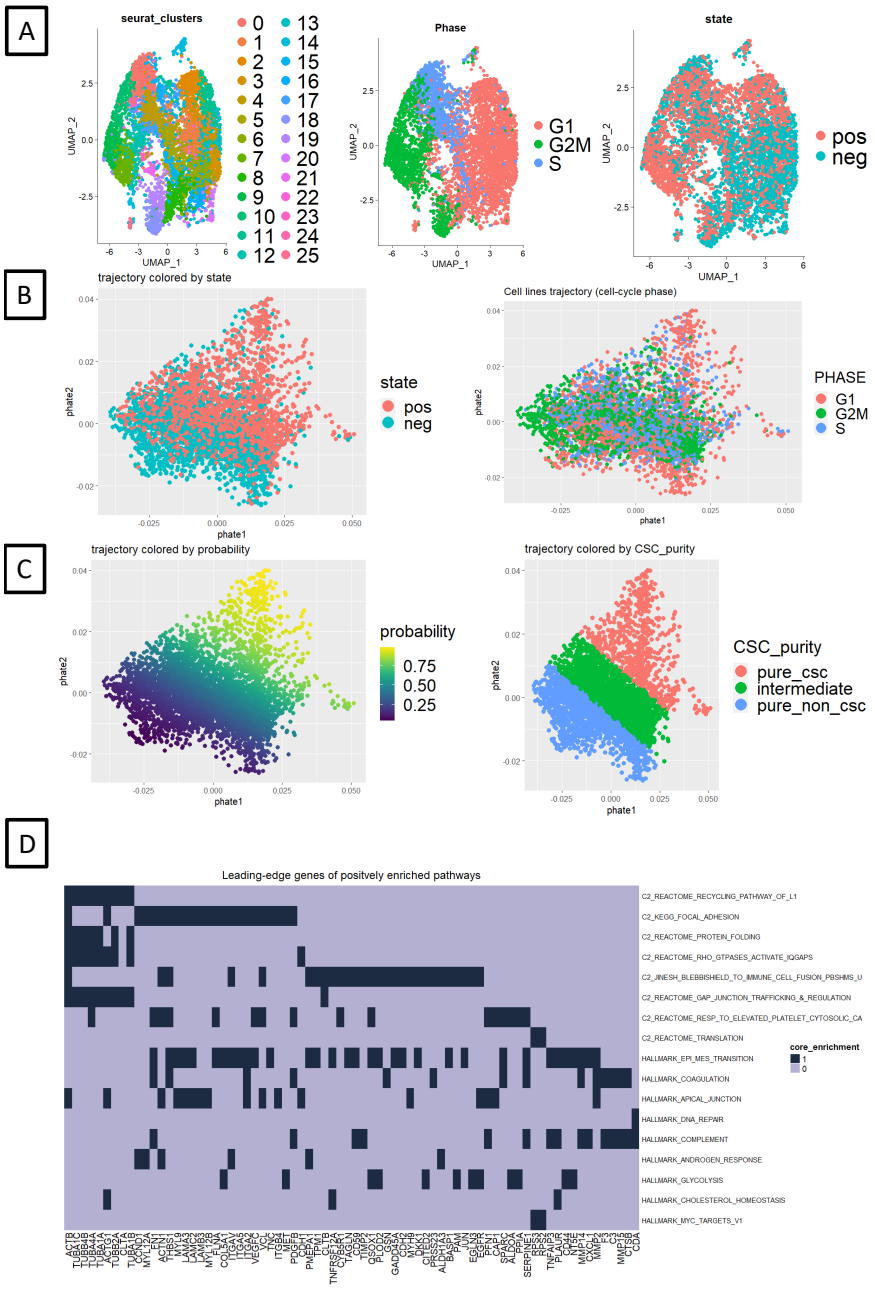


Figure 2.1 Cell line data results.

continuum but the distinction between non-CSC to CSC was weak since cell cycle phase affected the clustering (Figure 2.1A). As a next step, we eliminated the cell-cycle effect and we performed a trajectory analysis to capture both local and global nonlinear structure using an information-geometric distance between cells (Moon et al., 2019). Given recent evidence suggesting an inherent plasticity in cancer stem cells, we were interested in evaluating if there is a continuity between

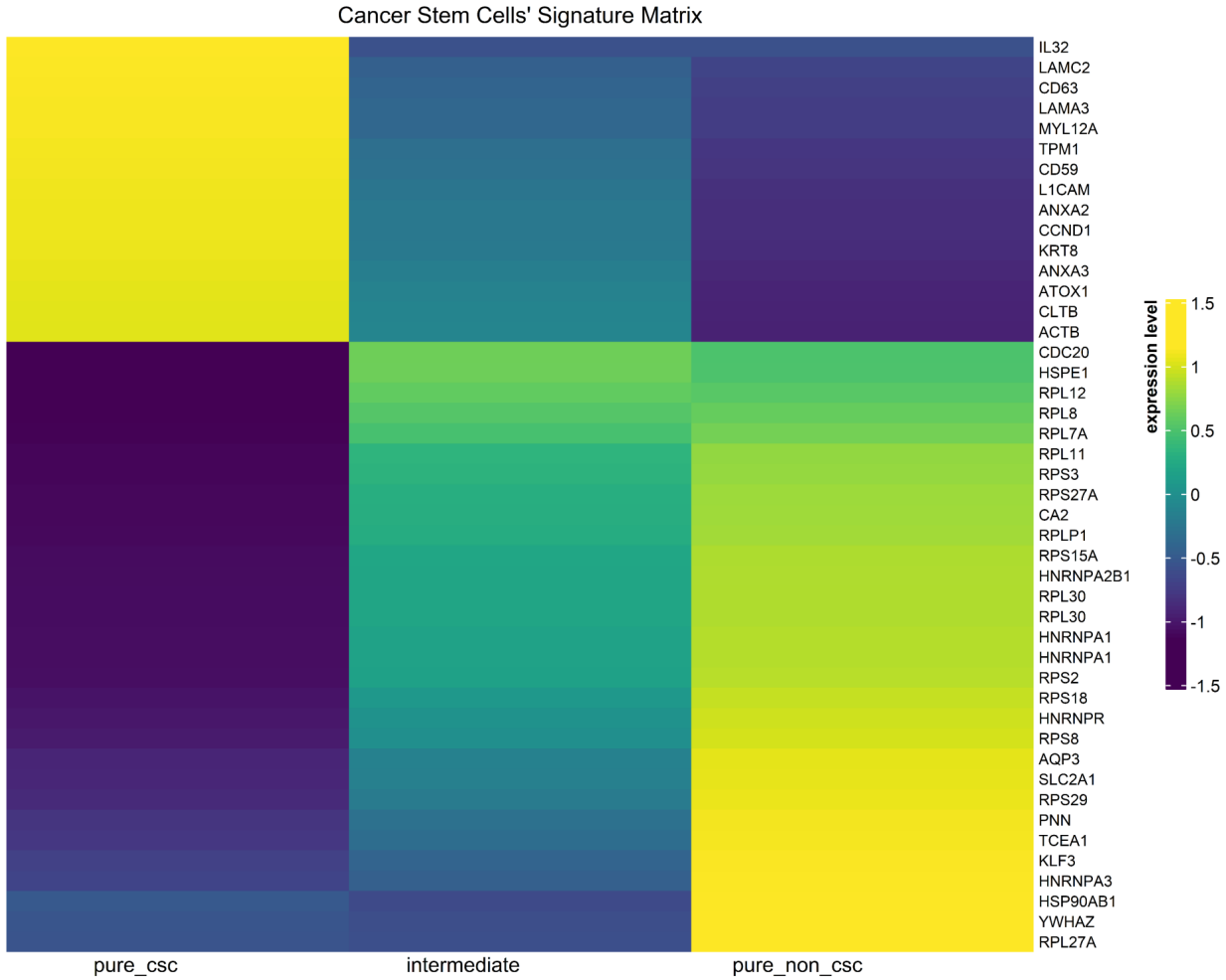


Figure 2.2 CSC signature genes matrix - Cell line data.

non-stem and stem cells in the tumor. Figure 2.1B demonstrates a spectrum of non-CSC to CSC, with an overlapping region in the middle, suggesting cells can progress from a non-stem-like to state to a stem-like state, supporting the hypothesis that CSC possesses plasticity-capacity and exist in a transitional CSC-state (Intermediate cells).

Finally, we used cell line data to generate a model to predict the probability that a cancer cell is a stem cell (or in a stem-like state) and develop a CSC-state gene expression signature. Using the trajectory coordinates of the cell line cells, we constructed a logistic regression model that provides the probability a cell is a cancer stem cell (Figure 2.1 right). We calculated the correlation of each gene's expression to the cells' predictive probabilities and used these values to rank each gene and to perform gene set enrichment analysis.

We found 29 enriched pathways in the C2 and Hallmark databases. The positively enriched pathways and contributing genes are demonstrated in the leading-edge analysis (Figure 2.1D). We tested which of the contributing genes are significantly expressed more in each predicted group of cells (CSC, Intermediate, Non-CSC) to construct a signature genes matrix (Figure 2.2). Together these data demonstrate a conserved cancer stem cell signature identified with single cell sequencing and novel bioinformatic techniques. These data nominate a subset of genes (ACTB, ANXA2, TPM1, MYL12A, CD63, CCDN1, CD59, ATOX1, LAMA3, LAMC2, L1CAM, KRT8, ANXA3,CLTB and IL32) as drivers of the cancer stem cell phenotype. Despite these cells being exclusively derived from epithelial cells, several of the CSC differentially expressed genes are associated predominantly with CAFs (TPM1, MYL12A, KRT8, CD63 and IL32), suggesting a mesenchymal state of CSC. These clusters were selected to further define a pure epithelial CSC signature in the primary tumor data of 10 patients.

2.2.2 Primary tumor data analysis

While patient-derived cell line data provides critical informatics and biologic data, it fails to capture the complexity and heterogeneity of HNC. We leveraged our access to fresh tumor specimens to perform scRNASeq techniques. Given the evidence of the tumor microenvironment playing a large role in maintenance of the cancer stem cell niche, we hypothesized that the cancer stem cell signature may differ between cell line and primary tumors, however cells in the CSC-state will have conserved signatures. To evaluate CSC signatures in primary tumors we analyzed 10 HNC harvested directly from the operative theatre. Tumors were then digested into single cell suspension and sorted by FACS for standard CSC markers (ALDH and CD44). scRNASeq was then performed on the enriched groups. Seurat clusters are shown in Figure 2.3A. Of the CD44/ALDH enriched cells, the deconvoluted epithelial tumor population was found to make up only a small proportion of the tumor bulk (5%) with the remaining cells representing the immune and stromal elements of the TME. To confirm the identity of the epithelial cell cluster, the cell line CSC expression data was normalized and mapped onto the primary tumor expression data. As seen in Figure 2.3A, the cell line data, in black, overlap with the epithelial cluster (cluster 9) confirming an epithelial expression

pattern. We then used RNAscope to show co-localization of top expressed epithelial genes within the tumor cell population to further confirm expression of the DEG genes in the primary tumor (Figure 2.3C). We considered isolating not only the epithelial but also the fibroblast cluster since CAF genes were found in the signature genes of CSC suggested by the cell line analysis. We also observe that PTPRC is low in fibroblasts of the sample and that the epithelial annotated cells are mapped on each fibroblast cluster (figure 2.3B). Hence, we proceed to investigate the expression profile of CSCs in both epithelial and fibroblast cells. Following the steps suggested by the analysis of the cell line data, we explore the trajectory of the epithelial and fibroblast cells.

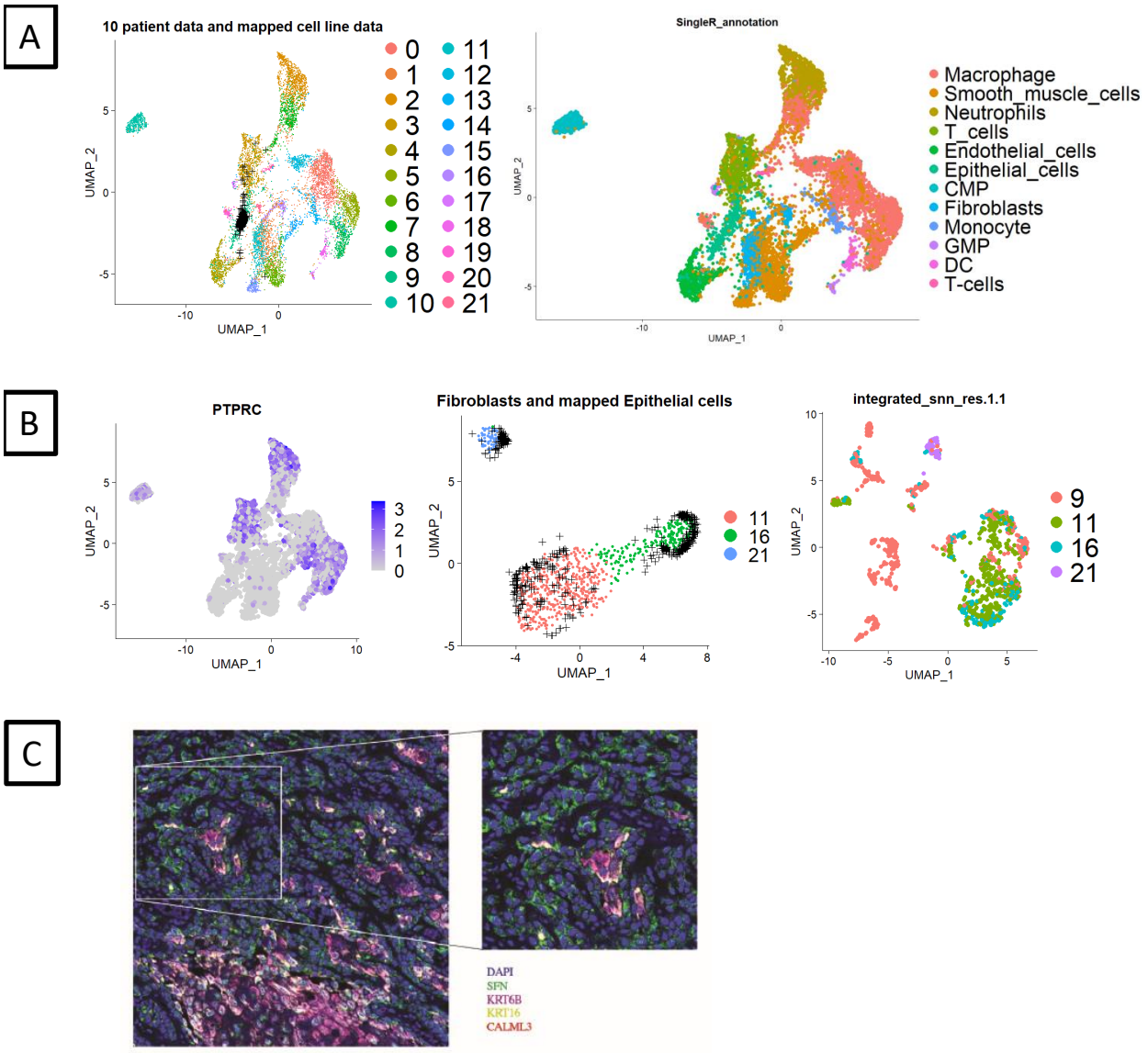


Figure 2.3 Primary tumor data analysis.

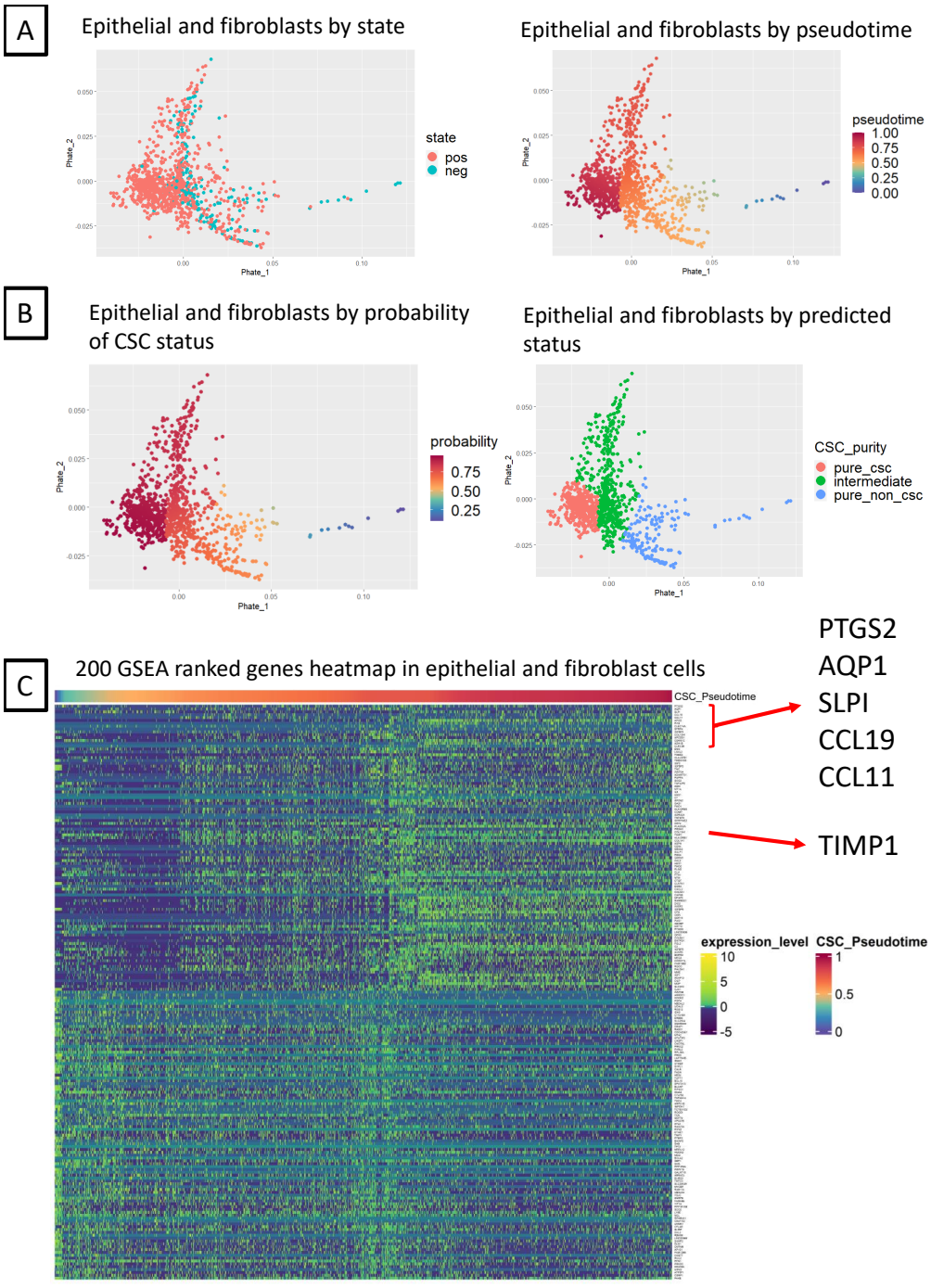


Figure 2.4 Epithelial and Fibroblast cells of primary tumor data.

Figure 2.4A demonstrates that there is a spectrum of CSCs, Intermediate and Non-CSCs on with two branches. To understand the ordering of cells on the trajectory we used the trajectory coordinates of each cell and their grouping based on ALDH and CD44 levels to extract their pseudotime (Street et al., 2018). We observe that cells with the highest pseudotime are located

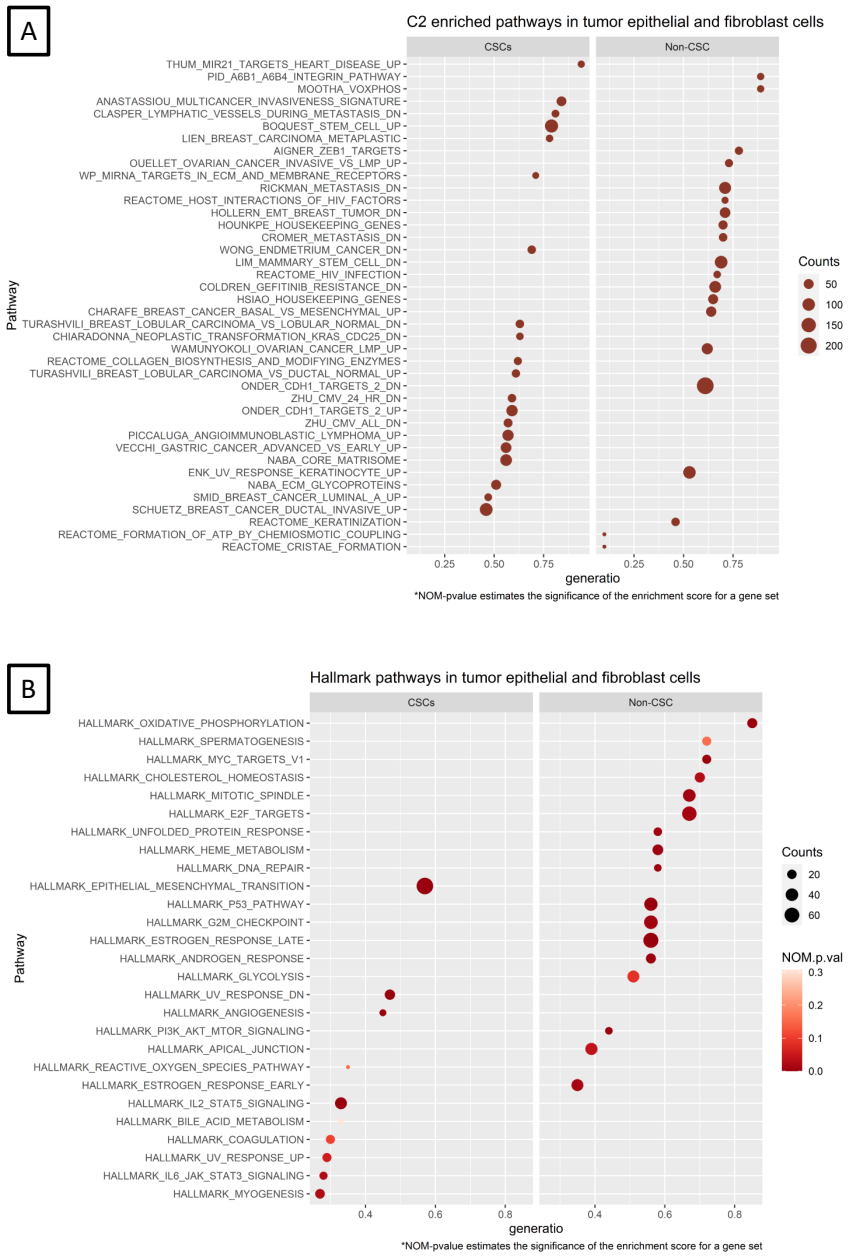


Figure 2.5 GSEA of primary tumor data.

at the leftmost part of the trajectory and have high ALDH and CD44 (sorted as CSC). For this reason, we assume that the purest of CSCs are located in the leftmost corner. To validate this assumption a logistic regression model was generated to predict the probability that a cell in the epithelial and fibroblast cluster is a CSC using the pseudotime information of the cell. Cells with the highest probability are indeed located in the leftmost corner (figure 2.4B). Additionally, genes of the fibroblast and epithelial cluster were ranked based on the correlation of their expression

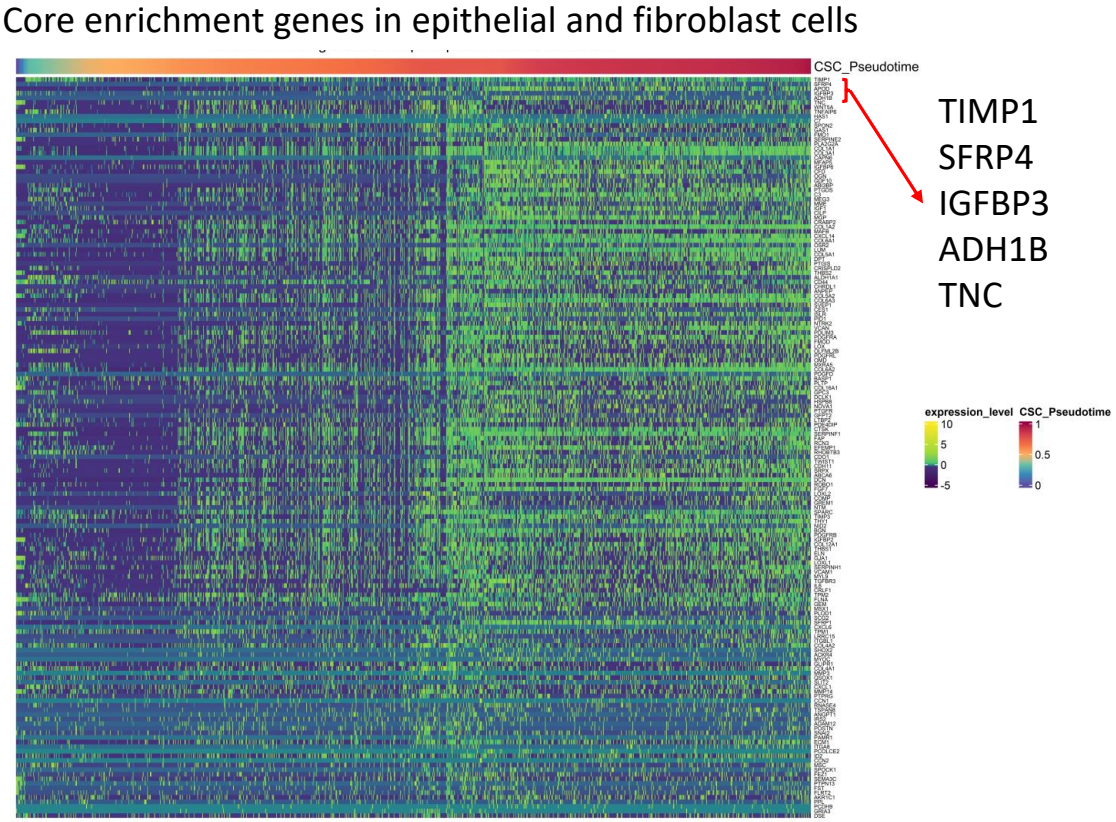


Figure 2.6 Heatmap of core enrichment genes of 10 patient data.

level with the pseudotime assigned to each cell. This ranking was used to perform gene set enrichment analysis. Positively enriched pathways are associated with the genes at the top of the list namely those that are positively correlated to pseudotime and hence the CSC cells. A heatmap of the expression of genes at the top and bottom of the ranked list shows a clustering of the tumor and epithelia cells based on pseudotime and the gradual transition from Non-CSC to CSC cells (figure 2.4C). The biologic behavior of the CSC subpopulation can be further described through the results of the GSEA of cell line and primary data. Permutation-based analysis was performed to calculate p-values and false discovery rates (FDR). Here we find that CSCs were enriched for the HALLMARK EPITHELIAL MESENCHYMAL TRANSITION, HALLMARK ANGIOGENESIS, C2 ANASTASSIO MULTICANCER INVASIVENESS SIGNATURE, and C2 BOQUEST STEM CELL UP compared to non-CSC. Interestingly, non-CSC were enriched for radiation response pathways (SMIRNOV RESPONSE TO IR 2HR UP) as well as the well-established HNC pathways, HALLMARK P53 PATHWAY, HALLMARK MYC TARGETS V1 and HALLMARK

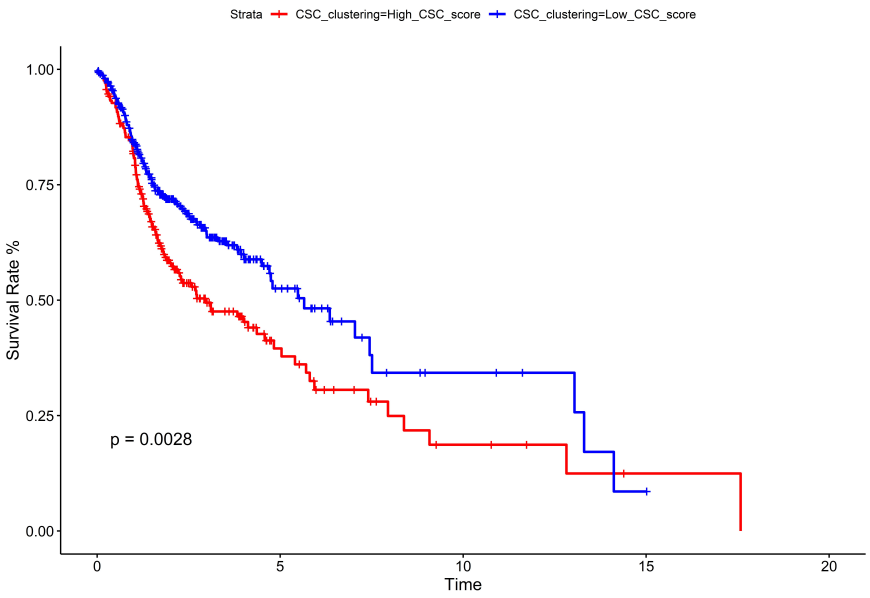


Figure 2.7 Survival rate of TCGA patients based on estimated CSC proportion.

PI3K AKT MTOR SIGNALING pathways (figure 2.5). Taken together, these findings further support that CSCs are critical components of EMT and play a crucial role in promoting tumor invasion, metastasis, and treatment resistance.

To further define the genes of interest, common genes across the significantly enriched gene sets were explored. Genes common to at least 2 significantly enriched pathways are SFRP4, ALDH1B, WNT5A, TIMP1, COL1A1, COL3A1, MFAP5, COL1A2, LUM, COL5A1, THBS2, COL5A2, COL6A3, VCAN, LOX, MXRA5, COL6A2, FAP, CDH11, DCN, SPOCK1. Given many of these genes are established mesenchymal genes (COL6A3, FAP, VCAN), this suggests that CSC may represent a critical subset of cells in a mesenchymal-state as part of the EMT.

2.2.3 Survival analysis of TCGA data

To test the significance of the CSC signature genes (figure 2.2) we utilized expression information of those genes in HPV patients of the TCGA database. Specifically, the CSC signature genes matrix was used to estimate the pure CSC, intermediate and pure Non-CSC proportion of cells in each patient via least trimmed square gene-expression deconvolution technique (Hao et al., 2018). Next, patients were grouped into two clusters via kmeans based on their estimated proportions. The two clusters separated patients with low pure CSC proportion and high pure CSC proportion. The

survival rate of patients in the cluster with low pure CSC proportion have a significanlty higher survival rate (p-value = 0.0028, figure 2.7)

2.3 Discussion

Our study provides new insights into the dynamic nature and plasticity of CSCs in head and neck cancer, and their potential involvement in the epithelial-to-mesenchymal transition (EMT) process. We identified multiple dynamic states of CSCs within our primary cell cultures of UMSCC HNC cell lines and 10 primary tumors, suggesting that CSCs exist in a state of dynamic equilibrium with their non-CSC counterparts. Our in vitro lineage tracing experiments further confirmed the plasticity of CSCs, and their ability to differentiate into non-CSCs and vice versa. Recent insight into CSC biology has moved away from them representing a distinct entity and more of a dynamic state. Our findings are consistent with previous studies that have shown the plasticity of CSCs in various cancer types, including breast cancer, colorectal cancer, and glioblastoma (Chaffer and Weinberg, 2011; Vermeulen et al., 2010; Wang et al., 2018). In addition, our study supports the hypothesis that the EMT process may be involved in the acquisition of stem cell-like properties by cancer cells and may enhance their ability to initiate and sustain tumors (Mani et al., 2008; Tam and Weinberg, 2013). This hypothesis is supported by our gene set enrichment analysis (GSEA) results, which identified enrichment of EMT-related gene sets in our CSC populations. Interestingly, as part of this mesenchymal transition, we found cells in the CSC-state had similar canonical expression patterns as CAFs. This was seen both in pure epithelial cell line data as well as in the primary data. Furthermore, we found that the TIMP1/CD63 pathway was differentially expressed in our CSC populations. TIMP1 and CD63 have both been characterized in CAFs, more so than epithelial cells. TIMP1 is a member of the tissue inhibitor of metalloproteinase (TIMP) family, regulating matrix metalloproteinases (MMPs), and have been critical mediators in cancer invasion and metastasis. TIMP1 has been shown to be overexpressed in several solid organ cancers, including breast, lung, prostate and ovarian. In addition to invasive characteristics, TIMP1 has been shown promote cancer cell survival, thus critical for cancer maintenance. CD63 is a member of the tetraspanin family of membrane proteins, thus associated with cell adhesion, migration, and signaling through the

regulation of cell surface receptor trafficking and the regulation of extracellular vesicles, which are important mediators of intercellular communication within the TME. Like TIMP1, CD63 has been found to be overexpressed in various types of cancer, including breast, lung, and melanoma. Together, both TIMP1 and CD63 are involved in several overlapping pathways that regulate EMT, specifically AKT/mTOR, WNT/b-catenin, integrins and CD44. TIMP1/CD63 has been studied in other cancers as part of the EMT as well as CSCs, suggesting that this pathway may play a role in the maintenance and plasticity of CSCs in HNC. Our study provides new evidence for the potential involvement of this pathway in HN CSC biology, and may open up new avenues for the development of targeted therapies for HNC and other cancers. Taken together, our findings highlight the dynamic and plastic nature of CSCs in HNC, and their potential involvement in the EMT process and the TIMP1/CD63 pathway. Our study may have implications for the development of personalized therapeutic strategies for this deadly disease. Further studies are needed to validate our findings in larger patient cohorts and to explore the clinical relevance of our results.

BIBLIOGRAPHY

Chaffer, C. L. and Weinberg, R. A. (2011). A perspective on cancer cell metastasis. *science*, 331(6024):1559–1564.

Chen, C.-Y., Ueha, S., Ishiwata, Y., Shichino, S., Yokochi, S., Yang, D., Oppenheim, J. J., Ogiwara, H., Deshimaru, S., Kanno, Y., et al. (2021). Combining an alarmin hmgn1 peptide with pd-l1 blockade results in robust antitumor effects with a concomitant increase of stem-like/progenitor exhausted cd8+ t cells. *Cancer immunology research*, 9(10):1214–1228.

Hao, Y., Yan, M., Lei, Y. L., and Xie, Y. (2018). Fast and robust deconvolution of tumor infiltrating lymphocyte from expression profiles using least trimmed squares. *bioRxiv*.

Mani, S. A., Guo, W., Liao, M.-J., Eaton, E. N., Ayyanan, A., Zhou, A. Y., Brooks, M., Reinhard, F., Zhang, C. C., Shipitsin, M., et al. (2008). The epithelial-mesenchymal transition generates cells with properties of stem cells. *Cell*, 133(4):704–715.

Moon, K. R., van Dijk, D., Wang, Z., Gigante, S., Burkhardt, D. B., Chen, W. S., Yim, K., van den Elzen, A., Hirn, M. J., Coifman, R. R., et al. (2019). Visualizing structure and transitions in high-dimensional biological data. *Nature Biotechnology*, 37(12):1482–1492.

Mroz, E. A., Tward, A. M., Hammon, R. J., Ren, Y., and Rocco, J. W. (2015). Intra-tumor genetic heterogeneity and mortality in head and neck cancer: analysis of data from the cancer genome atlas. *PLoS medicine*, 12(2):e1001786.

O'Brien, C. A., Pollett, A., Gallinger, S., and Dick, J. E. (2007). A human colon cancer cell capable of initiating tumour growth in immunodeficient mice. *Nature*, 445(7123):106–110.

Puram, S. V., Parikh, A. S., and Tirosh, I. (2018). Single cell rna-seq highlights a role for a partial emt in head and neck cancer. *Molecular & cellular oncology*, 5(3):e1448244.

Street, K., Risso, D., Fletcher, R. B., Das, D., Ngai, J., Yosef, N., Purdom, E., and Dudoit, S. (2018). Slingshot: cell lineage and pseudotime inference for single-cell transcriptomics. *BMC genomics*, 19:1–16.

Stuart, T., Butler, A., Hoffman, P., Hafemeister, C., Papalexi, E., III, W. M. M., Hao, Y., Stoeckius, M., Smibert, P., and Satija, R. (2019). Comprehensive integration of single-cell data. *Cell*, 177:1888–1902.

Sung, H., Ferlay, J., Siegel, R. L., Laversanne, M., Soerjomataram, I., Jemal, A., and Bray, F. (2021). Global cancer statistics 2020: Globocan estimates of incidence and mortality worldwide for 36 cancers in 185 countries. *CA: a cancer journal for clinicians*, 71(3):209–249.

Tam, W. L. and Weinberg, R. A. (2013). The epigenetics of epithelial-mesenchymal plasticity in cancer. *Nature medicine*, 19(11):1438–1449.

Vermeulen, L., De Sousa E Melo, F., Van Der Heijden, M., Cameron, K., De Jong, J. H., Borovski, T., Tuynman, J. B., Todaro, M., Merz, C., Rodermond, H., et al. (2010). Wnt activity defines colon cancer stem cells and is regulated by the microenvironment. *Nature cell biology*, 12(5):468–476.

Wang, Q., He, Z., Huang, M., Liu, T., Wang, Y., Xu, H., Duan, H., Ma, P., Zhang, L., Zamvil, S. S., et al. (2018). Vascular niche il-6 induces alternative macrophage activation in glioblastoma through hif-2α. *Nature communications*, 9(1):559.

CHAPTER 3

CLUSTERING AND VISUALIZATION OF SINGLE-CELL RNA-SEQ DATA USING PATH METRICS

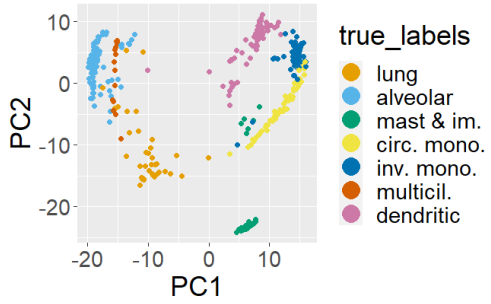
3.1 Introduction

The advance in single-cell RNA-seq (scRNA-seq) technologies in recent years has enabled the simultaneous measurement of gene expression at the single-cell level Saliba et al. (2014); Eberwine et al. (1992); Tang et al. (2009). This opens up new possibilities to detect previously unknown cell populations, study cellular development and dynamics, and characterize cell composition within bulk tissues. Despite its similarity with bulk RNAseq data, scRNAseq data tends to have larger variation and larger amounts of missing values due to the low abundance of initial mRNA per cell. To address these challenges, numerous computational algorithms have been proposed focusing on different aspects. Given a collection of single cell transcriptomes from scRNAseq, one of the most common applications is to identify and characterize subpopulations, e.g., cell types or cell states. Numerous clustering approaches have been developed such as k-means based methods SC3 Kiselev et al. (2017), SIMLR Wang et al. (2017), and RaceID Herman et al. (2018); hierarchical clustering based methods CIDR Lin et al. (2017), BackSPIN A et al. (2015), and pcaReduce žurauskienė and Yau (2016); graph based methods Rphenograph CLevine et al. (2015), SNN-Cliq Xu and Su (2015), SSNN-Louvain Zhu et al. (2020), Seurat Stuart et al. (2019), and scanpy Wolf et al. (2018); and deep-learning based methods scGNN Wang et al. (2021), scVI Lopez et al. (2018), ScDeepCluster Tian et al. (2019b), DANCE Ding et al. (2022). To visualize and characterize relationships between cell types, it is important to represent them in a low-dimensional space. Many low-dimensional embedding methods have been proposed including UMAP McInnes et al. (2018), t-SNE Van der Maaten and Hinton (2008), PHATE Moon et al. (2019), and LargeVis Tang et al. (2016). However, a key challenge for embedding methods is to simultaneously reduce cluster variance and preserve the global geometry, including the distances between clusters and cluster shapes. For example, Figure 3.4 illustrates the typical situation on a cell mixture dataset Tian et al. (2019a): the PCA embedding preserves the global geometry but clusters have high variance; clusters are better separated in the

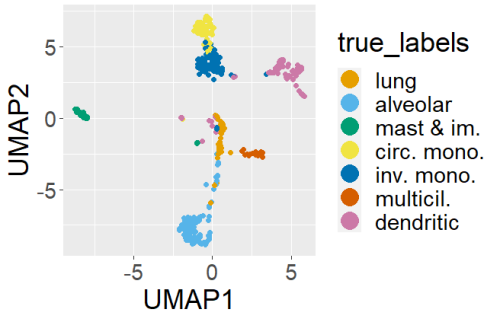
UMAP and t-SNE embeddings, but the global geometric structure of the clusters is lost.

When choosing a clustering algorithm, there is always an underlying tension between respecting data density and data geometry. Density-based methods such as DBSCAN cluster data by connecting together high-density regions, regardless of cluster shape. More traditional approaches such as k-means require that clusters are convex and geometrically well separated. However, in many real data, clusters tend to have both nonconvex/elongated geometry and a lack of robust density separation as shown in Figure 3.2b which consists of three elongated Gaussian distributions and a bridge connecting two of the distributions. The data set is challenging because it exhibits elongated geometry, but methods relying only on density will fail due to the bridge. Such characteristics are commonly observed in scRNA-seq data, especially for cells sampled from a developmental process, as cell types often trace out elongated structures and frequently lack robust density separation. This elongated geometry phenomena is due to the fact that all the cell types originate from stem cells through a trajectory-like differentiation process, and the bridge structures are created by the cells in the transition states. For example, circulating monocytes in the Tabula Muris (TM) lung data set Tabula Muris Consortium (2018) have an elongated cluster structure as illustrated by the PCA plot in Figure 3.1a, as do the ductal cells in the TM pancreatic data set (see Figure 3.1c). The UMAP plots of these same data sets illustrate the lack of robust density separation: for TM lung, there is a bridge connecting the alveolar and lung cell types, and also an overlap/bridge between the circulating and invading monocytes (see Figure 3.1b); for TM pancreatic, the pancreatic A and pancreatic PP cells are not well separated. The combination of elongation and poor density separation make clustering scRNA-seq data sets a challenging task.

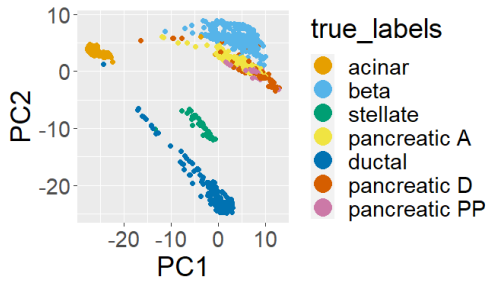
We propose an embedding method based on the *power weighted path metric* which is well suited to this difficult regime. These metrics balance density and geometry considerations in the data learning tasks such as clustering and semi-supervised learning Vincent and Bengio (2003); Bousquet et al. (2004); Sajama and Orlitsky (2005); Chang and Yeung (2008); Bijral et al. (2011); Moscovich et al. (2017); Mckenzie and Damelin (2019); Little et al. (2020a); Borghini et al. (2020). They have performed well in applications such as imaging Fischer et al. (2001); Zhang and Murphy



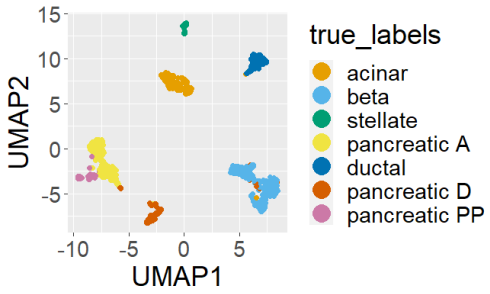
(a) PCA TM Lung



(b) UMAP TM Lung

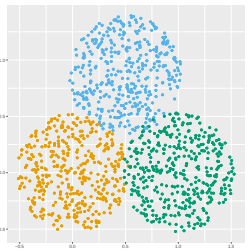


(c) PCA TM Pancreatic

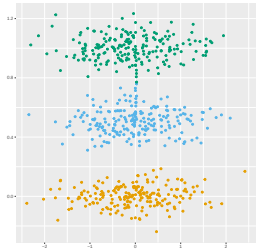


(d) UMAP TM Pancreatic

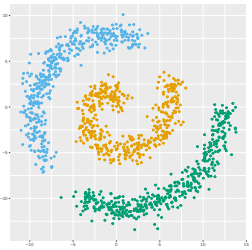
Figure 3.1 Tabula Muris data sets have elongated clusters in the PCA embedding and clusters connected with a bridge of points in the UMAP embedding.



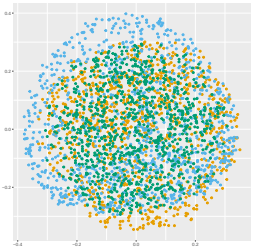
(a) Balls



(b) Elongated with Bridge



(c) Swiss Roll



(d) GL Manifold

Figure 3.2 Toy Data Sets. 3.2a and 3.2b show the 2-dimensional data sets. 3.2c plots the first two coordinates of the Swiss roll. 3.2d shows the 2-dimensional PCA plot of the SO(3) manifolds.

(2021); Little et al. (2020a); Mckenzie and Damelin (2019), but their usefulness for the analysis of scRNAseq data remains unexplored. Because these metrics are density-sensitive, they reduce cluster variance; in addition, these metrics also capture global distance information, and thus preserve global geometry; see Figure 3.4b. Using the path metric embedding to cluster the data thus yields a clustering method which balances density-based and geometric information.

3.2 Materials and Methods

We first introduce our theoretical framework in Section 3.2.1; Section 3.2.2 then describes the details of the proposed scPMP algorithm, and Section 3.2.3 describes metrics for assessment.

3.2.1 Path Metrics

We first define a family of power-weighted path metrics parametrized by $1 \leq p < \infty$.

Definition 1

Given a discrete data set X, the discrete p-power weighted path metric between $a, b \in X$ is defined as $\ell_p(a,b) := \inf_{(x_0,\ldots,x_s)} \left(\sum_{i=0}^{s-1} \left\|x_{i+1} - x_i\right\|_2^p\right)^{\frac{1}{p}}$, where the infimum is taken over all sequences of points $x_0, \ldots, x_s$ in X with $x_0 = a$ and $x_s = b$.

Note as $p \to \infty$, ℓ_p converges to the "bottleneck edge" distance

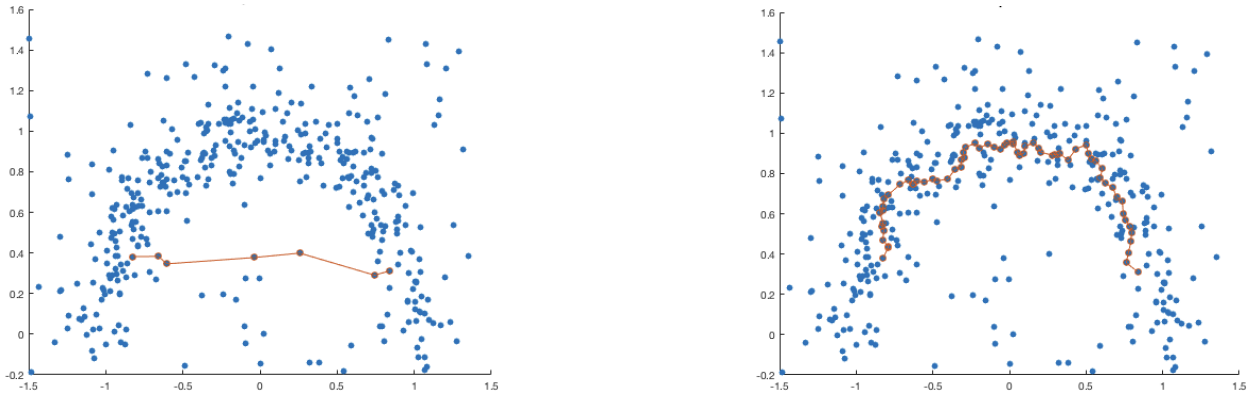
$$\ell_\infty(a,b) := \inf_{(x_0,\ldots,x_s)} \max_i \|x_{i+1} - x_i\|_2,$$

which is well studied in the computer science literature Pollack (1960); Hu (1961); Camerini (1978); Gabow and Tarjan (1988). Two points are close in ℓ_∞ if they are connected by a high-density path through the data, regardless of how far apart the points are. On the other hand, when $p = 1$, ℓ_1 reduces to Euclidean distance. If path edges are furthermore restricted to lie in a nearest neighbor graph, ℓ_1 approximates the geodesic distance between the points, i.e. the length of the shortest path lying on the underlying data structure, which is a highly useful metric for manifold learning Tenenbaum et al. (2000). The parameter p governs a trade-off between these two extremes, i.e. it determines how to balance density and geometry considerations when determining which data points should be considered close. The relationship between ℓ_p and density can be made precise. Assume n independent samples from a continuous, nonzero density function f supported on a d-dimensional, compact Riemannian manifold $\mathcal{M}$ (a manifold is a smooth, locally linear surface; see Lee (2018)). Then for $p > 1$, $\ell_p(a,b)$ converges (after appropriate normalization) to

$$\mathcal{L}_p(a,b) := \inf_\gamma \left(\int f(\gamma(t))^{-\frac{(p-1)}{d}} |\gamma'(t)|\, dt\right)^{\frac{1}{p}}, \tag{3.1}$$

as $n \to \infty$, where the infimum is taken over all smooth curves $\gamma : [0,1] \to \mathcal{M}$ connecting a, b Hwang et al. (2016); Groisman et al. (2022); Fernández et al. (2023). Note $|\gamma'(t)|$ is simply the arclength element on $\mathcal{M}$, so $\mathcal{L}_1$ reduces to the standard geodesic distance. When $p \neq 1$, one obtains a density-weighted geodesic distance. The optimal $\mathcal{L}_p$ path is not necessarily the most direct: a detour may be worth it if it allows the path to stay in a high-density region; see Figure 3.3. Thus the metric is *density-sensitive*, in that distances across high-density regions are smaller than distances across low-density regions; this is a desirable property for many machine learning tasks Chu et al. (2017), including trajectory estimation for developmental cells and cancer cells. However the metric is also *geometry preserving*, since it is computed by path integrals on $\mathcal{M}$. The parameter p controls the balance of these two properties: when p is small, $\mathcal{L}_p$ depends mainly on the geometry of the data, while for large p, $\mathcal{L}_p$ is primarily determined by data density.

Although path metrics are defined in a complete graph, i.e. Definition 1 considers *every* path in the data connecting a, b, recent work Little et al. (2020b); Groisman et al. (2018); Mckenzie and Damelin (2019); Chu et al. (2020) has established that it is sufficient to only consider paths in a K-nearest neighbors (KNN) graph, as long as $K \geq C \log n$ for a constant C depending on p, d, f, and the geometry of the data. By restricting to a KNN graph, all pairwise path distances can be computed in $O(Kn^2)$ with Dijkstra's algorithm.



(a) $p = 1.1$ (b) $p = 2$

Figure 3.3 Optimal ℓ_p path between two points in a moon data set.

Algorithm 3.1 scPMP.

Input: noisy data $\widetilde{X} \in \mathbb{R}^{n \times d}$, parameter p, number of clusters k
Optional input: $K_1, K_2, r_{\min}, r_{\max}, \tau$
(Defaults: $12, n \wedge 500, 3, 39, 0.01$)
Output: scPMP embedding $Y \in \mathbb{R}^{n \times r}$, label vector $\hat{\ell} \in [k]^n$

% Denoise data:
$x_i \leftarrow \frac{1}{K_1} \sum_{j \in \mathcal{N}_{i,K_1}} \widetilde{x}_i$

% Compute path metrics:
$\mathcal{G}^p_{K_2} \leftarrow K_2$NN graph on X with edge weights $\|x_i - x_j\|^p$
$D^p_{ij} \leftarrow$ length of shortest path connecting x_i, x_j in $\mathcal{G}^p_{K_2}$
$(D_{\mathrm{PM}})_{ij} \leftarrow (D^p_{ij})^{\frac{1}{p}}$

% Compute MDS embedding of path metrics:
$B \leftarrow -\frac{1}{2} J D^{(2)}_{\mathrm{PM}} J$
$\Lambda = \mathrm{diag}(\lambda_1, \ldots, \lambda_n) \leftarrow$ eigenvalues of B in descending order
$V = (v_1, \ldots, v_n) \leftarrow$ corresponding eigenvectors of B
$r \leftarrow$ index maximizing $\lambda_i / \lambda_{i+1}$ for i satisfying $r_{\min} \le i \le r_{\max}$, $\lambda_i / \lambda_1 \ge \tau$
$Y \leftarrow (\sqrt{\lambda_1} v_1, \ldots, \sqrt{\lambda_r} v_r) \in \mathbb{R}^{n \times r}$

% Cluster the data:
$\hat{\ell} \leftarrow$ constrained k-means(Y, k)

3.2.2 Algorithm

We consider a noisy data set of n data points $\widetilde{x}_1, \ldots, \widetilde{x}_n \in \mathbb{R}^d$, which form the rows of noisy data matrix $\widetilde{X} \in \mathbb{R}^{n \times d}$. We first denoise the data with a local averaging procedure, which has been shown to be advantageous for manifold plus noise data models García Trillos et al. (2019). More specifically, we replace $\widetilde{x}_i$ with its local average:

$$x_i := \frac{1}{K_1} \sum_{j \in \mathcal{N}_{i,K_1}} \widetilde{x}_j \quad , \quad \mathcal{N}_{i,K_1} = \{ j : \widetilde{x}_j \text{ is a } K_1\text{NN of } \widetilde{x}_i \} ,$$

and let $X \in \mathbb{R}^{n \times d}$ denote the denoised data matrix.

We then fix p and compute the p-power weighted path distance between all points in X to obtain pairwise distance matrix $D_{\mathrm{PM}} \in \mathbb{R}^{n \times n}$. More precisely, we let $\mathcal{G}^p_{K_2} = (X, E)$ be the graph on X where x_i, x_j are connected with edge weight $E_{ij} = \|x_i - x_j\|^p$ if x_i is a K_2NN of x_j or x_j is a K_2NN of x_i. We then compute D^p_{ij} as the total length of the shortest path connecting x_i, x_j in $\mathcal{G}^p_{K_2}$,

and define D_{PM} by $(D_{\text{PM}})_{ij} = (D_{ij}^p)^{\frac{1}{p}}$.

We next apply classical multidimensional scaling Borg and Groenen (2005) to obtain a low-dimensional embedding which preserves the path metrics. Specifically, we define the path metric MDS matrix $B = -\frac{1}{2}JD_{\text{PM}}^{(2)}J$ where $J = I_n - \frac{1}{n}\mathbf{1}\mathbf{1}^T$ is the centering matrix, $\mathbf{1} \in \mathbb{R}^n$ is a vector of all 1's, and $D_{\text{PM}}^{(2)}$ is obtained from D_{PM} by squaring all entries. We let the spectral decomposition of B be denoted by $B = V\Lambda V^T$, where $\Lambda = \text{diag}(\lambda_1, \ldots, \lambda_n)$, $V = (v_1, \ldots, v_n) \in \mathbb{R}^{n\times n}$ contain the eigenvalues and eigenvectors of B in descending order. The embedding dimension r is then chosen as the index i which maximizes the eigenratio λ_i/λ_{i+1} Lam and Yao (2012), with the following restrictions: we constrain $3 \leq i \leq 39$ and only consider ratios λ_{i+1}/λ_i between "large" eigenvalues, i.e. we require $\lambda_i/\lambda_1 \geq 0.01$. The scPMP embedding is then defined by $Y = (\sqrt{\lambda_1}v_1, \ldots, \sqrt{\lambda_r}v_r) \in \mathbb{R}^{n\times r}$.

Finally, we apply k-means to the scPMP embedding to obtain cluster labels. Specifically, we let $\hat{\ell}_i \in [k] = \{1, \ldots, k\}$ be the cluster label of x_i returned by running k-means on Y with k clusters and 20 replicates. Since k-means may return highly imbalanced clusters, cluster sample sizes were constrained to be at least $\sqrt{n}/2$. Specifically, if k-means returned a tiny cluster, k was increased to $k + 1$, and the tiny cluster merged with the closest non-trivial cluster. This entire procedure is summarized in the pseudocode in Algorithm 3.1.

We note that the computational bottleneck for Algorithm 3.1 is the computation and storage of all pairwise path distances, which has complexity $O(n^2 \log n)$ when $K_2 = O(\log n)$. However this quadratic cost can be avoided by utilizing a low rank approximation of the squared distance matrix via the Nystrom method Williams and Seeger (2001); Ghojogh et al. (2020); Platt (2005); Yu et al. (2012); Civril et al. (2006). For example, Shamai et al. (2020) propose a fast, quasi-linear implementation of MDS which only requires the computation of path distances from a set of q landmarks, so that the complexity of computing path distances is reduced to $O(qn \log n)$. Our implementation of scPMP includes the option to use this landmark-based approximation and is thus highly scalable.

We also note that an important consideration in the fully unsupervised setting is how to select

the number of clusters k. This is a rather ill-posed question with multiple reasonable answers due to hierarchical cluster structure. We do not focus on this in the current article, and Algorithm 3.1 assumes the number of clusters is given. However we emphasize that when k is unknown, the scPMP embedding offers a useful tool for selecting a reasonable number of clusters. For example, Line 21 of Algorithm 3.1 can be repeated for a range of candidate k values to obtain candidate clusterings $\widehat{\ell_k}$; $\widehat{k}$ can then be chosen so that $\widehat{\ell_k}$ optimizes a cluster validity criterion such as the silhouette criterion Kaufman and Rousseeuw (2009); Maechler et al. (2021). Alternatively, one could build a graph with distances computed in the scPMP embedding, and estimate k as the number of small eigenvalues of a corresponding graph Laplacian Von Luxburg (2007); Little et al. (2020a).

3.2.3 Assessment

We evaluate the performance of Algorithm 3.1 with respect to (1) cluster quality and (2) geometric fidelity on a collection of labeled benchmarking data sets with ground truth labels ℓ. There are many helpful metrics for the quality of the estimated cluster labels $\hat{\ell}$, and we compute the adjusted rand index (ARI), entropy of cluster accuracy (ECA), and entropy of cluster purity (ECP). Definitions of ECA and ECP can be found in Appendix B. We compare our clustering results with the output of k-means, DBSCAN Ester et al. (1996); Xu et al. (1998), k-means on t-SNE embedding Van der Maaten and Hinton (2008), DBSCAN on UMAP embedding McInnes et al. (2018) and for scRNAseq data sets additionally with the following scRNAseq clustering methods: SC3 Kiselev et al. (2017), scanpy Wolf et al. (2018), RaceID3 Grün et al. (2018), SIMRL Wang et al. (2017) and Seurat Stuart et al. (2019).

Assessing the geometric fidelity of the low-dimensional embedding Y is more delicate; we want to assess whether the embedding procedure preserves the global relative distances between clusters. We first compute the mean of each cluster as in Van der Maaten and Hinton (2008) using the ground truth labels, i.e. $\mu_j(X) = \frac{1}{|\mathcal{I}_j|}\sum_{i\in\mathcal{I}_j} x_i$ where $\mathcal{I}_j = \{i : \ell_i = j\}$; we then define $D_{\mu,X}(i,j) = \|\mu_{\ell_i}(X) - \mu_{\ell_j}(X)\|_2$. Similarly, we compute the means $\mu_j(Y)$ in the scPMP embedding, and define $D_{\mu,Y}(i,j) = \|\mu_{\ell_i}(Y) - \mu_{\ell_j}(Y)\|_2$; we then compare $D_{\mu,X}$ and $D_{\mu,Y}$. Specifically, we

define the geometric perturbation π by:

$$\pi(X, Y, \ell) = \min_c \frac{\left\|D_{\mu,X} - cD_{\mu,Y}\right\|_F^2}{\left\|D_{\mu,X}\right\|_F^2},$$

where $\|\cdot\|_F$ is the Frobenius norm. The c achieving the minimum is easy to compute, and one obtains

$$\pi(X, Y, \ell) = \frac{\left\|D_{\mu,X} - c^* D_{\mu,Y}\right\|_F^2}{\left\|D_{\mu,X}\right\|_F^2} \quad , \quad c* = \frac{\langle D_{\mu,X}, D_{\mu,Y}\rangle}{\left\|D_{\mu,Y}\right\|_F^2}.$$

We compare $\pi(X, Y, \ell)$ with the geometric perturbation of other embedding schemes for X, i.e. with $\pi(X, U, \ell)$ for U equal to the UMAP McInnes et al. (2018) and t-SNE Van der Maaten and Hinton (2008) embeddings. Note that π is not always a useful measure: for example if X consisted of concentric spheres sharing the same center, the metric would be meaningless, as the distance between cluster means would be zero. Nevertheless, in most cases π is a helpful metric for quantifying the preservation of global cluster geometry.

3.3 Results

We apply Algorithm 3.1 to both a collection of toy manifold data sets and a collection of scRNAseq data sets. Results are reported in Sections 3.3.1 and 3.3.2 respectively. The default parameter values reported in Algorithm 3.1 were used on all data sets.

3.3.1 Manifold Data

We apply Algorithm 1 for $p = 1.5, 2, 4$ to the following four manifold data sets:

Balls ($n = 1200$, $d = 2$, $k = 3$): Clusters were created by uniform sampling of 3 overlapping balls in $\mathbb{R}^2$; see Figure 3.2a.

Elongated with bridge (denoted EWB, $n = 620$, $d = 2$, $k = 3$): Clusters were created by sampling from 3 elongated Gaussian distributions. A bridge was added connecting two of the Gaussians; see Figure 3.2b.

Swiss roll ($n = 1275$, $d = 3$, $k = 3$): Clusters were created by uniform sampling from three distinct regions of a Swiss roll; 3-dimensional isotropic Gaussian noise ($\sigma = 0.75$) was then added to the data. Figure 3.2c shows the first two data coordinates.

Method	Balls	EWB	Swiss	SO(3)
k-means	**0.955**	-0.001	0.373	0.010
DBSCAN	0.055	0.550	**1**	**1**
UMAP+DBSCAN	0.600	0.645	**1**	**1**
t-SNE+k-means	0.895	0.359	**1**	0.532
Seurat	0.777	0.837	**1**	**1**
$PM_{1.5}$	0.921	0.489	**1**	0.501
PM_2	0.907	**0.990**	**1**	**1**
PM_4	0.781	0.584	**1**	**1**

Table 3.1 ARI for manifold data.

SO(3) manifolds ($n = 3000$, $d = 1000$, $k = 3$): For $1 \leq i \leq 3$, the 3-dimensional manifold $\mathcal{M}_i \subseteq \mathbb{R}^9$ is defined by fixing three eigenvalues $D_i = \text{diag}(\lambda_1, \lambda_2, \lambda_3)$ and then defining $\mathcal{M}_i = \cup_{V \in SO(3)} V D_i V^T$, where SO(3) is the special orthogonal group. After fixing D_i, we randomly sample from $\mathcal{M}_i$ by taking random orthonormal bases V of $\mathbb{R}^3$. A noisy, high-dimensional embedding was then obtained by adding uniform random noise with standard deviation $\sigma = 0.0075$ in 1000 dimensions. Figure 3.2d shows the first two principal components of the data, which exhibits no cluster separation.

The data sets were chosen to illustrate various cluster separability characteristics. For the balls, the clusters have good geometric separation but are not separable by density. For the Swiss roll and SO(3), the clusters have a complex and inter-twined geometry but are well separated in terms of density. For EWB, clusters are both elongated and lack robust density separability due to the bridge, and one expects that methods which rely too heavily on either geometry or density will fail. The ARIs achieved by Algorithm 3.1, k-means based methods, DBSCAN based methods, and Seurat are reported in Table 3.1. See Tables B.1 and B.2 in Appendix B for ECP and ECA. As expected, k-means out performs all methods on the balls but performs very poorly on all other data sets. DBSCAN and Seurat achieve perfect accuracy on the Swiss roll and SO(3) but perform rather poorly on the balls and EWB, although Seurat does noticeably better than DBSCAN. PM_2 is the only method which achieves a high ARI ($> 90\%$) and a low ECP and ECA (< 0.15) on all data sets.

Table 3.2 reports the geometric perturbation of the embedding produced by Algorithm 3.1 and

Method	Balls	EWB	Swiss	SO(3)
2d UMAP	0.001	0.006	0.305	0.071
rd UMAP	**0**	0.033	0.339	**0.054**
2d t-SNE	**0**	**0.004**	0.187	0.171
rd t-SNE	**0**	0.042	0.074	0.157
2d $PM_{1.5}$	**0**	0.033	**0.002**	0.103
rd $PM_{1.5}$	**0**	0.023	0.011	0.154
2d PM_2	**0**	0.146	0.025	0.156
rd PM_2	**0**	0.068	0.025	0.179
2d PM_4	0.003	0.191	0.056	0.194
rd PM_4	0.004	0.157	0.056	0.194

Table 3.2 Geometric perturbation for manifold data. The rd UMAP embeddings were computed with an embedding dimension of $r = 5$ for the balls, EWB, Swiss roll and $r = 7$ for SO(3), which corresponded to the estimated dimension for both $PM_{1.5}$ and PM_2. For t-SNE, $r = 3$ for all data sets.

compares with UMAP and t-SNE. Since Algorithm 3.1 generally selects an embedding dimension $r > 2$, to ensure a fair comparison the geometric perturbation was computed in both the 2d and r-dimensional (rd) embeddings for all methods, where for UMAP r is the dimension selected by Algorithm 3.1 and for t-SNE $r = 3$ (note $r \leq 3$ was required in Rtsne implementation). Overall $PM_{1.5}$ achieved the lowest geometric perturbation, although all methods had small perturbation on the Balls data set and t-SNE had the lowest perturbation on EWB. We point out however that for both the Swiss roll and SO(3), the metric may not be meaningful due to the complicated cluster geometry.

3.3.2 scRNAseq Data

We apply Algorithm 1 for $p = 1.5, 2, 4$ to the following synthetic scRNAseq data sets:

RNA mixture: Benchmarking scRNAseq data set from Tian et al. (2019a). RNAmix1 was processed with CEL-seq2 and has $n = 296$ cells and $d = 14687$ genes. RNAmix2 was processed with Sort-seq and has $n = 340$ cells and $d = 14224$ genes. For the creation of the two data sets, RNA was extracted in bulk for each of the following cell lines: H2228, H1975, HCC827. Then the RNA was mixed in $k = 7$ different proportions (each defining a ground truth cluster label), diluted to single cell equivalent amounts ranging from 3.75pg to 30pg, and processed using CEL-seq2 and SORT-seq. See here for Supplemental info including ground truth geometric structure.

Simulated beta: Simulated data set of $n = 473$ beta cells and $d = 2279$ genes, created based on SAVER Huang et al. (2018) and scImpute Li and Li (2018). First, we subset the Baron's Pancreatic data set Baron et al. (2016) to include only Beta cells. As in Li and Li (2018), we randomly choose 10% of the genes to operate as marker genes. Then, we split the cells to $k = 3$ clusters and each cluster is assigned a different group of marker genes. For each cluster we scale up the mean expression of its marker genes. Lastly, to simulate the drop out effect, as in Huang et al. (2018), we multiply each cell by an efficiency loss constant drawn by Gamma(10, 100). Using S to refer to the data matrix resulting from the above steps, the final simulated data X is obtained by letting X_{ij} be drawn from Poisson(S_{ij}).

In addition to the synthetic data, we evaluate the performance of Algorithm 3.1 on the following real scRNAseq data sets:

Cell mixture data set: Another benchmarking data set from Tian et al. (2019a) consisting of a mixture of $k = 5$ cell lines created with 10x sequencing platform. The cell line identity of a cell is also its true cluster label. The data set consists of $n = 3822$ cells and $d = 11786$ genes; we removed multiplets, based on the provided metadata file and kept 3000 most variable genes after SCT tranformation Hafemeister and Satija (2019); Choudhary and Satija (2022).

Baron's pancreatic: Human pancreatic data set generated by Baron et al. (2016). After quality control and SAVER imputation, there are $d = 14738$ genes and $n = 1844$ cells. For analysis purposes cells that belong in a group with less than 70 members were filtered out to reduce to $k = 8$ cell types. Also, we kept only the 3000 most variable genes after SCT tranformation Hafemeister and Satija (2019); Choudhary and Satija (2022). The cell types associated with each cell were obtained by an iterative hierarchical clustering method that restricts genes enriched in one cell type from being used to separate other cell types. The enriched markers in every cluster defined the cell type of the cells that belong in that cluster.

Tabula Muris data sets: Mouse scRNAseq data for different tissues and organs Tabula Muris Consortium (2018). We select the pancreatic data (TM Panc) with $n = 1444$ cells and $d = 23433$ genes and the lung data (TM Lung) with $n = 453$ cells and $d = 23433$ genes. Both

Method	RNA1	RNA2	TMLung	Beta	TMPanc	BaronPanc	PBMC4K	CellMix
SC3	0.637	0.827	0.798	0.969	0.894	0.767	0.889	**1**
scanpy	0.620	0.825	0.796	0.898	0.615	0.966	0.977	**1**
RaceID3	0.730	0.520	**0.900**	0.714	0.751	0.651	0.763	**1**
SIMLR	0.878	0.792	0.727	0.969	0.599	0.698	0.705	**1**
Seurat	0.792	0.667	0.843	0.901	0.547	0.941	0.889	0.993
Seurat_def	0.714	0.785	0.764	0.907	0.798	**0.971**	0.975	**1**
k-means	0.921	0.786	0.848	0.957	0.840	0.662	0.747	**1**
DBSCAN	**0.952**	0.826	0.587	0.541	0.734	0.724	0.889	**1**
UMAP+DBSCAN	0.926	0.892	0.619	0.946	0.893	0.848	0.974	**1**
t-SNE+k-means	0.943	0.915	0.753	0.928	0.620	0.641	0.596	0.878
$PM_{1.5}$	0.939	0.924	0.888	0.969	0.626	0.804	0.754	**1**
PM_2	0.939	**0.973**	0.808	0.969	**0.921**	0.969	0.757	**1**
PM_4	0.939	0.939	0.731	**0.975**	0.775	0.853	**0.978**	**1**

Table 3.3 ARI for RNA data.

data sets have $k = 7$ different cell types which were characterized by an FACS-based full length transcript analysis.

PBMC4k data set: This data set includes the gene expression of Peripheral Blood Mononuclear Cells. The raw data are available from 10X Genomics. After quality control, saver imputation, and removing the two smallest cell types, there are $d = 16655$ genes and $n = 4316$ cells in the dataset. Also, we merge CD8+ T-cells and CD4+ T-cells in one type named T-cells resulting in $k = 4$ cell types. The ground truth cell types are provided by SingleR annotation after marker gene verification in github.com/SingleR.

Details about the pre-processing of data sets can be found in Appendix A. For the following UMAP and t-SNE results, Linnorm normalization was applied without denoising, as this normalization gave the best results. Note Seurat_def refers to the results of the entire Seurat pipeline, whereas Seurat refers to the result of using Seurat clustering on data with the same processing and normalization as for PM. The embedding dimension r selected by Algorithm 3.1 ranged from 3 to 7 for $PM_{1.5}$ and PM_2, and from 3 to 11 for PM_4.

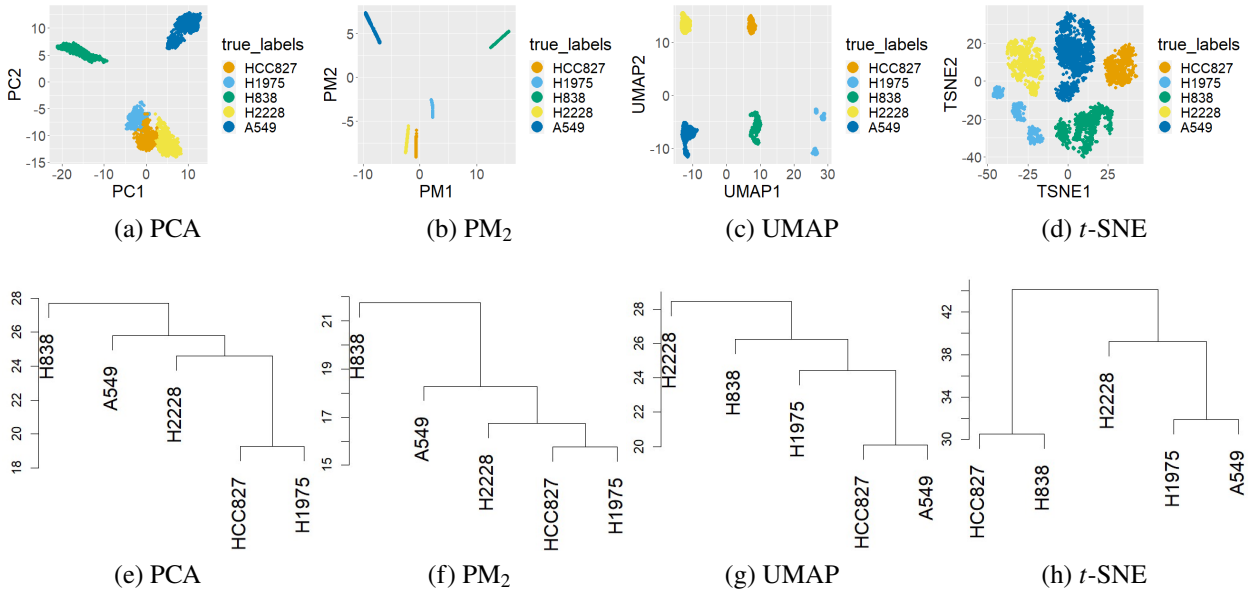
Table 3.3 reports the ARI achieved by Algorithm 3.1 and other methods; see Tables B.6 and B.5 in Appendix B for ECP and ECA. The path metric methods perform equally well or better than the rest of the methods. Once again PM_2 exhibits the best overall performance, with a high ARI ($\geq 90\%$) on all data sets except TM lung and PBMC4K; the next best method is PM_4, which

Method	RNA1	RNA2	TMLung	Beta	TMPanc	BaronPanc	PBMC4k	CellMix
2d UMAP	0.122	0.142	0.057	0.036	0.064	0.115	0.015	0.090
rd UMAP	0.160	0.131	0.092	0.023	0.036	0.129	0.027	0.050
2d t-SNE	0.059	0.054	0.042	0.025	0.048	0.206	0.038	0.061
rd t-SNE	0.035	0.054	0.027	0.010	0.040	0.229	0.050	0.033
2d $PM_{1.5}$	**0.010**	0.013	0.046	0.003	0.076	0.067	0.028	0.098
rd $PM_{1.5}$	0.017	**0.009**	**0.006**	**0**	**0.019**	**0.006**	**0.007**	**0.007**
2d PM_2	0.040	0.040	0.085	0.002	0.150	0.103	0.050	0.101
rd PM_2	0.048	0.036	0.029	0.002	0.051	0.010	0.013	0.008
2d PM_4	0.108	0.135	0.246	0.007	0.265	0.193	0.069	0.107
rd PM_4	0.100	0.082	0.083	0.007	0.099	0.027	0.029	0.008

Table 3.4 Geometric perturbation for RNA data. For rd UMAP $r = 7, 6, 5, 3, 5, 9, 3, 4$ for the various data sets, which was the maximum of the $PM_{1.5}$ dimension and the PM_2 dimension. For rd t-SNE $r = 3$.

achieves a high ARI on all but 3 data sets. SC3, RaceId3, and SIMLR had a low ARI ($< 90\%$) on 6 of the 8 data sets; scanpy, Seurat, k-means, and t-SNE+k-means had a low ARI on 5 of the 8 data sets; Seurat_def, UMAP+DBSCAN, and $PM_{1.5}$ had a low ARI for 4 of 8 data sets. These results indicate that incorporating both density-based and geometric information when determining similarity generally leads to more robust results for scRNA-seq data. Moreover, PM_2 achieves the best median ECP and median ECA values across all RNA data sets. Although the optimal balance depends on the data set (for example PBMC4K does best with $p = 4$, while TMLung does best with $p = 1.5$), path metrics with a moderate p exhibit the best performance across a wide range of data sets.

For BaronPanc we observe that Seurat_def achieves a slightly higher ARI than all the reported path metric methods ($p = 1.5, 2, 4$). However, a significant advantage of Algorithm 3.1 over Seurat is the high clustering performance on a wide range of sample sizes. To demonstrate our claim we compare the ARI results in different down-sampled versions of BaronsPanc. We selected a stratified sample of 50%, 25% and 10% of the cells of the BaronPanc data set. The results can be found in Table B.4 of Appendix B. We observed no ARI deterioration for Algorithm 3.1 for the 50% and 25% down-sampled data set and only a moderate decrease for the 10% down-sampled dataset (ARI of 0.67 at 10% downsampling for $p = 1.5$). On the contrary, there is significant ARI deterioration both for Seurat and Seurat_def; in particular, at 10% downsampling the ARI deteriorates to 0.405



(a) PCA (b) PM_2 (c) UMAP (d) t-SNE

(e) PCA (f) PM_2 (g) UMAP (h) t-SNE

Figure 3.4 Top: embeddings colored by true cell type. Bottom: average linkage dendrograms of cluster means.

for Seurat and to 0.185 for Seurat_def. Notice that in the 10% down-sampled data set, we use regular k-means for PM_2 to allow for the prediction of smaller sized clusters.

We also investigated whether we could learn the ground truth number of clusters by optimizing the silhouette criterion in the scPMP embedding, and compared this with the number of clusters obtained from Seurat using the default resolution; see Table B.3 in Appendix B. For 4 out of the 8 RNA data sets evaluated in this article (RNAMix1, RNAMix2, BaronPanc, and CellMix), this procedure on PM_2 yielded an estimate for k which matched the number of distinct annotated labels. On the other hand, Seurat correctly estimates the number of clusters for only 2 out of the 8 RNA data sets (RNAMix1 and TMLung).

Table 3.4 reports the geometric perturbation. We see that increasing p increases the geometric perturbation, with $PM_{1.5}$ yielding the smallest geometric perturbation on all data sets. Although $PM_{1.5}$ is the clear winner in terms of this metric, PM_2 still performed favorably with respect to UMAP and t-SNE. Indeed, rd PM_2 had lower geometric perturbation than UMAP on all but one data set (TMPanc), and lower geometric perturbation than t-SNE on the majority of data sets. Figure 3.4 shows the PCA, PM_2, UMAP, and t-SNE embeddings of the Cell Mix data set, as well

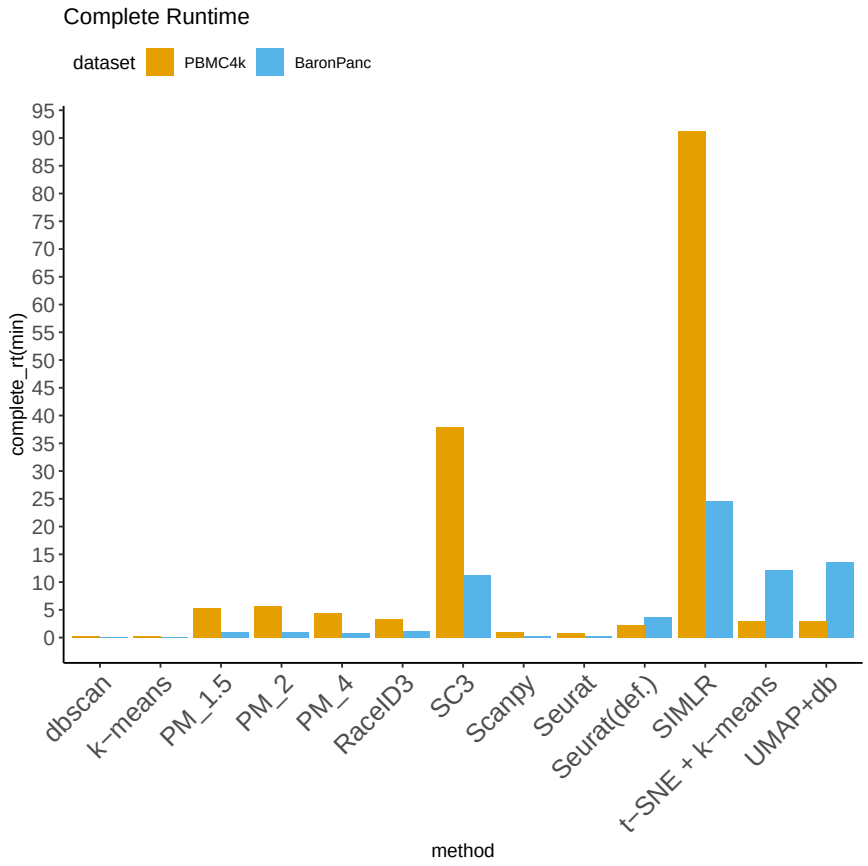


Figure 3.5 Processing and clustering time for PBMC4K and Baron's Pancreatic data sets.

as a tree structure on the clusters. The tree structure was obtained by first computing the cluster means in the embedding and then applying hierarchical clustering with average linkage to the means. The PCA tree (Figure 3.4(e)) was computed using 40 PCs so that it accurately reflects the global geometry of the clusters. Interestingly path metrics recover the same hierarchical structure on the clusters as PCA: the cell types HCC827 and H1975 are the most similar, and H838 is the most distinct. This is what one would expect given more extensive biological information about the cell types, since H838 is the only cell line here derived from metastatic site Lymph node on a male patient, while both HCC827 and H1975 originated from the primary site of female lung cancer patients. However, neither UMAP or t-SNE give the correct hierarchical representation of the clusters, because both methods struggle to preserve global geometric structure as observed in numerous studies Kobak and Berens (2019); Cooley et al. (2020).

Furthermore, Figure 3.5 records the runtime for processing and clustering (in minutes) of the

Baron's Pancreatic ($n = 1844$) and PBMC4K ($n = 4316$) data sets. For PBMC4k (our largest data set), we use the landmark-based approximation of path distances for scalability. All the PM methods run in less than a minute on BaronPanc and less than 6 minutes on PBMC4k; RaceID3, scanpy, and Seurat were also fast. SC3 and SIMLR had long runtimes, requiring 37.9 and 91.1 minutes respectively for PBMC4k.

3.4 Discussion

This article applies a new theoretical framework to the analysis of single cell RNA-seq data which is based on the computation of optimal paths. Path metrics encode both geometric and density-based information, and the resulting low-dimensional embeddings simultaneously preserve density-based cluster structure as well as global cluster orientation. The method exhibits competitive performance when applied to numerous benchmarks, and the implementation is scalable to large data sets. Although we investigated other choices of p, we found that $p = 2$ performed well on a wide range of RNA data sets, indicating that $p = 2$ is an appropriate balance between density and geometry for this application. Future research will explore ways to make the method more robust to noise and adapting the method to the semi-supervised context.

BIBLIOGRAPHY

A, Z., Muñoz-Manchado, A. B., Codeluppi, S., P, L., G, L. M., A, J., S, M., H, M., L, H., C, B., C, R., G, C.-B., J, H.-L., and and, L. S. (2015). Brain structure. Cell types in the mouse cortex and hippocampus revealed by single-cell RNA-seq. *Science (New York, N.Y.)*, 347:1138–1142.

Baron, M. et al. (2016). A single-cell transcriptomic map of the human and mouse pancreas reveals inter- and intra-cell population structure. *Cell Systems*, 3(4):346–360.

Bijral, A., Ratliff, N., and Srebro, N. (2011). Semi-supervised learning with density based distances. In *UAI*, pages 43–50.

Borg, I. and Groenen, P. J. (2005). *Modern multidimensional scaling: Theory and applications*. Springer Science & Business Media.

Borghini, E., Fernández, X., Groisman, P., and Mindlin, G. (2020). Intrinsic persistent homology via density-based metric learning. *arXiv preprint arXiv:2012.07621*.

Bousquet, O., Chapelle, O., and Hein, M. (2004). Measure based regularization. In *NIPS*, pages 1221–1228.

Camerini, P. (1978). The min-max spanning tree problem and some extensions. *Information Processing Letters*, 1(10-14).

Chang, H. and Yeung, D.-Y. (2008). Robust path-based spectral clustering. *Pattern Recognition*, 41(1):191–203.

Choudhary, S. and Satija, R. (2022). Comparison and evaluation of statistical error models for scrna-seq. *Genome Biology*, 23.

Chu, T., Miller, G., and Sheehy, D. (2017). Exploration of a graph-based density sensitive metric. *arXiv preprint arXiv:1709.07797*.

Chu, T., Miller, G., and Sheehy, D. (2020). Exact computation of a manifold metric, via Lipschitz embeddings and shortest paths on a graph. In *SODA*, pages 411–425.

Civril, A., Magdon-Ismail, M., and Bocek-Rivele, E. (2006). Ssde: Fast graph drawing using sampled spectral distance embedding. In *International Symposium on Graph Drawing*, pages 30–41. Springer.

CLevine, J., Simonds, E., Bendall, S., Davis, K., Amir, E.-a., Tadmor, M., Litvin, O., Fienberg, H., Jager, A., Zunder, E., Finck, R., Gedman, A., Radtke, I., Downing, J., Pe'er, D., and Nolan, G. (2015). Data-driven phenotypic dissection of aml reveals progenitor-like cells that correlate with prognosis. *Cell*.

Cooley, S. M., Hamilton, T., Ray, J. C. J., and Deeds, E. J. (2020). A novel metric reveals previously unrecognized distortion in dimensionality reduction of scrna-seq data. *bioRxiv*.

Ding, J., Wen, H., Tang, W., Liu, R., Li, Z., Venegas, J., Su, R., Molho, D., Jin, W., Zuo, W., et al. (2022). Dance: A deep learning library and benchmark for single-cell analysis. *bioRxiv*, pages 2022–10.

Eberwine, J., Yeh, H., Miyashiro, K., Cao, Y., Nair, S., Finnell, R., Zettel, M., and Coleman, P. (1992). Analysis of gene expression in single live neurons. *Proceedings of the National Academy of Sciences*, 89(7):3010–3014.

Ester, M., Kriegel, H.-P., Sander, J., Xu, X., et al. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In *Kdd*, volume 96, pages 226–231.

Fernández, X., Borghini, E., Mindlin, G., and Groisman, P. (2023). Intrinsic persistent homology via density-based metric learning. *Journal of Machine Learning Research*, 24(75):1–42.

Fischer, B., Zöller, T., and Buhmann, J. M. (2001). Path based pairwise data clustering with application to texture segmentation. In *International Workshop on Energy Minimization Methods in Computer Vision and Pattern Recognition*, pages 235–250. Springer.

Gabow, H. and Tarjan, R. (1988). Algorithms for two bottleneck optimization problems. *Journal of Algorithms*, 9:411–417.

García Trillos, N., Sanz-Alonso, D., and Yang, R. (2019). Local regularization of noisy point clouds: Improved global geometric estimates and data analysis. *Journal of Machine Learning Research*, 20(136):1–37.

Ghojogh, B., Ghodsi, A., Karray, F., and Crowley, M. (2020). Multidimensional scaling, sammon mapping, and isomap: Tutorial and survey. *arXiv preprint arXiv:2009.08136*.

Groisman, P., Jonckheere, M., and Sapienza, F. (2018). Nonhomogeneous Euclidean first-passage percolation and distance learning. *arXiv preprint arXiv:1810.09398*.

Groisman, P., Jonckheere, M., and Sapienza, F. (2022). Nonhomogeneous euclidean first-passage percolation and distance learning. *Bernoulli*, 28(1):255–276.

Grün, D. et al. (2018). Revealing dynamics of gene expression variability in cell state space. *Nature methods*, 17:45–49.

Hafemeister, C. and Satija, R. (2019). Normalization and variance stabilization of single-cell rna-seq data using regularized negative binomial regression. *Genome Biology*, 20(1).

Herman, J. S., Grün, D., et al. (2018). Fateid infers cell fate bias in multipotent progenitors from single-cell rna-seq data. *Nature methods*, 15(5):379.

Hu, T. (1961). Letter to the editor: The maximum capacity route problem. *Operations Research*, 9(6):898–900.

Huang, M., Wang, J., Torre, E., Dueck, H., Shaffer, S., Bonasio, R., Murray, J. I., Raj, A., Li, M., and Zhang, N. R. (2018). Saver: gene expression recovery for single-cell rna sequencing. *Nature methods*, 15(7):539–542.

Hwang, S., Damelin, S., and Hero, A. (2016). Shortest path through random points. *The Annals of Applied Probability*, 26(5):2791–2823.

Kaufman, L. and Rousseeuw, P. (2009). *Finding Groups in Data: An Introduction to Cluster Analysis*.

Kiselev, V. Y., Kirschner, K., Schaub, M. T., Andrews, T., Yiu, A., Chandra, T., Natarajan, K. N., Reik, W., Barahona, M., Green, A. R., and Hemberg, M. (2017). SC3: consensus clustering of single-cell RNA-seq data. *Nature Methods*, 14:483–486.

Kobak, D. and Berens, P. (2019). The art of using t-sne for single-cell transcriptomics. *Nature Communications*, 10:2041–1723.

Lam, C. and Yao, Q. (2012). Factor modeling for high-dimensional time series: inference for the number of factors. *The Annals of Statistics*, pages 694–726.

Lee, J. M. (2018). *Introduction to Riemannian manifolds*. Springer.

Li, W. V. and Li, J. J. (2018). An accurate and robust imputation method scimpute for single-cell rna-seq data. *Nature communications*, 9(1):1–9.

Lin, P., Troup, M., and Ho, J. W. K. (2017). CIDR: Ultrafast and accurate clustering through imputation for single-cell RNA-seq data. *Genome Biology*, 18.

Little, A., Maggioni, M., and Murphy, J. (2020a). Path-based spectral clustering: Guarantees, robustness to outliers, and fast algorithms. *Journal of Machine Learning Research*, 21(6):1–66.

Little, A., McKenzie, D., and Murphy, J. (2020b). Balancing geometry and density: Path distances on high-dimensional data. *arXiv preprint arXiv:2012.09385*.

Lopez, R., Regier, J., Cole, M. B., Jordan, M. I., and Yosef, N. (2018). Deep generative modeling for single-cell transcriptomics. *Nature methods*, 15(12):1053–1058.

Maechler, M., Rousseeuw, P., Struyf, A., Hubert, M., and Hornik, K. (2021). *cluster: Cluster Analysis Basics and Extensions*.

McInnes, L., Healy, J., and Melville, J. (2018). Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*.

Mckenzie, D. and Damelin, S. (2019). Power weighted shortest paths for clustering Euclidean data. *Foundations of Data Science*, 1(3):307.

Moon, K. R., van Dijk, D., Wang, Z., Gigante, S., Burkhardt, D. B., Chen, W. S., Yim, K., Elzen, A. v. d., Hirn, M. J., Coifman, R. R., Ivanova, N. B., Wolf, G., and Krishnaswamy, S. (2019). Visualizing structure and transitions in high-dimensional biological data. *Nature Biotechnology*, 37(12):1482–1492.

Moscovich, A., Jaffe, A., and B.Nadler (2017). Minimax-optimal semi-supervised regression on unknown manifolds. In *AISTATS*, pages 933–942.

Platt, J. (2005). Fastmap, metricmap, and landmark mds are all nyström algorithms. In *International Workshop on Artificial Intelligence and Statistics*, pages 261–268. PMLR.

Pollack, M. (1960). Letter to the editor: The maximum capacity through a network. *Operations Research*, 8(5):733–736.

Sajama and Orlitsky, A. (2005). Estimating and computing density based distance metrics. In *ICML*, pages 760–767.

Saliba, A.-E., Westermann, A. J., Gorski, S. A., and Vogel, J. (2014). Single-cell RNA-seq: advances and future challenges. *Nucleic Acids Research*, 42(14):8845–8860.

Shamai, G., Zibulevsky, M., and Kimmel, R. (2020). Efficient inter-geodesic distance computation and fast classical scaling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(1):74–85.

Stuart, T., Butler, A., Hoffman, P., Hafemeister, C., Papalexi, E., III, W. M. M., Hao, Y., Stoeckius, M., Smibert, P., and Satija, R. (2019). Comprehensive integration of single-cell data. *Cell*, 177:1888–1902.

Tabula Muris Consortium, Overall coordination, L. c. e. a. (2018). Single-cell transcriptomics of 20 mouse organs creates a tabula muris. *Nature*, 562:367–372.

Tang, F., Barbacioru, C., Wang, Y., Nordman, E., Lee, C., Xu, N., Wang, X., Bodeau, J., Tuch, B. B., Siddiqui, A., et al. (2009). mrna-seq whole-transcriptome analysis of a single cell. *Nature methods*, 6(5):377–382.

Tang, J., Liu, J., Zhang, M., and Mei, Q. (2016). Visualizing large-scale and high-dimensional data. In *Proceedings of the 25th international conference on world wide web*, pages 287–297.

Tenenbaum, J., Silva, V. D., and Langford, J. (2000). A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323.

Tian, L., Dong, X., Freytag, S., Lê Cao, K.-A., Su, S., JalalAbadi, A., Amann-Zalcenstein, D.,

Weber, T. S., Seidi, A., Jabbari, J. S., et al. (2019a). Benchmarking single cell rna-sequencing analysis pipelines using mixture control experiments. *Nature methods*, 16(6):479–487.

Tian, T., Wan, J., Song, Q., and Wei, Z. (2019b). Clustering single-cell rna-seq data with a model-based deep learning approach. *Nature Machine Intelligence*, 1(4):191–198.

Van der Maaten, L. and Hinton, G. (2008). Visualizing data using t-sne. *Journal of machine learning research*, 9(11).

Vincent, P. and Bengio, Y. (2003). Density-sensitive metrics and kernels. In *Snowbird Learning Workshop*.

Von Luxburg, U. (2007). A tutorial on spectral clustering. *Statistics and computing*, 17(4):395–416.

Wang, B., Zhu, J., Pierson, E., Ramazzotti, D., and Batzoglou, S. (2017). Visualization and analysis of single-cell RNA-seq data by kernel-based similarity learning. *Nature Methods*, 14:414–416.

Wang, J., Ma, A., Chang, Y., Gong, J., Jiang, Y., Qi, R., Wang, C., Fu, H., Ma, Q., and Xu, D. (2021). scgnn is a novel graph neural network framework for single-cell rna-seq analyses. *Nature communications*, 12(1):1882.

Williams, C. and Seeger, M. (2001). Using the nyström method to speed up kernel machines. In *Proceedings of the 14th annual conference on neural information processing systems*, number CONF, pages 682–688.

Wolf, F. A., Angerer, P., and Theis, F. J. (2018). SCANPY: large-scale single-cell gene expression data analysis. *Genome Biology*, 19.

Xu, C. and Su, Z. (2015). Identification of cell types from single-cell transcriptomes using a novel clustering method. *Bioinformatics*, 31(12):1974–1980.

Xu, X., Ester, M., Kriegel, H.-P., and Sander, J. (1998). A distribution-based clustering algorithm for mining in large spatial databases. In *Proceedings 14th International Conference on Data Engineering*, pages 324–331. IEEE.

Yip, S. H., Wang, P., Kocher, J.-P. A., Sham, P. C., and Wang, J. (2017). Linnorm: improved statistical analysis for single cell RNA-seq expression data. *Nucleic Acids Research*, 45(22):e179–e179.

Yu, H., Zhao, X., Zhang, X., and Yang, Y. (2012). Isomap using nyström method with incremental sampling. *Advances in Information Sciences & Service Sciences*, 4(12).

Zhang, S. and Murphy, J. (2021). Hyperspectral image clustering with spatially-regularized ultrametrics. *Remote Sensing*, 13(5):955.

Zhu, X., Zhang, J., Xu, Y., Wang, J., Peng, X., and Li, H.-D. (2020). Single-cell clustering based on shared nearest neighbor and graph partitioning. *Interdisciplinary Sciences: Computational Life Sciences*, 12:117–130.

žurauskienė, J. and Yau, C. (2016). pcaReduce: hierarchical clustering of single cell transcriptional profiles. *BMC Bioinformatics*, 17.

APPENDIX A

DATA PREPROCESSING

In this section the pre-processing of all RNA data sets is described. The main preprocessing steps are quality control, imputation with SAVER Huang et al. (2018), and normalization. Below we provide information about quality control and imputation and then we describe how we used those steps according to the guidelines of each method.

A.1 Data availability

The raw data of Cellmix and RNAmix are downloaded from GEO under accession code GSE118767, and the preprocessed data are available at their github repository. The PBMC4K data is available at 10x Genomics's website. The Baron's pancreatic data is available in GEO with the access code GSM2230757. The mouse tissue scRNAseq data sets are accessible on Figshare.

A.2 Main steps

Quality Control: Quality control is applied on RNAmix1, RNAmix2, Cellmix, BaronPanc, PMC4K, Beta. Specifically, cells where at most 200 genes are expressed are filtered out. Also, only genes that are expressed in more than 3 cells are included in the data set. In addition, cells with percentage of expressed mitochondrial genes greater than 20% are excluded. The data sets TMpanc and TMLung as found in Figshare have passed a quality control check with cutoffs of at least 500 genes and 50,000 reads, so no additional filtering was applied.

Imputation: Imputation with SAVER Huang et al. (2018) was applied to all RNA seq data sets apart from Cellmix. After removing multiplets the Cellmix data set included high quality data and every clustering method achieved high ARI, suggesting no need for further processing and imputation.

A.3 Preprocessing per method

Path metrics (PM), k-means, DBSCAN: After quality control and imputation, we normalize the data. RNAmix1, RNAmix2, TMLung, Beta, TMPanc, PBMC4K were row normalized and log transformed (data matrix had cells in rows and genes in columns). We then restrict to the top

2000 high variance genes. For the BaronPanc and CellMix, which have large sample size, SCT transformation was applied instead and the top 3000 variable genes were kept Hafemeister and Satija (2019); Choudhary and Satija (2022). When needed, we rescale genes where variances were extremely high. As a next step we apply PCA for dimension reduction, keeping the top 40 PC's. Finally, denoising is applied by replacing each point with the mean of its local neighborhood, using a neighborhood size of $K = 12$ points. For very large data sets, one may want to use a larger K.

UMAP+DBSCAN, t-SNE+k-means: After quality control and imputation, we apply Linnnorm Yip et al. (2017) to all data sets. Then, we restrict to the top 2000 high variance genes. When needed, we rescale genes with extremely high variance. Finally, we apply PCA for dimension reduction, keeping the top 40 PC's.

Seurat: For this method, we process the data as for PM and then use Seurat'sStuart et al. (2019) functions to find neighboring points and cluster them. Notice that here we adjust the parameter 'res', to retrieve the correct number of clusters.

Seurat_def: We follow the suggested processing and clustering workflow of Seurat Stuart et al. (2019) for all data sets. Notice that we normalize BaronPanc and CellMix with the SCT method Hafemeister and Satija (2019); Choudhary and Satija (2022). Then data sets are clustered with adjusted resolution parameter, to retrieve the correct number of clusters.

SC3: After quality control and imputation we normalize the information of every cell and multiply by 10000. Then we use the log of the data for clustering with SC3 Kiselev et al. (2017). Exception to this are the BaronPanc and CellMix data set, for which we use SCT normalization.

scanpy: After quality control and imputation we use the lognormalization of scanpy Wolf et al. (2018). Exception to this are the BaronPanc and CellMix data set, for which we use SCT normalization.

RaceID3: We apply quality control on the cells of the counts of the data set. RaceID3 Herman et al. (2018); Grün et al. (2018) applies filtering and normalization in one step, which we adjust to have about the same amount of cells and genes as with other methods. Notice that we do not apply imputation because imputed data would not be counts, which are the required input of RaceID3.

SIMLR: For SIMLR Wang et al. (2017) After quality control and imputation we normalize the information of every cell and multiply by 10000 and use the log of those data. Exception to this are the BaronPanc and CellMix data set, for which we use SCT normalization.

APPENDIX B

ADDITIONAL CLUSTERING RESULTS

Here we present more clustering evaluation results based on Entropy of Cluster Accuracy (ECA) and Entropy of cluster Purity (ECP). The ECA can quantify the variety of true labels within a predicted cluster and ECP can quantify the variety of predicted cluster labels within a true group.

Definition 2

Let N represent the number of true groups and M the number of predicted clusters. Let N_j be the number of true groups with data points within the j^{th} predicted cluster and similarly let M_j be the number of predicted clusters with data points within the j^{th} true group. Finally let $p(x_j)$ denote the proportion of data points belonging to the j^{th} true group that are within a given j^{th} predicted cluster and let $p_i(y_j)$ denote the proportion of data points of j^{th} predicted cluster that are within a given i^{th} true group. Then:

$$ECA = -\frac{1}{M}\sum_{i=1}^{M}\sum_{j=1}^{N_i} p_i(x_j)log(p(x_j))$$

$$ECP = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{M_i} p_i(y_j)log(p(y_j))$$

For a given clustering, low ECA means that data points in a predicted cluster originate from the same true group. On the other hand, low ECP indicates that almost all the data points in a true group were assigned the same clustering label. Use of ECP and ECA in clustering of scRNAseq data was also found in Tian et al. (2019a).

Method	Balls	EWB	Swiss	SO(3)
k-means	**0.082**	1.050	0.588	1.084
DBSCAN	0.385	0.114	**0**	**0**
UMAP+DBSCAN	0.941	0.695	**0**	**0**
t-SNE+ k-means	0.153	0.630	**0**	0.440
Seurat	0.255	0.193	**0**	**0**
$PM_{1.5}$	0.123	0.447	**0**	0.460
PM_2	0.142	**0.020**	**0**	**0**
PM_4	0.253	0.268	**0**	**0**

Table B.1 ECP for manifold data.

Method	Balls	EWB	Swiss	SO(3)
k-means	**0.082**	1.096	0.633	1.089
DBSCAN	0.362	0.231	**0**	**0**
UMAP+DBSCAN	0.200	**0.014**	**0**	**0**
t-SNE+ k-means	0.147	0.582	**0**	0.440
Seurat	0.250	0.183	**0**	**0**
$PM_{1.5}$	0.120	0.461	**0**	0.462
PM_2	0.138	0.020	**0**	**0**
PM_4	0.248	0.291	**0**	**0**

Table B.2 ECA for manifold data.

Method	RNA1	RNA2	TMLung	Beta	TMPanc	BaronPanc	PBMC4k	CellMix
Seurat_res=0.8	**7**	8	**7**	7	11	12	13	14
$PM_{1.5}$	12	11	8	4	15	9	**4**	**5**
PM_2	**7**	**7**	9	4	5	**8**	5	**5**
PM_4	8	8	16	4	5	7	**4**	**5**
True k	7	7	7	3	7	8	4	5

Table B.3 Predicted number of clusters for Seurat and Path metrics for RNA data (k is the true number of clusters).

Dataset	Seurat	Seurat_def	$PM_{1.5}$	PM_2	PM_4
100% of Baron's Pancreatic	0.941	0.971	0.804	0.969	0.853
50% of Baron's Pancreatic	0.880	0.844	0.969	0.969	0.969
25% of Baron's Pancreatic	0.973	0.705	0.973	0.973	0.973
10% of Baron's Pancreatic	0.410	0.185	0.674	0.939*	0.804

Table B.4 Downsampling results.

Method	RNA1	RNA2	TMLung	Beta	TMPanc	BaronPanc	PBMC4k	CellMix
SC3	0.289	0.114	0.228	0.058	0.132	0.368	0.328	0
Scanpy	0.481	0.242	0.314	0.147	0.309	0.093	**0.054**	0
RaceID3	0.336	0.621	**0.168**	0.342	**0.122**	0.181	0.207	0.000
SIMLR	0.163	0.294	0.263	0.057	0.407	0.104	0.190	0
Seurat	0.319	0.230	0.193	0.146	0.290	0.097	0.265	0
Seurat_def	0.256	0.270	0.423	0.128	0.289	0.112	0.106	0
k-means	0.147	0.268	0.221	0.080	0.194	0.164	0.193	0
DBSCAN	0.090	0.188	0.368	0.440	0.202	0.146	0.262	0
UMAP+db	**0.078**	0.151	0.449	**0.019**	0.124	**0.076**	0.163	0
t-SNE+ k-means	0.104	0.137	0.426	0.085	0.259	0.171	0.187	0.126
$PM_{1.5}$	0.110	0.146	0.180	0.061	0.305	0.147	0.196	0
PM_2	0.110	**0.071**	0.362	0.061	0.279	0.077	0.195	**0**
PM_4	0.110	0.123	0.230	0.048	0.156	0.159	0.096	0

Table B.5 ECA for RNA data.

Method	RNA1	RNA2	TMLung	Beta	TMPanc	BaronPanc	PBMC4k	CellMix
SC3	0.328	0.114	0.294	0.058	0.070	0.301	0.062	0
Scanpy	0.517	0.183	0.322	0.146	0.516	0.088	0.057	0
RaceID3	0.381	0.665	**0.182**	0.351	0.268	0.413	0.310	0
SIMLR	0.151	0.267	0.275	0.057	0.543	0.380	0.360	0
Seurat	0.292	0.282	0.230	0.150	0.540	0.122	0.053	0.027
Seurat_def	0.320	0.258	0.436	0.133	0.284	0.089	0.062	0
k-means	0.131	0.255	0.244	0.079	0.215	0.395	0.316	0
DBSCAN	**0.075**	0.141	0.404	0.135	0.138	0.109	**0.051**	0
UMAP+db	0.151	0.226	0.413	0.126	**0.061**	0.248	0.087	0
t-SNE+ k-means	0.102	0.133	0.437	0.089	0.494	0.402	0.451	0.147
$PM_{1.5}$	0.096	0.136	0.197	0.061	0.482	0.273	0.312	0
PM_2	0.096	**0.062**	0.323	0.061	0.158	**0.081**	0.308	**0**
PM_4	0.096	0.114	0.184	**0.048**	0.260	0.226	0.055	0

Table B.6 ECP for RNA data.

CHAPTER 4

SHARED NEAREST NEIGHBORS GRAPH BASED SPECTRAL CLUSTERING

4.1 Introduction

The exploration of the theoretical properties of spectral clustering on finite sample data started more than twenty years ago (Spielman and Teng, 1996; Guattery and Miller, 1998; Ng et al., 2001; Meilă and Shi, 2001) along with theoretical properties for increasing sample size (Luxburg et al., 2004). One of the advantages of spectral clustering is the interpretability of its performance on data points represented as vertices on graphs (kNN, ϵ-neighbor graph) that are connected with edges based on their similarity to other data points (von Luxburg, 2007). Although theoretical results of kNN graph-based clustering methods have been explored by Maier et al. (2009), the theoretical properties of SNN graph-based clustering combined with spectral clustering haven't been investigated yet. In the following sections, we use a similar framework as of Maier et al. (2009) to provide a range for the number of neighbors used for the construction of the SNN graph, such that exact cluster identification is achieved.

4.2 Framework

Our aim is to cluster a set of n points, $X_1, \ldots, X_n$, which have been drawn from some underlying density, p, of $\mathbb{R}^m$. For this task, we build the SNN graph of those points and use its Laplacian for spectral clustering. The number of true clusters is known and denoted as K.

4.2.1 The SNN graph

For the construction of the SNN graph, we first find the k nearest neighbors of each point X_i. Let $k\text{NN}(X_i)$ be the set of the first k nearest neighbors of X_i. Then, we connect two vertices X_i and X_j, if $X_i \in kNN(X_j)$ or if $X_j \in kNN(X_i)$. The weight, $W_{i,j}$, of their edge is the Jaccard similarity of $k\text{NN}(X_i)$, $k\text{NN}(X_j)$, i.e.

$$W_{i,j} = \frac{|kNN(X_i) \cap kNN(X_j)|}{|kNN(X_i) \cup kNN(X_j)|}. \tag{4.1}$$

The SNN graph is a symmetric graph denote as $G_{SNN}(k)$.

4.2.2 The SNN Spectral Clustering Algorithm

Algorithm 4.1 SNN Spectral Clustering Algorithm.

Input: $X \in \mathbb{R}^{n \times m}$, number of clusters K, nearest neighbors k, bandwidth h, density bound t, Denoise = (TRUE, FALSE), Laplacian = $(D - W, D^{-\frac{1}{2}} W D^{-\frac{1}{2}}, I - D^{-1}W)$, t, δ, ϵ_n

Output: Predicted clustering labels $\ell \in \mathbb{R}^n$

% k-nearest neighbors:

for i=1 to n **do**

    $kNN_i \leftarrow \{X_j : X_j \text{ is one of the } kNN \text{ of } X_i, \text{ based on Euclidean distance}\}$

% Shared Nearest Neighbors graph:

$W_{i,j} \leftarrow \frac{|kNN_i \cap kNN_j|}{|kNN_i \cup kNN_j|}$

$G_{SNN}(k) \leftarrow$ graph with adjacency matrix W

% Kernel estimation of density p:

for i=1 to n **do**

    $\hat{p}_n(X_i) \leftarrow \frac{1}{nh} \sum_{j=1}^{n} K(\frac{X_i - X_j}{h})$

% Denoising:

if Denoising = TRUE **then**

    Remove vertices and edges of X s.t.$\hat{p}_n(X) < t - 2\epsilon_n$

    Remove components with size less than δn

    $G'_{SNN}(k) \leftarrow$ denoised $G_{SNN}(k)$

    $W, n \leftarrow$ adjacency of $G'_{SNN}(k)$, number of vertices in $G'_{SNN}(k)$

% Eigendecomposition of SNN graph Laplacian:

$D_{i,i} \leftarrow \sum_{j=1}^{n} W_{i,j}$

$D \leftarrow$ diagonal$(D_{i,i})$

$L \leftarrow$ Laplacian

$V \leftarrow K$ eigenvector matrix of$L$

% Clustering of $X_1, \ldots, X_n$:

$\ell \leftarrow k$-means_plusplus(V, K)

Algorithm 4.2 Eigenvector matrix of L.

Input: L, K, Laplacian = $(D - W, D^{-\frac{1}{2}}WD^{-\frac{1}{2}}, I - D^{-1}W)$

Output: K eigenvector matrix of L

if $L = D^{-\frac{1}{2}}WD^{-\frac{1}{2}}$ **then**
    $\{\lambda_1, \lambda_2, ..., \lambda_K, ...\} \leftarrow$ eigenvalues of L such that $\lambda_1 > \lambda_2 > \lambda_3...$
    $V = [V_1, V_2, ..., V_K] \leftarrow V_i$ eigenvector of λ_i
    % Row normalization of V
    $v_i \leftarrow i$-th row of V
    $q \leftarrow (q_1, \ldots, q_n)$ where $q_i = (\sum_j v_{i,j}^2)^{1/2}$
    $\tilde{V} \leftarrow D_q^{-1}V$, where $D_q = diagonal(q)$
    $\ell \leftarrow k$-means_plusplus$(\tilde{V}, K)$
else
    $\{\lambda_1, \lambda_2, ..., \lambda_K, ...\} \leftarrow$ eigenvalues of L such that $\lambda_1 < \lambda_2 < \lambda_3...$
    $V = [V_1, V_2, ..., V_K] \leftarrow V_i$ eigenvector of λ_i

4.3 Theoretical results

The theoretical results presented in section 4.3 are divided into two cases; the noise-free case and the noisy case, based on Maier et al. (2009).

Noise-free case. In this case, we consider a probability distribution p, whose support consists of several high-density regions separated by a positive distance from each other. We consider that successful cluster identification means that each high-density region corresponds to a unique predicted cluster. Since there is no overlap between high-density regions, every point will belong to one cluster only. Hence, the denoising step of 4.1 will not remove any points from the SNN graph.

Noisy case. In this case, the high-density regions of p are connected by low-density regions. For a $t > 0$ we define the t-level set, $L(t)$, as the closure of all points $x \in \mathbb{R}^m$ with $p(x) \geq t$, i.e. $\overline{\{x : p(x) \geq t\}}$. We denote those components as $C^{(1)}, \ldots, C^{(K)}$. In the following results, we explore two approaches to the noisy case.

In the first approach, the true clusters are the sets $C^{(1)}, \ldots, C^{(K)}$. Points in low-density regions do not belong to any cluster and are removed. We consider that clusters are identified exactly by our algorithm when each connected component of $L(t)$ is included in a unique predicted cluster

$p_{\max}^{(i)}$	supremum of density attained by points of $C^{(i)}$
$p_{\min}^{(i)}$	infimum of density attained by points of $C^{(i)}$
$u^{(i)}$	lower bound on distance of cluster $C^{(i)}$ to other clusters
u^{ij}	distance between cluster $C^{(i)}$ and cluster $C^{(j)}$
$\rho(u^{(i)})$	probability of balls of radius $u^{(i)}$ in $C^{(i)}$
$\beta_{(i)}$	probability mass of cluster $C^{(i)}$
$G'_{SNN}(k)$	the SNN graph after denoising

Table 4.1 Notations.

and the ratio of noisy points to cluster points goes to zero. We also consider rough identification of clusters, when the clustering algorithm predicts components that contain points of a unique $C^{(i)}$ plus some noisy points that do not belong to any cluster. The following table includes notation used in sections . In the second approach, noisy points are not removed. Instead, they are defined as cluster points of the $L(t)$ component closest to them. For this approach, there are connections between subgraphs that correspond to true clusters and spectral clustering might mis-cluster points that are equidistant from two clusters. We provide results regarding the mis-clustering error.

Let the sets $C^{(1)}, \ldots, C^{(K)}$ be K disjoint, compact and connected subsets of $\mathbb{R}^m$. The boundary $\partial C^{(i)}$ of every $C^{(i)}$ is a smooth $(m-1)$-dimensional submanifold in $\mathbb{R}^m$. We will denote with $\kappa^{(i)}$ the minimal curvature radius of $\partial C^{(i)}$, which is equal to the inverse of the largest principal curvature of $\partial C^{(i)}$. Also we will denote with $\beta_{(i)} = \mu(C^{(i)}) = \int_{C^{(i)}} p d\lambda$, the probability mass of $C^{(i)}$, where λ is the Lebesgue measure in $\mathbb{R}^m$.

The following results about within cluster connectivity of a predicted cluster and isolation (disconnectivity) of each cluster will be proven using the collar set of each cluster. Specifically, the collar of $C^{(i)}$ is defined as $Col^{(i)}(\nu) = \{x \in C^{(i)} \mid dist(x, \partial C^{(i)}) \leq \nu\}$, with $\nu < \kappa^{(i)}$. Furthermore, we define the maximal covering radius to be $\nu_{max}^{(i)} = \max_{\nu \leq \kappa^{(i)}} \{\nu \mid C^{(i)} \setminus Col^{(i)}(\nu) \text{ is connected } \}$ and we denote with $u^{(i)}$ a lower bound of the distance of $C^{(i)}$ from $C^{(i)}$ with $j \neq i$. In the noise-free case, $u^{(i)}$ will be considered strictly greater than 0. Finally, the kNN radius of a point X_i is the maximum distance of X_i to a point in $k\text{NN}(X_i)$. The minimal kNN radius of a cluster $C^{(i)}$, $R_{min}^{(i)}$, is the minimal kNN radius of the points in $C^{(i)}$.

4.3.1 Noise-free case

Lemma 1 *(Within cluster connectedness in $G_{SNN}(k)$)*

Let $\mathcal{A}_n^{(i)}$ denote the event that the points of cluster C^i are connected in $G_{SNN}(k)$. For $z \in \Big(0, 2min\{u^{(i)}, \nu_{max}^{(i)}\}\Big)$,

$$P\Big((\mathcal{A}_n^{(i)})^c\Big) \leq n\beta_{(i)}P\Big(M \geq k\Big) + N\Big(1 - p_{\min}^{(i)}\eta_m z^m/4^m\Big)^{(n-1)}\Bigg(1 - \eta_m z^m/4^m\big(p_{\min}^{(i)} - np_{\max}^{(i)}\big)\Bigg), \quad (4.2)$$

for $M \sim Bin(n-1, p_{\max}^{(i)}\eta_m z^m)$.

Proof. Observe that if $R_{min}^{(i)} > z$, for some $z > 0$ and if for two points of $X_i, X_j \in C^{(i)}$ holds that $d(X_i, X_j) \leq z$, then we have that $X_i \in k\text{NN}(X_j)$ and $X_j \in k\text{NN}(X_i)$. Furthermore, if we can find a covering of $C^{(i)} \setminus Col^{(i)}(z/4)$ of a finite number of balls of radius $z/4$, where every ball contains at least two points of $C^{(i)}$, then points in neighboring balls have distance less than z. Hence they are in the list of k nearest neighbors of each other and every pair of points will have a shared neighbor. This implies that every point in $C^{(i)} \setminus Col^{(i)}(z/4)$ will be connected in $G_{SNN}(k)$. Notice that points in $Col^{(i)}(z/4)$ will have at most distance $3z/4$ from the balls of the covering and since every ball includes at least two points of the cluster, we conclude that the points of $Col^{(i)}(z/4)$ will be connected to $C^{(i)} \setminus Col^{(i)}(z/4)$. As a result, the points of $C^{(i)}$ will be connected in $G_{SNN}(k)$. Let $\mathcal{F}_z^{(i)}$ be the event that, given a covering of $C^{(i)} \setminus Col^{(i)}(z/4)$, there exists a ball that doesn't contain at least two points of $C^{(i)}$. Based on the above observation, $\{R_{min}^{(i)} > z\} \cap (\mathcal{F}_z^{(i)})^c$ implies the points of $C^{(i)}$ will be connected on $G_{SNN}(k)$. Therefore,

$$P\Big((\mathcal{A}_n^{(i)})^c\Big) \leq P\Big(\{R_{min}^{(i)} \leq z\}\Big) + P\Big(\mathcal{F}_z^{(i)}\Big) \quad (4.3)$$

For $\mathbf{P\Big(\{R_{min}^{(i)} \leq z\}\Big)}$: We define $N_s = |\{j \neq s | X_j \in B(X_s, z)\}|$, for $1 \leq s \leq n$. Then,

$$\text{if the event } \{R_{min}^{(i)} \leq z\} \text{ is true} \Longrightarrow$$
$$\exists X_s \in C^{(i)} \text{ s.t. } max\{d(y, X_s) \mid y \in kNN(X_s)\} \leq z \Longrightarrow$$
$$\exists X_s \in C^{(i)} \text{ s.t. } N_s \geq k.$$

Therefore, $\{R^{(i)}_{min} \leq z\} \subseteq \bigcup_{s=1}^{n}\Big\{\{N_s \geq k\} \cap \{X_s \in C^{(i)}\}\Big\}$ and then we have:

$$P\Big(\{R^{(i)}_{min} \leq z\}\Big) \leq \sum_{s=1}^{n} P\Big(N_s \geq k \mid X_s \in C^{(i)}\Big)P\Big(X_s \in C^{(i)}\Big) \leq n\beta_{(i)}P\Big(M \geq k\Big).$$

Here $N_s \mid \{X_s \in C^{(i)}\} \sim Bin(n-1, \mu(B(X_s, z))$ and $\mu(B(X_s, z)) \leq \sup_{x\in C^{(i)}} \mu(B(x, z)) \leq p^{(i)}_{\max}\eta_m z^m$. Hence, $P\Big(N_s \geq k\Big) < P\Big(M \geq k\Big)$, for $M \sim Bin(n-1, p^{(i)}_{\max}\eta_m z^m)$ and η_m the volume of the m-dimensional unit ball.

For $\mathbf{P}\Big(\mathcal{F}^{(\mathbf{i})}_{\mathbf{z}}\Big)$: Since $C^{(i)}$ is compact and connected, we can find a covering of $C^{(i)} \setminus Col^{(i)}(z/4)$ with N balls, $B_1(z/4), \ldots, B_N(z/4)$ of radius $z/4$. Let us denote, $P_{X_j,B_s} = P(X_j \in B_s(z/4) \mid X_j \in C^{(i)})P(X_j \in C^{(i)}))$ the probability that the point X_j is in $C^{(i)}$ and in the ball $B_s(z/4)$. Then,

$$\begin{aligned}
P\Big(\mathcal{F}^{(i)}_z\Big) &= P\Big(\{\exists s, 1 \leq s \leq N, \text{ s.t. } B_s(z/4) \text{ has less than two points of } C^{(i)}\}\Big)\\
&\leq \sum_{s=1}^{N} P\Big(\{B_s(z/4) \text{ has less than two points of } C^{(i)}\}\Big)\\
&= \sum_{s=1}^{N} P\Big(\{B_s(z/4) \text{ has no points of } C^{(i)}\}\Big) + \sum_{s=1}^{N} P\Big(\{B_s(z/4) \text{ has exactly one point of } C^{(i)}\}\Big)\\
&= \sum_{s=1}^{N}\prod_{j=1}^{n}\Big(1 - P_{X_j,B_s}\Big) + \sum_{s=1}^{N}\sum_{j=1}^{n} P_{X_j,B_s}\prod_{q\neq j}\Big(1 - P_{X_q,B_s}\Big)
\end{aligned} \tag{4.4}$$

Now we notice that,

$$P_{X_j,B_s} = \mu(B_s(z/4)) \leq \sup_{B_q\subset C^{(i)}} \mu(B_q(z/4)) \leq p^{(i)}_{\max}\eta_m z^m/4^m \text{ and,} \tag{4.5}$$

$$P_{X_j,B_s} = \mu(B_s(z/4)) \geq \inf_{B_q\subset C^{(i)}} \mu(B_q(z/4)) \geq p^{(i)}_{\min}\eta_m z^m/4^m \tag{4.6}$$

where $p^{(i)}_{\max}$ is the supremum of density attained by points of $C^{(i)}$ and $p^{(i)}_{\min}$ the infimum of the density attained by points of $C^{(i)}$. From inequalities 4.5, 4.6, we can write 4.4 as,

$$\begin{aligned}
P\Big(\mathcal{F}^{(i)}_z\Big) &\leq N\Big(1 - p^{(i)}_{\min}\eta_m z^m/4^m\Big)^n + nNp^{(i)}_{\max}\eta_m z^m/4^m\Big(1 - p^{(i)}_{\min}\eta_m z^m/4^m\Big)^{(n-1)}\\
&= N\Big(1 - p^{(i)}_{\min}\eta_m z^m/4^m\Big)^{(n-1)}\Big(1 - p^{(i)}_{\min}\eta_m z^m/4^m + np^{(i)}_{\max}\eta_m z^m/4^m\Big)\\
&= N\Big(1 - p^{(i)}_{\min}\eta_m z^m/4^m\Big)^{(n-1)}\Big(1 - \eta_m z^m/4^m\big(p^{(i)}_{\min} - np^{(i)}_{\max}\big)\Big)
\end{aligned}$$

For the covering we will use a standard $\frac{z}{4}-$packing. Also, since $\frac{z}{4} \leq \frac{\nu_{max}^{(i)}}{2}$, balls of radius $z/8$ around the packing centers are disjoint subsets of $C^{(i)}$. Consequently, the total volume of the N balls will be bounded by the volume of cluster $C^{(i)}$ and hence $N\eta_m \frac{z^m}{8^m} \leq Vol(C^{(i)})$. Finally we get the following bound for the number of covering balls, N,

$$N \leq \frac{Vol(C^{(i)})}{\eta_m \frac{z^m}{8^m}} \tag{4.7}$$

□

Now we will explore the connectivity of points from different clusters in $G_{SNN}(k)$. We say that $C^{(i)}$ is isolated on $G_{SNN}(k)$, if there is no edge between sample points of $C^{(i)}$ and any other cluster.

Lemma 2 *(Between clusters connectivity in $G_{SNN}(k)$)*
Let $\mathcal{I}_n^{(i)}$ be the event that $C^{(i)}$ is isolated from all other clusters in $G_{SNN}(k)$, then for $k \leq \rho(u^{(i)})n/2 - 2\log(\beta_{(i)}n)$

$$P\left(\left(\mathcal{I}_n^{(i)}\right)^c\right) \leq \sum_{i=1}^{K} e^{-\frac{n-1}{2}\left(\frac{\rho(u^{(i)})}{2} - \frac{k-1}{n-1}\right)}. \tag{4.8}$$

Proof. For the construction of the $G_{SNN}(k)$, we practically start with constructing a symmetric kNN graph, G_{kNN}, and then we remove the edges of points that do not have common neighbors in their k-nearest neighbors lists. This implies that points that aren't connected on the G_{kNN} will not be connected on the $G_{SNN}(k)$. Hence,

$$\begin{aligned}
\left(\mathcal{I}_n^{(i)}\right)^c &\implies \left\{C^{(i)} \text{ is not isolated in } G_{kNN}\right\} \\
P\left(\left(\mathcal{I}_n^{(i)}\right)^c\right) &\leq P\left(\left\{C^{(i)} \text{ not isolated in } G_{kNN}\right\}\right) \\
&\leq P\left(\left\{R_{\max}^{(i)} \geq u^{(i)}\right\}\right) + \sum_{j\neq i} P\left(\left\{R_{\max}^{(j)} \geq u^{(ij)}\right\}\right) \\
&\leq P\left(\left\{R_{\max}^{(i)} \geq u^{(i)}\right\}\right) + \sum_{j\neq i} P\left(\left\{R_{\max}^{(j)} \geq u^{(j)}\right\}\right) \\
&\leq \sum_{i=1}^{K} e^{-\frac{n-1}{2}\left(\frac{\rho(u^{(i)})}{2} - \frac{k-1}{n-1}\right)}
\end{aligned}$$

where we get the final bound for $k < \rho(u^{(i)})n/2 - 2\log(\beta^{(i)}n)$ and using Preposition 6 and Lemma 7 from Maier et al. (2009). Here, $u^{(i)} \leq u^{ij}$ and $\rho(u^{(i)})$ is the probability of balls of radius $u^{(i)}$ in $C^{(i)}$. □

Lemma 3 *(Range of k for within-cluster connectedness)*

If $k \geq 4^{m+1}\log\left(2np_{max}^{(i)}Vol(C^{(i)})8^m\right)$ *and* $k \leq (n-1)p_{max}^{(i)}\eta_m \min\{(u^{(i)})^m, (\nu_{max}^{(i)})^m\}$ *then,*

$$P\Big((\mathcal{A}_n^{(i)})^c\Big) \leq \Big(2 + \frac{1}{4^m}\frac{n^2}{n-1}\Big)e^{-\frac{(k-1)p_{min}^{(i)}}{4^{m+1}p_{max}^{(i)}}} \tag{4.9}$$

Proof. Our overall goal is to find appropriate values for k such that $P\Big((\mathcal{A}_n^{(i)})^c\Big)$ has an upper bound that goes to zero as n goes to infinity. We will find upper bounds for the two terms of the inequality 4.2.

We will use the following inequalities in the proof:

(Hoeffding's inequality). Let $M \sim Bin(n,p)$ and define $\alpha = \frac{k}{n-1}$. Then,

$$\alpha \geq p, P\Big(M \geq k\Big) \leq e^{-nK(\alpha||p)}, \tag{4.10}$$

where $K(\alpha||p) = \alpha\log\big(\frac{a}{p}\big) + (1-\alpha)\log\big(\frac{1-a}{1-p}\big)$ is the Kullback-Leibler divergence of $(\alpha, 1-\alpha)$ and $(p, 1-p)$.

(1st logarithmic inequality.)

$$\log(x) \geq \frac{x-1}{x}, \text{ for } x > 0 \tag{4.11}$$

(2nd logarithmic inequality.)

$$\log(1-x) \leq x, \text{ for } x \leq 1. \tag{4.12}$$

For the first term of 4.2, we use inequality 4.10 for $p = p_{max}^{(i)}\eta_m z^m$ and $\alpha = \frac{k}{n-1}$. Now, assuming that $p < \alpha$ and that $k \leq n-1$ we get,

$$n\beta_{(i)}P\Big(M \geq k\Big) \leq e^{-(n-1)\Big(\alpha\log\big(\frac{a}{p}\big)+(1-\alpha)\log\big(\frac{1-a}{1-p}\big)\Big)} \leq e^{-(n-1)\Big(\alpha\log\big(\frac{a}{p}\big)+p-\alpha\Big)}, \text{ by 4.11.}$$

Let $\theta = \eta_m z^m/\alpha$ then we have,

$$\begin{aligned} n\beta_{(i)}P\Big(M \geq k\Big) &\leq n\beta_{(i)}e^{-k\Big(\log\big(\frac{1}{\theta p_{max}^{(i)}}\big)+\theta p_{max}^{(i)}-1\Big)\Big)} \\ &= e^{\log(n\beta_{(i)})-k\Big(\log\big(\frac{1}{\theta p_{max}^{(i)}}\big)+\theta p_{max}^{(i)}-1\Big)\Big)} \\ &\leq e^{-\frac{k}{2}\Big(\log\big(\frac{1}{\theta p_{max}^{(i)}}\big)+\theta p_{max}^{(i)}-1\Big)\Big)}, \end{aligned} \tag{4.13}$$

for k such that $\log(n\beta_{(i)}) \leq \frac{k}{2}\big(\log\big(\frac{1}{\theta p^{(i)}_{max}}\big) + \theta p^{(i)}_{max} - 1\big)$. That way we attain a lower bound for k:

$$k \geq \frac{2\log(n\beta_{(i)})}{\log\big(\frac{1}{\theta p^{(i)}_{max}}\big) + \theta p^{(i)}_{max} - 1} \tag{4.14}$$

For the second term of 4.2 we have,

$$\begin{aligned}
&N\Big(1 - p^{(i)}_{\min}\eta_m z^m/4^m\Big)^{(n-1)}\Big(1 - \eta_m z^m/4^m\big(p^{(i)}_{\min} - np^{(i)}_{\max}\big)\Big) = \\
&\Big(1 - \eta_m z^m/4^m\big(p^{(i)}_{\min} - np^{(i)}_{\max}\big)\Big)e^{(n-1)\log(1-p^{(i)}_{\min}\eta_m z^m/4^m)+\log(N)} \leq \\
&\Big(1 - \eta_m z^m/4^m\big(p^{(i)}_{\min} - np^{(i)}_{\max}\big)\Big)e^{(n-1)\log(1-p^{(i)}_{\min}\eta_m z^m/4^m)+\log\left(\frac{Vol(C^{(i)})}{\eta_m \frac{z^m}{8^m}}\right)}, \text{ by 4.7.}
\end{aligned}$$

Now we use the substitution $\eta_m z^m = \theta\alpha = \frac{\theta k}{n-1}$

$$\begin{aligned}
&N\Big(1 - p^{(i)}_{\min}\eta_m z^m/4^m\Big)^{(n-1)}\Big(1 - \eta_m z^m/4^m\big(p^{(i)}_{\min} - np^{(i)}_{\max}\big)\Big) \leq \\
&\Big(1 + \frac{\theta k}{4^m(n-1)}\big(np^{(i)}_{\max} - p^{(i)}_{\min}\big)\Big)e^{-\frac{\theta k p^{(i)}_{\min}}{4^m}+\log\left(\frac{Vol(C^{(i)})8^m(n-1)}{\theta k}\right)} \leq \\
&\Big(1 + \frac{\theta n}{4^m(n-1)}\big(np^{(i)}_{\max} - p^{(i)}_{\min}\big)\Big)e^{-\frac{\theta k p^{(i)}_{\min}}{4^m}+\log\left(\frac{Vol(C^{(i)})8^m n}{\theta}\right)} \leq \\
&\Big(1 + \frac{\theta n}{4^m(n-1)}\big(np^{(i)}_{\max} - p^{(i)}_{\min}\big)\Big)e^{-\frac{k}{2}\frac{\theta p^{(i)}_{\min}}{4^m}},
\end{aligned} \tag{4.15}$$

where the last step of the inequality 4.15 holds if $-\frac{\theta k p^{(i)}_{\min}}{4^m} + \log\big(\frac{Vol(C^{(i)})8^m n}{\theta}\big) \leq -\frac{k}{2}\frac{\theta p^{(i)}_{\min}}{4^m}$ or equivalently if,

$$k \geq \frac{4^m 2}{\theta p^{(i)}_{min}}\log\big(\frac{Vol(C^{(i)})8^m n}{\theta}\big) \tag{4.16}$$

To bound inequality 4.13 with the upper bound of 4.15 we need,

$$\Big(1 + \frac{\theta n}{4^m(n-1)}\big(np^{(i)}_{\max} - p^{(i)}_{\min}\big)\Big)e^{-\frac{k}{2}\frac{\theta p^{(i)}_{\min}}{4^m}} \geq e^{-\frac{k}{2}\left(\log\left(\frac{1}{\theta p^{(i)}_{max}}\right)+\theta p^{(i)}_{max}-1\right)}$$

for which it suffices to have, $\frac{k}{2}\frac{\theta p^{(i)}_{\min}}{4^m} \leq \frac{k}{2}\Big(\log\big(\frac{1}{\theta p^{(i)}_{max}}\big) + \theta p^{(i)}_{max} - 1\Big)$ or equivalently $-\log(\gamma) \geq 1 + \gamma - \frac{\gamma}{4^m}\frac{p^{(i)}_{min}}{p^{(i)}_{max}}$ for $\gamma = \theta p^{(i)}_{max}$. The above is satisfied for values of γ that $-\log(\gamma) \geq 1 + \gamma - \frac{3\gamma}{4}$, since $\frac{p^{(i)}_{min}}{4^m p^{(i)}_{max}} \leq \frac{1}{4}$. Such values of γ could be $\frac{1}{10}$, $\frac{1}{2}$ and others. We will use $\gamma = \frac{1}{2}$, which will result to $\theta = \frac{1}{2p^{(i)}_{max}}$ and the probability inequality in Lemma 1 can be rewritten as:

$$P\Big((\mathcal{A}^{(i)}_n)^c\Big) \leq 2\Big(1 + \frac{n}{n-1}\frac{1}{4^m 2p^{(i)}_{max}}\big(np^{(i)}_{\max} - p^{(i)}_{\min}\big)\Big)e^{-\frac{kp^{(i)}_{min}}{4^{m+1}p^{(i)}_{max}}} \leq \Big(2 + \frac{1}{4^m}\frac{n^2}{n-1}\Big)e^{-\frac{(k-1)p^{(i)}_{min}}{4^{m+1}p^{(i)}_{max}}}$$

Now we return to the range of k. We substitute the chosen value for θ in 4.14 and 4.16 and we observe that the largest of the two lower bounds is the one in 4.16. This is because $\beta_{(i)} = \mu(C^{(i)}) \leq p_{max}^{(i)} Vol(C^{(i)})$.

Hence, we conclude that an appropriate lower bound for k is $k \geq 4^{m+1} \log\big(2np_{max}^{(i)} Vol(C^{(i)}) 8^m\big)$. From the assumption $\eta_m z^m = \theta\alpha$ and that $z \leq 2\min\{u^{(i)}, v_{max}^{(i)}\}$ we get an upper bound for k. $k < (n-1)p_{max}^{(i)} \eta_m z^m \Leftrightarrow k < (n-1)p_{max}^{(i)} \eta_m \min\{(u^{(i)})^m, (v_{max}^{(i)})^m\}$. □

The theoretical results we derived so far provide us with probabilities of having connected components in the graph $G_{SNN}(k)$, isolated from other components that each correspond to a specific cluster. Now, we want to explore the probability that the step of spectral clustering on $G_{SNN}(k)$ of algorithm 4.1 will yield exact identification of clusters.

Theorem 1 *(Optimal k for exact identification of clusters)*
For $k = c\big(1 + \frac{4\log(n) + (n-1)\rho_{min}}{2 + \frac{p_{min}}{4^m p_{max}}}\big)$, *where c such that k will satisfy* $k \geq 4^{m+1} \log\big(2np_{max}^{(i)} Vol(C^{(i)}) 8^m\big)$ *and* $k \leq \min\big\{\rho(u^{(i)})n/2 - 2\log(\beta_{(i)} n), (n-1)p_{max}^{(i)} \eta_m \min\{(u^{(i)})^m, (v_{max}^{(i)})^m\}\big\}$, *for* $i \in \{1, \ldots, K\}$, *the algorithm 4.1 achieves exact cluster identification with probability*

$$P\Big(\mathcal{Q}_n\Big) \geq 1 - K\Big(2 + \frac{1}{4^m}\frac{n^2}{n-1}\Big) e^{-\frac{(k-1)p_{min}}{4^{m+1} p_{max}}} - K^2 e^{-\frac{n-1}{2}\left(\frac{\rho_{min}}{2} - \frac{k-1}{n-1}\right)}$$

Proof. We will start our proof with some notation. Let $\mathcal{A}_n = \bigcap_{i=1}^{K} \mathcal{A}_n^{(i)}$, the event that for each cluster its points are connected on the graph. Correspondingly, let $\mathcal{I}_n = \bigcap_{i=1}^{K} \mathcal{I}_n^{(i)}$ be the event that every cluster is isolated from the other clusters on the graph $G_{SNN}(k)$. Then, we denote with $\mathcal{Q}_n$ the intersection of $\mathcal{A}_n$ and $\mathcal{I}_n$.

If $\mathcal{Q}_n$ is true, then with an appropriate permutation of rows, the adjacency matrix W of $G_{SNN}(k)$ will be block-diagonal and each block will correspond to a cluster. The first step of our proof will be, to explain how each of the Laplacian option will utilize the block structure of W to achieve exact clustering. The second step of our proof is to find an optimal choice for k and an upper bound for the probability of event $(\mathcal{Q}_n)^c$.

For $\mathbf{L} = \mathbf{D} - \mathbf{W}$ and $\mathbf{L} = \mathbf{I} - \mathbf{D}^{-1}\mathbf{W}$: According to Propositions 2 and 4 of von Luxburg (2007), the unnormalized Laplacian and the random walk Laplacian have eigenvalue zero with multiplicity

equal to the number of connected components on the graph. Furthermore, the eigenspace of eigenvalue zero is spanned by indicator vectors $\mathbb{1}_{C^{(i)}}$, for $i \in \{1, \ldots, K\}$.

If $G_{SNN}(k)$ has a connected component for all the points that belong to a unique cluster, and the components are isolated from components that correspond to different clusters, the matrix W will have a block structure and every block will include all the points of one cluster.

The matrix V with columns the eigenvectors of L will hence be of the form $V = BM$, where M is an orthogonal matrix and B is a block diagonal matrix with columns the indicator eigenvectors $\mathbb{1}_{C^{(i)}}$, for $i \in \{1, \ldots, K\}$. Now, notice that V will also have a block diagonal structure and the rows of every block will be equal. On the other hand, due to the block structure of V, columns of V of different blocks will be perpendicular to each other. Hence, kmeans on V will choose one row from each block as the centroid of a cluster, and as a result points of the same block will be clustered together.

For $\mathbf{L = D^{-\frac{1}{2}} W D^{-\frac{1}{2}}}$: We observe that $L = D^{-\frac{1}{2}} W D^{-\frac{1}{2}}$ and $L_{sym} = I - D^{-\frac{1}{2}} W D^{-\frac{1}{2}}$ have the same eigenvectors but different eigenvalues. Specifically, if v is an eigenvector for the eigenvalue λ of L_{sym} then v is an eigenvector for the eigenvalue $-\lambda$ of L.

According to Proposition 4 of von Luxburg (2007), L_{sym} has eigenvalue zero with multiplicity equal to the number of connected components in the graph and the eigenspace of zero is spanned by the vectors $\{D^{-\frac{1}{2}} \mathbb{1}_{C^{(1)}}, \ldots, D^{-\frac{1}{2}} \mathbb{1}_{C^{(K)}}\}$.

In this case, the matrix V of eigenvectors of L will be of the form $V = D^{-\frac{1}{2}} BM$, where M is an orthogonal matrix, and B is a block diagonal matrix with columns the indicator eigenvectors $\mathbb{1}_{C^{(i)}}$, for $i \in \{1, \ldots, K\}$. This time the rows of V that correspond to points of the same component can be seen as vectors that aren't identical, but, interestingly, have the same direction and different lengths. For this reason in algorithm 4.1 we do not apply kmeans on V, but instead on $\tilde{V}$ which is equal to V after row-normalization. We observe that the rows of $\tilde{V}$ that correspond to points of the same component will be equal. The columns of $\tilde{V}$ of different blocks will be perpendicular to each other. Hence, kmeans on $\tilde{V}$ will choose one row from each block as the centroid of a cluster, and as a result points of the same block will be clustered together.

Moving on to the second step of the proof, we will calculate an upper bound for $P\Big((\mathcal{Q}_n)^c\Big)$. Let $\rho_{min} = \min_{i=1,\dots,K} \rho(u^{(i)})$, $p_{min} = \min_{i=1,\dots,K} p^{(i)}_{min}$ and $p_{max} = \max_{i=1,\dots,K} p^{(i)}_{max}$. We notice that by Lemma 3, for $k \geq 4^{m+1}\log\big(2np^{(i)}_{max}Vol(C^{(i)})8^m\big)$ and $k \leq (n-1)p^{(i)}_{max}\eta_m \min\{(u^{(i)})^m, (\nu^{(i)}_{max})^m\}$ for $i = 1,\dots,K$, we have that,

$$\begin{aligned} P\Big((\mathcal{A}_n)^c\Big) \leq \sum_{i=1}^{K} P\Big((\mathcal{A}_n^{(i)})^c\Big) &\leq \sum_{i=1}^{K}\Big(2 + \frac{1}{4^m}\frac{n^2}{n-1}\Big)e^{-\frac{(k-1)p^{(i)}_{min}}{4^{m+1}p^{(i)}_{max}}} \\ &\leq \sum_{i=1}^{K}\Big(2 + \frac{1}{4^m}\frac{n^2}{n-1}\Big)e^{-\frac{(k-1)p_{min}}{4^{m+1}p_{max}}} \\ &= K\Big(2 + \frac{1}{4^m}\frac{n^2}{n-1}\Big)e^{-\frac{(k-1)p_{min}}{4^{m+1}p_{max}}} \end{aligned} \tag{4.17}$$

If we additionally have that $k \leq \rho(u^{(i)})n/2 - 2\log(\beta_{(i)}n)$ for every i in $\{1,\dots,K\}$, then by Lemma 2

$$\begin{aligned} P\Big((\mathcal{I}_n)^c\Big) \leq \sum_{i=1}^{K} P\Big(\big(\mathcal{I}_n^{(i)}\big)^c\Big) &\leq K\sum_{i=1}^{K} e^{-\frac{n-1}{2}\left(\frac{\rho(u^{(i)})}{2} - \frac{k-1}{n-1}\right)} \\ &\leq K\sum_{i=1}^{K} e^{-\frac{n-1}{2}\left(\frac{\rho_{min}}{2} - \frac{k-1}{n-1}\right)} \\ &= K^2 e^{-\frac{n-1}{2}\left(\frac{\rho_{min}}{2} - \frac{k-1}{n-1}\right)}. \end{aligned} \tag{4.18}$$

Hence,

$$P\Big((\mathcal{Q}_n)^c\Big) \leq P\Big((\mathcal{A}_n)^c\Big) + P\Big((\mathcal{I}_n)^c\Big) \leq K\Big(2 + \frac{1}{4^m}\frac{n^2}{n-1}\Big)e^{-\frac{(k-1)p_{min}}{4^{m+1}p_{max}}} + K^2 e^{-\frac{n-1}{2}\left(\frac{\rho_{min}}{2} - \frac{k-1}{n-1}\right)}$$

Now we want to choose k such that the bounds of 4.17, 4.18 will be of the same order. Equivalently we want,

$$ne^{-\frac{(k-1)p_{min}}{4^{m+1}p_{max}}} = e^{-\frac{n-1}{2}\left(\frac{\rho_{min}}{2} - \frac{k-1}{n-1}\right)},$$

which holds for $k = 1 + \frac{4\log(n)+(n-1)\rho_{min}}{2+\frac{p_{min}}{4^m p_{max}}}$. In conclusion, choosing $k = c\big(1 + \frac{4\log(n)+(n-1)\rho_{min}}{2+\frac{p_{min}}{4^m p_{max}}}\big)$, for a constant c such that k will satisfy the conditions for inequalities 4.17, 4.18, will let $P\Big((\mathcal{Q}_n)^c\Big)$ go to zero exponentially with n. □

4.3.2 Noisy case

4.3.2.1 First approach to noisy case: Remove low-density points

We now explore the connectedness and isolation properties of clusters predicted by 4.1 when there are noise points in our sample - points that do not belong to any cluster and have low density. In this case, we apply spectral clustering on $G'_{SNN}(k)$, the graph that doesn't include any points with $\hat{p}(x) < t - 2\epsilon_n$ and their edges. The clusters of the $L(t-2\epsilon_n)$ are denoted as $C^{(i)}(2\epsilon_n)$. The value ϵ_n is the error in density estimation. We choose to work with clusters of the $L(t-2\epsilon_n)$ to ensure that the points of the $L(t)$ set will not be removed from the $G'_{SNN}(k)$. Additionally, ϵ_n has the property that $dist(C^{(i)}(2\epsilon_n), C^{(j)}(2\epsilon_n)) \geq u^{(i)}$ for every $i, j \in \{1, \ldots, K\}$. We denote with $\tilde{R}^{(i)}_{min}, \tilde{R}^{(i)}_{max}$ the minimal and maximal kNN radius of $C^{(i)}(2\epsilon_n)$ and with $\tilde{\beta}_{(i)}$ the mass of $C^{(i)}(2\epsilon_n)$, $\mu(C^{(i)}(2\epsilon_n))$. Finally, $\tilde{\rho}(u^{(i)})$ is the probability of balls of radius $u^{(i)}$ in $C^{(i)}(2\epsilon_n)$. The aim of this section is to illustrate how to extend the connectedness and isolation results for the clusters $C^{(i)}$ $i = 1, \ldots, K$ in the noisy case. In more detail, we prove that $G'_{SNN}(k)$ will have so many connected components as the number of clusters K and each component on the graph $G'_{SNN}(k)$ will correspond to a unique topological component of the $L(t-2\epsilon_n)$ set. Furthermore, the component of $G'_{SNN}(k)$ corresponding to $C^{(i)}(2\epsilon_n)$ will include all the points of $C^{(i)}$ and some additional points, but it will be isolated from all other components. Hence the adjacency matrix of $G'_{SNN}(k)$ will again be block-diagonal and spectral clustering will predict clusters that include the points of $C^{(i)}$ plus some boundary points. We further prove that as the sample size n increases, the ratio of boundary to cluster points will go to zero and allow spectral clustering to yield a grouping only of true cluster points achieving that way exact clustering. Let $\mathcal{D}^{(i)}_n$ be the event that $|\hat{p}_n(X_i) - p_n(X_i)| \leq \epsilon_n$ for every X_i, $i = 1, \ldots, n$.

Lemma 4 *(Range of k for within-cluster connectedness in noisy case 1)*
Let $\mathcal{A}^{(i)}_n$ denote the event that the points of cluster $C^{(i)}$ are connected in $G'_{SNN}(k)$. If $k \geq 4^{m+1}\log\left(2np^{(i)}_{max}Vol(C^{(i)})8^m\right)$ and $k \leq (n-1)p^{(i)}_{max}\eta_m \min\{(u^{(i)})^m, (\nu^{(i)}_{max})^m\}$ then,

$$P\left((\mathcal{A}^{(i)}_n)^c\right) \leq \left(2 + \frac{1}{4^m}\frac{n^2}{n-1}\right)e^{-\frac{(k-1)t}{4^{m+1}p^{(i)}_{max}}} + P\left(\mathcal{D}^c_n\right)$$

Proof. We assume that $\mathcal{D}_n^{(i)}$ holds, and we use the steps of the proof of Lemma 1. There are no difference in the bounds for $P\Big(\{R_{min}^{(i)} \leq z\}\Big) \leq n\beta_{(i)}P(M \geq k)$. We recall that $\mathcal{F}_z^{(i)}$ is the event that, given a covering of $C^{(i)}(2\epsilon_n) \setminus Col^{(i)}(z/4)$, there exists a ball that doesn't contain at least two points of $C^{(i)}(2\epsilon_n)$. In the noisy case, this event can happen either if some ball in the covering contains less than two points or if some points of $C^{(i)}$ were discarded. If the event $\mathcal{D}_n^{(i)}$ holds for a point x then $\hat{p}(x) > t - \epsilon_n$, then x will not be removed at the denoising step. Additionally, $p_{min} = t$ in $C^{(i)}$. Consequently,

$$\mathbf{P}\Big(\mathcal{F}_{\mathbf{z}}^{(\mathbf{i})}\Big) \leq N\Big(1 - t\eta_m z^m/4^m\Big)^{(n-1)}\left(1 - \eta_m z^m/4^m\big(t - np_{\max}^{(i)}\big)\right) + P\Big(\mathcal{D}_n^c\Big) \tag{4.19}$$

We find the final bound of $P\Big((\mathcal{A}_n^{(i)})^c\Big)$ by following the proof of Lemma 3 and using the inequality 4.19. □

Lemma 5 *(Cluster size probability)*

Let $\mathcal{B}_n^{(i)}$ denote the event that there are more than δn sample points from cluster $C^{(i)}$. If $\beta_{(i)} > \delta$ then,

$$P\big(\big(\mathcal{B}_n^{(i)}\big)^c\big) \leq e^{-\frac{1}{2}n\beta_{(i)}\left(\frac{\beta_{(i)}-\delta}{\beta_{(i)}}\right)^2}$$

Proof Same as in Maier et al. (2009) Lemma 4. □

Lemma 6 *(Density estimation error)*

Let $\mathcal{E}_n^{(i)}$ denote the event that there are less than δn points in all the boundary points sets $C^{(j)}(2\epsilon_n) \setminus C^{(j)}$ together. If $\sum_{j=1}^{K} \mu(C^{(j)}(2\epsilon_n) \setminus C^{(j)}) < \delta/2$, we have $P\big(\big(\mathcal{E}_n^{(i)}\big)^c\big) \leq e^{-\delta n/8}$

Proof Same as in Maier et al. (2009) Lemma 5. □

Proposition 1 *(Cluster connectedness in $G'_{SNN}(k)$)*

Let $C_n^{(i)}$ be the event that in the denoised graph $G'_{SNN}(k)$ it holds that:

- all sample points of $C_n^{(i)}$ are contained in the graph

-the sample points of $C_n^{(i)}$ are connected in the graph

-there is no component of the graph that consists only of points outside the $L(t)$ set.

Then under the conditions that

1. $\beta_{(i)} > 2\delta$
2. ϵ_n *sufficiently small such that* $\mu(\bigcup_{i=1}^{K} C^{(j)}(2\epsilon_n) \setminus C^{(j)}) \leq \delta/2$
3. $k \geq 4^{m+1} \log\left(2np_{max}^{(i)} Vol(C^{(i)}) 8^m\right)$ *and*
4. $k \leq (n-1)p_{max}^{(i)} \eta_m \min\{(u^{(i)})^m, (\nu_{max}^{(i)})^m\}$

and for sufficiently large n we obtain

$$P\left(\left(C_n^{(i)}\right)^c\right) \leq \left(2 + \frac{1}{4^m}\frac{n^2}{n-1}\right) e^{-\frac{(k-1)t}{4^{m+1}p_{max}^{(i)}}} + 2e^{\frac{-\delta n}{8}} + 2P\left(\mathcal{D}_n^c\right) \tag{4.20}$$

Proof. We observe that:
$P\left(\left(C_n^{(i)}\right)^c\right) \leq P\left(\left(\mathcal{A}_n^{(i)}\right)^c\right) + P\left(\left(\mathcal{B}_n^{(i)}\right)^c\right) + P\left(\left(\mathcal{E}_n^{(i)}\right)^c\right) + P\left(\left(\mathcal{D}_n^{(i)}\right)^c\right)$ and use 4,5 and 6. □

Lemma 7 *(Between clusters connectivity in* $G'_{SNN}(k)$*)*

Let $\mathcal{I}_n^{(i)}$ *be the event that* $C^{(i)}(2\epsilon_n)$ *is isolated from all other clusters in* $G'_{SNN}(k)$*, then for* $k \leq \rho(u^{(i)})n/2 - 2\log(\tilde{\beta}_{(i)}n)$

$$P\left(\left(\mathcal{I}_n^{(i)}\right)^c\right) \leq \sum_{i=1}^{K} e^{-\frac{n-1}{2}\left(\frac{\rho(u^{(i)})}{2} - \frac{k-1}{n-1}\right)} + P\left(\mathcal{D}_n^c\right). \tag{4.21}$$

Proof. We follow the proofs of Proposition 6 and Lemma 7 from Maier et al. (2009). □

Proposition 1 and Lemma 7 will now be used to find a range for k for the rough identification of clusters $C^{(i)}(2\epsilon_n)$ with spectral clustering on $G'_{SNN}(k)$. With the term rough identification, we mean that all points of each true cluster $C^{(i)}$ belong to the same predicted cluster, which may also have some additional points that do not belong to any cluster. Two important conditions for the results of exact cluster identification are the following:

Condition 1:

a) $k \geq 4^{m+1} \log\left(2np_{max}^{(i)} Vol(C^{(i)}) 8^m\right)$ and

b) $k \leq \min\left\{\rho(u^{(i)})n/2 - 2\log(\beta_{(i)}n), (n-1)p_{max}^{(i)}\eta_m \min\{(u^{(i)})^m, (\nu_{max}^{(i)})^m\}\right\}$,

Condition 2:

a) $\beta_{(i)} > 2\delta$

b) p is three times continuously differentiable with uniformly bounded derivatives

c) ϵ_n sufficiently small such that $\mu(\bigcup_{i=1}^{K} C^{(j)}(2\epsilon_n) \setminus C^{(j)}) \leq \delta/2$.

Theorem 2 *(Rough cluster identification in noisy case 1)*

If condition 2 holds, an optimal choice of k for the identification of clusters $C^{(i)}$ is $k = c\big(1 + \frac{4\log(n)+(n-1)\rho_{min}}{2+\frac{t}{4^m p_{max}}}\big)$, for a constant c such that k will satisfy condition 1 for every $i \in \{1, \ldots, K\}$. Also for a kernel density estimator $\hat{p}_n$ with bandwidth h there are constants C_1, C_2 such that if $h^2 \leq C_1\epsilon_n$:

$$P\Big(\mathcal{Q}_n\Big) \geq 1 - K\Big(2 + \frac{1}{4^m}\frac{n^2}{n-1}\Big)e^{-\frac{(k-1)t}{4^{m+1}p_{max}}} + K^2 e^{-\frac{n-1}{2}\left(\frac{\rho_{min}}{2} - \frac{k-1}{n-1}\right)} + 2e^{-n\frac{\delta}{8}} + 3e^{-C_2 n h^m \epsilon_n^2},$$

where $\mathcal{Q}_n$ is the event that the algorithm 4.1 roughly identifies all clusters $C^{(i)}$.

Proof. If $\mathcal{I}_n^{(i)}$ is true for all clusters $C^{(i)}(2\epsilon_n)$ then for every i such that $1 \leq i \leq K$, there will be no connections between the subgraph of $G'_{SNN}(k)$ containing points of cluster $C^{(i)}$ and any other cluster. Furthermore, if $\mathcal{C}_n^{(i)}$ holds, then all points that belong to $C^{(i)}$ are connected on $G'_{SNN}(k)$ and points outside $C^{(i)}$ are discarded or connected to points of $C^{(i)}$. If $\mathcal{I}_n^{(i)}$ and $\mathcal{C}_n^{(i)}$ hold for every cluster, then the adjacency matrix W of this graph will be block diagonal. Each block will correspond to a cluster $C^{(i)}(2\epsilon_n)$ and spectral clustering will roughly identify all $C^{(i)}$. To prove that, we follow the same argument regarding the different types of Laplacians as in the proof of Theorem 1.

Let $\mathcal{Q}_n$ be the event that that clusters are roughly identified by algorithm 4.1, $\mathcal{C}_n = \bigcap_{i=1}^{K} \mathcal{C}_n^{(i)}$ and $\mathcal{I}_n = \bigcap_{i=1}^{K} \mathcal{I}_n^{(i)}$. Then using Proposition 1 and Lemma 7 we obtain,

$$P\Big((\mathcal{Q}_n)^c\Big) \leq P\Big((\mathcal{C}_n)^c\Big) + P\Big((\mathcal{I}_n)^c\Big) \leq K\Big(2 + \frac{1}{4^m}\frac{n^2}{n-1}\Big)e^{-\frac{(k-1)t}{4^{m+1}p_{max}}} + K^2 e^{-\frac{n-1}{2}\left(\frac{\rho_{min}}{2} - \frac{k-1}{n-1}\right)} + 2e^{\frac{-n\delta}{8}} + 3P\Big((\mathcal{D}_n)^c\Big),$$

where $\rho_{min} = \min_{i=1,\ldots,K} \rho(u^{(i)})$ and $p_{max} = \max_{i=1,\ldots,K} p_{max}^{(i)}$.

According to Lemma 9 of Maier et al. (2009) if $p \in C^2(\mathbb{R}^m)$ with $||p||_\infty = p_{max}$ and $p'(x) \neq 0$ for x in the neighborhood of $\{p = t\}$ then for sufficiently small ϵ_n

$$\mu(\bigcup_{i=1}^{K} C^{(j)}(2\epsilon_n) \setminus C^{(j)}) \leq C \sum_{i=1}^{K} vol(\partial C^{(i)}) p_{max} \epsilon_n,$$

for some constant C. Under those conditions for p and Theorem 3.1.7 of Prakasa Rao (1983), there exist constants C_1, C_2 such that when we choose bandwidth h for the estimation of density p that satisfies $h^2 \leq C_1\epsilon_n$ we get that $P\Big((\mathcal{D}_n)^c\Big) \leq e^{C_2 n h^m \epsilon_n^2}$. Hence, under the conditions for k of Proposition 1 and Lemma 7,

$$P\Big((\mathcal{Q}_n)^c\Big) \leq K\Big(2 + \frac{1}{4^m}\frac{n^2}{n-1}\Big)e^{-\frac{(k-1)t}{4^{m+1}p_{max}}} + K^2 e^{-\frac{n-1}{2}\left(\frac{\rho_{min}}{2} - \frac{k-1}{n-1}\right)} + 2e^{-n\frac{\delta}{8}} + 3e^{-C_2 n h^m \epsilon_n^2}. \quad (4.22)$$

To find an appropriate value of k so that 4.22 holds, we follow the same argument as in the proof of Theorem 1 and noticing that last term of the bound of $P((\mathcal{Q}_n)^c)$ is independent of k. We find that an appropriate value for k will be $k = c\big(1 + \frac{4\log(n)+(n-1)\rho_{min}}{2+\frac{t}{4^m p_{max}}}\big)$, for a constant c such that condition 1 is satisfied. Again $P\Big((\mathcal{Q}_n)^c\Big)$ goes to zero exponentially with n. □

It is important to notice that as n increases the boundary points found in $C^{(i)}(2\epsilon_n) \setminus C^{(j)}, i = 1, \ldots, K$ will decrease and will be significantly less than the points of the L(t) set, leading to cleaner predicted clusters. Actually $C^{(i)}(2\epsilon_n)$ will collapse to $C^{(i)}$. We will refer to the term exact identification when rough identification of clusters is achieved and the ratio of number of points that do not belong to any cluster to number of cluster points goes to zero.

Theorem 3 *(Exact cluster identification in noisy case 1)*
Let p be three times continuously differentiable with uniformly bounded derivatives and let $\hat{p}_n$ be a kernel density estimator with bandwidth $h_n = h_0(logn/n)^{\frac{1}{m+4}}$ for some $h_0 > 0$. For a suitable $\epsilon_0 > 0$ set $\epsilon_n = \epsilon_0(logn/n)^{\frac{2}{m+4}}$. Then there exist constants c_1, c_2 such that for $n \to \infty$ and $c_1 logn \leq k \leq c_2 n$ we obtain cluster $C^{(i)}$ is exactly identified by algorithm 4.1 almost surely.

Proof. According to Proposition 8 of von Luxburg (2007) if $N_{cluster}$ is the number of cluster points and $N_{NoCluster}$ is the number of points that do not belong to any cluster, then for ϵ_n that goes to zero as n goes to infinity and $\beta = \sum_{i=1}^{K} \beta_{(i)}$, there exist a constant $\bar{D}$ such that for large n,

$$P\Big(N_{NoCluster}/N_{cluster} > 4\frac{\bar{D}}{\beta}\epsilon_n \mid C_n\Big) \leq e^{-\frac{1}{4}\bar{D}\epsilon_n n} + e - n\frac{\beta}{8} + P\Big((\mathcal{D}_n)^c\Big). \quad (4.23)$$

We can choose ϵ_0 such that $h_n^2 \leq C\epsilon_n$ for a suitable constant C. Then there exist $C_2 > 0$ with $P\Big((\mathcal{D}_n)^c\Big) \leq e^{-C_2 n h_n^m \epsilon_n^2}$. Notice that $n h_n^m \epsilon_n^2 = h_0^m \epsilon_0 n\big(\frac{logn}{n}\big)^{\frac{m}{m+4}}\big(\frac{logn}{n}\big)^{\frac{4}{m+4}} = h_0^m \epsilon_0 logn$. Hence,

$\sum_{i=1}^{\infty} P\Big((\mathcal{D}_n)^c\Big) < \infty$. Furthermore by inequality 4.23 we have $\sum_{i=1}^{\infty} P\Big(N_{NoCluster}/N_{cluster} > 4\frac{\bar{D}}{\beta}\epsilon_n \mid C_n\Big) < \infty$. Following similar proof as of Theorem 2 we can find constants c_1, c_2 such that for $c_1 logn \leq k \leq c_2 n$ cluster $C^{(i)}$ will be roughly identified almost surely, and as a result the event C_n will also occur almost surely. Consequently, $N_{NoCluster}/N_{cluster} \to 0$ alomost surely.

4.3.2.2 Second approach to noisy case - No removal of points

As before we denote the connected components of the t-level of the density p by $C^{(1)}, ..., C^{(K)}$. For the rest in the support of p (i.e. $B = \text{supp}\{p\} \setminus \cup_{i=1}^{K} C^{(i)}$), we denote:

$$\tilde{C}^{(i)} = \{x \in B \,:\, i = \text{argmin}_{1\leq j\leq K} d(x, C^{(j)})\}, \text{ for } 1 \leq i \leq K.$$

The ith cluster consists of $C^{(i)}$ and $\tilde{C}^{(i)}$ and no point is removed. Let cluster i be denoted as the set $\bar{C}^{(i)} = C^{(i)} \cup \tilde{C}^{(i)}$. We describe the noisy case as the event that the minimal distance of points between $C^{(i)}$ and $\tilde{C}^{(i)}$ is zero. For this reason, for the minimum density of points in $\bar{C}^{(i)}$, p_{min} , will hold that $p_{min} > 0$. Consequently, $\bar{C}^{(i)}$ is connected topologically. The following results correspond to clustering with algorithm 4.1 and the choice of graph Laplacian to be the unnormalized, i.e. $L = D - W$. Let us denote the "ideal" version by $\tilde{L} = \tilde{D} - \tilde{W}$, namely; $\tilde{W}$ is the revised version of W by removing all connections between different clusters. So $\tilde{W}$ is a block diagonal matrix (up to permutation of the nodes), with each block corresponding to a true cluster. Table 4.2 provides notations for this case.

Lemma 8 *(Connectedness of $\bar{C}^{(i)}$)*

Let $C_n^{(i)}$ be the event that the sample points in $\bar{C}^{(i)}$ are connected. Then under the conditions:

1. $k \geq 4^{m+1} log(2np_{max}^{(i)} Vol(\bar{C}^{(i)}) 8^m)$ *and*
2. $k < 2(n-1)p_{max}^{(i)} \eta_m 4^m \min\{(d_n^{(i)})^m, (\nu_{max}^{(i)})^m\}$, *we obtain*

$$P\Big((C_n^{(i)})^c\Big) \leq (2 + \frac{1}{4^m}\frac{n^2}{n-1}) e^{-\frac{(k-1)p_{min}^{(i)}}{4^{m+1}p_{max}^{(i)}}}. \tag{4.24}$$

Proof. We find a covering of $\bar{C}^{(i)}(p_{min}^{(i)} + \epsilon_n)$ of $N \leq \frac{Vol(\bar{C}^{(i)})}{\eta_m \frac{z^m}{8^m}}$ balls with radius $z/4$, where $z \in 4(0, \min\{d_n^{(i)}, \nu_{max}^{(i)}\})$. If each of the covering balls contains at least two points of $\bar{C}^{(i)}$ and

$\bar{C}^{(i)}$	$C^{(i)} \cup \tilde{C}^{(i)}$
$p^{(i)}_{\max}$	supremum of density attained by points of $\bar{C}^{(i)}$
$p^{(i)}_{\min}$	infimum of density attained by points of $\bar{C}^{(i)}$
$u^{(i)}$	lower bound on distance of $C^{(i)}$ to all $C^{(j)}$ with $j \neq i$
u^{ij}	distance between r $C^{(i)}$ and $C^{(j)}$
$\rho(u^{(i)})$	probability mass of balls of radius $u^{(i)}$ in $C^{(i)}$
ρ_{min}	$\min_{i=1,\ldots,K} \rho(u^{(i)})$
$\tilde{\rho}(u^{(i)})$	probability mass of balls of radius $u^{(i)}$ in $\tilde{C}^{(i)}$
$\tilde{\rho}_{min}$	$\min_{i=1,\ldots,K} \tilde{\rho}(u^{(i)/2})$
$\beta_{(i)}$	probability mass of $C^{(i)}$
β_{max}	$\max_{1=1,\ldots,K} \beta_{(i)}$
$\tilde{\mu}_{(i)}$	probability mass of $\tilde{C}^{(i)}$
$\tilde{\mu}_{max}$	$\max_{1=1,\ldots,K} \tilde{\mu}_{(i)}$
$\hat{R}^{(i)}_{min}$	minimal kNN radius of $\bar{C}^{(i)}$
$\tilde{R}^{(i)}_{max}, R^{(i)}_{max}$	maximal kNN radius of $\tilde{C}^{(i)}, C^{(i)}$
$d^{(i)}_n$	minimum distance of $\bar{C}^{(i)}(p^{(i)}_{min} + \epsilon_n)$ from $\partial\bar{C}^{(i)}$
$\kappa^{(i)}$	the minimal curvature radius of $\partial\bar{C}^{(i)}$
$\nu^{(i)}_{max}$	$\max_{\nu<\kappa^{(i)}} \{\nu \mid \bar{C}^{(i)} \setminus Col^{(i)}(\nu)$is connected$\}$ and $Col^{(i)}$ is the collar of $\bar{C}^{(i)}$

Table 4.2 Notations.

$\{\hat{R}^{(i)}_{min} > z\}$ then neighboring covering balls will contain points that share common neighbors and hence they will be connected. We follow the arguments of the proofs of Lemma 1 and Lemma 3 to obtain

$$P\big((C^{(i)}_n)^c\big) \leq (2 + \frac{1}{4^m}\frac{n^2}{n-1}) e^{-\frac{(k-1)p^{(i)}_{min}}{4^{m+1}p^{(i)}_{max}}}$$

under the stated conditions and with $p^{(i)}_{\max}, p^{(i)}_{\min}$ to be the supremum and the infimum of density attained by points of $\bar{C}^{(i)}$. □

Lemma 9 *(Isolation of every $\tilde{C}^{(i)}$ from all $C^{(j)}$)*

Let $\tilde{\mathcal{I}}_n$ denote the event that every point in any $\tilde{C}^{(i)}$ is not connected to points in any $C^{(j)}$ for $j \neq i$. Then for $k < \frac{\tilde{\rho}_{min} n}{2} - 2\max\{log(\tilde{\mu}_{max} n), log(\beta_{max} n)\}$ we obtain

$$P\big((\tilde{\mathcal{I}}_n)^c\big) \leq K^2 e^{-\frac{n-1}{2}\left(\frac{\tilde{\rho}_{min}}{2} - \frac{k-1}{n-1}\right)} \tag{4.25}$$

Proof. Let $\tilde{\mathcal{I}}_n^{(i)}$ denote the event that every point of $\tilde{C}^{(i)}$ is not connected to points in any $C^{(j)}$ for $j \neq i$. Then $\tilde{C}^{(i)}$ is connected to some $C^{(j)}$ for $j \neq i$, if either the event $\{\tilde{R}_{max}^{(i)} > u^{(i)}\}$ occurs or $\bigcup_{j\neq i}\{R_{max}^{(j)} > u^{(ij)}/2\}$ occurs. Following the steps of the proof of Lemma 2 we obtain under the conditions that $k < \frac{\tilde{\rho}(u^{(i)})}{2} - 2log(\tilde{\mu}_{(i)}n)$ and for every $j \neq i$ that $k < \frac{\rho(u^{(j)/2})}{2} - 2log(\beta_{(j)}n)$. We reach the stated results observing that $\tilde{\rho}(u_{(i)}) > \tilde{\rho}(u_{(i)}/2) \geq \tilde{\rho}_{min}, \rho(u_{(j)}) > \tilde{\rho}(u_{(j)}/2) \geq \tilde{\rho}_{min}$ and that $P\big((\tilde{\mathcal{I}}_n)^c\big) \leq \sum_{i=1}^{K} P\big((\tilde{\mathcal{I}}_n^{(i)})^c\big)$. □

Proposition 2 *(Distance between the eigenspaces of L and $\tilde{L}$)*

Under the conditions,

1. $k \geq 4^{m+1}log(2np_{max}^{(i)}Vol(\bar{C}^{(i)})8^m)$ *for every* $i, 1 \leq i \leq K$ *and*
2. $k < 2(n-1)p_{max}^{(i)}\eta_m 4^m \min\{(d_n^{(i)})^m, (v_{max}^{(i)})^m\}$ *for every* $i, 1 \leq i \leq K$ *and*
3. $k < \frac{\tilde{\rho}_{min}n}{2} - 2\max\{log(\tilde{\mu}_{max}n), log(\beta_{max}n)\}$,

there exists an orthogonal matrix $O \in \mathbb{R}^{K\times K}$ *such that*

$$||UO - \tilde{U}||_F \leq \frac{2^{\frac{5}{2}}\sqrt{K}\sum_{i=1}^{K}\tilde{n}_i}{\lambda^+(\tilde{L})},$$

with probability at least $1 - K(2 + \frac{1}{4^m}\frac{n^2}{n-1})e^{-\frac{(k-1)p_{min}}{4^{m+1}p_{max}}} + K^2 e^{-\frac{n-1}{2}\left(\frac{\rho_{min}+\tilde{\rho}_{min}}{2} - \frac{2(k-1)}{n-1}\right)}$, *where* $U, \tilde{U}$ *are eigenvector matrices of the K smallest eigenvalues of* $L, \tilde{L}$*, respectively,* $\tilde{n}_i$ *is the size of* $\tilde{C}^{(i)}$ *and* $\lambda^+(\tilde{L})$ *is the* $K+1$ *smallest eigenvalue of* $\tilde{L}$.

Proof. Let C_n be the event that every cluster $\bar{C}^{(i)}$ is connected on $G_{SNN}(k)$. Then if k satisfies the the conditions of Lemma 8 for every cluster we obtain

$$P\big((C_n)^c\big) \leq K(2 + \frac{1}{4^m}\frac{n^2}{n-1})e^{-\frac{(k-1)p_{min}}{4^{m+1}p_{max}}}, \tag{4.26}$$

where $p_{min} = \min_{i=1,\ldots,K} p_{min}^{(i)}$ and $p_{max} = \min_{i=1,\ldots,K} p_{max}^{(i)}$. Removing any edges that connect subgraphs of different clusters will result in a block diagonal adjacency matrix $\tilde{W}$. As we explored in proof of Theorem 1 this is equivalent with exact cluster identification by spectral clustering.

Furthermore, let $\mathcal{I}_n$ be the event that every set $C^{(i)}$ is isolated from other sets $C^{(j)}$ for $j \neq i$ we obtain following the proof of Lemma 2 that if $k < \frac{\rho_{min}n}{2} - 2log(\beta_{max}n)$ then

$$P\big((\mathcal{I}_n)^c\big) \leq K^2 e^{-\frac{n-1}{2}\left(\frac{\rho_{min}}{2} - \frac{k-1}{n-1}\right)}. \tag{4.27}$$

Since $\frac{\rho_{min}n}{2} - 2log(\beta_{max}n) \geq \frac{\tilde{\rho}_{min}n}{2} - 2\max\{log(\tilde{\mu}_{max}n), log(\beta_{max}n)\}$ we observe that the if $k < \frac{\tilde{\rho}_{min}n}{2} - 2\max\{log(\tilde{\mu}_{max}n), log(\beta_{max}n)\}$ the upper bounds of inequality 4.27 and inequality 4.25 will hold. Now let us denote with S_n the event that the only connections between any pair of clusters $C^{(i)}, C^{(j)}$ are from the samples in $\tilde{C}^{(i)}$ and $\tilde{C}^{(j)}$. Then,

$$P\big((\mathcal{S}_n)^c\big) \leq P\big((C_n)^c\big) + P\big((\mathcal{I}_n)^c\big) + P\big((\tilde{\mathcal{I}}_n)^c\big) \tag{4.28}$$

and if k satisfies the condition for the upper bounds of those probabilities we obtain that,

$$P\big((\mathcal{S}_n)^c\big) \leq K(2 + \frac{1}{4^m}\frac{n^2}{n-1})e^{-\frac{(k-1)p_{min}}{4^{m+1}p_{max}}} + K^2 e^{-\frac{n-1}{2}\left(\frac{\rho_{min}+\tilde{\rho}_{min}}{2} - \frac{2(k-1)}{n-1}\right)} \tag{4.29}$$

Additionally, Theorem 2 in YU et al. (2015) we have that there exists an orthogonal matrix $O \in \mathbb{R}^{K\times K}$ such that

$$||UO - \tilde{U}||_F \leq \frac{2^{\frac{3}{2}}\sqrt{K}||L - \tilde{L}||_2}{\lambda^+(\tilde{L})},$$

where $U, \tilde{U}$ are the eigenvector matrices of the K smallest eigenvalues of $L, \tilde{L}$, respectively, $||L-\tilde{L}||_2$ is the spectral norm, and $\lambda^+(\tilde{L})$ is the smallest non-zero eigenvalue of $\tilde{L}$ (i.e. the $K+1$ smallest eigenvalues of $\tilde{L}$). Now we would like to bound $||L - \tilde{L}||_2$. First of all,

$$||L - \tilde{L}||_2 \leq ||D - \tilde{D}||_2 + ||W - \tilde{W}||_2,$$

and $||W - \tilde{W}||_2$ is determined by the between-cluster connections. So if the event $\mathcal{S}_n$ occurs and because Jaccard similarity ≤ 1 we notice that

$$||W - \tilde{W}||_2 \leq \sum_{i=1}^{K} \tilde{n}_i$$

and

$$||L - \tilde{L}||_2 \leq 2\sum_{i=1}^{K} \tilde{n}_i,$$

where $\tilde{n}_i$ is the sample size of $\tilde{C}^{(i)}$. Hence, under the condition for k we conclude that

$$||UO - \tilde{U}||_F \leq \frac{2^{\frac{5}{2}}\sqrt{K}\sum_{i=1}^{K}\tilde{n}_i}{\lambda^+(\tilde{L})},$$

with probability at least $1 - K(2 + \frac{1}{4^K}\frac{n^2}{n-1})e^{-\frac{(k-1)p_{min}}{4^{m+1}p_{max}}} + K^2 e^{-\frac{n-1}{2}\left(\frac{\rho_{min}+\tilde{\rho}_{min}}{2} - \frac{2(k-1)}{n-1}\right)}$. □

Definition 3

The mis-clustering error is defined as

$$M_n = \min_{\sigma \in S_K} \frac{1}{n} \sum_{i=1}^{n} \mathbf{1}(\sigma(\tilde{q}_i) \neq q_i),$$

where q_i is the true cluster label of ith data, $\tilde{q}_i$ is the estimated one and S_K is the set of all possible permutations of {1, . . . K}.

Theorem 4 *(Mis-clustering error bound)*

Under the conditions,

1. $k \geq 4^{m+1} log(2np_{max}^{(i)} Vol(\bar{C}^{(i)}) 8^m)$ *for every* $i, 1 \leq i \leq K$ *and*
2. $k < 2(n-1)p_{max}^{(i)} \eta_m 4^m \min\{(d_n^{(i)})^m, (v_{max}^{(i)})^m\}$ *for every* $i, 1 \leq i \leq K$ *and*
3. $k < \frac{\tilde{\rho}_{min} n}{2} - 2\max\{log(\tilde{\mu}_{max} n), log(\beta_{max} n)\}$,

we obtain

$$M_n \leq \frac{256 n^* K}{n} \frac{(\sum_{i=1}^{K} \tilde{n}_i)^2}{\lambda^+(\tilde{L})^2}$$

with probability at least $1 - K(2 + \frac{1}{4^m} \frac{n^2}{n-1}) e^{-\frac{(k-1)p_{min}}{4^{m+1} p_{max}}} + K^2 e^{-\frac{n-1}{2}\left(\frac{\rho_{min} + \tilde{\rho}_{min}}{2} - \frac{2(k-1)}{n-1}\right)}$.

Here, $n^* = \max_{1 \leq i \leq K} n_i$ *for the size of clusters* $\bar{C}^{(i)}$, $n_i, 1 \leq i \leq K$.

Proof. We notice that $\{\tilde{q}_i\}_{i=1}^n$ are obtained by running k-means on $U \in \mathbb{R}^{n \times K}$. Let us define their associated centroids by $\{\tilde{h}_i\}_{i=1}^n$. Note also that $\{\tilde{h}_i\}_{i=1}^n$ have K unique vectors. Further define

$$\tilde{H} = \text{argmin}_{H \in \mathbb{R}^{n \times K}:\ \text{has } K \text{ unique rows}} ||U - H||_F^2.$$

It is clear that $\tilde{H} = \begin{pmatrix} \tilde{h}_1^T \\ \tilde{h}_2^T \\ \vdots \\ \tilde{h}_n^T \end{pmatrix} \in \mathbb{R}^{n \times K}$. We define the set $A = \left\{1 \leq i \leq n :\ ||\tilde{h}_i - e_i^T \tilde{U} O^T||_2 \geq \frac{1}{\sqrt{2n^*}}\right\}$, where $\{e_i\}_{i=1}^n$ is the standard basis of $\mathbb{R}^n$ and $n^* = \max_{1 \leq i \leq K} n_i$ where n_i is the sample size from the ith cluster $C^{(i)} \cup \tilde{C}^{(i)}$. Then,

$$||\tilde{h}_i - e_i^T \tilde{U} O^T||_2 < \frac{1}{\sqrt{2n^*}}, \text{ for } i \notin A. \tag{4.30}$$

Also,

$$\tilde{U} = \begin{pmatrix} \frac{1}{\sqrt{n_1}}\mathbf{1}_{n_1} & 0 & \cdots & 0 \\ 0 & \frac{1}{\sqrt{n_2}}\mathbf{1}_{n_2} & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & \frac{1}{\sqrt{n_K}}\mathbf{1}_{n_K} \end{pmatrix} P, \tag{4.31}$$

with P being orthogonal. By 4.31

$$||e_i^T \tilde{U} O^T - e_j^T \tilde{U} O^T||_2 \geq \sqrt{\frac{2}{n^*}} \tag{4.32}$$

Combining 4.30 and 4.32

$$||\tilde{h}_i - e_j^T \tilde{U} O^T||_2 \geq ||e_i^T \tilde{U} O^T - e_j^T \tilde{U} O^T||_2 - ||\tilde{h}_i - e_i^T \tilde{U} O^T||_2 > ||e_i^T \tilde{U} O^T - e_j^T \tilde{U} O^T||_2 - \frac{1}{\sqrt{2n^*}} > \frac{1}{\sqrt{2n^*}}$$

Consequently,

$$\begin{aligned} M_n \leq \frac{1}{n}|A| \leq \frac{1}{n}\sum_{i\in A}\mathbf{1} \leq \frac{1}{n}2n^*\sum_{i\in A}||\tilde{h}_i - e_i^T\tilde{U}O^T||_2^2 &\leq \frac{2n^*}{n}||\tilde{H} - \tilde{U}O^T||_F^2 \\ &\leq \frac{2n^*}{n}\left(||\tilde{H} - U||_F + ||U - \tilde{U}O^T||_F\right)^2 \\ &\leq \frac{8n^*}{n}||U - \tilde{U}O^T||_F^2 \\ &= \frac{8n^*}{n}||UO - \tilde{U}||_F^2 \\ &\leq \frac{8n^*}{n}\left(\frac{2^{\frac{5}{2}}\sqrt{K}\sum_{i=1}^K \tilde{n}_i}{\lambda^+(\tilde{L})}\right)^2 \\ &\leq \frac{256n^*K}{n}\frac{(\sum_{i=1}^K \tilde{n}_i)^2}{\lambda^+(\tilde{L})^2}, \end{aligned}$$

using Proposition 2 with probability at least $1 - K(2 + \frac{1}{4^m}\frac{n^2}{n-1})e^{-\frac{(k-1)p_{min}}{4^{m+1}p_{max}}} + K^2 e^{-\frac{n-1}{2}\left(\frac{\rho_{min}+\tilde{\rho}_{min}}{2} - \frac{2(k-1)}{n-1}\right)}$.

□

4.4 A general algorithm for tuning of clustering method parameters

To construct an SNN graph as we described in section 4.2.1 one must first construct a kNN graph and then remove edges between kNN neighbors that do not share any of their neighbors. The parameter k affects the structure of the SNN graph and hence any other method that is based on

it, as for example the algorithm 4.1. A lot of SNN graph-based clustering methods (Stuart et al., 2019; Xu and Su, 2015) do not use data-driven ways to decide on the value of k.

Bellow we introduce a general cross-validation tuning algorithm called $k$cv tuning that provides an optimal choice of a clustering parameter based on the data information. We apply this algorithm to find an optimal choice for the parameter k of the number of nearest neighbors used in the SNN spectral clustering algorithm 4.1 in simulated data of different signal-to-noise levels and data feature structure.

The simulated data are described in section 4.4.2 and the tools used to assess the performance of algorithm 4.3 are introduced in section 4.4.3. The performance results are summarized in section 4.4.4.

4.4.1 $k$cv tuning algorithm

The introduced cross-validation method suggests a tuning of a parameter k of a clustering method, based on the idea that when the clustering is optimal, points in the same cluster can predict with high accuracy features of points in the same cluster. A similar methodology has been used for the tuning of model parameters in Li et al. (2020).

The $k$cv tuning algorithm works in N folds.

- Given a dataset X, in every fold a version of X is created by randomly removing 10% of the entries of X.

- Then, Singular Value Thresholding is applied to each version to extract its low rank approximation and hence a completed matrix, $\hat{A}$.

- The chosen clustering algorithm is applied on $\hat{A}$ for multiple values of k.

- Next, the missing entries of X are predicted. Specifically, a missing feature of a data point in the predicted cluster c, associated with a specific value k, is predicted by data points of X that belong also to cluster c.

- The optimal k is chosen to be the one that attained the lowest average prediction error across the N folds.

Notice that for the completion of A^q, we first find its SVD, i.e $A^q = UDV^T$ and then we use SVD

thresholding to obtain the final result $\hat{A}^q = \frac{1}{m} U D_{\hat{k}} V^T, D_{\hat{k}} = diag(D_{1,1}, D_{2,2}, ..., D_{\hat{k},\hat{k}}, 0, .., 0)$.

Algorithm 4.3 kcv_tuning.

Input: $X \in \mathbb{R}^{n \times m}$, number of clusters K, clustering function G, nearest neighbors k, number of folds N, prediction method (mean, ols, lasso), low-rank approximation threshold $\hat{k}$, training percentage p
Output: Optimal number of nearest neighbors, k_{optimal}

for $q = 1$ to N **do**
    $A^q \leftarrow X$ with (1-p)% of entries randomly replaced by 0
    $I_q \leftarrow \{(i, j) : x_{i,j}$ replaced with 0 in $A^q\}$

    % A^q completion via Singular Value Thresholding
    $\hat{A}^q \leftarrow$ SVD on X with threshold $\hat{k}$

    % Evaluation of the performance of k
    for $k = 10$ to $\lceil \frac{n}{3} \rceil$, by 20 **do**
        $\ell_k = G(\hat{A}^q, K, k) \leftarrow$ predicted membership
        for $\{i, j\} \in I_q$ **do**
            $C_i \leftarrow$ points in the same cluster as point i
            $Y = [X_{rj}]$ for r $\in C_i$
            $Z = [X_{rt}]$ for r $\in C_i$ and $t \neq j$

            % Prediction of missing values
            if prediction_method = mean **then**
                $\hat{x}_{i,j} \leftarrow \bar{Y}$
            else if prediction_method = ols **then**
                $\hat{\beta} \leftarrow argmin_\beta\{||Y - \beta Z||^2\}$
                $\hat{x}_{ij} \leftarrow \hat{\beta} Z$
            else if prediction_method = lasso **then**
                $\lambda_0, \hat{\beta} \leftarrow argmin_{\lambda,\beta}\{||Y - \beta Z||^2 + \lambda||\beta||_1\}$
                $\hat{x}_{ij} \leftarrow \hat{\beta} Z$

        % Mean prediction error of k in fold q
        $L_{k,q} = \frac{1}{|I_q|} \sum_{(i,j) \in I_Q} (x_{ij} - \hat{x}_{ij})^2$

% Mean prediction error of k
$L_k = \frac{1}{N} \sum_{q=1}^{N} L_{k,q}$

% Optimal k
$k_{optimal} = argmin_k(L_k)$

4.4.2 Simulations

We test the performance of the algorithm 4.3 when used for the tuning of parameter k of algorithm 4.1, on various Multivariate Gaussian (MG) data. We consider a simulated data set of n data points $x_1, x_2, \ldots, x_n \in \mathbb{R}^m$, that are grouped in K clusters. The data are independently sampled from a Multivariate Gaussian mixture model:

$$\sum_{i=1}^{K} \pi_i N(\mu_i, \Sigma_i).$$

The coordinates of the centers of the Gaussian mixture follow the standard normal distribution,

$$\mu_{ij} \overset{i.i.d.}{\sim} N(0,1) \text{ for } i = 1, \ldots, K, j = 1, \ldots, m,$$

and the clusters have equal sizes, i.e $\pi_1 = \cdots = \pi_K = 1/K$. We consider three types of covariance matrices Σ_k:

1. the simple case where $\Sigma_1 = \cdots = \Sigma_K = ms \cdot I$

2.the case where the $\Sigma_1 = \cdots = \Sigma_K = ms \cdot \Sigma$ and the corresponding precision matrix $\Omega = \Sigma^{-1}$ is tridiagonal. This case simulates a chain dependency between features of the data. Specifically, $\Sigma = \{\sigma_{i,j}\}$, with $\sigma_{i,j} = 0.5^{|i-j|}$.

3. the case where the $\Sigma_1 = \cdots = \Sigma_K = ms \cdot \Sigma$, the corresponding precision matrix $\Omega = \Sigma^{-1}$ is sparse and simulates a network dependency between features of the data. For the construction of Ω we follow the simulation procedure of Li and Gui (2005).

For the assessment of the $k$cv tuning algorithm, we simulated Multivariate Gaussian data of n=1000 points and $m = 10$ or 50 features. The tables below represent the data setting considered based on the type of covariance matrix used.

4.4.3 Assessment methods

We introduce the Normalized Prediction Accuracy function that measures the performance of a value k for the clustering of a data set, utilizing the average prediction loss. Additionally, we define the ARI Relative Ratio that measures how close the choice of the value k suggested by 4.3 is from the value that achieves maximum ARI.

m	s=0.1	s=0.3	s=0.5
10	setting 1	setting 2	setting 3
50	setting 4	setting 5	setting 6

(a) Simple MG data

m	s=0.1	s=0.2	s=0.3
10	setting 7	setting 8	setting 9
50	setting 10	setting11	setting 12

(b) MG data with tridiagonal precision matrix

m	s = 0.06	s = 0.16	s = 0.23
10	setting 13	setting14	setting 15
50	setting 16	setting17	setting 18

(c) MG data with network feature dependency

Table 4.3 Settings of simulated data.

Definition 4 *(Normalized Prediction Accuracy - NPA)*

Let $\mathcal{K}$ to be the set of values for the parameter k of the number of nearest neighbors. Let L_k^i be the the mean prediction error of k associated with the prediction of the held out entries of the data matrix X, for the simulation iteration i. The Normalized Prediction Accuracy, F, of k for the iteration i is :

$$F(k,i) = 1 - \frac{L_k^i - min_{j\in\mathcal{K}}(L_j^i)}{max_{j\in\mathcal{K}}(L_j^i) - min_{j\in\mathcal{K}}(L_j^i)}$$

Definition 5 *(ARI Relative Ratio)*

Let $\tilde{k} = argmax_{k\in\mathcal{K}}(ARI)$, when running the Snn_Spectral_Clustering algorithm. Let $\hat{k}$ be the optimal k based on the kcv_tuning algorithm. The ARI Relative Ratio for iteration i, $R(i)$, is:

$$R(i) = \frac{ARI(\tilde{k})_i - ARI(\hat{k})_i}{ARI(\tilde{k})_i},$$

where $ARI(k)_i$ is the Adjusted Rand Index for the clustering produced using k nearest neighbors in iteration i.

4.4.4 Simulation Results

The simulated data were used to tune the parameter k of the SNN spectral clustering algorithm introduced in section 4.2.2. Below we describe the performance results of the $k$cv tuning algorithm.

To summarize the NPA results of a particular value of k, we use the mean NPA of this value over a round of 1000 simulations. The maximum mean NPA is achieved for the value of k that the $k$cv tuning algorithm suggests as optimal. It is interesting to observe whether the mean NPA

is maximized for the same value of k that would achieve the maximum mean ARI over a round of simulations. We notice that this depends on the prediction method used along with the structure of feature dependency.

For data settings 1-6 (figures A.2, A.4) we observe that the ARI of SNN spectral algorithm is increasing for $10 \leq k \leq 50$ and for larger values of k, the ARI remains about the same. The maximum ARI is achieved for $k = 330$, i.e. the largest k we consider during tuning. The NPA curve is also maximized at $k = 330$ for mean, ols, and lasso prediction. Mean and lasso perform better than ols and achieve lower values of ARI ratios. This is because ols uses information from all features introducing more variance in the prediction, since those settings simulate data sets with independent features. Mean uses information only from one feature and lasso selects a few of the features and hence performs better than ols.

For data settings 7-9 and 13-15 the maximum ARI achieved by the SNN spectral clustering algorithm is within the range of [10,90] (figures B.2, B.4, C.2). Applying prediction with ols or mean, fails to tune k within this range, in contrast to lasso. This is because simulated data of type 7-15 do not have independent features, every feature depends on 2-5 other features. The ols prediction will utilize information of every feature and will introduce more error in the prediction and the mean prediction takes into consideration only the information of one feature, whereas lasso will use information only of the features correlated to the one of interest. For this reason, lasso performs better for types 7-9 and 13-15.

In settings 10-12 and 16-18, although feature selection methods like lasso are expected to perform better than ols and mean, it is observed that they have similar performance. Here the simulated data have low signal-to-noise ratio and hence a ridge regression prediction or elastic net might perform better.

The mean and median ARI Relative Ratios provide an estimate of the proportion of the difference between the maximum ARI and the ARI achieved by clustering using the tuned k. The settings with larger variance factor s as shown in table 4.3 have higher mean and median ARI Relative Ratios than settings with lower s. In more detail, we observe that for settings 1-6, the ARI of

SNN spectral clustering after tuning is about 1% to 10% less than the optimum ARI when using mean and lasso prediction for tuning (tables A.1, A.2). For settings 7-9, lasso achieves the best tuning results with ARI that is 3% to 14% smaller than the optimum ARI (table B.1). The chain dependency of a higher number of features makes settings 10-12 harder to tune for k. The tuned k obtains an ARI difference from the optimum between 3% to 20% (tables B.2). However, for the final set of simulations (settings 13-18) the features of each dataset can be represented as a network and every feature will depend on 5 (setting 13-15) or 12 other features (settings 16-18). In this case, the ARI of SNN spectral clustering after tuning is only 0.6% to 10% lower than the optimum (tables C.1, C.2).

4.5 Conclusions

In this chapter, we conducted an investigation into the clustering performance of an SNN graph-based method. For this method, we build the SNN graph as a subgraph of a kNN graph based on the Jaccard similarity of $k$nn neighbors of vertices. The parameter k affects the structure of the SNN graph and hence the clustering performance. Our goal was to determine for which values of k, the SNN spectral clustering algorithm can achieve true cluster identification with high probability. Our results suggest that in both the noise-free and the noisy case, one needs to select k of the order cn to maximize the probability of cluster identification, in contrast to random geometric graph literature that suggests k of order $\log n$ (Brito et al., 1997). Furthermore, we introduce a general cross-validation tuning method for parameters of clustering algorithms. We use this method to tune the number of nearest neighbors k of the SNN spectral clustering algorithm for a variety of simulated data types and find that the accuracy of clustering results after using the tuned value is 1% to 20% lower than the accuracy achieved by the optimum k and depends on the feature dependency of the data.

BIBLIOGRAPHY

Brito, M. R., Chávez, E. L., Quiroz, A. J., and Yukich, J. E. (1997). Connectivity of the mutual k-nearest-neighbor graph in clustering and outlier detection. *Statistics & Probability Letters*, 35(1):33–42.

Guattery, S. and Miller, G. L. (1998). On the quality of spectral separators. *SIAM Journal on Matrix Analysis and Applications*, 19(3):701–719.

Li, H. and Gui, J. (2005). Gradient directed regularization for sparse Gaussian concentration graphs, with applications to inference of genetic networks. *Biostatistics*, 7(2):302–317.

Li, T., Levina, E., and Zhu, J. (2020). Network cross-validation by edge sampling. *Biometrika*, 107(2):257–276.

Luxburg, U., Bousquet, O., and Belkin, M. (2004). Limits of spectral clustering. *Advances in neural information processing systems*, 17.

Maier, M., Hein, M., and von Luxburg, U. (2009). Optimal construction of k-nearest-neighbor graphs for identifying noisy clusters. *Theoretical Computer Science*, 410(19):1749–1764.

Meilă, M. and Shi, J. (2001). A random walks view of spectral segmentation. In *International Workshop on Artificial Intelligence and Statistics*, pages 203–208. PMLR.

Ng, A., Jordan, M., and Weiss, Y. (2001). On spectral clustering: Analysis and an algorithm. *Advances in neural information processing systems*, 14.

Prakasa Rao, B. (1983). Nonparametric functional estimation. Probability and Mathematical Statistics: A Series of Monographs and Textbooks. Academic Press.

Spielman, D. A. and Teng, S.-H. (1996). Spectral partitioning works: Planar graphs and finite element meshes. In *Proceedings of 37th conference on foundations of computer science*, pages 96–105. IEEE.

Stuart, T., Butler, A., Hoffman, P., Hafemeister, C., Papalexi, E., III, W. M. M., Hao, Y., Stoeckius, M., Smibert, P., and Satija, R. (2019). Comprehensive integration of single-cell data. *Cell*, 177:1888–1902.

von Luxburg, U. (2007). A tutorial on spectral clustering. *Statistics and Computing*, 17.

Xu, C. and Su, Z. (2015). Identification of cell types from single-cell transcriptomes using a novel clustering method. *Bioinformatics*, 31(12):1974–1980.

YU, Y., WANG, T., and SAMWORTH, R. J. (2015). A useful variant of the davis—kahan theorem for statisticians. *Biometrika*, 102(2):315–323.

APPENDIX A

PERFORMANCE ON GAUSSIAN DATA WITH DIAGONAL COVARIANCE MATRIX

prediction	setting	Median	Mean	Sd.Error
mean	1	0.007	0.010	0.001
	2	0.021	0.042	0.004
	3	0.053	0.099	0.010
ols	1	0.006	0.011	0.001
	2	0.024	0.053	0.005
	3	0.055	0.109	0.010
lasso	1	0.007	0.012	0.001
	2	0.023	0.044	0.004
	3	0.051	0.102	0.010

Table A.1 ARI ratios summary for settings 1, 2 and 3.

prediction	setting	Median	Mean	Sd.Error
mean	4	0.009	0.012	0.001
	5	0.030	0.074	0.008
	6	0.114	0.280	0.023
ols	4	0.009	0.019	0.002
	5	0.049	0.151	0.016
	6	0.132	0.309	0.024
lasso	4	0.009	0.011	0.001
	5	0.030	0.078	0.010
	6	0.100	0.249	0.021

Table A.2 ARI ratios summary for settings 4, 5 and 6.

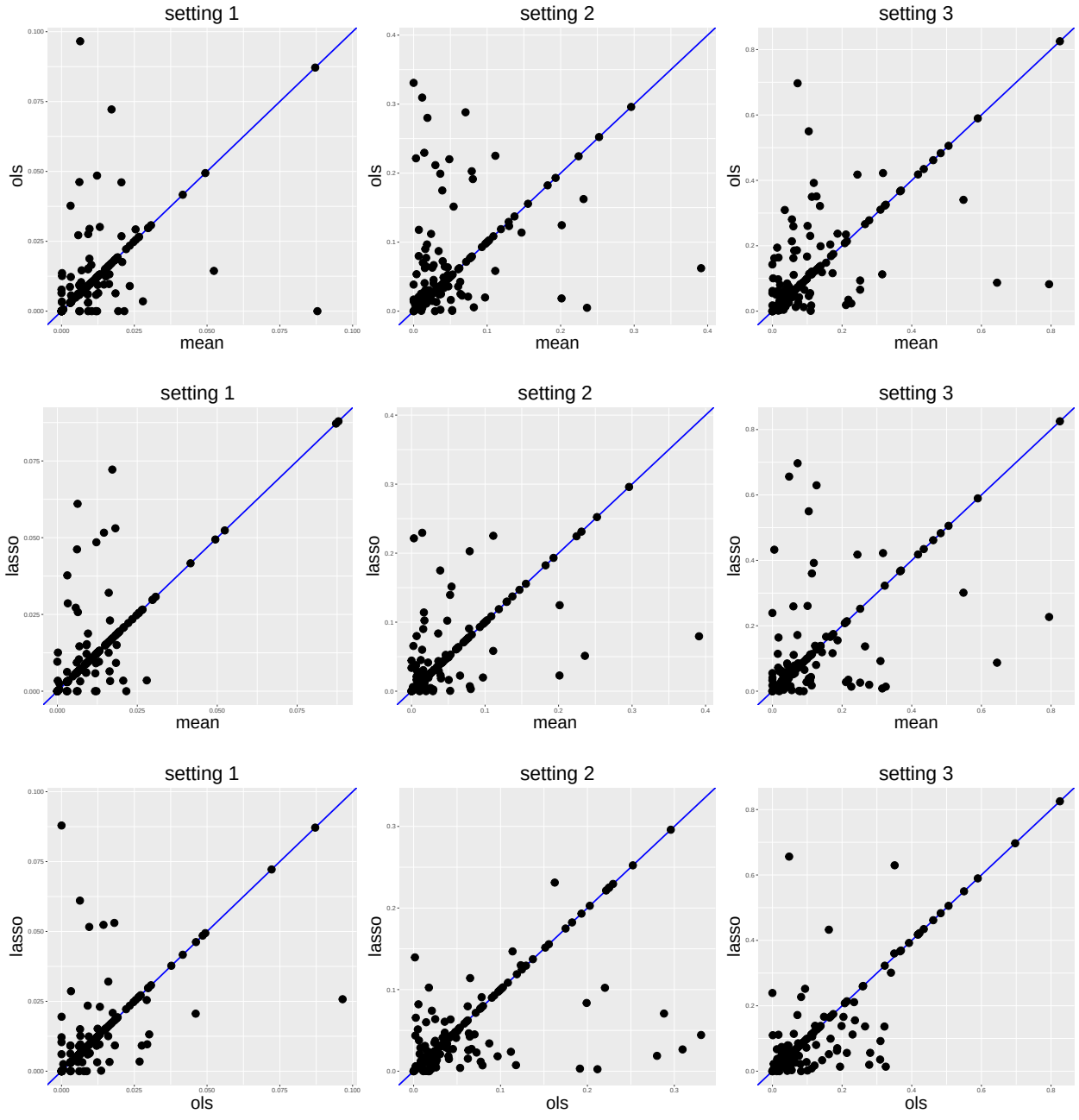


Figure A.1 ARI ratios comparisons for 1, 2, 3.

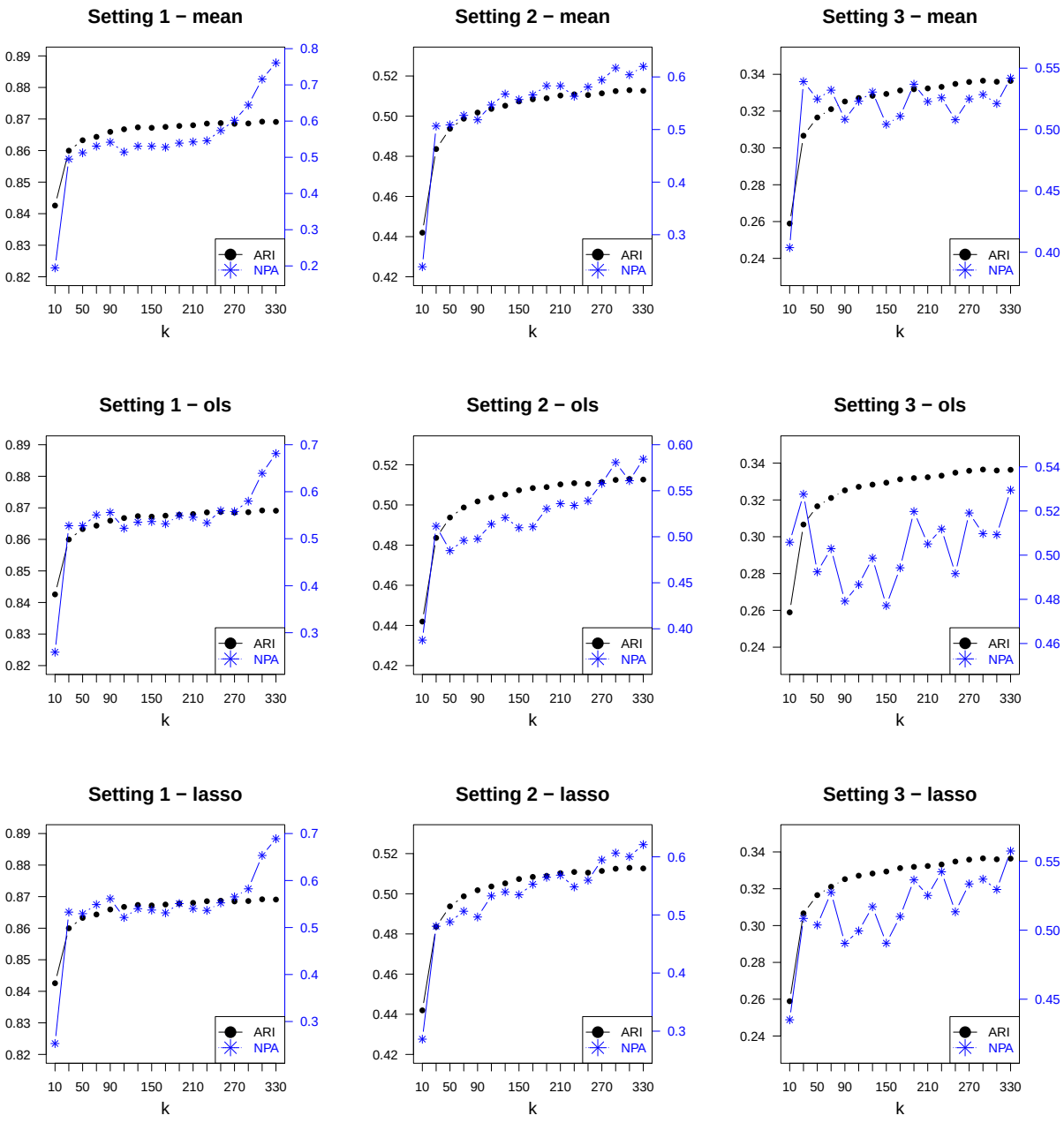


Figure A.2 ARI and NPA vs k for settings 1, 2 and 3.

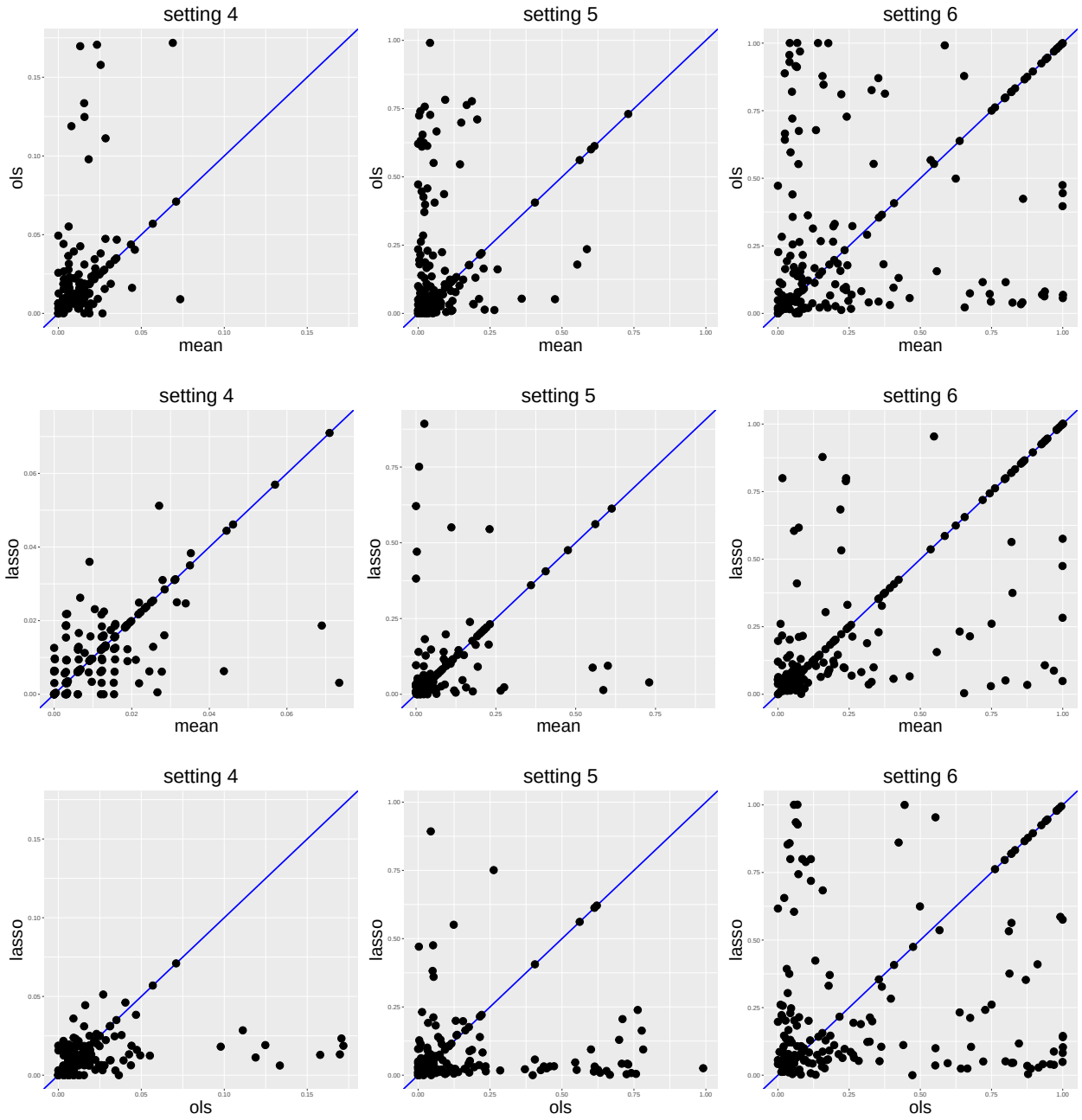


Figure A.3 ARI ratios comparisons for 4, 5, 6.

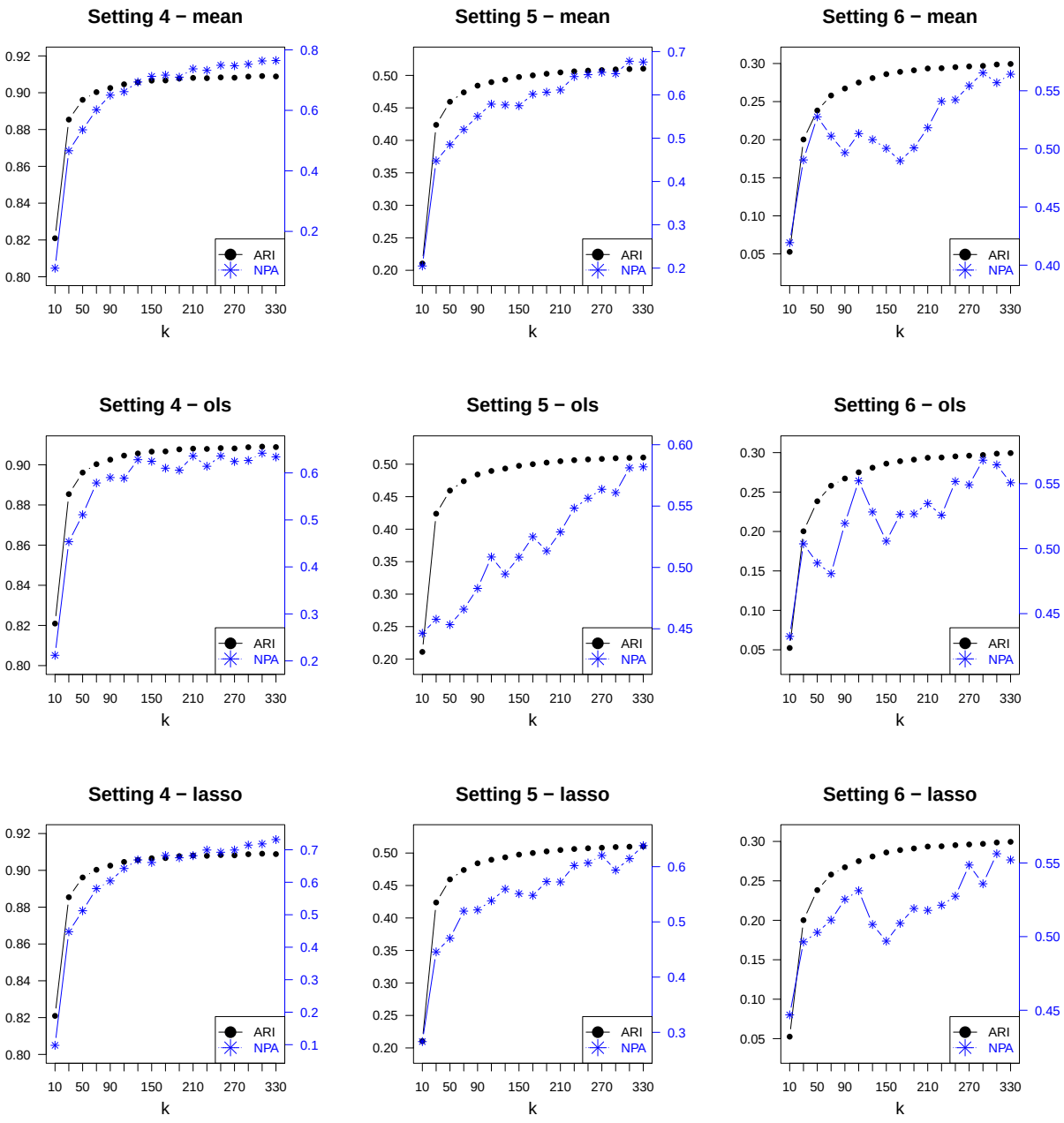


Figure A.4 ARI and NPA vs k for settings 4, 5 and 6.

APPENDIX B

PERFORMANCE ON GAUSSIAN DATA WITH TRIDIAGONAL PRECISION MATRIX

prediction	setting	Median	Mean	Sd.Error
mean	7	0.029	0.046	0.004
	8	0.094	0.116	0.008
	9	0.118	0.158	0.011
ols	7	0.023	0.036	0.003
	8	0.067	0.101	0.008
	9	0.102	0.143	0.011
lasso	7	0.015	0.030	0.003
	8	0.060	0.094	0.007
	9	0.099	0.141	0.010

Table B.1 ARI ratios summary for settings 7, 8 and 9.

prediction	setting	Median	Mean	Sd.Error
mean	10	0.038	0.048	0.003
	11	0.171	0.201	0.012
	12	0.243	0.281	0.015
ols	10	0.025	0.040	0.003
	11	0.163	0.198	0.012
	12	0.102	0.143	0.011
lasso	10	0.027	0.038	0.002
	11	0.147	0.202	0.013
	12	0.219	0.271	0.016

Table B.2 ARI ratios summary for settings 10, 11 and 12.

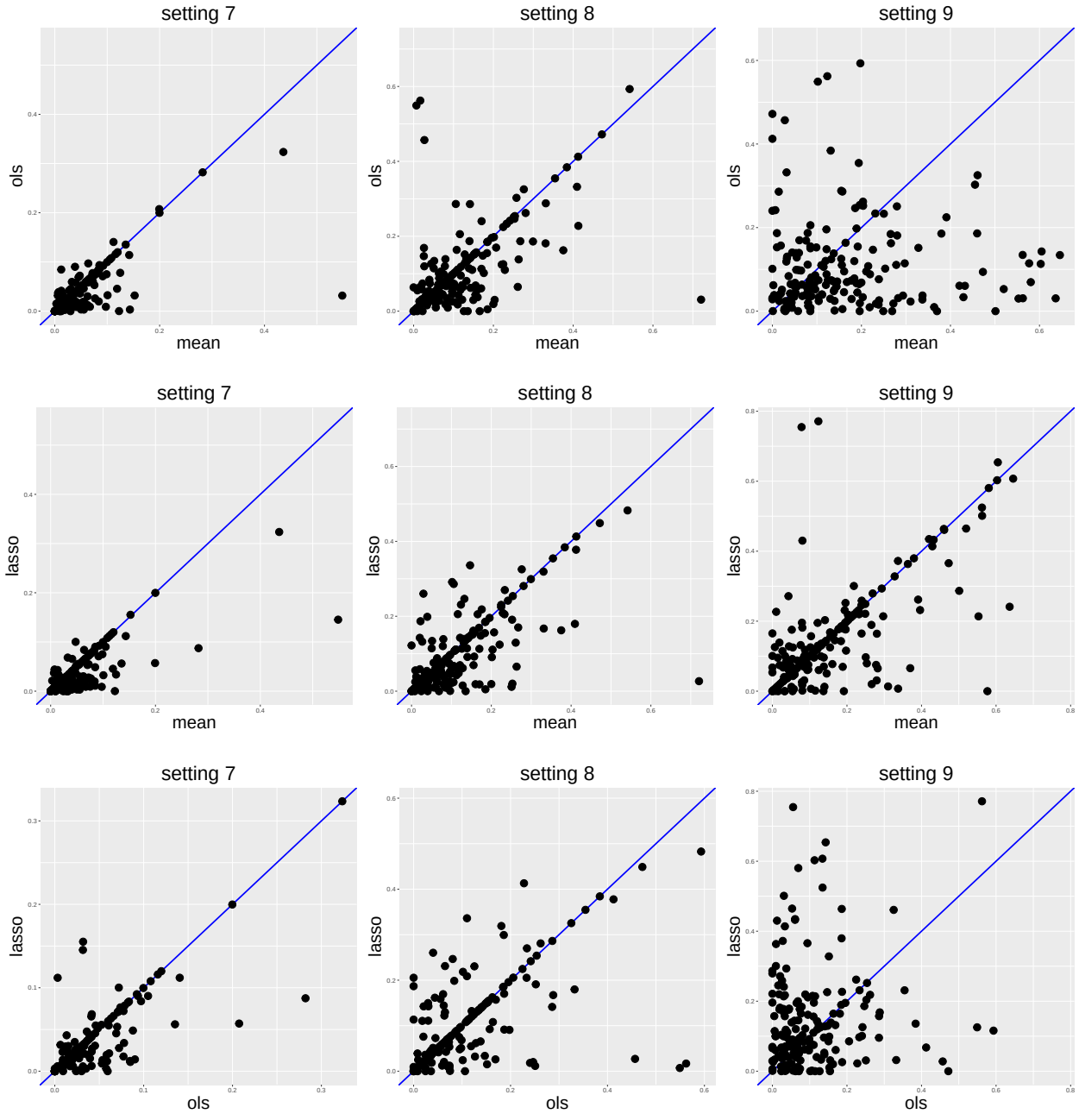


Figure B.1 ARI ratios comparisons for 7, 8, 9.

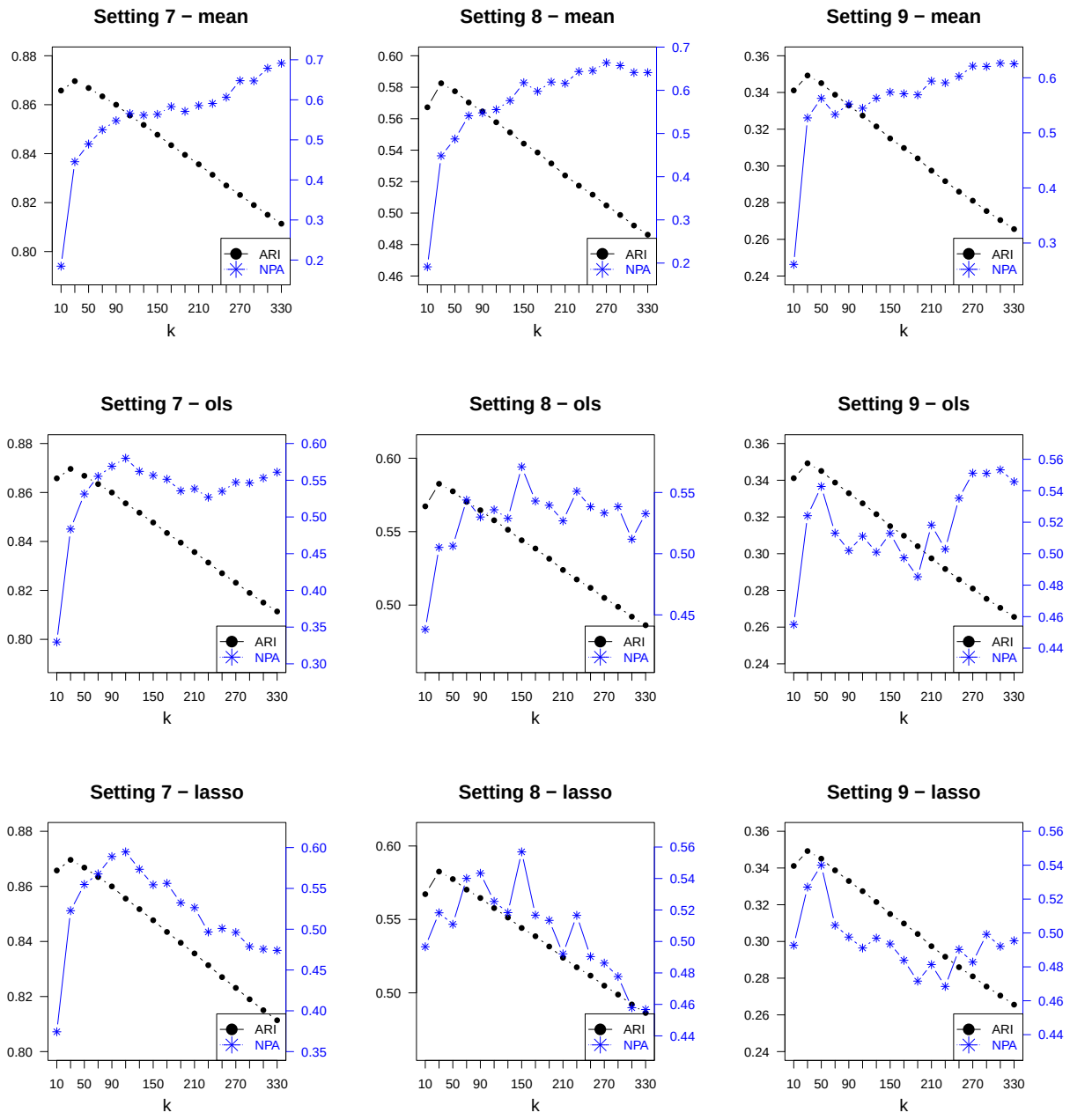


Figure B.2 ARI and NPA vs k for settings 7, 8 and 9.

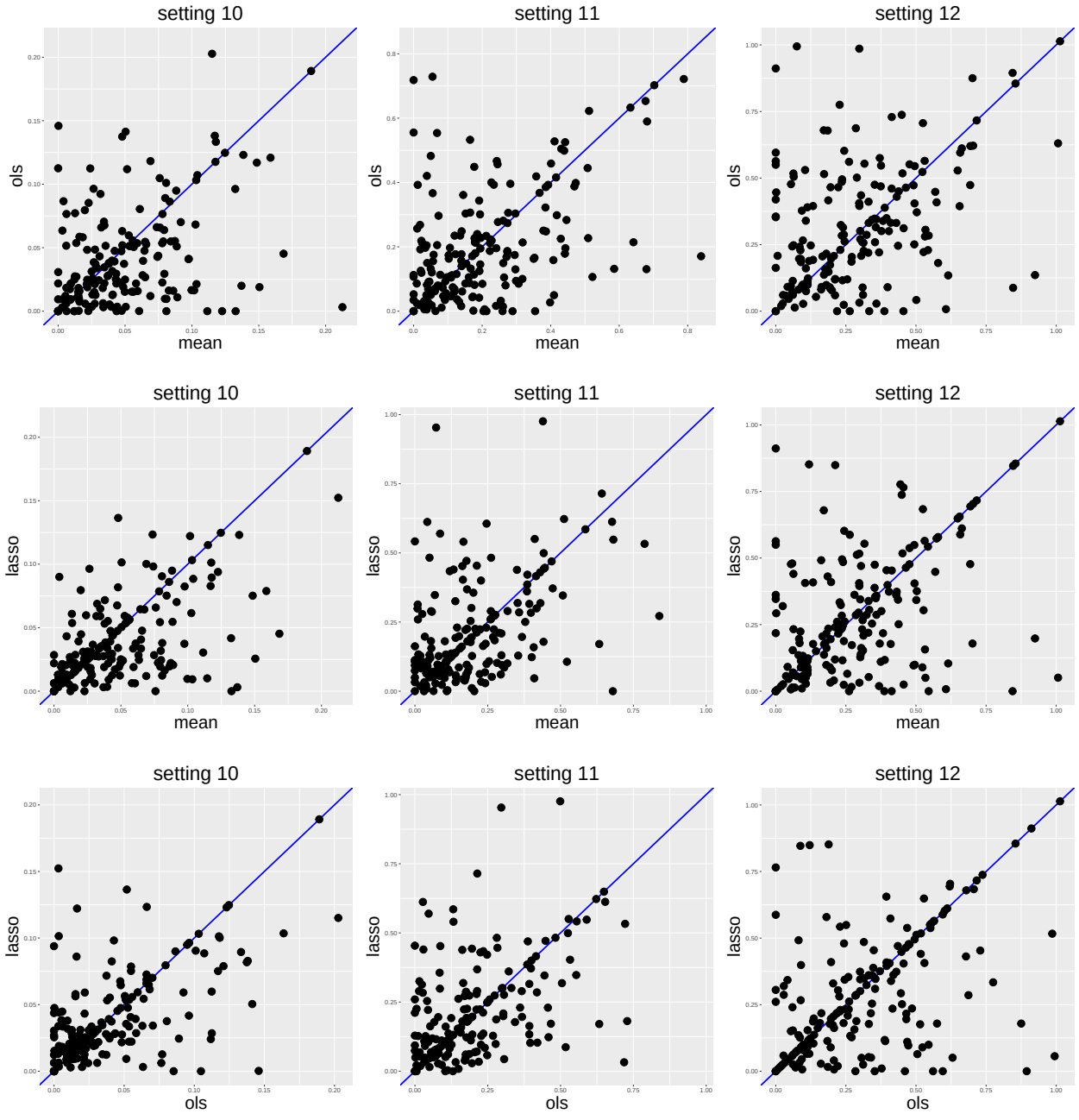


Figure B.3 ARI ratio comparisons for settings 10, 11 and 12.

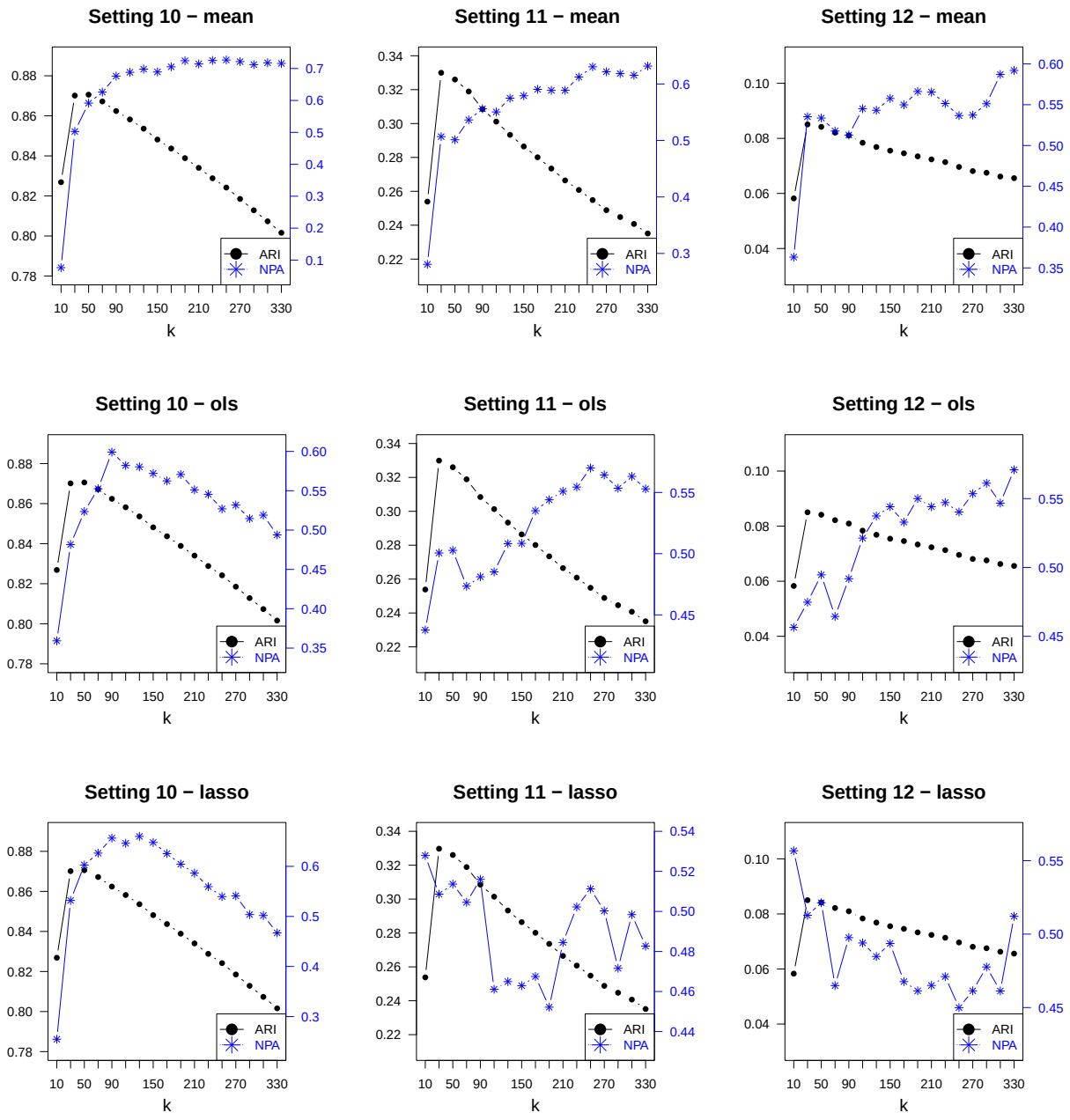


Figure B.4 ARI and NPA vs k for settings 10, 11 and 12.

APPENDIX C

PERFORMANCE ON GAUSSIAN DATA WITH NETWORK OF FEATURES

prediction	setting	Median	Mean	Sd.Error
mean	13	0.015	0.026	0.003
	14	0.058	0.078	0.006
	15	0.088	0.121	0.009
ols	13	0.012	0.020	0.002
	14	0.048	0.071	0.005
	15	0.077	0.105	0.008
lasso	13	0.009	0.017	0.001
	14	0.047	0.072	0.006
	15	0.072	0.104	0.008

Table C.1 ARI ratios summary for settings 13, 14 and 15.

prediction	setting	Median	Mean	Sd.Error
mean	16	0.006	0.006	0.000
	17	0.027	0.036	0.003
	18	0.050	0.107	0.012
ols	16	0.006	0.008	0.001
	17	0.029	0.056	0.006
	18	0.057	0.147	0.014
lasso	16	0.006	0.007	0.000
	17	0.027	0.053	0.006
	18	0.061	0.164	0.015

Table C.2 ARI ratios summary for settings 16, 17 and 18.

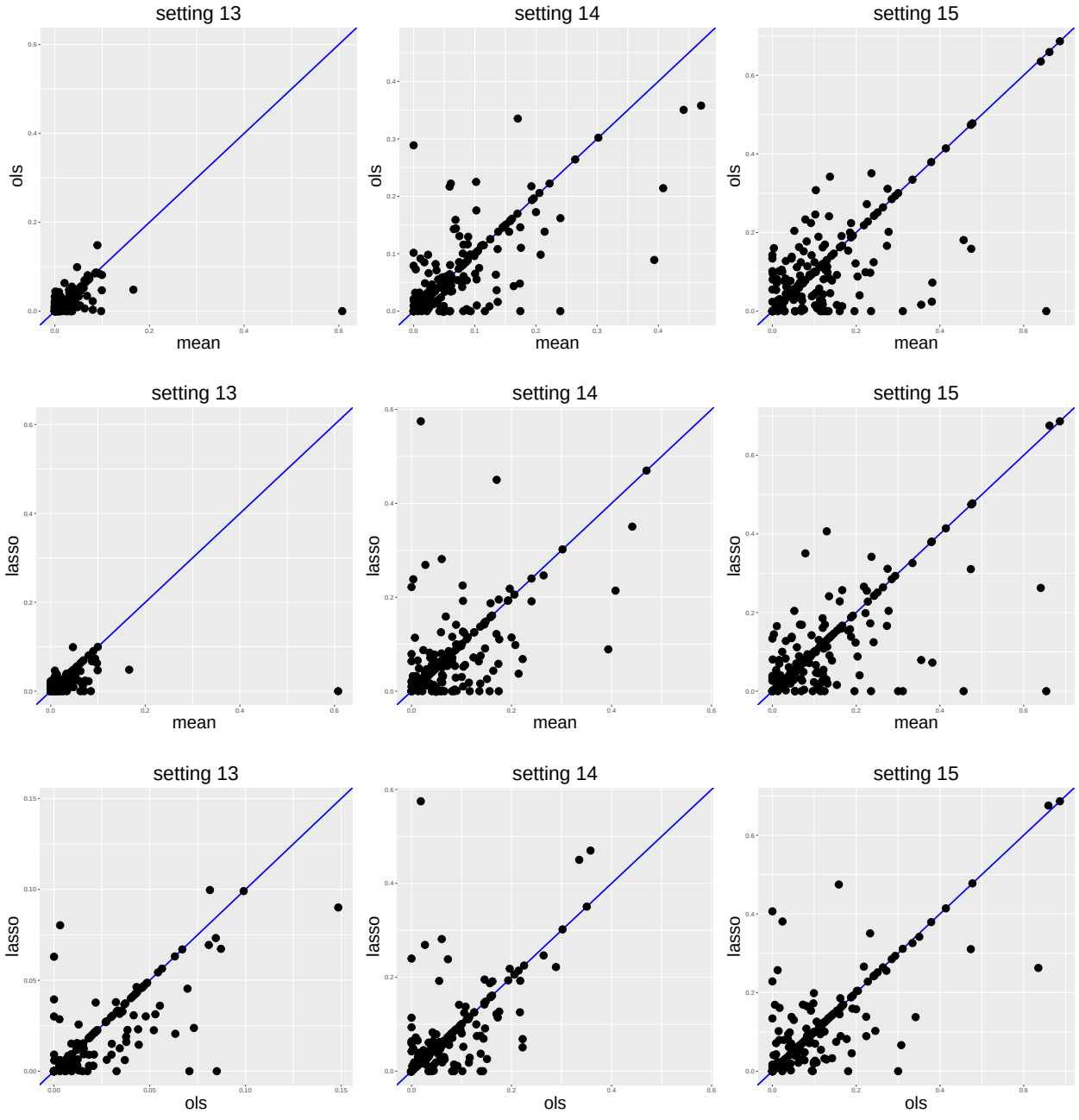


Figure C.1 ARI ratio comparisons for settings 13, 14 and 15.

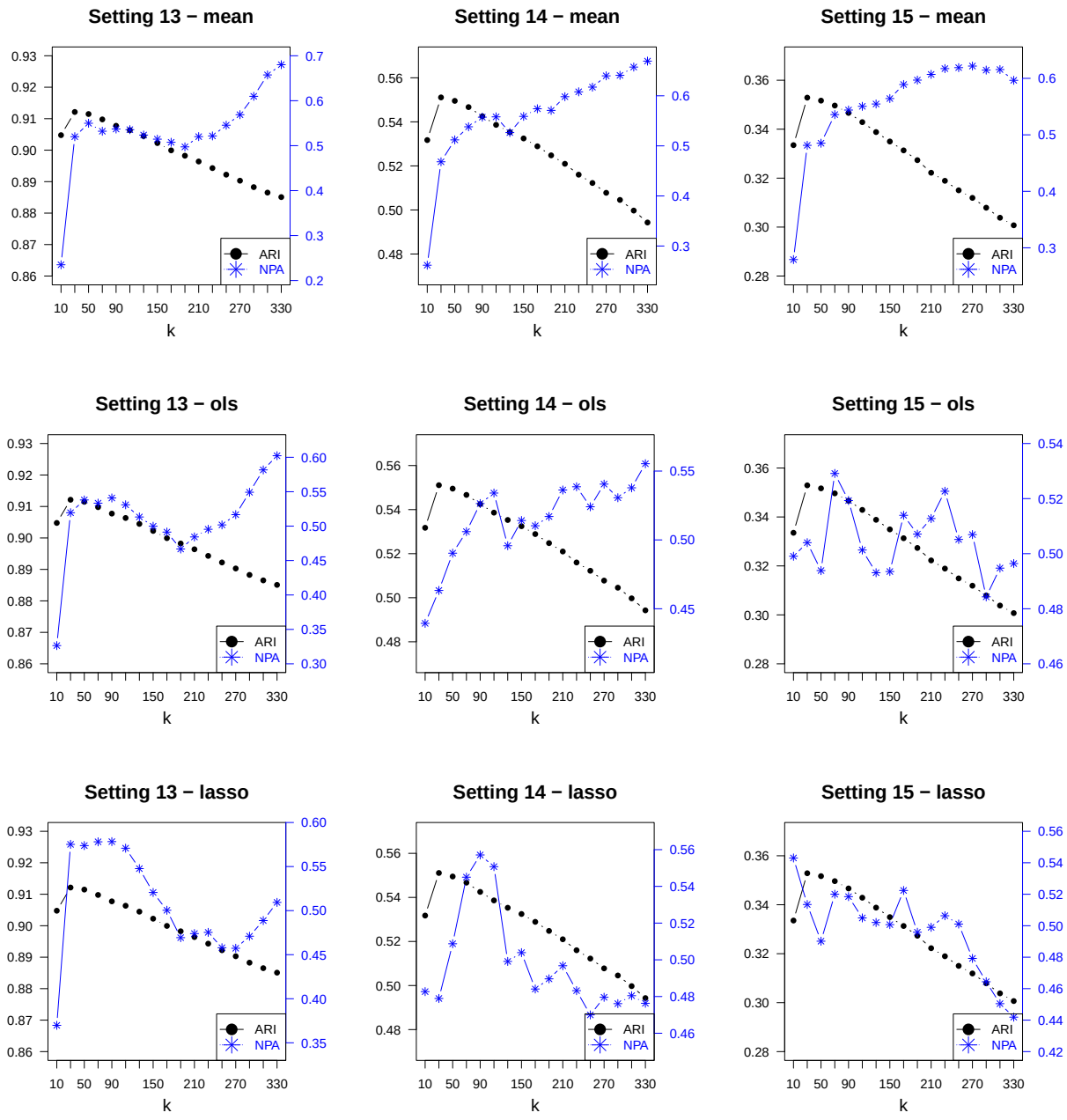


Figure C.2 ARI and NPA vs k for settings 13, 14 and 15.

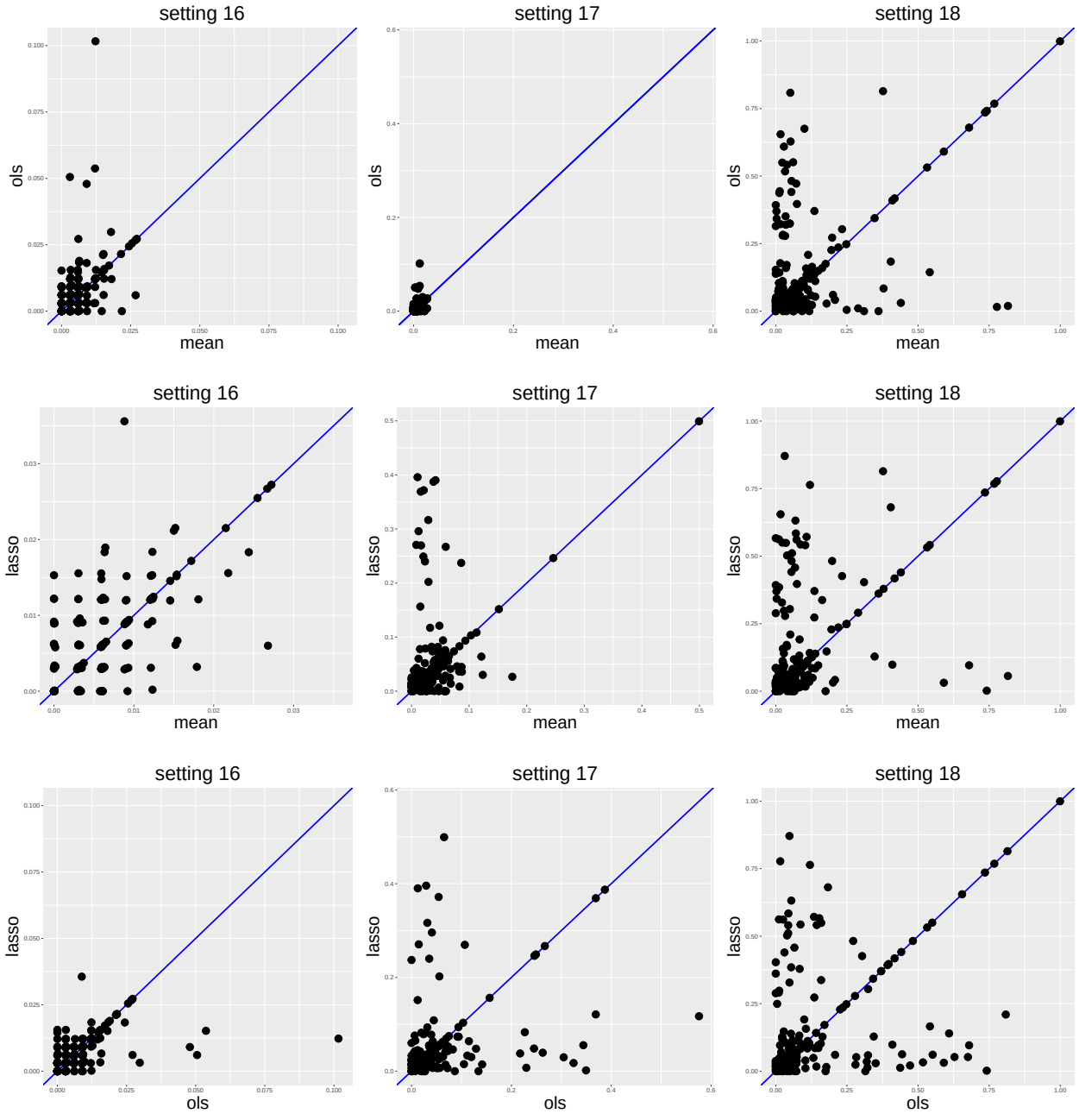


Figure C.3 ARI ratio comparisons for settings 16, 17 and 18.

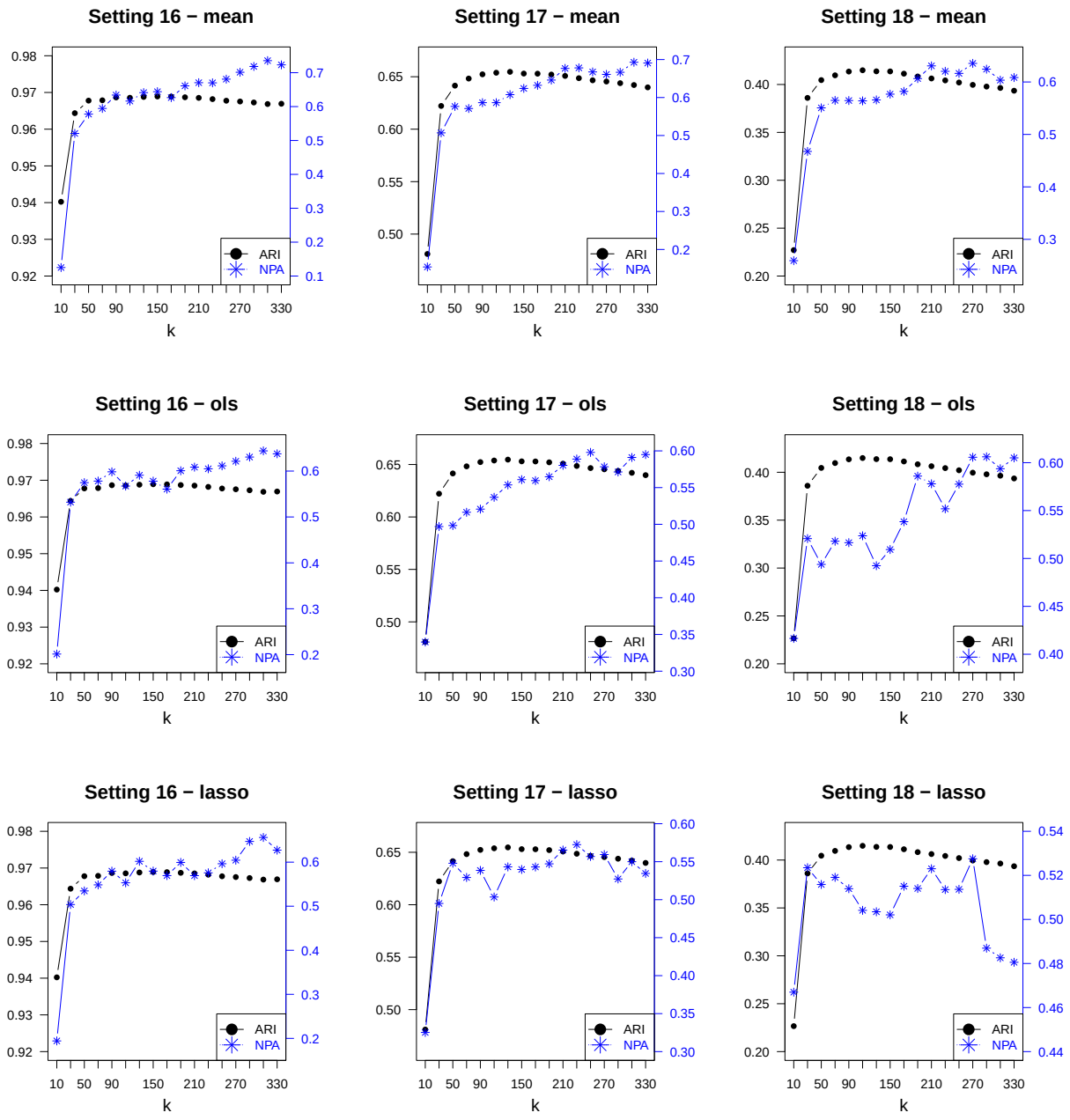


Figure C.4 ARI and NPA vs k for settings 16, 17 and 18.