Enhancing experimental signals in single-cell RNA-sequencing data using graph signal processing

Daniel B. Burkhardt[1,†], Jay S. Stanley III[3,†], Ana Luisa Pertigoto[4], Scott A. Gigante[1,2,3], Kevan C. Herold[4], Guy Wolf[5‡], Antonio Giraldez[1,‡], David van Dijk[1,2,‡*], Smita Krishnaswamy[1,2,‡*]

[1]Department of Genetics; [2]Department of Computer Science;
[3]Computational Biology & Bioinformatics Program;
[4]Departments of Immunobiology and Internal Medicine; Yale University, New Haven, CT, USA
[5]Department of Mathematics and Statistics, Université de Montréal, Montreal, QC, Canada

[*]Corresponding authors. E-mail: smita.krishnaswamy@yale.edu, david.vandijk@yale.edu
[†] These authors contributed equally. [‡] These authors contributed equally.

Abstract

Quantifying the differences in gene expression and cellular composition between single-cell RNA-sequencing (scRNA-seq) data sets presents an analytical challenge due to biological and technical noise. To facilitate analysis of multiple scRNA-seq experiments, we developed MELD: Manifold Enhancement of Latent Dimensions. MELD leverages tools from graph signal processing to learn the underlying, or latent, dimension within the data corresponding to the differences between data sets. We call this dimension the Enhanced Experimental Signal (EES). MELD learns the EES by filtering the categorical experimental label in the graph frequency domain to recover a smooth signal with continuous values. We also present a novel clustering algorithm combing the graph Fourier transform with the EES to identify cells that are transcriptionally similar and exhibit uniform response to a perturbation via an adapted vertex frequency clustering. Together, these methods can be used to identify signature genes that vary strongly between conditions, characterize clusters of cells with similar responses to the experiment, and quantify the degree to which each cluster is affected by a given perturbation. We demonstrate the advantages of MELD analysis using a combination of biological and synthetic data sets. MELD is implemented in Python and code is available at https://github.com/KrishnaswamyLab/MELD.

1 Introduction

As single-cell RNA-sequencing (scRNA-seq) has become more accessible, design of single cell experiments has become increasingly complex. Moving beyond profiling of cellular heterogeneity under a single condition, researchers are beginning to use scRNA-seq to compare cellular states across multiple experimental conditions. These experiments are powerful scientific tools because each assay generates thousands of independent measurements of gene expression per condition. However, quantifying the differences between single cell data sets presents an analytical challenge. There is often a large overlap between single-cell profiles across conditions due to subtle effect size, incomplete experimental penetrance, and the presence of shared cell types across conditions. Furthermore, single cell data sets are prone to biological and technical noise due to transcriptional heterogeneity and inefficient capture of mRNA in single cells. As a result, the signal of an experimental perturbation is small with respect to the biological and technical variation in an experiment.

Although several methods exist for merging single-cell data sets[1,2], identifying cell types[3], and quantifying differential expression between experimental conditions[4–6], to the best of our knowledge there is only one method, ClusterMap, specifically designed to quantify differences across single-cell data sets[7]. Most published analyses of multiple scRNA-seq samples, including ClusterMap, follow the same basic steps[4,7–13]. First, data sets are merged applying either batch normalization[12,13] or a simple concatenation of data matri-

ces[4,7–11]. Next, clusters are identified by grouping either sets of cells or modules of genes. Finally, within each cluster, the cells from each condition are are used to calculate various measures. Commonly, the reported statistics are fold-change of cluster proportion across conditions and differential expression of genes or gene modules within each cluster. Although the recently described ClusterMap algorithm reverses the merging and clustering steps[7], the overwhelming trend is a reliance on clustering prior to comparison.

Clustering prior to sample comparison results in a key limitation: existing clustering algorithms for scRNA-seq are based on global expression variation within a data set and are blind to the effect of a perturbation on each cell. Because the granularity of a cluster (*i.e.* the number of cells in the cluster) is somewhat arbitrary, this means the resolution of the initial clustering may not correspond to the resolution of the perturbation response. For example, a cluster may combine multiple cell subtypes each of which exhibit varying responses to the experimental perturbation.

Instead, to quantify the differences between experimental conditions, it would be helpful to find groups of cells that are prototypical of experimental or control conditions in the single-cell population, even if they form small or rare groups. Thus, we effectively want a quantification (*i.e.* a score) of how prototypical each cell is of the control or experimental condition. Such a score would identify the cells and populations that are the most or least affected by an experimental perturbation. We term this score the *Enhanced Experimental Signal* (EES).

One way to derive such a score would be probabilistic. We could, for example, build a probabilistic model of the experimental perturbation, and then examine each cell and quantify the likelihood that it came from the experimental or control condition. However, cells exist in a continuum of states and a probabilistic approach would require modelling a high dimensional complex continuous probability distribution over the cellular state space. Such an approach would require making strong parametric assumptions and would be computationally intractable.

To avoid this, we sought a nonparametric approach beginning by modelling the cellular state space using graphs. Graphs have been applied in scRNA-seq analysis for visualization[14,15], imputation[16,17], and clustering[18,19]. Here we propose to use methods from graph signal processing (GSP) that, despite their proven strength in other domains, have not often been used in biomedical data analysis[20].

The key advantage of GSP is the access to a set of tools for processing *graph signals*, which are functions defined over the nodes in a graph. These tools are extended from classical signal processing and give access to many functions such as those used for filtering or frequency analysis. In our case, we want to infer an EES that characterizes how prototypical a given cell is of each experimental condition. For example, in a simple experiment with one experimental condition and one control condition, we would like the EES to be +1 or -1 for cells that are most likely to arise in the experimental or control condition, respectively, and 0 for cells equally likely to arise in either condition.

To derive this score, we start with the condition from which each cell is sampled and use this to define a *Raw Experimental Signal* (RES). In our simple two-sample example, the RES would be defined as -1 for cells from the control condition and +1 for cells in the experimental condition. However, due to the technical and biological challenges listed above, the RES is not a perfect measure of likelihood that a cell would be observed in one condition or another. We would like to derive a similar likelihood for transcriptionally similar cells, but cells that are adjacent on the cell similarity graph may have different RES values. The RES provides useful information about the experiment, but we would like to remove noise from the signal.

To derive a score of prototypicality from this raw signal, we developed MELD (Manifold Enhancement of Latent Dimensions). MELD low-pass filters or smooths the RES on the cell state graph and converts the categorical RES into continuous values which, for our simple case, vary smoothly between -1 and 1. These values represent the ideal EES we described above.

We show in the following sections that this framework of treating the experimental label as a graph signal and then filtering this signal has many useful properties for analysis of experimental conditions in scRNA-seq. First, it can be used as a measure of transcriptional response to a perturbation on a cell-by-cell basis. Second, it can be used to identify gene signatures of a perturbation by examining gene trends with the EES. Finally, we leverage this framework to develop a clustering algorithm that identifies populations of cells that are transcriptionally similar and exhibit uniform response to a perturbation. To demonstrate these advantages, we apply MELD to a variety of biological data sets, including T-cell receptor stimulation, mutations in the developing zebrafish embryo, undirected differentiation of human embryonic stem

cells, and a newly generated data set of interferon-gamma stimulation in human pancreatic islets. In each case, we demonstrate the ability of MELD to identify trends across experimental conditions, and identify instances where MELD improves over standard analytic techniques.

2 Results

2.1 Overview of MELD algorithm

The goal of the MELD algorithm is to use a manifold model of cellular states across experimental conditions to learn an *Enhanced Experimental Signal* (EES) that quantifies how prototypical each cell is of each experimental condition. This can be loosely thought of as the likelihood a given cell state was observed in each experiment. This is a challenging problem as scRNA-seq measurements produce data sets with roughly 20,000-30,000 gene measurements in tens of thousands of cells. However, the space of biologically possible cellular states is much smaller than the space of all possible combinations of gene expression. Hence, the set of possible gene expression states represent a small section of the ambient space. To learn this cellular space from the single cell gene expression profiles, we use the manifold assumption[21] to construct a simplified data model: a graph that preserves the salient features of the original data set. We then deduce the EES in a controlled way over this graph.

The MELD algorithm computes the EES using the following steps:

1. A graph is constructed over the single cell data set where the nodes are cells and edges exist between cells that have similar transcriptional profiles.

2. The experimental label of each cell, which indicates the sample origin of the cell, is modeled over the cell similarity graph as a discrete signal that we call the *Raw Experimental Signal (RES)*.

3. MELD filters biological and technical noise from the RES to infer the EES, which reflects how prototypical each cell is of each condition.

4. The EES is used to identify cell populations that are prototypical of each condition and to infer gene trends of the experimental perturbation.

The first step of the MELD algorithm is to create a cell similarity graph. There are many ways to construct such a graph; MELD is agnostic to the specific construction used. By default, MELD constructs a graph with edge weights between cells calculated as similarity*using a variant of the radial basis kernel called the α-decay kernel, first proposed by Moon et al.[15]. This can be interpreted as a smooth k-Nearest Neighbors (kNN) kernel. However, in cases where batch normalization is required, we first apply a variant of Mutual Nearest Neighbors (MNN) to merge the data sets[1].

Next, MELD uses the input experimental label to create the RES on the graph. For simple two-sample experimental cases, cells from the control condition are assigned a value of -1 and cells from the experimental signal are assigned +1 (**Fig. 1a**). For more complex cases, such as in a time course or a series of drug titrations, the raw signal can be defined ordinally as the stage or timepoint of collection or dosage of a drug. Alternatively, the RES can be defined as a multi-dimensional signal *e.g.* when representing treatment with multiple different drugs.

Although the RES contains useful information, it is also noisy because of variability in the experimental treatment, biological heterogenity, and inefficent mRNA capture from single cells. Cells with matching transcriptional profiles (*i.e.* neighbors in a cell similarity graph) are assumed to be in the same cellular state and thus are assummed to be affected similarly to the experimental perturbation (stimulation for example). However, neighboring cells often have different RES values. Thus, one type of noise in the RES is rapid fluctuation in values between cells that are proximal on the graph (**Fig. 1a**). This is also referred to as high frequency noise.

The analysis of a signal's frequency composition relies on the Fourier transform, a fundamental tool of signal processing. In classical signal processing, which focuses on regularly-structured data like audio or video, the Fourier Transform decomposes signals into a weighted sum of sines and cosines of varying periodicity. These sums are called Fourier bases, and they are useful because such bases can be used to reversibly decompose many functions. Once decomposed, it is possible to analyze and manipulate the frequency composition of signals. However, in contrast to the classical setting where signals are defined over a regularly structured space, such as time, the RES is defined over an graph, which

*Here, similarity is a mathematical measure that can be thought of as the inverse of a distance metric. Hence, as the distance between two cells increases, their similarity decreases.

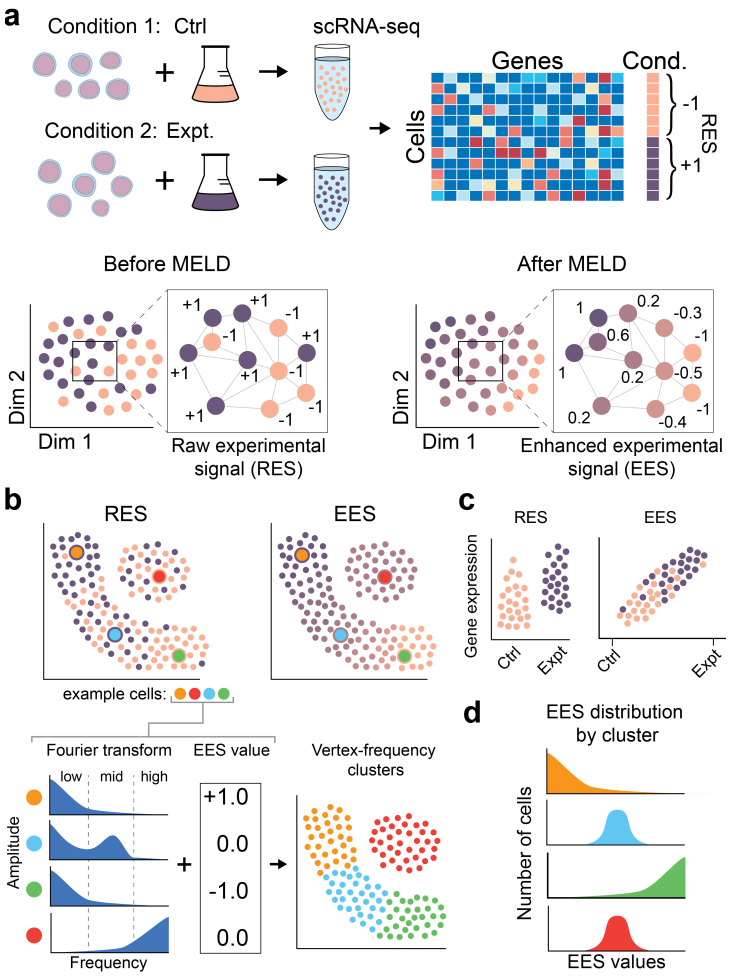

Figure 1: Overview of the MELD algorithm. (**a**) MELD quantifies the effect of an experimental perturbation by denoising the Raw Experimental Signal (RES) on the cell state graph to learn the Enhanced Experimental Signal (EES). (**b**) The Windowed Graph Fourier Transform and EES values at four example points shows distinct patterns between a transitional (blue) and unaffected (red) cell. This information is used for Vertex Frequency Clustering. (**c**) Ordering cells by the EES reveals gene expression changes of the experimental condition. (**d**) Examining the distribution of EES scores in vertex-frequency clusters identifies cell populations most affected by a perturbation.

can be irregular. This irregularity means that the classical Fourier transform cannot be used to analyze the RES or any other graph signals. To analyze the frequency content of signals defined over irregular data, such as the RES, we turn to the *graph Fourier transform.*

The graph Fourier transform is constructed via direct analogy to the classical Fourier transform[20]. On a cell similarity graph, the graph Fourier transform may be used to analyze the frequency content of the RES in terms of a weighted sum of the eigenvectors of the graph Laplacian, $\mathcal{L}$. These vectors, referred to as the *graph Fourier basis*, encode global trends in variation analogously to the periodic sine and cosine functions in the classical Fourier transform. We thus refer to an eigenvector as a graph frequency or harmonic, and the relative fluctuation or frequency of each eigenvector is described by its corresponding eigenvalue. In the graph Fourier transform of the RES, the weight (or Fourier coefficient) associated with each frequency describes the contribution of that eigenvector to the overall RES behavior. For example, rapid fluctuations across neighboring cells in the RES are represented by large magnitude Fourier coefficients for high frequency eigenvectors. These concepts are explained more thoroughly in Section 4.2, but the key notion here is that the graph Fourier transform provides information about the frequency composition of graph signals, such as the RES.

One assumption for recovering an EES is that it must be *smooth* (**Fig. 1b**). As before, this assumption regards high frequency Fourier coefficients as noise. Low frequency components are assumed to be the underlying signal trends (*i.e.* gradual changes from a node to its neighbors)[22]. Under this assumption, a low-pass filter, which removes high frequency components from a signal according to some cutoff frequency, is used to recover a latent signal. Here, we use latent to describe the underlying biological process(es) changing across experimental conditions. For example, in an experiment where T cells are treated with anti-CD3/CD28 beads, the latent signal corresponds to the level of activation of each cell.

One such low-pass filter that has seen success in the machine learning literature is *Laplacian regularization*[23–31]. This strategy, is expressed as the optimization

$$\mathbf{y} = \underset{\mathbf{z}}{\operatorname{argmin}} \underbrace{\|\mathbf{x} - \mathbf{z}\|_2^2}_{\text{a}} + \underbrace{\beta \mathbf{z}^T \mathcal{L} \mathbf{z}}_{\text{b}}, \tag{1}$$

Here, the regularization acts as a low-pass filter for input graph signals $\mathbf{x}$. The optimization over the variable $\mathbf{z}$ is broken into two parts: (a) reconstruction, calculated as the euclidean distance between $\mathbf{x}$ and $\mathbf{z}$; and (b) a *Laplacian quadratic form* that calculates the smoothness of $\mathbf{z}$ using the graph Laplacian $\mathcal{L}$. The quadratic form of the Laplacian is the canonical measure of signal smoothness on a graph. This property is made explicit by its equivalence with the *total variation*

$$\beta \mathbf{z}^T \mathcal{L} \mathbf{z} = \beta \sum_{i,j} W_{ij} (\mathbf{z}(i) - \mathbf{z}(j))^2. \tag{2}$$

The relation (2) illustrates the interpretation of the filter defined over the vertices, or nodes, of the graph, called the

vertex domain. This is an alternative representation to the signal defined over eigenvectors of the Laplacian, called the spectral domain. In the vertex domain, signal values at cell i and cell j are compared and weighted by the strength of the corresponding cells' connection on the graph (given by the weight matrix entry W_{ij}). To minimize this quantity, one must make $y(i) - y(j)$ small for points that are connected (*i.e.* $W_{ij} > 0$).

The Laplacian quadratic form emits a second interpretation as the *norm of the graph gradient*,

$$\beta \mathbf{z}^T \mathcal{L} \mathbf{z} = \beta \|\nabla_G \mathbf{z}\|_2^2 \,. \tag{3}$$

This interpretation reveals that the Laplacian regularization (1) is merely a minimization of the *graph gradient*, which is a first order derivative from point to point. For a given cell i, ∇_G yields a value for each of its neighbors that corresponds to the difference between cell i and its adjacent cells. The squared norm of this, $\|\nabla_G \mathbf{z}\|_2^2$, seeks to minimize the total energy in the derivative. Thus, when the derivative is small, i.e. changes between cells are small, then this term is small. These interpretations of the Laplacian regularization give vertex intuitions for the frequency behavior of (1).

However, low-pass filters are not a panacea. Indeed, low frequency noise (such as background noise) is common and will be exacerbated by low-pass filtering. In **Fig. S2a**, we illustrate such an example by blindly separating a medium frequency signal from a low frequency signal. Such a technique could be used to analyze both signals in a denoised setting if they originate from experimental design. On the other hand, basal processes like cell cycle can lead to low-frequency noise[32], thus this technique could be used to isolate a medium frequency signal from both high and low frequency corruption. In MELD we propose a new class of graph filters that is adaptable to graph and signal noise context, given by the following equation:

$$\mathbf{y} = \underset{\mathbf{z}}{\operatorname{argmin}} \|x - \mathbf{z}\|_2^2 + \mathbf{z}^T \mathcal{L}_* \mathbf{z} \tag{4}$$
$$\text{where } \mathcal{L}_* = [\beta \mathcal{L} - \alpha \mathbf{I}]^\rho \,.$$

To interpret this optimization, note that x corresponds to an input RES, y is an EES, $\mathcal{L}$ is a graph Laplacian, $\mathbf{I}$ is the identity matrix, and each of α, β, and ρ are parameters that control the spectral translation, reconstruction penalty, and filter order. These problems are typically solved by starting with $\hat{y} = x$. In contrast to previous works using Laplacian filters, these parameters allow analysis of signals that are contaminated by noise across the frequency spectrum. We address each of these in more detail in Section 4.3, where we analyze parameter choices in the spectral domain (see **Fig. S2**). Finally, we note that for the low-pass filter given by $\alpha = 0$ and $\rho = 1$, $\mathcal{L}_* = [\beta \mathcal{L} - \alpha \mathbf{I}]^\rho = \beta \mathcal{L}$. Thus, the canonical Laplacian regularization (1) is a subfilter of the MELD optimization (4).

Regardless of parameter selection for Equation 4, MELD proceeds by supplying the RES as x to obtain the EES y. The optimization in equation (4) may be solved in many ways; if many RES are to be examined, MELD solves the problem in terms of a matrix inversion which makes subsequent filters easy to apply. On the other hand, if only one signal is to be examined, MELD considers the corresponding spectral representation (see 4.2.2), and uses a Chebyshev polynomial approximation to quickly obtain the EES. With the EES, it is now possible to address a common problems in single-cell analysis, such as quantifying the effect on an experimental perturbation on gene expression.

2.2 The EES improves inference of perturbation effects on gene expression

Commonly, a researcher wants to know how gene expression changes between two experimental conditions or wants to identify a gene expression signature of a given process. When using the RES (i.e. directly comparing gene expression between samples), the data is organized categorically. This limits analysis to calculating summary statistics such as mean or variance of gene expression within each category. Furthermore, it is impossible to identify non-linear or non-monotonic changes in gene expression between two samples. One major advantage of applying MELD for analysis of experimental perturbations in scRNA-seq is that the EES is a quantitative vector that varies continuously and interpolates between the two discrete conditions. The cells that are most prototypical of each condition have the most extreme EES values and cells equally likely to be observed in either condition occupy the middle of the spectrum. The continuous nature of the EES makes it possible to order cells by EES values and identify continuous changes in gene expression between the most extreme cell states (**Fig. 1c**).

The EES effectively increases the resolution of the experimental data and enables the recovery of complex non-linear and non-monotonic trends in gene expression with the experimental condition. Even if only two conditions (such as an experiment and control) are measured, MELD can infer

which cells exhibit a weak or intermediate response to an experiment. This increased resolution provides the power to regress complex non-linear trends in expression relative to the EES. We demonstrate this on simulated data using only two samples (**Fig. S3**). In this simulated experiment we generate high-dimensional data emulating a biological transition between two terminal cell states through an intermediate transitional population. One of the genes in this simulation has peak expression in the intermediate cell state, but low expression in the terminal states. We show that directly comparing expression of this gene between samples using the RES would find no difference between samples. However the MELD-inferred EES reveals the true pattern of gene expression (**Fig. S3h**).

Beyond examining trends of single genes, one often wants to know which genes are the most strongly affected by an experimental perturbation. These strongly affected genes are often called the gene signature of an experiment. However, due to technical and biological noise in the experiment, simply calculating fold-change in expression between conditions often fails to recover capture meaningful changes in gene expression. A key advantage of the EES is that it provides a continuous measure of the experimental signal, which makes it possible to identify gene signatures by ranking genes by their statistical association with the EES (**Fig. 1c**). We previously developed kNN-DREMI (*k*-Nearest Neighbors conditional Density Resampled Estimate of Mutual Information)[33,34] to quantify such trends in scRNA-seq. To characterize signatures of an experiment, we can calculate the kNN-DREMI on all genes against the EES and rank them by their scores. For example, in Section 2.4, we use this approach to identify the gene signature of T cell activation and show that this signature is enriched for genes known to play a role in activation. It is also possible to quantify changes in expression by calculating fold-change only between the cells that are most prototypical of each condition. In Section 2.5, we take this approach to calculate fold-change in expression between cells with the top and bottom 20% of EES values and reveal specific responses within zebrafish cell types to Cas9 mutagenesis of chordin. We anticipate that using the EES to quantify gene signatures of an experiment will be a major use-case for MELD.

2.3 Vertex-frequency clustering identifies patterns of heterogeneity in high dimensional data

Another common goal for analysis of experimental scRNA-seq data is to identify subpopulations of cells that are responsive to the experimental treatment. Existing methods cluster cells by transcriptome alone and then attempt to quantify the degree to which these clusters are differentially represented in the two conditions. However, this is problematic because the granularity, or sizes, of these clusters may not correspond to the sizes of the cell populations that respond similarly to experimental treatment. Conveniently, GSP offers an approach to identifying clusters in scRNA-seq data sets that are transcriptionally similar and respond similarly to an experimental perturbation.

A naive approach to identify such clusters would be to simply concatenate the EES to the gene expression data as an additional feature and cluster on these combined features. However, we show that this would not correctly identify subpopulations with respect to their experimental response. Supplemental Figure **S4** provides a simulated case for this, generated using a Gaussian mixture model which separates two cell types along Dim 2 based on their responsiveness to a treatment on Dim 1. Traditional analysis may identify two clusters (based on the binary RES); alternatively, clustering based on *k*-means (**Fig. S4a**) and spectral clustering (**Fig. S4b**) revealed 4 clusters, and Louvain (**Fig. S4c**) returned 5 clusters. Each of these clusterings identify the pure populations resulting from the reservoirs of prototypical cells in condition 1 and condition 2 (which progress along Dim 1), but each fails to treat the non responsive population appropriately, breaking it into two or three pieces.

This inability to separate unaffected and transitioning cells has two fundamental causes. First, if one considers the cell similarity graph alone, the transitioning population of our simulation appears as a smearing of the two pure circles; k-means and methods that assume circular structure will thus partition the transition as parts of the reservoirs. Second, if one considers the EES, the purely mixed population has a similar value (0) as the transitioning population. Because of this, the two populations get clustered together.

However, we conjecture that in this example there are 4 meaningful clusters: two each given by the pure cells from condition 1 and 2, one that is a partition of the purely mixed cells (along Dim 2), and the final cluster is the transitioning population between the two pure reservoirs. While this is

merely an illustrative example, our biological analysis will show that analogous situations occur in real experiments.

As no contemporary method is suitable for finding this transitioning structure, we developed a method that uses the graph Fourier domain to cluster cells based on their latent geometry as well as their behavior under the EES (S4d). In particular, we cluster using local frequency profiles of the RES around each cell. This paradigm is motivated by the utility of analyzing cells based on different classes of heterogeneity. This method, which we call *vertex-frequency clustering*, is an adaptation of the signal-biased spectral clustering proposed by Shuman et al.[35].

Briefly, the method considers sums of many scales of spectrograms generated from the RES. Each spectrogram is obtained by translating a window function that considers neighbors of a specific scale support at each vertex, then taking the resulting Fourier transform of that windowed signal. The result of this operation is a vertex-frequency analysis matrix that is N-cells by N-frequencies. Each scale is then activated using a nonlinear transformation and summed with its previous activated scales. Finally, the summation is concatenated with the EES vector, and k-means is used to cluster the cells based on their multiscale vertex-frequency characteristics. Vertex-frequency clustering separates the value in the EES from its spectral characteristics and allows one to consider both the local spectra as well as the signal value. By considering both vertex and frequency information, one may distinguish between populations which are purely heterogeneous and populations which are in transition.

The algorithm briefly proposed above is discussed in further detail in methods Section 4.4. In particular, we detail a fast implementation using the recently proposed fast graph Fourier transform[36] and the diffusion operator. In the following sections, we demonstrate MELD filtering and vertex-frequency clustering on biological data.

2.4 MELD identifies a biologically relevant signature of T cell activation

To demonstrate the ability of MELD to identify a biologically relevant EES, we applied the algorithm to 5740 Jurkat T cells cultured for 10 days with and without anti-CD3/anti-CD28 antibodies published by Datlinger et al.[11] (**Fig. 2a**). The goal of the experiment was to characterize the transcriptional signature of T cell Receptor (TCR) activation. We selected this data because it relatively simple: the experiment profiles a single cell type, yet exhibits a heterogeneous continuum of experimental responses. We visualized the data using PHATE, a visualization and dimensionality reduction tool we developed for single-cell RNA-seq data (**Fig. 2b**)[15]. We observed a large degree of overlap in cell states between the experimental and control conditions, as noted in the original study[11]. This noise is both technical and biological. Approximately 76% of the cells were transfected with gRNAs targeting proteins in the TCR pathway, leading to some cells in the stimulated condition lacking key effectors of activation. The expectation for these cells is to appear transcriptionally unactivated despite originating from the stimulated experimental condition. In other words, although the RES for these cells is +1 (originating from the stimulated condition), the EES of these cells is expected to be closer to -1 (prototypical of the unstimulated condition).

To obtain a signature of T cell activation, Datlinger et al.[11] devised an *ad hoc* iterative clustering approach whereby cells were first clustered by the gRNA observed in that cell and then further clustered by the gene targeted. In each cluster, the median gene expression was calculated and the first principle component was used as the dimension of activation. The 165 genes with the highest component loadings were defined as signature genes and used to judge the level of activation in each cell. We reasoned that MELD would be able to identify an EES of TCR activation at single cell resolution without relying on clustering or access to information about the gRNA observed in each cell.

Applying MELD to the data, we observe a continuous spectrum of scores across the data set (**Fig. 2b**). As expected, the regions enriched for cells from the stimulated condition have higher EES values representing highly activated cells, and the converse is true for regions enriched for unstimulated cells. To ensure that the EES represents a biologically relevant dimension of activation, we looked for genes with a high mutual information with the EES using kNN-DREMI[16]. To facilitate comparison with the results of Datlinger et al.[11], we used EnrichR[37] to perform gene set enrichment analysis on the 165 genes with the top kNN-DREMI scores (**Fig. 2c,e**). We found comparable enrichment for gene sets related to T cell activation, T cell differentiation, and TCR response (**Fig. 2d**) and identify an overlap of 53 genes between the MELD-inferred and published signatures. We find that in the GO sets of T cell activation, T cell differentiation, and T cell receptor signalling, the MELD signatures includes as many or more genes for each GO term. Furthermore, our signature includes genes known to be affected by TCR stimulation

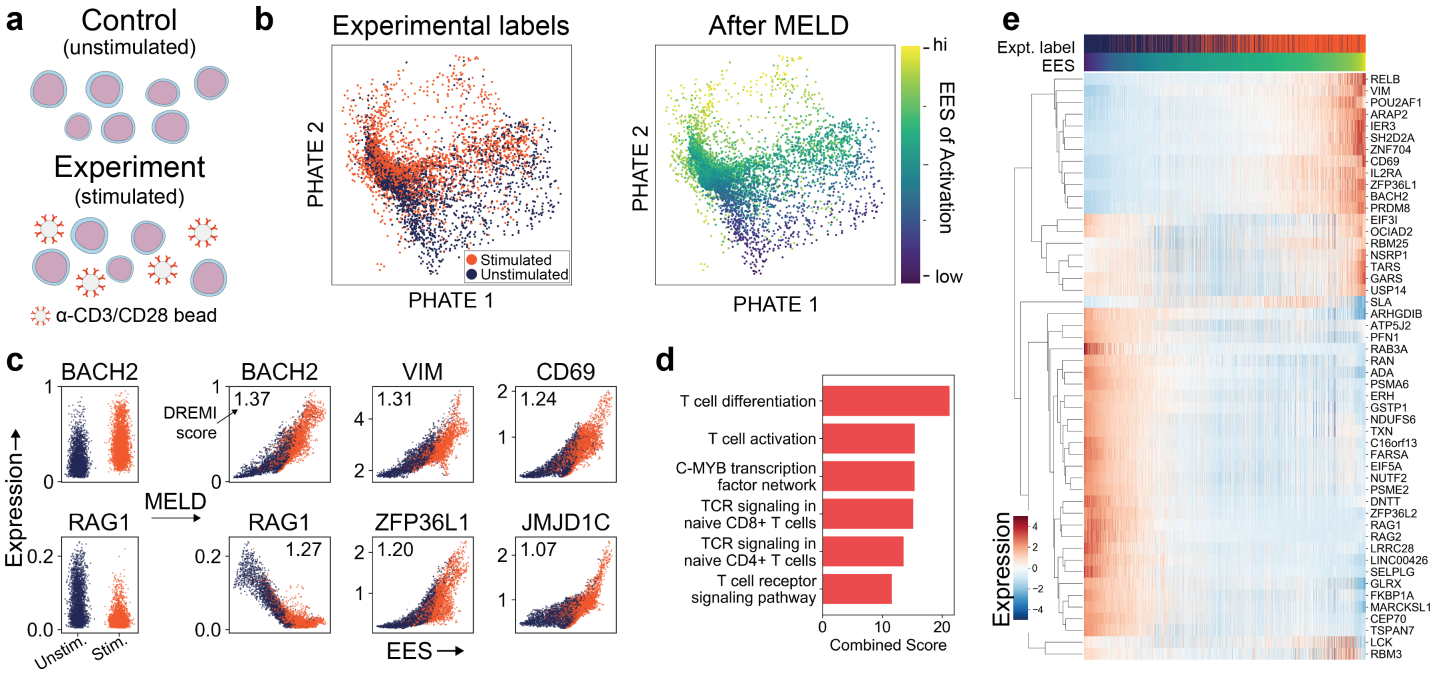

Figure 2: MELD recovers signature of TCR activation. (**a**) Jurkat T-cells were stimulated with α-CD3/CD28 coated beads for 10 days before collection for scRNA-seq. (**b**) Examining a PHATE plot, there is a large degree of overlap in cell state between experimental conditions. However, after MELD it is clear which cells states are prototypical of each experimental condition. (**c**) Relationship between gene expression and TCR activation state is revealed when cells are ordered by the EES instead of grouped by experimental condition. (**d**) Signature genes identified by top 1% of kNN-DREMI scores are enriched for annotations related to TCR activation. (**e**) Z-scored expression of select signature genes ordered by the EES reveals patterns of up- and downregulation. Notice a subset of genes exhibit non-monotonic expression patterns, such as USP14 and NSRP1. Identifying such trends would be impossible without MELD.

but not present in the Datlinger et al.[11] signature list, such as down regulation of RAG1 and RAG2[38]. These results show that MELD is capable of identifying a biologically relevant dimension of T cell activation at the resolution of single cells.

2.5 Characterizing genetic loss-of-function mutations in the developing zebrafish

To demonstrate the utility of GSP in the analysis of complex data sets composed of multiple cell types, we applied MELD to a recently published chordin loss-of-function experiment in zebrafish using CRISPR/Cas9 (**Fig. 3**)[12]. In this system, loss of chordin function results in a ventralization phenotype characterized by expansion of the ventral mesodermal tissues at the expense of the dorsally-derived neural tissues[39–41]. In Wagner et al.[12], zebrafish embryos were injected at the 1-cell stage with Cas9 and gRNAs targeting either chordin (*chd*), a BMP-antagonist required for developmental patterning, or tyrosinase (*tyr*), a control gene required for pigmentation but not expected to affect cell composition at these stages. Embryos were collected for scRNA-seq at 14-16 hours-post-fertilization (hpf). Similar to the T cell data set above, we expect incomplete penetrance of the perturbation in this data set because not all cells in the experimental condition will share the same mutation.

To characterize the effect of chordin mutagenesis, Wagner et al.[12] projected cells from each sample onto 28 clusters obtained from a reference wild-type data set. Within each cluster, the fold-change of cells from the *tyr*-injected to *chd*-injected condition was calculated and MAST[4] was used to calculate differentially expressed genes. A drawback of this approach is the restriction of analysis of the experimental effect to clusters, instead of single cells. This means that there is no way to detect divergent responses across subpopulations within clusters. Here, we demonstrate the ability of MELD to detect such occurrences and show how VF clustering detects groups of cells with similar responses to an experimental perturbation.

First, we used MELD to derive an EES of response to chordin loss-of-function. Here, cells with high EES values correspond to cells prototypical of the *chd* samples and

low EES values correspond to cells prototypical of the *tyr* samples (**Fig. 3a**). To identify the effect of mutagenesis on various cell populations, we first examined the distribution of EES scores across the 28 cell state clusters generated by Wagner et al.[12] for this data set (**Fig. 3b**). As expected, we found that Mesoderm – Lateral Plate (MLP), Tailbud – Presomitic Mesoderm (TPM), Hatching Gland (HG), and Mesoderm – Blood Island (MBI) had the highest average EES values, matching the observed expansion of the mesoderm and blood tissues in the embryos injected with *chd* gRNAs[12]. The cells with the lowest EES values were the Optic Primordium (OP), Differentiating Neurons (DN), Neural – Diencephalon (NDI), and Notochord (NTC). This is interpreted as finding these tissues in a *tyr* embryo, but not in a *chd* embryo, matching observed deficiencies of these tissues in the absence of chordin[39–41]. These results confirm that MELD is able to identify the effect of experimental perturbations across many cell types.

2.6 VF clustering identifies subpopulations in the Tailbud - Presomitic Mesoderm cluster

An advantage of using MELD is the ability to examine the distribution of scores within a cluster to understand the range of responses. In analyzing the chordin loss-of-function experiment, we observed that the Tailbud – Presomitic Mesoderm (TPM) cluster exhibited the largest range of EES values. This large range suggests that there are cells in this cluster with many different responses to *chd* mutatgenesis. To investigate this effect further, we generated a PHATE plot of the cluster (**Fig. 3c**). In this visualization, we observed many different branches of cell states each with varying ranges of MELD scores. We used vertex-frequency clustering to identify clusters of cells that are transcriptionally similar and exhibit a homogeneous response to perturbation (**Fig. 3d**).

We identified four subclusters within the PSM cluster. Using established markers[13], we identified these clusters as immature adaxial cells, mature adaxial cells, the presomitic mesoderm, and forming somites (**Fig. 3c, S5**). Examining the

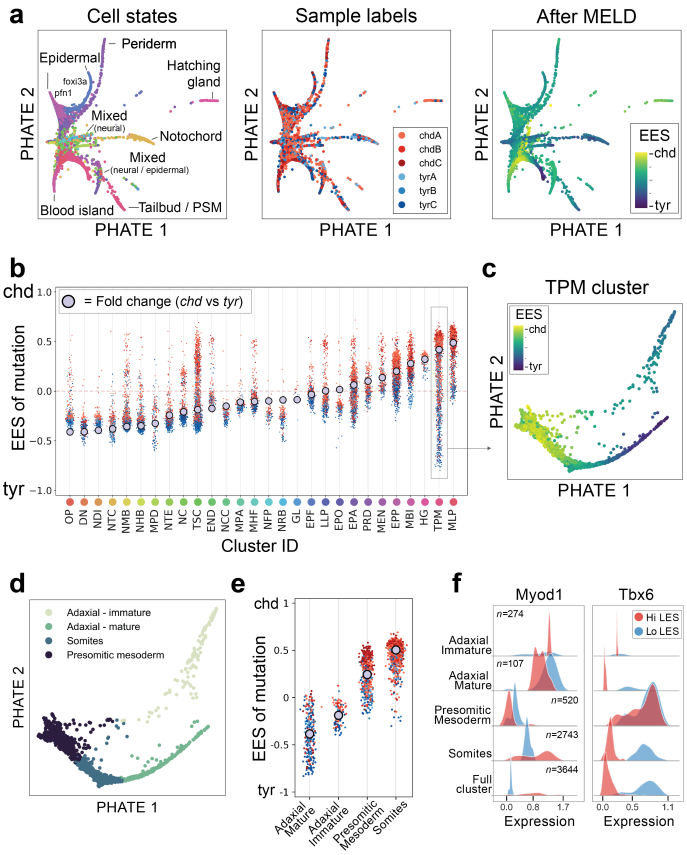

Figure 3: Characterizing chordin Cas9 mutagenesis with MELD. (**a**) PHATE shows a high degree of overlap of sample labels across cell types. Applying MELD to the mutagenesis vector reveals regions of cell states enriched in the *chd* or *tyr* conditions. (**b**) Using published cluster assignments[†], we show that the EES quantifies the effect of the experimental perturbation on each cell, providing more information than calculating fold-change in the number of cells between conditions in each cluster (grey dot), as was done in the published analysis. Color of each point corresponds to the sample labels in panel (a). Generally, average EES value aligns with the fold-change metric. However, we can identify clusters, such as the TPM or TSC, with large ranges of EES values indicating non-uniform response to the perturbation. (**c**) Visualizing the TPM cluster using PHATE, we observe several cell states with mostly non-overlapping EES values. (**d**) Vertex Frequency Clustering identifies four cell types in the TPM. (**e**) We see the range of EES values in the TPM cluster is due to subpopulations with divergent responses to the *chd* perturbation. (**f**) Changes in gene expression within subclusters is lost when only considering the full cluster, as was done in the published analysis.

distribution of EES scores within each cell type, we conclude that the large range of EES values within the TPM cluster is due to largely non-overlapping distributions of scores within each of these subpopulations (**Fig. 3e**). The mature and immature adaxial cells, which are muscle precursors, have low EES values. This indicates depletion of these cells in the *chd* condition which matches observed depletion of myotomal cells in chordin mutants[39]. Conversely, the presomitic meso-

[†]Abbreviations: MLP: Lateral plate, TPM: Tailbud - Presomitic mesoderm, HG: Hatching gland, MBI: Blood island, EPP: Epidermal - pfn1, MEN: Endothelial, PRD: Periderm, EPA: Epidermal anterior, EPO: Otic placode, LLP: Lateral line, EPF: Epidermal - foxi3a, GL: Germline, NRB: Rohon beard, NFP: Floorplate, MHF: Heart field, MPA: Pharyngeal arch, NCC: Neural crest - crestin, END: Endoderm, TSC: Tailbud - spinal cord, NC: Neural crest, NTE: Telencephalon, MPD: Pronephric duct, NHB: Hindbrain, NMB: Midbrain, NTC: Notocord, NDI: Diencephalon, DN: Neurons, OP: Optic

derm and forming somites have high EES values, indicating that these cells are prototypically enriched in a chordin mutant. Indeed, expansion of these presomitic tissues is observed in siblings of the *chd* embryos[12].

Another advantage of vertex-frequency clustering is that we can now calculate differential expression of genes within these populations of cells that we infer have homogeneous responses to a perturbation. Examining the distribution of genes within each of the identified subclusters, we find different trends in expression within each group (**Fig. 3f**). For example, Myod1, a marker of adaxial cells, is lowly expressed in the presomitic mesoderm and in the somites, but highly expressed in the adaxial cells. Attempting to compare the difference in expression of this gene in the entire cluster would be obfuscated by differences in abundance of each cell subpopulation between samples. We find a similar trend with Tbx6, a marker of the presomitic mesoderm, which is not expressed in adaxial cells and mature somites (**Fig. 3f**). Note that if we had merely compared the fold-change in abundance in the *chd* vs *tyr* conditions, as was done in the published analysis, we would have completely missed this effect and instead only observed that there is a 2-fold change in abundance of this cluster between samples. These results demonstrate the advantage of using MELD and vertex frequency clustering to quantify the effect of genetic loss-of-function perturbations in a complex system with many cell types.

2.7 MELD identifies a dimension corresponding to latent developmental time

Next, we applied MELD to enhance the experimental signals of time course data. Because it is not currently possible to measure the whole transcriptome of single cells continuously through development, several strategies exist to determine putative orderings of cells from snapshots of gene expression. This ordering is often called pseudotime. However, most existing methods learn pseudotime by identifying a trajectory through the data, then ordering cells along the trajectory[43–47]. Indeed, MELD does not attempt to learn a trajectory or set of trajectories through the data. MELD is agnostic to the number of beginning or end points or branches or even the existence of trajectory structures in the data. Instead, the goal of learning the EES in time courses is simply to infer an ordering of cells by latent developmental time. These orderings are useful for understanding how gene expression or cell type composition changes over time.

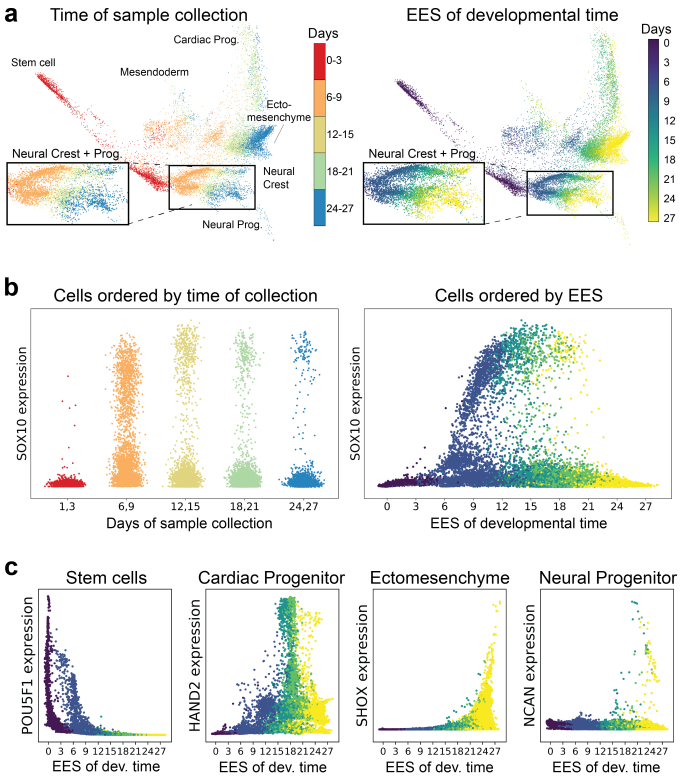

Figure 4: MELD captures latent developmental time. (**a**) PHATE visualization of a 27-day time course of human stem cells grown as embryoid bodies (EBs) colored by time of sample collection (left) or the EES (right). (**b**) Ordering cells by the EES reveals temporal trends in gene expression that are not apparent in the raw gene expression or after MAGIC[16]. (**c**) Examining expression of marker genes for various stem and progenitor populations, we observe a concordance between the EES values and experimental days (*i.e.* expression of POU5F1 in cells with EES values of 0-10 matches observed expression of this gene from 0-10 days of EB culture[42].)

To determine the efficacy of MELD to identify a biologically accurate temporal ordering of cells, we applied MELD to 16,825 cells captured over a 27-day time course of human embryonic stem cell differentiation as embryoid bodies (EBs, **Fig. 4**)[15]. Here, the EES corresponds to the latent developmental time scaled between 0 and 27, the time period of the experiment in days. To test if these EES time values represent a biologically relevant ordering of cells, we imputed gene expression using MAGIC[16] and examined the trends of marker genes with the EES (**Fig. 4b**). We found that the ordering of gene expression during EB differentiation for individual genes matches previously reported expression patterns. In fact, we find that the EES values matches real time for expression of many genes. For example, we observed POU5F1/OCT4 expression in cells with EES values between 0 and 12 days matching reported expression of this gene in human EBs for at least 10 days (**Fig. 4b**)[42]. Additionally, the

expression of NC markers like SOX10 and FOXD3 has been described in EBs starting at 10 days, matching observed up-regulation of these genes in cells with EES values between 10 and 22[42]. Furthermore, using the EES we observed the patterning of the mesendoderm and subsequent cardiac progenitors with mesendoderm markers such as FOXA2 expression between EES days 6-18 in a subset of cells (data not shown) followed by peak expression of caridac progenitor marker HAND2 between EES days 18-20 (**Fig. 4c**). These results demonstrate the utility of MELD for the inference of latent developmental time.

2.8 Identifying the effect of IFNγ stimulation on pancreatic islet cells

Next we used MELD to characterize previously unknown biology in a newly generated data set of pancreatic islet cells grown in culture for 24 hours with and without interferon-gamma (IFNγ). We chose this system because of its relevance to auto-immune diseases of the pancreas such as Type I Diabetes mellitus (T1D). The pathogenesis of T1D is generally understood to be caused by T cell mediated destruction of beta cells in the pancreatic islets[48] and previous reports suggest that islet-infiltrating T cells secrete IFNγ during the onset of T1D[49]. It has also been described that IFNγ-expressing T cells mediate rejection of pancreatic islet allografts[50]. Previous studies have characterized the effect of these cytokines on pancreatic beta-cells using bulk RNA-sequencing[51], but no studies have addressed this system at single cell resolution.

To better understand the effect of immune cytokines on islet cells, we cultured islet cells from three donors for 24 hours with and without IFNγ and collected cells for scRNA-seq. After filtering, we retained 5,708 cells for further analysis. Examining the expression of marker genes for major cell types of the pancreas, we observed a noticeable batch effect associated with the donor, driven by the maximum expression of glucagon, insulin, and somatostatin in alpha, beta, and delta cells respectively (**Fig. S6a**). To correct for this difference while preserving the relevant differences between samples, we applied the MNN kernel described in Section 4.1.1 to merge cells from each donor. Examining PHATE plots after batch correction, we observed three distinct populations of cells corresponding to alpha, beta, and delta cells (**Fig. 5a**).

To quantify the effect of IFNγ treatment across these cell types, we first applied MELD to the RES of IFNγ treatment

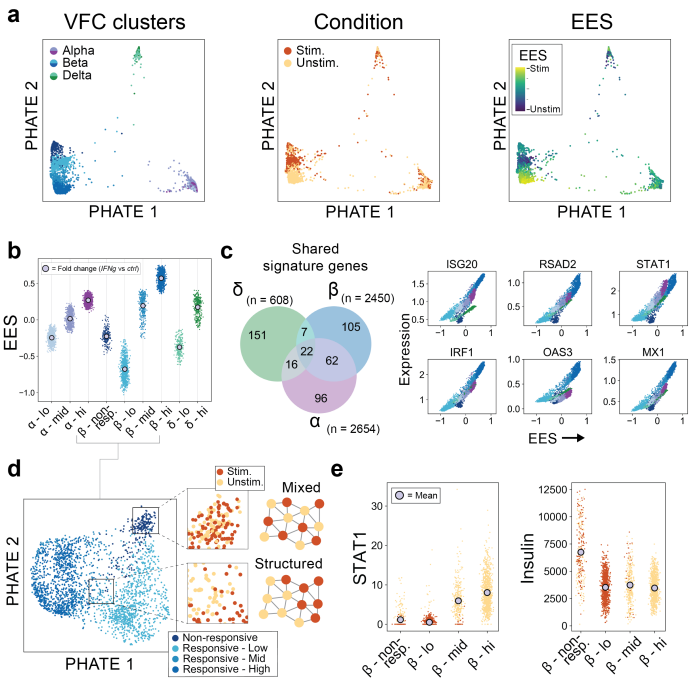

Figure 5: MELD characterizes the response to IFNγ in pancreatic islet cells. (**a**) PHATE visualization of pancreatic islet cells cultured for 24 hours with or without IFNγ. Vertex-frequency clustering identifies nine clusters corresponding to alpha, beta, and delta cells. (**b**) Examining the EES in each cluster, we observe that beta cells have a wider range of responses than alpha or delta cells. (**c**) We identify the signature of IFNγ stimulation by calculating kNN-DREMI scores of each gene with the EES. We find a high degree of overlap of the top 1% of genes by kNN-DREMI score between alpha and beta cells. (**d**) Examining the four beta cell clusters more closely, we observe two populations with intermediate EES values. These populations are differentiated by the structure of the RES in each cluster (outset). In the non-responsive cluster, the RES has very high frequency unlike the low frequency pattern in the transitional Responsive - mid cluster. (**e**) We find that the non-responsive cluster has low expression of IFNγ-regulated genes such as STAT1 despite containing roughly equal numbers of unstimulated (n=123) and stimulated cells (n=146). This cluster is marked by approximately 2.5-fold higher expression of insulin.

to calculate the EES of IFNγ stimulation(**Fig. 5a**). We then applied vertex-frequency clustering to identify nine subpopulations of cells. Using established marker genes of islet cells[52], we determined that these clusters correspond to alpha, beta, and delta cells (**Fig. 5a,b**, **Fig. S6b**). First, we sought to characterize the gene expression signature of IFNγ treatment across these cell types. Using kNN-DREMI[16] to identify genes with a strong association with the EES, we observe strong activation of genes in the JAK-STAT pathway including STAT1 and IRF1[53] and in the IFN-mediated antiviral response including MX1, OAS3, ISG20, and RSAD2[54–56] (**Fig. 5c**). The activation of both of these pathways has been previously reported in beta cells in response to IFNγ[57,58].

Furthermore, we observe a high degree of overlap in the IFNγ response between alpha and beta cells, but less between delta cells and either alpha or beta cells. Examining the genes with the top 1% of kNN-DREMI scores (n=196), we find 62 shared genes in the signatures of alpha and beta cells, but only 22 shared by alpha, beta, and delta cells. To confirm the validity of our gene signatures, we use EnrichR[37] to perform gene set enrichment analysis on the 196 signature genes and find strong enrichment for terms associated with interferon signalling pathways (**Fig. S6c**). From these results we conclude that although IFNγ leads to upregulation of the canonical signalling pathways in all three cell types, the responses to stimulation are subtly different between delta cells and alpha or beta cells.

We next examined the distribution of EES values within each of the clusters identified by vertex-frequency clustering (**Fig. 5b**). Interestingly, choosing k=9 clusters, we find two clusters of beta cells with intermediate EES values. These clusters are cleanly separated on the PHATE plot of all islet cells (**Fig. 5a**) and together represent the largest range of EES scores in the data set. To further inspect these clusters, we generated a new PHATE plot of the cells in the four beta cell clusters (**Fig. 5d**). Examining the distribution of RES values in these intermediate cell types, we find that one cluster, that we label as non-responsive, exhibits high frequency distribution of RES values indicative of a population of cells that does not respond to an experimental treatment (**Fig. 5d** - outset). The Responsive - mid cluster matches our characterization of a transitional population with a structured distribution of RES values. Supporting this characterization, we find a lack of upregulation in IFNγ-regulated genes such as STAT1 in the non-responsive cluster, similar to the cluster of beta cells with the lowest EES values **Fig. 5e**.

Seeking to understand the difference between the nonresponsive beta cells and the responsive populations, we calculated the Wasserstein distance between expression of genes in the non-responsive clusters and all others. The gene with the greatest difference in expression was insulin, the marker of beta cells, which was approximately 2.5-fold increased in the non-responsive cells (**Fig. 5e**). This cluster of cells bears resemblance to a recently described "extreme" population of beta cells that exhibit elevated insulin mRNA levels and are found to be more abundant in diabetic mice[59]. Given that these cells appear non-responsive to IFNγ stimulation and exhibit extreme expression of insulin suggests that the presence of abnormally high insulin in a beta cell prior to IFNγ exposure inhibits the IFNγ response pathway through an unknown mechanism. Confirming this hypothesis will require further experimental validation.

Here, we applied MELD to a new data set to identify the signature of IFNγ stimulation across alpha, beta, and delta cells. Furthermore, we used vertex frequency clustering to identify a population of beta cells with high insulin expression that appears unaffected by IFNγ stimulation. Together, these results demonstrate the utility of MELD analysis to reveal novel biological insights in a clinically-relevant biological experiment.

3 Discussion

When performing multiple scRNA-seq experiments in various experimental and control conditions, researchers often seek to characterize the cell types or groups of genes that change from one condition to another. However, quantifying these differences is a challenging task due to the subtlety of most biological effects relative to the biological and technical noise inherent to single cell data. To overcome this hurdle, we designed MELD to enhance the experimental signal in scRNA-seq data sets and to characterize the differences between samples.

MELD uses the framework of Graph Signal Processing to learn a signal over a cell similarity graph that indicates how prototypical that cell is of each experiment. In this context, the similarity graph is built from the gene expression profiles of all samples. Next, the label that indicates the sample origin of each cell is defined as a signal over the graph. This signal is called the Raw Experimental Signal (RES). MELD filters this signal to remove biological and technical noise to infer the Enhanced Experimental Signal (EES). The EES can be used to identify individual cells that are the most prototypical of each sample and the individual patterns in gene expression that are associated with these changes. The EES can also be used to identify groups of cells that do not change between experimental conditions. Moreover, we demonstrate that using the EES, it is possible to identify non-linear and non-monotonic changes in gene expression that would be lost through a direct comparison of expression between two samples. These benefits can be applied to arbitrary experimental designs, as long as the categorical condition labels can be ordered on a number line (e.g. dosage of treatment, time of collection, biological sex).

Existing strategies for quantifying the effect on an experi-

mental perturbation generally focus on clustering cells based on gene expression alone, then calculating statistics, such as differential expression within clusters or fold-change in cells from each cluster between samples[4,7–13]. However, we show in Section 2.5 that this approach can fail to identify the divergent responses of subpopulations of cells within a cluster. To identify clusters of cells with cohesive responses to a perturbation, we introduce a novel clustering algorithm, called vertex-frequency clustering. Using the raw and enhanced experimental signals, we derive clusters of cells that are transcriptionally similar and exhibit uniform response to an experimental perturbation. We show that this strategy is capable of differentiating between groups of cells that exhibit intermediate responses to a perturbation and cells that are transitioning to a different state as a result of a perturbation.

We demonstrate the advantages of MELD analysis across synthetic and biological data sets. When analyzing the effect of T cell receptor stimulation, we derive a biological signature of T cell activation, and identify non-monotonic gene trends that would be hidden by direct comparison of expression between conditions. Next, we use MELD to quantify the effect of CRISPR/Cas9 mutagenesis in a single-cell experiment in the zebrafish embryo. Here, we demonstrate that the EES provides deeper insight into the effect of chordin loss-of-function than the published analysis. We identify subpopulations with the Tailbud – Presomitic Mesoderm cluster that each have divergent responses to the mutation. We further demonstrated the applicability of MELD to quantify latent developmental time in a time course of stem cell differentiation. Finally, we presented a new data set of pancreatic islet cells with and without stimulation with interferon-gamma. Here, we quantified the degree to which canonical islet alpha, beta, and delta cell populations respond to stimulation and found the response more similar between alpha and beta cells than delta. Furthermore, we identified a subpopulation of beta cells marked by extremely high insulin expression that appears to be unaffected by the experimental IFNγ stimulus. Together, these results demonstrate the utility of MELD to characterize diverse biological phenomenon. MELD analysis is a powerful tool for quantifying scRNA-seq experiments and generating new hypotheses from single-cell data sets.

The flexibility of MELD to analyze arbitrary signals over a cell similarity graph suggest several future applications in scRNA-seq analysis. For example, in **Fig. S2** we demonstrate the ability of MELD to extract convoluted signals of different frequencies on a graph. These two signals might represent cell cycle effect, experimental signal, and technical noise. By tracking genes that vary with cell cycle, for example, we could remove this trend from the experiment to improve the identification of gene signatures of an experimental perturbation. Another potential application of MELD is the comparison of multiple experimental meta-variables. One can imagine an experiment where cells are exposed to combinations of drugs in varying concentrations with the goal of understanding how these combinations of drugs interact. By building a unified cell similarity graph across conditions, one could deconvolve the signals of each component of the treatment and then calculate a measure of association, such as mutual information, to identify which drugs elicit similar or divergent effects alone or in combination. This flexibility makes MELD an ideal analytical tool for scRNA-seq experiments across biological systems.

4 Algorithm

Previous works have attempted to uncover a *pseudotime* dimension for single cell data using various methods for cell ordering[21,43,46,60]. Others have attempted to denoise or visualize data by diffusion[15,16]. A common theme amongst these methods is the abstraction of single cell data into a similarity graph, which encodes the relationships structure of cells in an experiment. MELD builds upon this framework by using the spectrum of the *graph Laplacian* to extract latent features in single cell data.

4.1 Preliminaries and Graph Construction

A graph $\mathcal{G} = (V, E)$ is a mathematical abstraction consisting of vertices $v_i \in V$ and edges $(i, j) \in E : v_i, v_j \in V$. Graphs are a flexible tool for modelling structures irregular and regular; graphs can model structures from logic, mathematical groups, and infinite geometries, to circuits, social networks, and neuronal geometry. In MELD, we use weighted, undirected graphs defined by choosing single cells as vertices.

On the weighted, undirected graphs that model single-cells the edge set E is replaced by a symmetric weight matrix W. The entry W_{ij} encodes the weighted similarity between cell v_i and v_j, determined by a comparison function between the corresponding vectors $\mathbf{x_i}$ and $\mathbf{x_j}$ from the data set X. Many choices have been suggested for determining W_{ij} (e.g., Mateos et al.[61]). MELD can be run on graphs for any choice of W, although graph geometry is vital to the algorithm. To

construct a graph from input data we use an affinity kernel, which encodes similarities or affinities between data points, to determine edge weights. In most cases, we use an adaptive alpha-decaying kernel proposed by Moon et al.[15] for this purpose. However, in cases where batch, density, and technical artifacts confound graph construction, we also use anisotropic and mutual nearest neighbor kernels. These constructions are discussed in further detail in section 4.1.1.

Finally, we introduce a few additional key graph concepts used in MELD. First, D is the *degree* matrix. This matrix contains the total connectivity of every node as described by the degree $d(i) = \sum_j^N W_{ij}$. D is a diagonal matrix, where D_{ii} is the degree $d(i)$ of the vertex v_i. From the degree we define the *graph Laplacian*,

$$\mathcal{L} := D - W. \tag{5}$$

In MELD, we use the graph Laplacian to analyze and process *graph signals*, which for a graph with N vertices are functions that map from vertices to real values, i.e., $f : V \mapsto \mathbb{R}$. By abuse of notation, we interchange this functional notion of a signal with its discretized realization as an N-dimensional vector $\mathbf{f} \in \mathbb{R}^N$.

4.1.1 Kernel Selection

The graph construction at the core of MELD relies on a quantitative notion of neighborhoods in the data, which are encoded by a symmetric nonnegative kernel function $k(x, y)$. Such kernel functions are often used in manifold learning approaches (see Moon et al.[21] and references therein) to capture intrinsic data geometries that approximate underlying manifold models from the data. A wide variety of kernels have been proposed over the years for the task of formulating appropriate kernels for capturing meaningful notions of locality and data neighborhoods. We refer the readers to Coifman and Lafon[62], Coifman and Hirn[63], Berry and Sauer[64], Bermanis et al.[65,66], Marshall and Coifman[67], Marshall and Hirn[68], Lindenbaum et al.[69] and Lindenbaum et al.[70] for relevant examples and discussions.

In MELD, we mainly use the kernel proposed in Moon et al.[15], defined as

$$K_{k,\alpha}(x,y) = \frac{1}{2}\exp\left(-\left(\frac{\|x-y\|_2}{\varepsilon_k(x)}\right)^{\alpha}\right) + \frac{1}{2}\exp\left(-\left(\frac{\|x-y\|_2}{\varepsilon_k(y)}\right)^{\alpha}\right), \tag{6}$$

where x, y are data points, $\varepsilon_k(x), \varepsilon_k(x)$ are the distance from x, y to their k-th nearest neighbors (correspondingly), and $alpha$ is a parameter that controls the decay rate (i.e., locality) of the kernel. This construction generalizes the popular Gaussian kernel, which is typically used in manifold learning, but also has some disadvantages alleviated by the α-decaying kernel, as explained in Moon et al.[15].

While the kernel in (6) provides an effective way of capturing neighborhood structure in data, it is susceptible to batch effects. For example, when data is collected from multiple patients, subjects, or environments (generally referred to as "batches"), such batch effects can cause affinities within each batch are often much higher than between batches, thus artificially creating separation between them rather than follow the underlying biological state. To alleviate such effects, we adjust the kernel construction as follows when applied to multi-batch data. First, within each batch, the affinities are computed using (6). Then, across batches, we compute slightly modified affinities as

$$K'_{k,\alpha}(x,y) = \min\left\{\exp\left(-\left(\frac{\|x-y\|_2}{\varepsilon'_k(x)}\right)^{\alpha}\right), \exp\left(-\left(\frac{\|x-y\|_2}{\varepsilon'_k(y)}\right)^{\alpha}\right)\right\},$$

where $\varepsilon'_k(x)$ are now computed via the k-th nearest neighbor of x in the batch containing y (and vice versa for $\varepsilon'_k(y)$). Next, a rescaling factor γ_{xy} is computed such that

$$\sum_{z \in \text{batch}(y)} \gamma_{xy} K'_{k,\alpha}(x, z) \leq \beta \sum_{z \in \text{batch}(x)} K_{k,\alpha}(x, z)$$

for every x and y, where $\beta > 0$ is a user configurable parameter. This factor gives rise to the rescaled kernel

$$K'_{k,\alpha,\beta}(x, y) = \begin{cases} \gamma_{xy} K'_{k,\alpha}(x, y) & \text{if batch}(x) = \text{batch}(y) \\ K'_{k,\alpha}(x, y) & \text{otherwise.} \end{cases}$$

Finally, the full kernel is then computed as

$$K'_{k,\alpha}(x, y) = \min\left\{K'_{k,\alpha,\beta}(x, y), K'_{k,\alpha,\beta}(y, x)\right\},$$

and used to set the weight matrix for the constructed graph over the data. Notice that this construction is a well defined extension of (6), as it reduces back to that kernel when only a single batch exists in the data.

4.2 Graph Signal Processing Background

The graph Laplacian eigensystem $\mathcal{L} = \Psi\Lambda\Psi^{-1}$ with eigenvalues $\Lambda := \{0 = \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_N\}$ and corresponding eigenvectors $\Psi := \{\psi_i\}_{i=1}^N$ has been used with great success in spectral clustering[71], graph sparsification[72], and

dimensionality reduction[73].[‡] Recently, a number of works in the emergent field of *graph signal processing* (GSP) have shown that the Laplacian eigensystem is useful for analyzing, manipulating, and inferring data that resides on a graph[20]. Tools such as wavelet transforms[74], windowed Fourier transforms[35], and uncertainty principles[75] have been extended to graphs via analogy with the classical Fourier transform and eigenfunctions of the Laplacian. In MELD, we use graph signal processing to infer and enhance latent dimensions in scRNA-seq data. In the following sections, we provide an introduction to this field.

4.2.1 Classical Fourier

To begin, the graph Laplacian $\mathcal{L}$ introduced above is a discrete analog of the *Laplace operator* ∇^2. For some continuously differentiable real-valued function $f \in C^k(\mathbb{R}^N)$, the Laplacian, $\nabla^2 f$, is a scalar-valued function that yields a sum of the unmixed second partial derivatives around a point [§]. Recall from univariate Calculus that the second derivative $f''(x)$ for some function $f : \mathbb{R} \mapsto \mathbb{R}$ corresponds to the curvature of f at the point x. The Laplacian is a multivariate generalization of this notion, measuring the total curvature in all directions around a point.

The Laplace operator is found throughout physics and mathematics, where its uses include the heat equation, which is used for modelling diffusion, heat flow, and brownian motion. Solving the heat equation served as the catalyst for the genesis of modern Fourier analysis, as general solutions to the equation eluded mathematics for many years prior to 1807 when Joseph Fourier introduced the concept of spectral decompositions. Prior to Fourier, simple solutions to the heat equation were known if the initial heat source was a sine or a cosine. These solutions are called eigenfunctions - when the Laplace operator is applied to these functions, the result is the same function times a scalar eigenvalue. These eigenfunctions are the basis of the Fourier Transform.

In the Fourier Transform, arbitrary square integrable functions[¶] are decomposed into the orthonormal basis of periodic sines and cosines.[‖] These summations have various physical connections but the salient notion of the Fourier transform is a change of variables into a dual space that encodes the *frequency* of the function.

In audio, frequency encodes pitch. In images, frequencies describe edges and patterns. Common to all of these interpretations is that frequency is related to *smoothness*. We say that a function is *smooth* if one is unlikely to encounter a dramatic change in value across neighboring points. A simple way to imagine this is to look at the *zero-crossings* of a function. A fundamental example of this tool is found in one dimensional sin waves of various frequencies, i.e. $\sin ax$ for $a = 2^k, k \in \mathbb{N}$. For $k = 0$, the wave crosses the x-axis (a zero-crossing) when $x = \pi$. When we double the frequency at $k = 1$, our wave is now twice as likely to cross the zero and is thus less smooth than $k = 0$. This simple zero-crossing intuition for smoothness is relatively powerful, as we will see shortly.

4.2.2 The Graph Fourier Transform

Next, we'll show that our notions of smoothness and frequency are readily applicable to data that is not regularly structured, such as single-cell data. Before, we mentioned that the graph Laplacian $\mathcal{L}$ is a discrete analog of ∇^2. Let's make this transparent by deriving the graph Laplacian from first principles. For a graph $\mathcal{G}$ on N vertices, its graph Laplacian $\mathcal{L}$ and an arbitrary graph signal $\mathbf{f} \in \mathbb{R}^N$, we use equation (5) to write

$$\begin{aligned}(\mathcal{L}\,\mathbf{f})\,(i) &= ([D - W]\,\mathbf{f})\,(i) \\ &= d(i)\mathbf{f}(i) - \sum_j W_{ij}\mathbf{f}(j) \\ &= \sum_j W_{ij}\,(\mathbf{f}(i) - \mathbf{f}(j))\,. \end{aligned} \tag{7}$$

As the graph Laplacian is a weighted sum of differences of a function around a vertex, we may interpret it analogously to

[‡]Note that in this discussion we abuse notation by treating Λ as an ordered set of Laplacian eigenvalues and as the diagonal matrix with entries from the elements of this set. Similarly, Ψ is both the set of column eigenvectors $\{\psi_i\}_{i=1}^N$ as well as the $N \times N$ matrix $[\psi_1 \psi_2 \cdots \psi_N]$ with eigenvector as a column.

[§]The Laplace operator maps functions from C^k to C^{k-2}. Functions in C^k are *continuously differentiable* i.e. for any real valued function $f \in C^k(\mathbb{R}^N)$, the derivative $f'(x)$ is defined and differentiable.

[¶]*Square integrable* functions have a finite integral when taken over the real line. $L^2(\mathbb{R})$ is the set of all real-valued square integrable functions.

[‖]Various forms of the Fourier transform exist. In the continuous setting, it is typical to use Euler's identity to decompose functions using the complex exponential. For some function $f(t)$ we have its Fourier transform $\hat{f}(\xi) = \int_{-\infty}^{\infty} f(t) e^{-2\pi i t \xi} dt$.

its continuous counterpart as the curvature of a graph signal. Another common interpretation made explicit by derivation (7) is that $(\mathcal{L}\mathbf{f})(i)$ measures the *local variation* of a function at vertex i.

Local variation naturally leads to the notivation of *total variation*,

$$\mathbf{TV}(\mathbf{f}) = \sum_{i,j} W_{ij}(\mathbf{f}(i) - \mathbf{f}(j))^2,$$

which is effectively a sum of all local variations. $\mathbf{TV}(\mathbf{f})$ describes the global smoothness of the graph signal $\mathbf{f}$. In this setting, the more smooth a function is, the lower the value of the variation. This quantity is more fundamentally known as the *Laplacian quadratic form*,

$$\mathbf{f}^T \mathcal{L}\, \mathbf{f} = \sum_{i,j} W_{ij}(\mathbf{f}(i) - \mathbf{f}(j))^2. \tag{8}$$

Clearly, the graph Laplacian can be used as an operator and in a quadratic form to measure the smoothness of a function defined over a graph. One effective tool for analyzing operators is to examine their eigensystems. As the Laplacian is a square, symmetric matrix, the spectral theorem tells us that its eigenvectors $\Psi := \{\psi_i\}_{i=1}^N$ form an orthonormal basis for $\mathbb{R}^N$. Furthermore, the Courant-Fischer theorem establishes that the eigenvalues λ_i of $\mathcal{L}$ are local minima of $\mathbf{f}^T\mathcal{L}\mathbf{f}$ when $\mathbf{f}^T\mathbf{f} = 1$ and $\mathbf{f} \in U$ as $\dim(U) = i = 1, 2, \cdots N$. At each eigenvalue λ_i this function has $\mathbf{f} = \psi_i$. In summary, the eigenvectors of the graph Laplacian (1) are an orthonormal basis and (2) minimize the Laplacian quadratic form for a given dimension.

Henceforth, we use the term *graph Fourier basis* interchangeably with graph Laplacian eigenvectors, as this basis can be thought of as an extension of the classical Fourier modes to irregular domains. In particular, the ring graph eigenbasis is composed of sinusoidal eigenvectors, as they converge to discrete Fourier modes in one dimension. The graph Fourier basis thus allows one to define the *graph Fourier transform* (GFT) by direct analogy to the classical Fourier transform.

The GFT of a signal f is given by

$$\begin{aligned}\hat{f}(\lambda_\ell) &= \sum_i f(i)\psi_\ell^T(i) \\ &= \langle \mathbf{f}, \psi_\ell \rangle.\end{aligned}$$

We will also write this GFT as the matrix-vector product

$$\hat{\mathbf{f}} = \Psi^T \mathbf{f}. \tag{9}$$

As this transformation is unitary, the inverse graph Fourier transform (IGFT) is $\mathbf{f} = \Psi\hat{\mathbf{f}}$. Although the graph setting presents a new set of challenges for signal processing, many classical signal processing notions such as filterbanks and wavelets have been extended to graphs using the GFT. We use the GFT to process, analyze, and cluster experimental signals from single-cell data using a novel graph filter construction and a new harmonic clustering method.

4.3 The MELD Filter

In MELD, we seek to extract latent features from experimental signals. To do this, we employ a novel graph filter construction that can be used for denoising and deconvolution. To begin, we review the notion of filtering with focus on graphs, and demonstrate the filter in a low-pass setting. Next, we demonstrate the expanded version of the MELD filter and provide an analysis of its parameters. Finally, we provide a simple solution to the MELD filter that allows fast computation.

4.3.1 Filters on graphs

In their simplest forms, filters can be thought of as devices that alter the spectrum of their input. Filters can be used as bases, as is the case with wavelets, and they can be used to directly manipulate signals by changing the frequency response of the filter. For example, many audio devices contain an equalizer that allows one to change the amplitude of bass and treble frequencies. Simple equalizers can be built simply by using a set of filters called a filterbank. In MELD, we use a tunable filter to amplify latent features on a single-cell graph.

Mathematically, graph filters work analogously to classical filters. Particularly, a filter takes in a signal and attenuates it according to a frequency response function. This function accepts frequencies and returns a response coefficient. This is then multiplied by the input Fourier coefficient at the corresponding frequency. The entire filter operation is thus a reweighting of the input Fourier coefficients. In low-pass filters, the function only preserves frequency components below a threshold. Conversely, high-pass filters work by removing frequencies below a threshold. Bandpass filters transfer frequency components that are within a certain range of a central frequency. The tunable filter in MELD is capable of producing any of these responses.

As graph harmonics are defined on the set Λ, it is common

to define them as functions of the form $h : [0, \max(\Lambda)] \mapsto [0, 1]$. For example, a low pass filter with cutoff at λ_k would have $h(x) > 0$ for $x < \lambda_k$ and $h(x) = 0$ otherwise. By abuse of notation, we will refer to the diagonal matrix with the filter h applied to each Laplacian eigenvalue as $h(\Lambda)$, though h is not a set-valued or matrix-valued function. Filtering a signal $\mathbf{f}$ is clearest in the spectral domain, where one simply takes the multiplication $\hat{\mathbf{f}}_{\text{filt}} = h(\Lambda)\hat{\mathbf{f}} = h(\Lambda)\Psi^*\mathbf{f}$.

Finally, it is worth using the above definitions to define a vertex-valued operator to perform filtering. As a graph filter is merely a reweighting of the graph Fourier basis, one can construct the *filter matrix*,

$$H = \Psi h(\Lambda)\Psi^T. \tag{10}$$

A simple manipulation using equation (9) will verify that $H\mathbf{f}$ is the IGFT of $\hat{\mathbf{f}}_{\text{filt}}$. This filter matrix will be used to solve the MELD filter in approximate form for computational efficiency.

4.3.2 Laplacian Regularization

A simple assumption for recovering a latent signal from raw measurements is *smoothness*. In this model the latent signal is assumed to have a low amount of neighbor to neighbor variation. *Laplacian regularization*[23–31] is a simple technique that targets signal smoothness via the optimization

$$\mathbf{y} = \underset{\mathbf{z}}{\text{argmin}} \underbrace{\|\mathbf{x} - \mathbf{z}\|_2^2}_{\text{a}} + \underbrace{\beta \mathbf{z}^T \mathcal{L} \mathbf{z}}_{\text{b}}. \tag{11}$$

Laplacian regularization is a subproblem of the MELD filter that we will discuss for low-pass filtering. In the above, a reconstruction penalty (a) is considered alongside the Laplacian quadratic form (b), which is weighted by the parameter β. The Laplacian quadratic form may also be considered as the norm of the *graph gradient*, i.e.

$$\beta \mathbf{z}^T \mathcal{L} \mathbf{z} = \beta \|\nabla_G \mathbf{z}\|_2^2.$$

Thus one may view Laplacian regularization as a minimization of the edge-derivatives of a function while preserving a reconstruction. Because of this form, this technique has been cast as *Tikhonov regularization*[31,76], which is a common regularization to enforce a high-pass filter to solve inverse problems in regression. In our results we demonstrate a MELD filter that may be reduced to Laplacian regularization using a squared Laplacian.

In section 4.3.1 we introduced filters as functions defined over the Laplacian eigenvalues ($h(\Lambda)$) or as vertex operators (equation 10). Minimizing optimization 11 reveals a similar form for Laplacian regularization. To begin,

$$\begin{aligned}
\mathbf{y} &= \underset{\mathbf{z}}{\text{argmin}} \|\mathbf{x} - \mathbf{z}\|_2^2 + \beta \mathbf{z}^T \mathcal{L} \mathbf{z} \\
&= \underset{\mathbf{z}}{\text{argmin}} (\mathbf{x} - \mathbf{z})^T (\mathbf{x} - \mathbf{z}) + \beta \mathbf{z}^T \mathcal{L} \mathbf{z} \\
&= \underset{\mathbf{z}}{\text{argmin}} \mathbf{x}^T \mathbf{x} + \mathbf{z}^T \mathbf{z} - 2\mathbf{x}^T \mathbf{z} + \beta \mathbf{z}^T \mathcal{L} \mathbf{z}
\end{aligned}$$

Substituting $y = z$, we next differentiate with respect to y and set this to 0,

$$\begin{aligned}
0 &= \nabla_{\mathbf{y}} (\mathbf{x}^T \mathbf{x} + \mathbf{y}^T \mathbf{y} - 2\mathbf{y}^T \mathbf{x} + \beta \mathbf{y}^T \mathcal{L} \mathbf{y}) \\
&= 2\mathbf{y} - 2\mathbf{x} + 2\beta \mathcal{L} \mathbf{y} \\
\mathbf{x} &= (\mathbf{I} + \beta \mathcal{L})\mathbf{y},
\end{aligned}$$

so the solution to problem 11 is

$$\mathbf{y} = (\mathbf{I} + \beta \mathcal{L})^{-1} \mathbf{x}. \tag{12}$$

As the input x is a graph signal in the vertex domain, the least squares solution (12) is a filter matrix $H_{\text{reg}} = (I + \beta \mathcal{L})^{-1}$ as discussed in section 4.3.1. The spectral properties of Laplacian regularization immediately follow as

$$\begin{aligned}
H_{\text{reg}} &= (\mathbf{I} + \beta \mathcal{L})^{-1} \\
&= \Psi \frac{1}{1 + \beta \Lambda} \Psi^T.
\end{aligned} \tag{13}$$

Thus Laplacian regularization is a graph filter with frequency response $h_{\text{reg}}(\lambda) = (1 + \beta\lambda)^{-1}$. Figure S2b shows that this function is a low-pass filter on the Laplacian eigenvalues with cutoff parameterized by β.

4.3.3 Tunable Filtering with MELD

Though simple low-pass filtering with Laplacian regularization is a powerful tool for many machine learning tasks, we sought to develop a filter that is flexible and capable of filtering noise of any frequency. To accomplish these goals, we introduce the MELD filter:

$$\begin{aligned}
\mathbf{y} &= \underset{z}{\text{argmin}} \|\mathbf{x} - \mathbf{z}\|_2^2 + \mathbf{z}^T \mathcal{L}_* \mathbf{z} \\
&\text{where } \mathcal{L}_* = [\beta \mathcal{L} - \alpha \mathbf{I}]^\rho .
\end{aligned} \tag{14}$$

This filter expands upon Laplacian regularization by the addition of a new smoothness structure. Early and related work

proposed the use of a power Laplacian smoothness matrix S in a similar manner as we apply here[31], but little work has since proven its utility. In our construction, α is referred to as modulation, β acts as a reconstruction penalty, and ρ is filter order. These parameters add a great deal of versatility to the MELD filter, and we demonstrate their spectral and vertex effects in Figure **S2**, as well as provide mathematical analysis of the MELD parameters in section 4.3.4. Finally, in section 4.3.5 we discuss an implementation of the filter.

4.3.4 Parameter Analysis

A similar derivation as section 4.3.2 quickly reveals the filter matrix

$$H_{\text{MELD}}(\lambda) = [\mathbf{I} + (\beta\mathcal{L} - \alpha\mathbf{I})^{\rho}]^{-1}\,. \tag{15}$$

which has the frequency response

$$h_{\text{MELD}}(\lambda) = \frac{1}{1 + (\beta\lambda - \alpha)^{\rho}}. \tag{16}$$

Thus, the value of MELD parameters in the vertex optimization (14) has a direct effect on the graph Fourier domain. First, we note by inspection that $h_{\text{MELD}}(\lambda) = h_{\text{reg}}(\lambda)$ for $\alpha = 0$ and $\rho = 1$ (see equation 13). Thus the MELD filter is a superset of graph filters in which Laplacian regularization is a special case.

It is clear that β acts analogously in (16) as it does in the subfilter (13). In each setting, β steepens the cutoff of the filter and shifts it more towards its central frequency (**Fig. S2b**). In the case of $\alpha = 0$, this frequency is $\lambda_1 = 0$. This is done by scaling all frequencies by a factor of β. For stability reasons, we choose $\beta > 0$, as a negative choice of β yields a high frequency amplifier.

The parameters α and ρ change the filter from low pass to band pass or high pass. Figure **S2** highlights the effect on frequency response of the filters and showcases their vertex effects in simple examples. We begin our mathematical analysis with the effects of ρ.

ρ powers the Laplacian harmonics. This steepens the frequency response around the central frequency of the MELD filter and, for even values, makes the function square-integrable. Higher values of ρ lead to sharper tails (**Fig. S2c, S2e**), limiting the frequency response outside of the target band, but with increased response within the band. For technical reasons we do not consider odd-valued $\rho > 1$ when $\alpha > 0$ or $\rho \notin \mathbb{N}$. Indeed, though the parameters β and α do not disrupt the definiteness of $\mathcal{L}_*$ (thus $\mathcal{L}_*$ is defined for $\rho \notin \mathbb{N}$), odd-valued and fractional matrix powers of $\mathcal{L}_*$ result in hyperbolic and unstable filter discontinuities. When $\alpha = 0$, these discontinuities are present only at $\lambda = 0$ and are thus stable. However, when $\alpha > 0$, the hyperbolic behavior of the filter is unstable as these discontinuities now lie within the Laplacian spectrum. Finally, ρ can be used to make a high pass filter by setting it to negative values (**Fig. S2f**).

For the integer powers used in MELD, a basic vertex interpretation of ρ is available. Each column of $\mathcal{L}^k$ is *k-hop localized*, meaning that $\mathcal{L}^k_{ij}$ is non-zero if and only if the there exists a path length k between vertex i and vertex j (for a detailed discussion of this property, see Hammond et al.[74], section 5.2.) Thus, for $\rho \in \mathbb{N}$, the operator $\mathcal{L}^{\rho}$ considers variation over a hop distance of ρ. This naturally leads to the spectral behavior we demonstrate in Figure **S2c**, as signals are required to be smooth over longer hop distances when $\alpha = 0$ and $\rho > 1$.

The parameter α removes values from the diagonal of $\mathcal{L}$. This results in a *modulation* of frequency response by translating the Laplacian harmonic that yields the minimal value for problem (14). This allows one to change the target frequency when $\rho > 1$, as α effectively modulates a band-pass filter. As graph frequencies are positive, we do not consider $\alpha < 0$. In the vertex domain, the effect of α is more nuanced. We study this parameter for $\alpha > 0$ by considering a modified Laplacian $\mathcal{L}_*$ with $\rho = 1$. However, due to hyperbolic spectral behavior for odd-valued ρ, $\alpha > 0$ is ill-performing in practice, so this analysis is merely for intuitive purposes, as similar results extend for $\rho > 1$.

For mathematical analysis of α, $\mathcal{L}_*$ is applied as an operator (equation 7) to an arbitrary graph signal $\mathbf{f}$ defined on a graph G. Expanding $(\mathcal{L}_*\mathbf{f})(i)$ we have the following

$$\begin{aligned}
(\mathcal{L}_*\mathbf{f})(i) &= ([\beta(D - W) - \alpha\mathbf{I}]\,\mathbf{f})(i) \\
&= \beta(D\mathbf{f} - W\mathbf{f} - \frac{\alpha}{\beta}\mathbf{f})(i) \\
&= \beta\left[(d(i) - \frac{\alpha}{\beta})\mathbf{f}(i) - \sum_j W_{ij}\mathbf{f}(j)\right] \\
&= \beta\left[\sum_j (W_{ij} - \frac{\alpha}{N\beta})\mathbf{f}(i) - \sum_j W_{ij}\mathbf{f}(j)\right] \\
&= \beta\sum_j W_{ij}\left[(1 - \frac{\alpha}{d(i)\beta})\mathbf{f}(i) - \mathbf{f}(j)\right].
\end{aligned} \tag{17}$$

Relation (17) establishes the vertex domain effect of α, which corresponds to a reweighting of the local variation at vertex i by a factor of $1 - \frac{\alpha}{d(i)\beta}$. The intuition that follows is that positive α allows disparate values of $\mathbf{f}$ around each vertex to minimize problem (14), which leads to greater response for high frequency harmonics. We demonstrate this modulation in Figure **S2d**.

To conclude, we propose a filter parameterized by reconstruction β (**Fig. S2b**), order ρ (**Fig. S2c, S2e**), and modulation α (**Fig. S2d**). The parameters α and β are limited to be strictly greater than or equal to 0. When $\alpha = 0$, ρ may be any integer, and it adds more low-frequencies to the frequency response as it becomes more positive. On the other hand, if ρ is negative and $\alpha = 0$, ρ controls a high pass filter. When $\alpha > 0$, ρ must be even-valued and the MELD filter becomes a band-pass filter. In standard use cases we propose to use the parameters $\alpha = 0, \beta = 1,$ and $\rho = 2$. All of our biological results were obtained using this parameter set, which gives a square-integrable low-pass filter. As these parameters have direct spectral effects, their implementation in an efficient graph filter is straightforward and presented in section 4.3.5.

4.3.5 Implementation

A naive implementation of the MELD algorithm would apply the matrix inversion presented in equation 15. This approach is untenable for the large single-cell graphs that MELD is designed for, as H_{MELD}^{-1} will have many elements, and, for high powers of ρ or non-sparse graphs, extremely dense. A second approach to solving Equation 14 would diagonalize $\mathcal{L}$ such that the filter function in Equation 16 could be applied directly to the Fourier transform of input raw experimental signals. This approach has similar shortcomings as eigendecomposition is substantively similar to inversion. Finally, a speedier approach might be to use conjugate gradient or proximal methods. In practice, we found that these methods are not well-suited for MELD filtering.

Instead of gradient methods, we use Chebyshev polynomial approximations of $h_{\text{MELD}}(\lambda)$ to rapidly approximate and apply the MELD filter. These approximations, proposed by Hammond et al.[74] and Shuman et al.[77], have gained traction in the graph signal processing community for their efficiency and simplicity. Briefly, a truncated and shifted Chebyshev polynomial approximation is fit to the frequency response of a graph filter. For analysis, the approximating polynomials are applied as polynomials of the Laplacian multiplied by the signal to be filtered. As Chebvyshev polynomials are given by a recurrence relation, the approximation procedure reduces to a computationally efficient series of matrix-vector multiplications. For a more detailed treatment one may refer to Hammond et al.[74] where the polynomials are proposed for graph filters. For application of the MELD filter to a small set of input raw experimental signals, Chebyshev approximations offer the simplest and most efficient implementation of our proposed algorithm. For sufficiently large sets of RES, the computational cost of obtaining the Fourier basis directly may be less than repeated application of the approximation operator; in these cases, we diagonalize the Laplacian either approximately through randomized SVD or exactly using eigendecomposition, depending on user preference. Then, one simply constructs $H_{\text{MELD}} = \Psi h_{\text{MELD}}(\Lambda)\Psi^T$ to analyze raw experimental signals.

4.3.6 Summary

In summary, we have proposed a family of graph filters based on a generalization of Laplacian regularization. This family is parameterized by the modulation α, which controls the target graph harmonics, β, which adjusts the reconstruction weight, and ρ, which defines the squareness of the resulting MELD filter. We provide analysis of these three filters and a consideration of implementation considerations in sections 4.3.4 and 4.3.5 respectively.

MELD is implemented in Python 3 and is built atop the `scprep`, `graphtools`, and `pygsp` packages. We developed `scprep` efficiently process single cell data, and `graphtools` was developed for construction and manipulation of graphs built on data. Fourier analysis and Chebyshev approximations are implemented using functions from the `pygsp` toolbox[78]. These packages are available through the `pip` package manager. One may obtain the MELD package on github at `https://github.com/KrishnaswamyLab/MELD` and on `pip` as `meld`.

4.4 Vertex-frequency clustering

The goal of vertex-frequency clustering is to find a partition of a graph that sufficiently separates dissimilar vertices with respect to some observed signal. We use a technique proposed in Shuman et al.[35] based on a graph generalization of

the classical Short Time Fourier Transform. This generalization will allow us to simultaneously localize signals in both frequency and vertex domains. The output of this transform will be a spectrogram that we then use to cluster the graph.

4.4.1 The Short Time Fourier Transform (STFT)

The STFT of a signal is obtained by partitioning the signal into short equal sized segments of time and computing a Fourier transform on each segment. The result is a set of frequency coefficients for each segment, describing the frequency content of the signal as it changes over time. Plots of these functions are called spectrograms.

The computation of an STFT proceeds by multiplying a *window* by the input signal. The window is typically only non zero in a small contiguous region of time, determined by translating a *window function*, $g \in L^2(\mathbb{R})$ to a time $u \in \mathbb{R}$. The translation operator $T_u : L^2(\mathbb{R}) \to L^2(\mathbb{R})$ is defined via convolution of a a function $f \in L^2(\mathbb{R})$ with a dirac delta δ_u:

$$(T_u f)(t) := (f * \delta_u)(t) = g(t-u). \tag{18}$$

The window is modulated at frequencies $\xi \in \mathbb{R}$ via the modulation operator $M_\xi : L^2(\mathbb{R}) \to L^2(\mathbb{R})$, which multiplies a function $f \in L^2(\mathbb{R})$ by a Fourier basis function:

$$(M_\xi f)(t) := e^{2\pi i \xi t} f(t). \tag{19}$$

Later, we use the fact that these two operations can be defined in the Fourier domain via the convolution theorem, i.e. $\widehat{T_u f}(\xi) = (\hat{f} \cdot \hat{\delta_u})(\xi) = e^{-2\pi i \xi t}\hat{f}(\xi)$ and $\widehat{M_k f}(\xi) = (\hat{f} * \hat{\delta_k})(\xi) = \hat{f}(\xi - k)$.

With these tools we can translate and modulate a window function to produce a *windowed Fourier atom*:

$$g_{u,\xi}(t) := (M_\xi T_u g)(t) = g(t-u)e^{2\pi i \xi t}. \tag{20}$$

The inner product of each windowed Fourier atom with the signal f yields the **STFT**, a set of frequency coefficients

$$Sf(u,\xi) := \langle f, g_{u,\xi} \rangle = \int_{-\infty}^{\infty} f(t)[g(t-u)]^* e^{-2\pi i \xi t} dt. \tag{21}$$

One way to interpret this transform $Sf(u,\xi)$ is taking the Fourier transform of time-slices of the input signal evaluated at each frequency ξ.

4.4.2 The Windowed Graph Fourier Transform (WGFT)

Recent works[35] generalize the windowed Fourier transform to graph signals. In order to define translation and modulation for graph signals, a convolution operator must be defined.[35] construct the *generalized convolution* by extension of the convolution theorem, i.e. that convolution in the vertex(time) domain is equivalent to multiplication in the Fourier domain. Then we have for two signals $f, g \in \mathbb{R}^N$ on a graph $\mathcal{G}$ with N vertices and Laplacian $\mathcal{L} = \mathcal{U}\Lambda\mathcal{U}^{-1}$

$$(f * g)(n) := \sum_{\ell=0}^{N-1} \hat{f}(\lambda_\ell)\hat{g}(\lambda_\ell)U_\ell(n). \tag{22}$$

We rewrite this equation more succinctly in vector notation:

$$f * g = \mathcal{U}\,\mathrm{diag}(\hat{g})\hat{f}, \tag{23}$$

which amounts to taking the inverse Fourier transform of the product of f and g in the spectral domain, implying the key analogy to the classical setting

$$\widehat{f * g} = \hat{f}\hat{g}. \tag{24}$$

A graph translation operator quickly follows from generalized convolution. For translation we translate the function f to a vertex $i \in \{1, 2, ..., N\}$ via $T_i : \mathbb{R}^N \to \mathbb{R}^N$

$$(T_i f)(n) := \sqrt{N}(f * \delta_i)(n) \tag{25}$$

which uses the generalized convolution of equation 22 to convolve f with a dirac delta localized to vertex i.[35] demonstrate that this operator inherits various properties of its classical counterpart; however, the operator is not isometric and is affected by the graph that it is built on. Furthermore, for signals that are not tightly localized in the vertex domain and on graphs that are not directly related to Fourier harmonics (e.g. the circle graph), it is not clear what graph translation implies.

In addition to translation, a *generalized modulation operator* is defined[35] as $M_k : \mathbb{R}^N \to \mathbb{R}^N$ for frequencies $k \in \{0, 1, ..., N-1\}$.

$$(M_k f)(n) := \sqrt{N} f(n) U_k(n) \tag{26}$$

This formulation is analogous in construction to classical modulation, defined as multiplication by an eigenfunction (equation 19). Classical modulation translates signals in

the Fourier domain; because of the discrete nature of the graph Fourier domain, this property is only weakly shared between the two operators. Instead, the generalized modulation M_k translates the *DC component* of f, $\hat{f}(0)$, to λ_k, i.e. $\widehat{(M_k f)}(\lambda_k) = \hat{f}(0)$. Furthermore, for any function f whose frequency content is localized around λ_0, $(M_k f)$ is localized in frequency around λ_k.[35] details this construction and provides bounds on spectral localization and other properties.

With these two operators, a graph windowed Fourier atom is constructed[35] for any window function $g \in \mathbb{R}^N$

$$g_{i,k}(n) := (M_k T_i g)(n) = N U_k(n) \sum_{\ell=0}^{N-1} \hat{g}(\lambda_\ell) U_\ell^*(i) U_\ell(n). \tag{27}$$

We can then build a spectrogram $Q = (q_{ik}) \in \mathbb{R}^{NxN}$ by taking the inner product of each $g_{i,k} \forall i \in \{1, 2, ..., N\} \wedge \forall k \in \{0, 1, ..., N-1\}$ with the target signal f

$$q_{ik} = Sf(i,k) := \langle f, g_{i,k} \rangle. \tag{28}$$

As with the classical windowed Fourier transform, one could interpret this as segmenting the signal by windows and then taking the Fourier transform of each segment

$$q_i = \langle (T_i g .* f), U \rangle \tag{29}$$

where $.*$ is the element-wise product.

4.4.3 Description of vertex-frequency clustering algorithm

In order to generate the matrix Q we need a suitable window function. We use the normalized heat kernel

$$\hat{g}(\lambda) = C e^{-t\lambda}, \tag{30}$$

$$C = ||g||_2^{-1}. \tag{31}$$

By translating this kernel, multiplying it with our target signal f and taking the Fourier transform of the result, we obtain a windowed graph Fourier transform of f that is localized based on the *diffusion distance*[35,75] from each vertex to every other vertex in the graph.

For an input RES $\mathbf{x}$, signal-biased spectral clustering proceeds as follows:

1. Generate the window matrix P_t, which contains as its columns translated and normalized heat kernels at the scale t
2. Column-wise multiply $X_t = P .* \mathbf{x}$; the i-th column of X_t is an entry-wise product of the i-th window and $\mathbf{x}$.
3. Take the Fourier Transform of each column of X_t. This matrix, $\hat{C}_t$ is the normalized WGFT matrix.

This produces a single WGFT for the scale t. At this stage, Shuman et al.[35] proposed to saturate the elements of $\hat{C}_t$ using the activation function $\tanh(|\hat{C}_t|)$ (where $|\,.\,|$ is an element-wise absolute value). Then, k-means is performed on this saturated output to yield clusters. This operation has connections to spectral clustering as the features that k-means is run on are coefficients of graph harmonics.

We build upon this approach to add robustness, sensitivity to sign changes, and scalability. Particularly, vertex-frequency clustering builds a set of activated spectrograms at different window scales using the procedure outlined above. Then, the entire set is combined through summation and the filtered input signal $\mathbf{y}$ is concatenated as an additional feature. Finally, k-means is performed on this matrix.

The multiscale approach we have proposed has a number of benefits. Foremost, it removes the complexity of picking a window-size. Second, using the actual input signal as a feature allows the clustering to consider both frequency and sign information in the raw experimental signal. For scalability, we leverage the fact that P_t is effectively a diffusion operator and thus can be built efficiently by treating it as a Markov matrix and normalizing the graph adjacency by the degree.

5 Methods

5.1 Processing and analysis of the T-cell datasets

Gene expression counts matrices prepared by Datlinger et al.[11] were accessed from the NCBI GEO database accession GSE92872. 3,143 stimulated and 2,597 unstimulated T-cells were processed in a pipeline derived from the published supplementary software. First, artificial genes corresponding to gRNAs were removed from the counts matrix. Genes observed in fewer than five cells were removed. Cell with a library size higher than 35,000 UMI / cell were removed. To filter dead or dying cells, expression of all mitochondrial genes was z-scored and cells with average z-score expression greater than 1 were removed. As in the published analysis, all mitochondrial and ribosomal genes

were excluded. Filtered cells and genes were library size normalized and square-root transformed. To impute gene expression, MAGIC was run using default parameters. To build a cell-state graph, 100 PCA dimensions were calculated and edge weights between cells were calculated using an alpha-decay kernel as implemented in the Graphtools library (www.github.com/KrishnaswamyLab/graphtools) using knn=10 and decay=20. To infer the EES, MELD was run on the cell state graph using the stimulated / unstimulated labels and input with the smoothing parameter $\beta = 1$. To identify genes that vary with the MELD vector, kNN-DREMI[16] scores were calculated between each gene and the EES vector using default parameters as implemented in scprep (www.github.com/KrishnaswamyLab/scprep). GO term enrichment was performed using EnrichR with the genes having the top 1% of kNN-DREMI scores used as input.

5.2 Processing and analysis of the chordin datasets

Gene expression counts matrices prepared by Wagner et al.[12] (the chordin dataset) were downloaded from NCBI GEO (GSE112294). 16079 cells from *chd* embryos injected with gRNAs targeting chordin and 10782 cells from *tyr* embryos injected with gRNAs targeting tyrosinase were accessed. Lowly expressed genes detected in fewer than 5 cells were removed. Cells with library sizes larger than 15000 UMI / cell were removed. Counts were library-size normalized and square root transformed. Cluster labels included with the counts matrices were used for cell type identification.

During preliminary analysis, a group of 24 cells were identified originating exclusively from the *chd* embryos. Despite an average library size in the bottom 12% of cells, these cells exhibited 546-fold, 246-fold, and 1210-fold increased expression of Sh3Tc1, LOC101882117, and LOC101885394 respectively. To the best of our knowledge, the function of these genes in development is not described. These cells were annotated by Wagner et al.[12] as belonging to 7 cell types including the Tailbud – Spinal Cord and Neural – Midbrain. These cells were excluded from further analysis.

To generate a cell state graph, 100 PCA dimensions were calculated from the square root transformed filtered gene expression matrix of both datasets. Edge weights between cells on the graph were calculated using an alpha-decay kernel with parameters knn=10, decay=10. MAGIC was used to impute gene expression values using t=7. MELD was run using the *tyr* or *chd* label as input. To identify subpopulations of the Tailbud - Presomitic Mesoderm cluster, we applied Vertex Frequency Clustering with k=4. Cell types were annotated using sets of marker genes curated by Farrell et al.[13]. Changes in gene expression for the top and bottom 20% of cells by EES values in the four clusters were compared.

5.3 Generation, processing and analysis of the pancreatic islet datasets

Single cell RNA-sequencing was performed on human β cells from three different islet donors in the presence and absence of IFNγ. The islets were received on three different days. Cells were cultured for 24 hours with 25ng/mL IFNγ (R&D Systems) in CMRL 1066 medium (Gibco) and subsequently dissociated into single cells with 0.05% Trypsin EDTA (Gibco). Cells were then stained with FluoZin-3 (Invitrogen) and TMRE (Life Technologies) and sorted using a FACS Aria II (BD). The three samples were pooled for the sequencing. Cells were immediately processed using the 10X Genomics Chromium 3' Single Cell RNA-sequencing kit at the Yale Center for Genome Analysis. The raw sequencing data was processed using the 10X Genomics Cell Ranger Pipeline.

Data from all three donors was concatenated into a single matrix for analysis. First, cells not expressing insulin, somatostatin, or glucagon were excluded from analysis using donor-specific thresholds. The data was square root transformed and reduced to 100 PCA dimensions. Next, we applied our MNN batch-effect correction kernel to create a graph across all three donors with parameters knn=5, decay=30. This graph was then used for PHATE and MAGIC. The EES was calculated using MELD with default parameters. To identify cell types, we performed Vertex Frequency Clustering using k=8. To identify signature genes of IFNγ stimulation, we calculated kNN-DREMI scores for all genes with the EES vector and kept genes with the top 1% of scores. To identify genes that were differentially expressed in the beta - nonresponsive cluster, we calculated the Wasserstein distance (also called Earth Mover's distance) between expression of each gene in the nonresponsive cluster and all other clusters.

References

[1] Laleh Haghverdi, Aaron T. L. Lun, Michael D. Morgan, and John C. Marioni. Batch effects in single-cell RNA-

sequencing data are corrected by matching mutual nearest neighbors. *Nature Biotechnology*, April 2018. ISSN 1546-1696. doi: 10.1038/nbt.4091.

[2] Andrew Butler, Paul Hoffman, Peter Smibert, Efthymia Papalexi, and Rahul Satija. Integrating single-cell transcriptomic data across different conditions, technologies, and species. *Nature Biotechnology*, 36(5):411–420, May 2018. ISSN 1546-1696. doi: 10.1038/nbt.4096.

[3] Vladimir Yu Kiselev, Andrew Yiu, and Martin Hemberg. Scmap: Projection of single-cell RNA-seq data across data sets. *Nature Methods*, 15(5):359–362, May 2018. ISSN 1548-7105. doi: 10.1038/nmeth.4644.

[4] Greg Finak, Andrew McDavid, Masanao Yajima, Jingyuan Deng, Vivian Gersuk, Alex K. Shalek, Chloe K. Slichter, Hannah W. Miller, M. Juliana McElrath, Martin Prlic, Peter S. Linsley, and Raphael Gottardo. MAST: A flexible statistical framework for assessing transcriptional changes and characterizing heterogeneity in single-cell RNA sequencing data. *Genome Biology*, 16, 2015. ISSN 1474-7596. doi: 10.1186/s13059-015-0844-5.

[5] Charlotte Soneson and Mark D. Robinson. Bias, robustness and scalability in single-cell differential expression analysis. *Nature Methods*, February 2018. ISSN 1548-7105. doi: 10.1038/nmeth.4612.

[6] Jesse Min Zhang, Govinda M. Kamath, and David N. Tse. Towards a post-clustering test for differential expression. *bioRxiv*, page 463265, November 2018. doi: 10.1101/463265.

[7] Xin Gao, Deqing Hu, Madelaine Gogol, and Hua Li. ClusterMap: Comparing analyses across multiple Single Cell RNA-Seq profiles. *bioRxiv*, page 331330, June 2018. doi: 10.1101/331330.

[8] Anoop P. Patel, Itay Tirosh, John J. Trombetta, Alex K. Shalek, Shawn M. Gillespie, Hiroaki Wakimoto, Daniel P. Cahill, Brian V. Nahed, William T. Curry, Robert L. Martuza, David N. Louis, Orit Rozenblatt-Rosen, Mario L. Suvà, Aviv Regev, and Bradley E. Bernstein. Single-cell RNA-seq highlights intratumoral heterogeneity in primary glioblastoma. *Science*, 344 (6190):1396–1401, June 2014. ISSN 0036-8075, 1095-9203. doi: 10.1126/science.1254257.

[9] Itay Tirosh, Benjamin Izar, Sanjay M. Prakadan, Marc H. Wadsworth, Daniel Treacy, John J. Trombetta, Asaf Rotem, Christopher Rodman, Christine Lian, George Murphy, Mohammad Fallahi-Sichani, Ken Dutton-Regester, Jia-Ren Lin, Ofir Cohen, Parin Shah, Diana Lu, Alex S. Genshaft, Travis K. Hughes, Carly G. K. Ziegler, Samuel W. Kazer, Aleth Gaillard, Kellie E. Kolb, Alexandra-Chloé Villani, Cory M. Johannessen, Aleksandr Y. Andreev, Eliezer M. Van Allen, Monica Bertagnolli, Peter K. Sorger, Ryan J. Sullivan, Keith T. Flaherty, Dennie T. Frederick, Judit Jané-Valbuena, Charles H. Yoon, Orit Rozenblatt-Rosen, Alex K. Shalek, Aviv Regev, and Levi A. Garraway. Dissecting the multicellular ecosystem of metastatic melanoma by single-cell RNA-seq. *Science*, 352(6282):189–196, April 2016. ISSN 0036-8075, 1095-9203. doi: 10.1126/science.aad0501.

[10] Diego Adhemar Jaitin, Assaf Weiner, Ido Yofe, David Lara-Astiaso, Hadas Keren-Shaul, Eyal David, Tomer Meir Salame, Amos Tanay, Alexander van Oudenaarden, and Ido Amit. Dissecting Immune Circuits by Linking CRISPR-Pooled Screens with Single-Cell RNA-Seq. *Cell*, 167(7):1883–1896.e15, December 2016. ISSN 0092-8674. doi: 10.1016/j.cell.2016.11.039.

[11] Paul Datlinger, André F. Rendeiro, Christian Schmidl, Thomas Krausgruber, Peter Traxler, Johanna Klughammer, Linda C. Schuster, Amelie Kuchler, Donat Alpar, and Christoph Bock. Pooled CRISPR screening with single-cell transcriptome readout. *Nature Methods*, January 2017. ISSN 1548-7091. doi: 10.1038/nmeth.4177.

[12] Daniel E. Wagner, Caleb Weinreb, Zach M. Collins, James A. Briggs, Sean G. Megason, and Allon M. Klein. Single-cell mapping of gene expression landscapes and lineage in the zebrafish embryo. *Science*, page eaar4362, April 2018. ISSN 0036-8075, 1095-9203. doi: 10.1126/science.aar4362.

[13] Jeffrey A. Farrell, Yiqun Wang, Samantha J. Riesenfeld, Karthik Shekhar, Aviv Regev, and Alexander F. Schier. Single-cell reconstruction of developmental trajectories during zebrafish embryogenesis. *Science*, 360 (6392):eaar3131, June 2018. ISSN 0036-8075, 1095-9203. doi: 10.1126/science.aar3131.

[14] Caleb Weinreb, Samuel Wolock, Allon M. Klein, and Bonnie Berger. SPRING: A kinetic interface for vi-

sualizing high dimensional single-cell expression data. *Bioinformatics*, 34(7):1246–1248, April 2018. ISSN 1367-4803. doi: 10.1093/bioinformatics/btx792.

[15] Kevin R. Moon, David van Dijk, Zheng Wang, Scott Gigante, Daniel Burkhardt, William Chen, Antonia van den Elzen, Matthew J. Hirn, Ronald R. Coifman, Natalia B. Ivanova, Guy Wolf, and Smita Krishnaswamy. Visualizing Transitions and Structure for Biological Data Exploration. *bioRxiv*, page 120378, June 2018. doi: 10.1101/120378.

[16] David van Dijk, Roshan Sharma, Juozas Nainys, Kristina Yim, Pooja Kathail, Ambrose J. Carr, Cassandra Burdziak, Kevin R. Moon, Christine L. Chaffer, Diwakar Pattabiraman, Brian Bierie, Linas Mazutis, Guy Wolf, Smita Krishnaswamy, and Dana Pe'er. Recovering Gene Interactions from Single-Cell Data Using Data Diffusion. *Cell*, 174(3):716–729.e27, July 2018. ISSN 0092-8674. doi: 10.1016/j.cell.2018.05.061.

[17] Karthik Shekhar, Sylvain W. Lapan, Irene E. Whitney, Nicholas M. Tran, Evan Z. Macosko, Monika Kowalczyk, Xian Adiconis, Joshua Z. Levin, James Nemesh, Melissa Goldman, Steven A. McCarroll, Constance L. Cepko, Aviv Regev, and Joshua R. Sanes. Comprehensive Classification of Retinal Bipolar Neurons by Single-Cell Transcriptomics. *Cell*, 166(5):1308–1323.e30, August 2016. ISSN 0092-8674, 1097-4172. doi: 10.1016/j.cell.2016.07.054.

[18] Jacob H. Levine, Erin F. Simonds, Sean C. Bendall, Kara L. Davis, El-ad D. Amir, Michelle D. Tadmor, Oren Litvin, Harris G. Fienberg, Astraea Jager, Eli R. Zunder, Rachel Finck, Amanda L. Gedman, Ina Radtke, James R. Downing, Dana Pe'er, and Garry P. Nolan. Data-Driven Phenotypic Dissection of AML Reveals Progenitor-like Cells that Correlate with Prognosis. *Cell*, 162(1):184–197, July 2015. ISSN 0092-8674. doi: 10.1016/j.cell.2015.05.047.

[19] Chen Xu and Zhengchang Su. Identification of cell types from single-cell transcriptomes using a novel clustering method. *Bioinformatics (Oxford, England)*, 31(12):1974–1980, June 2015. ISSN 1367-4811. doi: 10.1093/bioinformatics/btv088.

[20] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst. The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains. *IEEE Signal Processing Magazine*, 30(3):83–98, May 2013. ISSN 1053-5888. doi: 10.1109/MSP.2012.2235192.

[21] Kevin R. Moon, Jay S. Stanley, Daniel Burkhardt, David van Dijk, Guy Wolf, and Smita Krishnaswamy. Manifold learning-based methods for analyzing single-cell RNA-sequencing data. *Current Opinion in Systems Biology*, 7:36–46, February 2018. ISSN 2452-3100. doi: 10.1016/j.coisb.2017.12.008.

[22] S. Deutsch, A. Ortega, and G. Medioni. Manifold denoising based on spectral graph wavelets. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4673–4677, March 2016. doi: 10.1109/ICASSP.2016.7472563.

[23] Rie K. Ando and Tong Zhang. Learning on Graph with Laplacian Regularization. In B. Schölkopf, J. C. Platt, and T. Hoffman, editors, *Advances in Neural Information Processing Systems 19*, pages 25–32. MIT Press, 2007.

[24] Kilian Q Weinberger, Fei Sha, Qihui Zhu, and Lawrence K. Saul. Graph Laplacian Regularization for Large-Scale Semidefinite Programming. In B. Schölkopf, J. C. Platt, and T. Hoffman, editors, *Advances in Neural Information Processing Systems 19*, pages 1489–1496. MIT Press, 2007.

[25] X. He, M. Ji, C. Zhang, and H. Bao. A Variance Minimization Criterion to Feature Selection Using Laplacian Regularization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(10):2013–2025, October 2011. ISSN 0162-8828. doi: 10.1109/TPAMI.2011.44.

[26] X. Liu, D. Zhai, D. Zhao, G. Zhai, and W. Gao. Progressive Image Denoising Through Hybrid Graph Laplacian Regularization: A Unified Framework. *IEEE Transactions on Image Processing*, 23(4):1491–1503, April 2014. ISSN 1057-7149. doi: 10.1109/TIP.2014.2303638.

[27] J. Pang, G. Cheung, A. Ortega, and O. C. Au. Optimal graph laplacian regularization for natural image denoising. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2294–2298, April 2015. doi: 10.1109/ICASSP.2015.7178380.

[28] Jiahao Pang and Gene Cheung. Graph Laplacian Regularization for Image Denoising: Analysis in the Continuous Domain. *IEEE Transactions on Image Processing*, 26(4):1770–1785, April 2017. ISSN 1057-7149, 1941-0042. doi: 10.1109/TIP.2017.2651400.

[29] Dengyong Zhou and Bernhard Schölkopf. A regularization framework for learning from graph data. In *ICML workshop on statistical relational learning and Its connections to other fields*, volume 15, pages 67–8, 2004.

[30] Jihun Ham, Daniel D Lee, and Lawrence K Saul. Semisupervised alignment of manifolds. In *AISTATS*, pages 120–127, 2005.

[31] Mikhail Belkin, Irina Matveeva, and Partha Niyogi. Regularization and semi-supervised learning on large graphs. In *International Conference on Computational Learning Theory*, pages 624–638. Springer, 2004.

[32] Martin Barron and Jun Li. Identifying and removing the cell-cycle effect from single-cell rna-sequencing data. *Scientific reports*, 6:33892, 2016.

[33] Smita Krishnaswamy, Matthew H. Spitzer, Michael Mingueneau, Sean C Bendall, Oren Litvin, Erica Stone, Dana Pe'er, and Garry P Nolan. Conditional Density-based Analysis of T cell Signaling in Single Cell Data. *Science (New York, N.Y.)*, 346(6213):1250689, November 2014. ISSN 0036-8075. doi: 10.1126/science.1250689.

[34] David van Dijk, Scott Gigante, Kevin Moon, Alexander Strzalkowski, Katie Ferguson, Jess Cardin, Guy Wolf, and Smita Krishnaswamy. Modeling Dynamics of Biological Systems with Deep Generative Neural Networks. *arXiv:1802.03497 [cs, stat]*, February 2018.

[35] David I Shuman, Benjamin Ricaud, and Pierre Vandergheynst. Vertex-frequency analysis on graphs. *Applied and Computational Harmonic Analysis*, 40(2): 260–291, March 2016. ISSN 1063-5203. doi: 10.1016/j.acha.2015.02.005.

[36] L. Le Magoarou, R. Gribonval, and N. Tremblay. Approximate Fast Graph Fourier Transforms via Multilayer Sparse Approximations. *IEEE Transactions on Signal and Information Processing over Networks*, 4(2): 407–420, June 2018. ISSN 2373-776X. doi: 10.1109/TSIPN.2017.2710619.

[37] Maxim V. Kuleshov, Matthew R. Jones, Andrew D. Rouillard, Nicolas F. Fernandez, Qiaonan Duan, Zichen Wang, Simon Koplev, Sherry L. Jenkins, Kathleen M. Jagodnik, Alexander Lachmann, Michael G. McDermott, Caroline D. Monteiro, Gregory W. Gundersen, and Avi Ma'ayan. Enrichr: A comprehensive gene set enrichment analysis web server 2016 update. *Nucleic Acids Research*, 44(Web Server issue):W90–W97, July 2016. ISSN 0305-1048. doi: 10.1093/nar/gkw377.

[38] L. A. Turka, D. G. Schatz, M. A. Oettinger, J. J. Chun, C. Gorka, K. Lee, W. T. McCormack, and C. B. Thompson. Thymocyte expression of RAG-1 and RAG-2: Termination by T cell receptor cross-linking. *Science*, 253 (5021):778–781, August 1991. ISSN 0036-8075, 1095-9203. doi: 10.1126/science.1831564.

[39] M. Hammerschmidt, F. Pelegri, M. C. Mullins, D. A. Kane, F. J. van Eeden, M. Granato, M. Brand, M. Furutani-Seiki, P. Haffter, C. P. Heisenberg, Y. J. Jiang, R. N. Kelsh, J. Odenthal, R. M. Warga, and C. Nusslein-Volhard. Dino and mercedes, two genes regulating dorsal development in the zebrafish embryo. *Development*, 123(1):95–102, December 1996. ISSN 0950-1991, 1477-9129.

[40] Stefan Schulte-Merker, Kevin J. Lee, Andrew P. McMahon, and Matthias Hammerschmidt. The zebrafish organizer requires *chordino*. *Nature*, 387(6636):862–863, June 1997. ISSN 1476-4687. doi: 10.1038/43092.

[41] Shannon Fisher and Marnie E. Halpern. Patterning the zebrafish axial skeleton requires early *chordin* function. *Nature Genetics*, 23(4):442–446, December 1999. ISSN 1546-1718. doi: 10.1038/70557.

[42] Tamar Dvash, Yoav Mayshar, Henia Darr, Michael McElhaney, Douglas Barker, Ofra Yanuka, Karen J. Kotkow, Lee L. Rubin, Nissim Benvenisty, and Rachel Eiges. Temporal gene expression during differentiation of human embryonic stem cells and embryoid bodies. *Human Reproduction*, 19(12):2875–2883, December 2004. ISSN 0268-1161. doi: 10.1093/humrep/deh529.

[43] Cole Trapnell, Davide Cacchiarelli, Jonna Grimsby, Prapti Pokharel, Shuqiang Li, Michael Morse, Niall J. Lennon, Kenneth J. Livak, Tarjei S. Mikkelsen, and John L. Rinn. The dynamics and regulators of cell fate decisions are revealed by pseudotemporal ordering

of single cells. *Nature Biotechnology*, 32(4):381–386, April 2014. ISSN 1087-0156. doi: 10.1038/nbt.2859.

[44] Sean C. Bendall, Kara L. Davis, El-ad David Amir, Michelle D. Tadmor, Erin F. Simonds, Tiffany J. Chen, Daniel K. Shenfeld, Garry P. Nolan, and Dana Pe'er. Single-Cell Trajectory Detection Uncovers Progression and Regulatory Coordination in Human B Cell Development. *Cell*, 157(3):714–725, April 2014. ISSN 0092-8674. doi: 10.1016/j.cell.2014.04.005.

[45] Joshua D. Welch, Alexander J. Hartemink, and Jan F. Prins. SLICER: Inferring branched, nonlinear cellular trajectories from single cell RNA-seq data. *Genome Biology*, 17:106, May 2016. ISSN 1474-760X. doi: 10.1186/s13059-016-0975-3.

[46] Laleh Haghverdi, Maren Büttner, F. Alexander Wolf, Florian Buettner, and Fabian J. Theis. Diffusion pseudotime robustly reconstructs lineage branching. *Nature Methods*, 13(10):845–848, October 2016. ISSN 1548-7091. doi: 10.1038/nmeth.3971.

[47] Wouter Saelens, Robrecht Cannoodt, Helena Todorov, and Yvan Saeys. A comparison of single-cell trajectory inference methods: Towards more accurate and robust tools. *bioRxiv*, page 276907, March 2018. doi: 10.1101/276907.

[48] Anastasia Katsarou, Soffia Gudbjörnsdottir, Araz Rawshani, Dana Dabelea, Ezio Bonifacio, Barbara J. Anderson, Laura M. Jacobsen, Desmond A. Schatz, and Åke Lernmark. Type 1 diabetes mellitus. *Nature Reviews Disease Primers*, 3:17016, March 2017. ISSN 2056-676X. doi: 10.1038/nrdp.2017.16.

[49] V. Ablamunits, D. Elias, T. Reshef, and I. R. Cohen. Islet T cells secreting IFN-gamma in NOD mouse diabetes: Arrest by p277 peptide treatment. *Journal of Autoimmunity*, 11(1):73–81, February 1998. ISSN 0896-8411. doi: 10.1006/jaut.1997.0177.

[50] Andrew S. Diamond and Ronald G. Gill. An Essential Contribution by IFN-γ to CD8+ T Cell-Mediated Rejection of Pancreatic Islet Allografts. *The Journal of Immunology*, 165(1):247–255, July 2000. ISSN 0022-1767, 1550-6606. doi: 10.4049/jimmunol.165.1.247.

[51] Miguel Lopes, Burak Kutlu, Michela Miani, Claus H. Bang-Berthelsen, Joachim Størling, Flemming Pociot, Nathan Goodman, Lee Hood, Nils Welsh, Gianluca Bontempi, and Decio L. Eizirik. Temporal profiling of cytokine-induced genes in pancreatic β-cells by meta-analysis and network inference. *Genomics*, 103(4):264–275, April 2014. ISSN 0888-7543. doi: 10.1016/j.ygeno.2013.12.007.

[52] Mauro J. Muraro, Gitanjali Dharmadhikari, Dominic Grün, Nathalie Groen, Tim Dielen, Erik Jansen, Leon van Gurp, Marten A. Engelse, Francoise Carlotti, Eelco J.P. de Koning, and Alexander van Oudenaarden. A Single-Cell Transcriptome Atlas of the Human Pancreas. *Cell Systems*, 3(4):385–394.e3, October 2016. ISSN 2405-4712. doi: 10.1016/j.cels.2016.09.002.

[53] Chilakamarti V Ramana, M. Pilar Gil, Robert D Schreiber, and George R Stark. Stat1-dependent and -independent pathways in IFN-γ-dependent signaling. *Trends in Immunology*, 23(2):96–101, February 2002. ISSN 1471-4906. doi: 10.1016/S1471-4906(01)02118-4.

[54] Anthony J. Sadler and Bryan R. G. Williams. Interferon-inducible antiviral effectors. *Nature reviews. Immunology*, 8(7):559–568, July 2008. ISSN 1474-1733. doi: 10.1038/nri2314.

[55] Katherine A. Fitzgerald. The Interferon Inducible Gene: Viperin. *Journal of Interferon & Cytokine Research*, 31(1):131–135, January 2011. ISSN 1079-9907. doi: 10.1089/jir.2010.0127.

[56] Zhiwei Zheng, Lin Wang, and Jihong Pan. Interferon-stimulated gene 20-kDa protein (ISG20) in infection and disease: Review and outlook. *Intractable & Rare Diseases Research*, 6(1):35–40, February 2017. ISSN 2186-3644. doi: 10.5582/irdr.2017.01004.

[57] Monica Hultcrantz, Michael H. Hühn, Monika Wolf, Annika Olsson, Stella Jacobson, Bryan R. Williams, Olle Korsgren, and Malin Flodström-Tullberg. Interferons induce an antiviral state in human pancreatic islet cells. *Virology*, 367(1):92–101, October 2007. ISSN 0042-6822. doi: 10.1016/j.virol.2007.05.010.

[58] Andrew F. Stewart, Mehboob A. Hussain, Adolfo García-Ocaña, Rupangi C. Vasavada, Anil Bhushan, Ernesto Bernal-Mizrachi, and Rohit N. Kulkarni. Human β-Cell Proliferation and Intracellular Signaling: Part 3. *Diabetes*, 64(6):1872–1885, June 2015. ISSN 0012-1797. doi: 10.2337/db14-1843.

[59] Lydia Farack, Matan Golan, Adi Egozi, Nili Dezorella, Keren Bahar Halpern, Shani Ben-Moshe, Immacolata Garzilli, Beáta Tóth, Lior Roitman, Valery Krizhanovsky, and Shalev Itzkovitz. Transcriptional Heterogeneity of Beta Cells in the Intact Pancreas. *Developmental Cell*, 48(1):115–125.e4, January 2019. ISSN 1534-5807. doi: 10.1016/j.devcel.2018.11.001.

[60] Manu Setty, Michelle D. Tadmor, Shlomit Reich-Zeliger, Omer Angel, Tomer Meir Salame, Pooja Kathail, Kristy Choi, Sean Bendall, Nir Friedman, and Dana Pe'er. Wishbone identifies bifurcating developmental trajectories from single-cell data. *Nature Biotechnology*, 34(6):637–645, June 2016. ISSN 1087-0156. doi: 10.1038/nbt.3569.

[61] Gonzalo Mateos, Santiago Segarra, Antonio G Marques, and Alejandro Ribeiro. Connecting the dots: Identifying network structure via graph signal processing. *arXiv preprint arXiv:1810.13066*, 2018.

[62] Ronald R. Coifman and Stéphane Lafon. Diffusion maps. *Applied and Computational Harmonic Analysis*, 21(1):5–30, July 2006. ISSN 1063-5203. doi: 10.1016/j.acha.2006.04.006.

[63] Ronald R Coifman and Matthew J Hirn. Diffusion maps for changing data. *Applied and computational harmonic analysis*, 36(1):79–107, 2014.

[64] Tyrus Berry and Timothy Sauer. Local kernels and the geometric structure of data. *Applied and Computational Harmonic Analysis*, 40(3):439–469, 2016.

[65] Amit Bermanis, Guy Wolf, and Amir Averbuch. Diffusion-based kernel methods on euclidean metric measure spaces. *Applied and Computational Harmonic Analysis*, 41(1):190–213, 2016.

[66] Amit Bermanis, Guy Wolf, and Amir Averbuch. Cover-based bounds on the numerical rank of gaussian kernels. *Applied and Computational Harmonic Analysis*, 36(2): 302–315, 2014.

[67] Nicholas F Marshall and Ronald R Coifman. Manifold learning with bi-stochastic kernels. *arXiv preprint arXiv:1711.06711*, 2017.

[68] Nicholas F Marshall and Matthew J Hirn. Time Coupled Diffusion Maps. *arXiv preprint arXiv:1608.03628*, 2016.

[69] Ofir Lindenbaum, Arie Yeredor, Moshe Salhov, and Amir Averbuch. MultiView Diffusion Maps. *arXiv:1508.05550 [cs, stat]*, August 2015.

[70] Ofir Lindenbaum, Moshe Salhov, Arie Yeredor, and Amir Averbuch. Kernel scaling for manifold learning and classification. *arXiv preprint arXiv:1707.01093*, 2017.

[71] Andrew Y Ng, Michael I Jordan, and Yair Weiss. On spectral clustering: Analysis and an algorithm. In *Advances in Neural Information Processing Systems*, pages 849–856, 2002.

[72] Daniel A Spielman and Shang-Hua Teng. Spectral sparsification of graphs. *SIAM Journal on Computing*, 40 (4):981–1025, 2011.

[73] Mikhail Belkin and Partha Niyogi. Laplacian eigenmaps and spectral techniques for embedding and clustering. In *Advances in Neural Information Processing Systems*, pages 585–591, 2002.

[74] David K Hammond, Pierre Vandergheynst, and Rémi Gribonval. Wavelets on graphs via spectral graph theory. *Applied and Computational Harmonic Analysis*, 30 (2):129–150, 2011.

[75] Nathanael Perraudin, Benjamin Ricaud, David Shuman, and Pierre Vandergheynst. Global and local uncertainty principles for signals on graphs. *arXiv preprint arXiv:1603.03030*, 2016.

[76] Nathanaël Perraudin, Johan Paratte, David Shuman, Lionel Martin, Vassilis Kalofolias, Pierre Vandergheynst, and David K. Hammond. GSPBOX: A toolbox for signal processing on graphs. *ArXiv e-prints*, August 2014.

[77] David I Shuman, Pierre Vandergheynst, and Pascal Frossard. Chebyshev polynomial approximation for distributed signal processing. In *Distributed Computing in Sensor Systems and Workshops (DCOSS), 2011 International Conference on*, pages 1–8. IEEE, 2011.

[78] Nathanaël Perraudin, Nicki Holighaus, Peter L Søndergaard, and Peter Balazs. Designing Gabor windows using convex optimization. *arXiv preprint arXiv:1401.6033*, 2014.

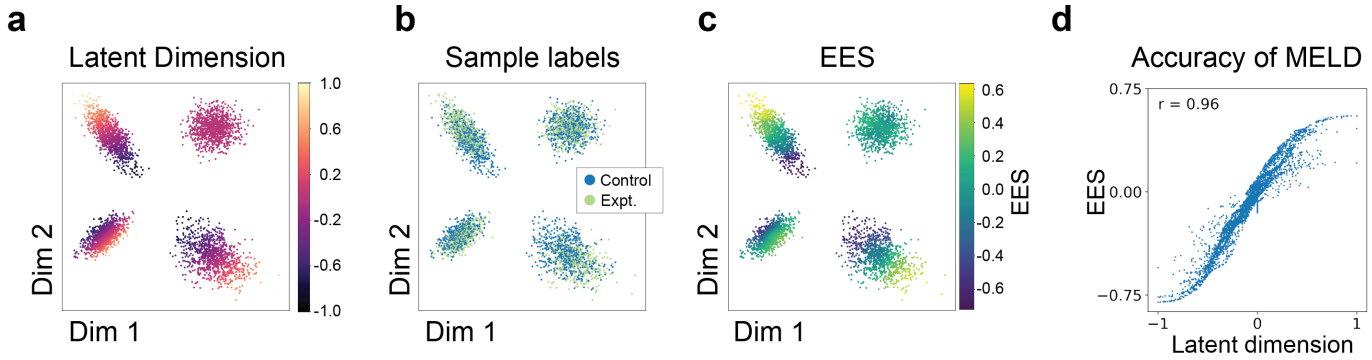

Figure S1: MELD captures the latent experimental signals across clusters. (**a**) In many scRNA-seq experiments, there is not one, but many populations of cells. Each of these populations, or cell types, may respond to an experimental perturbation differently. We simulated four Gaussian clouds of various sizes and densities and created artificial latent dimensions across each population. The scale of this dimension is arbitrarily defined over the interval [-1,1], Note that the axis of greatest variation within a population does not always match the dimension corresponding to the experimental response, as in the lower left cluster. Furthermore, some populations of cells may not respond to the experimental perturbation, as in the upper right cluster. (**b**) To simulate the results of noisy experimental sampling of these cell populations, we assigned experimental labels to cells such that cells with high latent dimension values are more likely to come from the experimental condition and cells with low latent dimension values are likely to come from the control experiment. These labels are used as the Raw Experimental Signal (RES). (**c**) MELD identifies an Enhanced Experimental Signal (EES) in each cluster. (**d**) Comparing the EES to the ground truth Latent Dimension, we find very strong correlation between the EES and the true experimental signal.

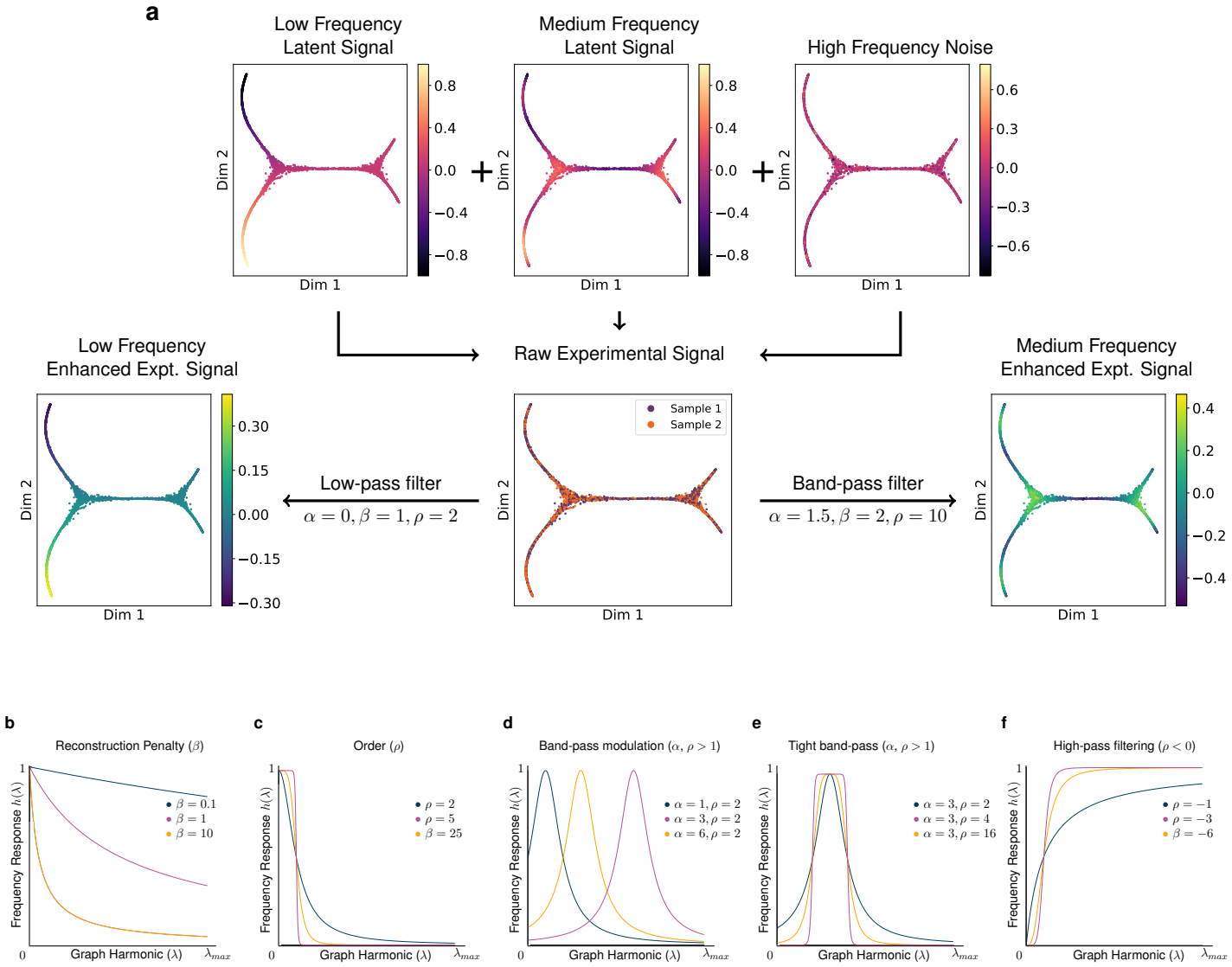

Figure S2: Source Separation and Parameter Analysis with the MELD filter. (**a**) A raw experimental signal (center) is obtained that is a binarized observation of a low frequency latent signal (top left), a medium frequency latent signal (top middle), and high frequency noise (top right). Analysis of the RES alone is intractable as it is corrupted by noise and experimental binarization. MELD low-pass filters (bottom left) to separate a longitudinal trajectory and band-pass filters (bottom right) to yield the periodic signature of the medium frequency latent signal. Parameters used for this analysis are supplied beneath the corresponding arrows. (**b**) Reconstruction penalty β controls a low-pass filter. For this demonstration, $\alpha = 0, \rho = 1$. This filter is equivalent to Laplacian regularization. (**c**) Order ρ controls the filter squareness. This parameter is used in the low-pass filter of (**a**). For this demonstration, $\beta = 1, \alpha = 0$. (**d**) Band-pass modulation via α. When ρ is even valued, α modulates the central frequency of a band-pass filter. This parameter is used in (**a**) to separate a medium-frequency source from a low-frequency source. (**e**) α and ρ combine to make square band-pass filters. For (**d**) and (**e**), $\beta = 1$. (**f**) Negative values of ρ yield a high-pass filter. For (**b-f**), Laplacian harmonics for a general normalized Laplacian are plotted on the x-axis. The frequency response of the filter given by the colored parameters is on the y-axis.

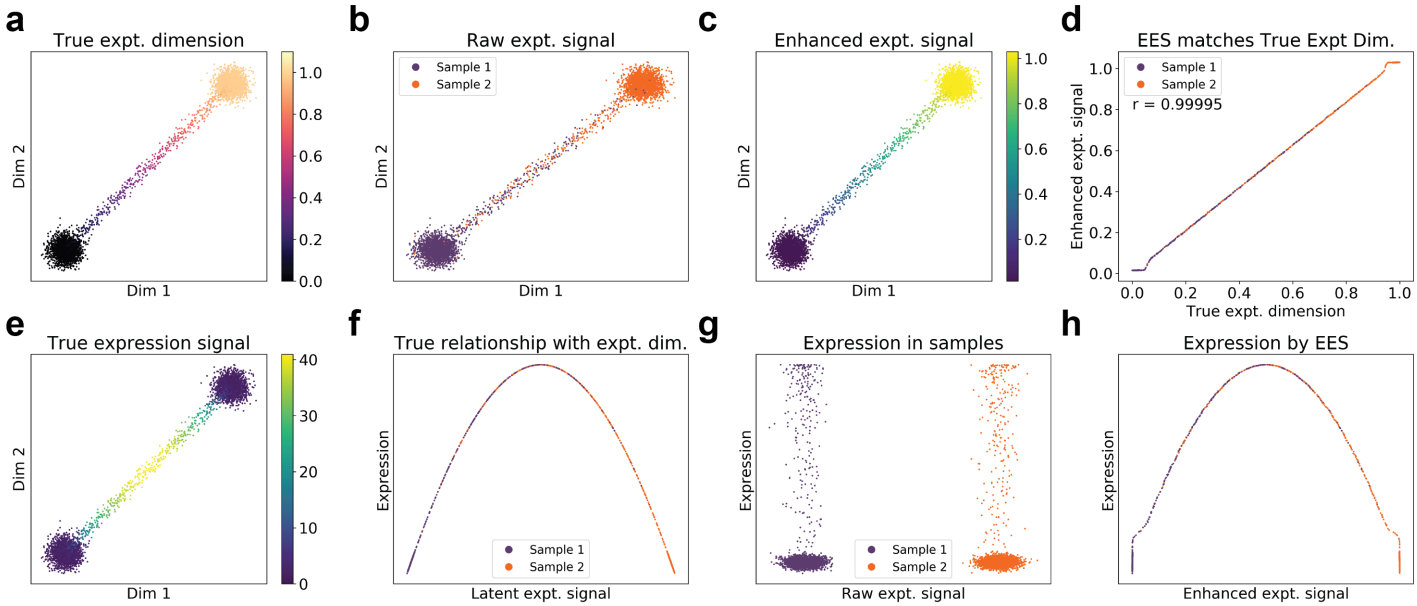

Figure S3: MELD can capture a ground-truth non-linear gene expression signature. (**a**) To demonstrate the ability for MELD to capture non-linear, and non-monotonic gene expression signatures, we simulated a simple 100-dimensional datatset with two terminal cell states connected by an intermediate, transitional spectrum of cells with added noise. The true latent experimental dimension (corresponding to the progression instigated by an experimental condition) is a smooth progression from the left to the right terminal cell state. (**b**) To simulate the results of noisy experimental sampling of these cell populations, we assigned experimental labels to cells such that cells with high latent dimension values are more likely to come from the experimental condition and cells with low latent dimension values are likely to come from the control experiment. These labels are used as the Raw Experimental Signal (RES). (**c**) MELD identifies an Enhanced Experimental Signal (EES). (**d**) Comparing the EES to the ground truth experimental dimension, we find very strong recovery of the true experimental signal. (**e**) To simulate a non-linear gene expression pattern of a single gene, we created an artificial gene expression signal that is low in the terminal cell states, but peaks in the intermediate transitional cells. (**f**) Plotting the expression of the artifical gene as a function of the true experimental dimension, we can observe the non-linear nature of the artificial expression signal. (**g**) Using only the sample labels to characterize expression of this gene in our simulated dataset, we observe no difference in expression of the gene between conditions. (**h**) Only when plotting the expression as a function of the enhance experimental signal can we observe the non-linear nature of the expression. This would be hidden without MELD.

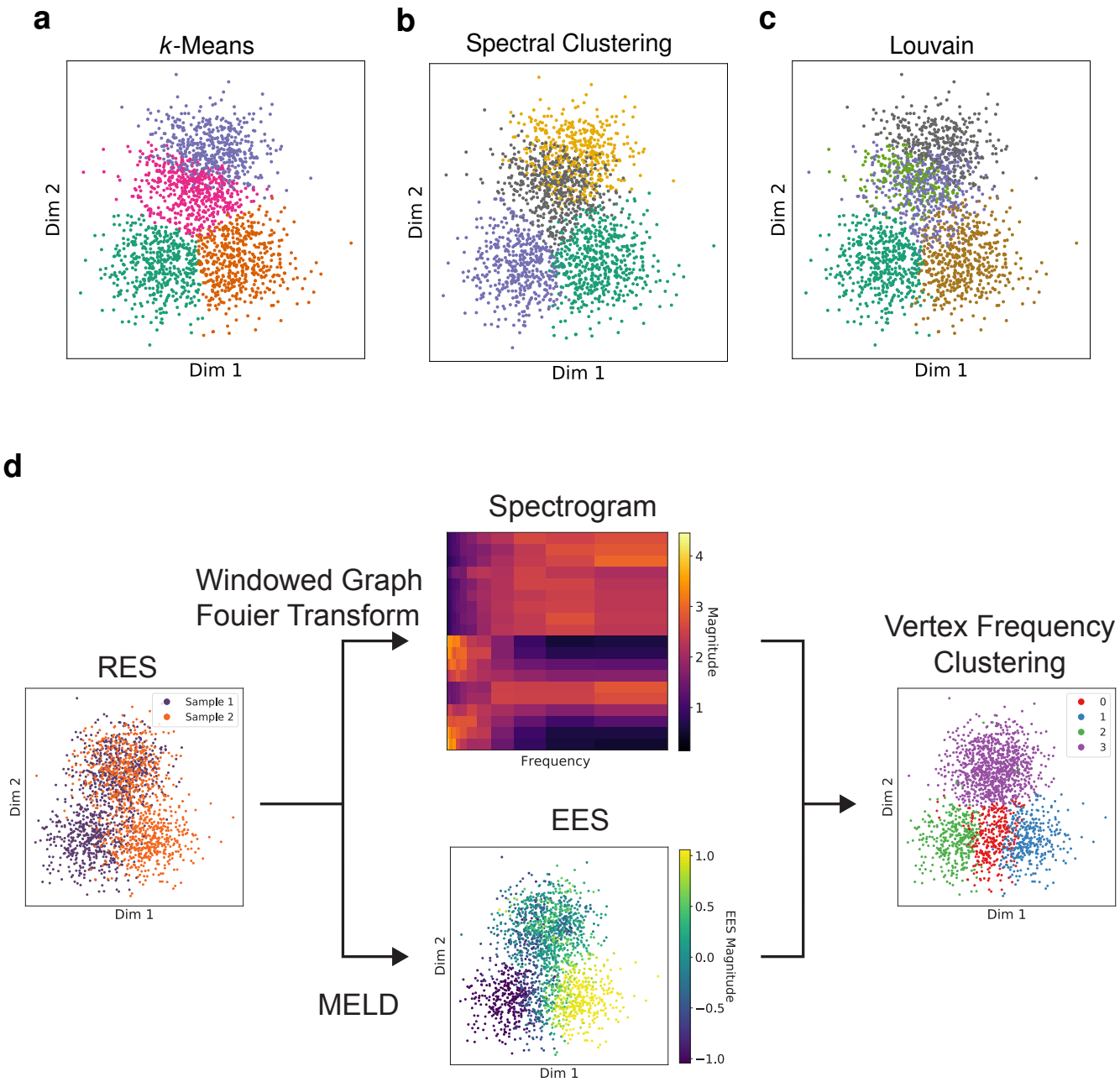

Figure S4: Vertex-Frequency clustering with MELD. A Gaussian mixture model was used to generate N = 1000 points in a mixture of three Gaussian distributions. This experiment is representative of a two-cell type experiment (split by Dim 2) in which one sample changes (bottom clusters) along Dim 1 due to the experiment while the other remains mixed (top clusters). (**a**) *k*-Means clustering separates the left and right experimental groups but splits the upper group erroneously. (**b**) Spectral clustering replicates the performance of *k*-Means in this example. (**c**) Louvain modularity clustering splits the mixture into five groups, with the same lower separations as before but with three groups in the upper cell type. **(d)** Vertex-Frequency clustering recovers a new cluster type. Briefly, the RES (left) is used for (1) a windowed graph Fourier Transform to obtain vertex-frequency information (above, logarithmically downsampled for clarity) and (2) MELD, which generates a continuous profile of the simulated experimental effect. These measures are concatenated together and clustered with *k*-Means. The clusters (right) separate the two cell types (purple and green/red/blue), and finds a separate grouping of cells that are in transition from green to blue, shown in red. One may see that in the spectrogram the green and blue groups are found on relatively low frequency patterns (bottom half of spectrogram, mostly black bands), whereas the medium frequency transition is well separated (middle of bottom bands). The well-mixed, nonresponsive population is entirely high frequency (top half).

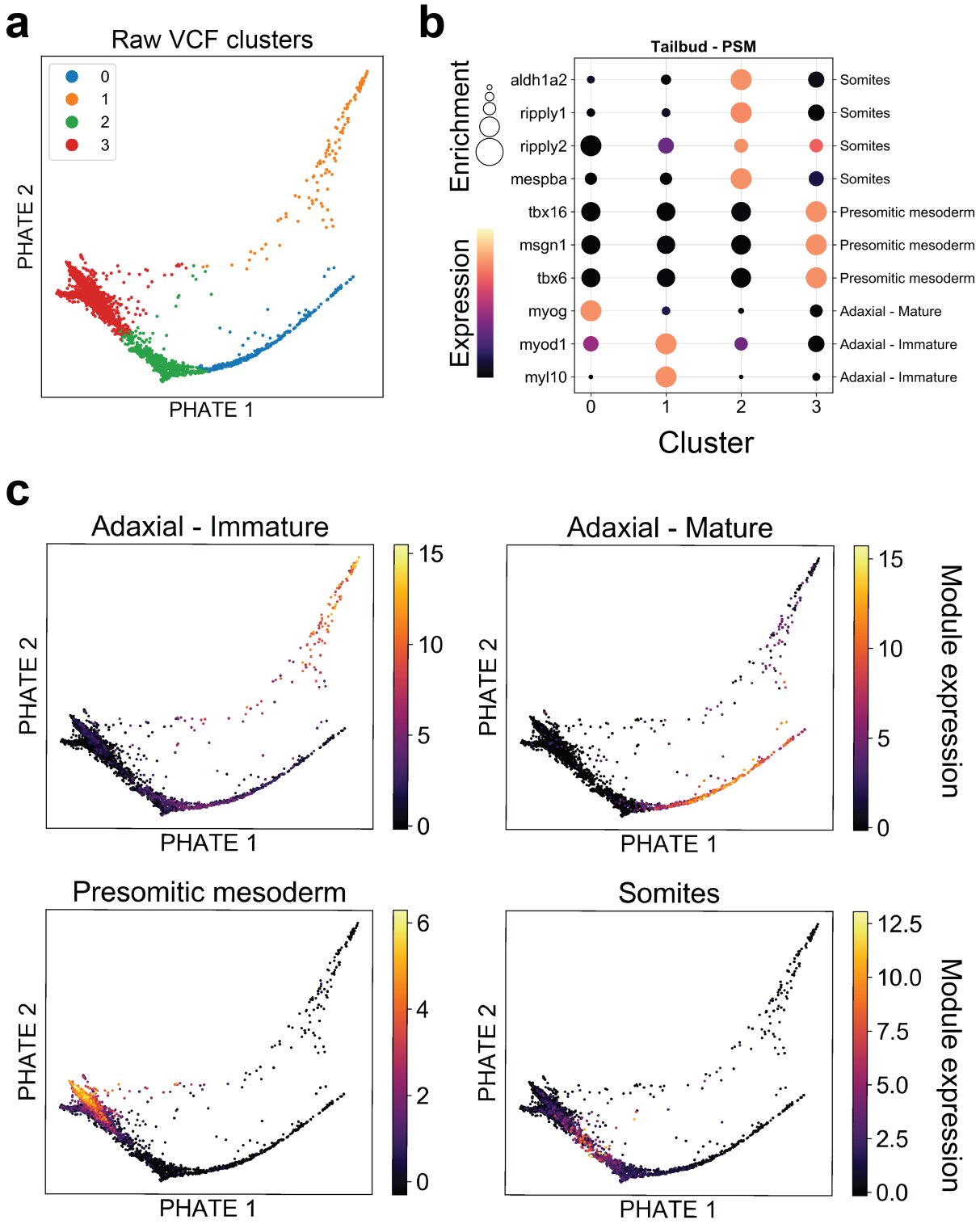

Figure S5: Characterization of vertex-frequency clusters in the Tailbud - Presomitic Mesoderm Cluster (**a**) Raw vertex-frequency cluster assignments on a PHATE visualization. (**b**) Normalized expression of previously identified marker genes of possible subtypes of the Tailbud - Presomitic Mesoderm[13]. The color of the dot for each gene in each cluster indicates the expression level after MAGIC and the size of the dot corresponds to the normalized Wasserstein distance between expression within cluster to all other clusters. (**c**) Average z-score transformed expression of genes associated with each cell type is plotted on a PHATE visualization of the Tailbud - Presomitic Mesoderm Cluster.

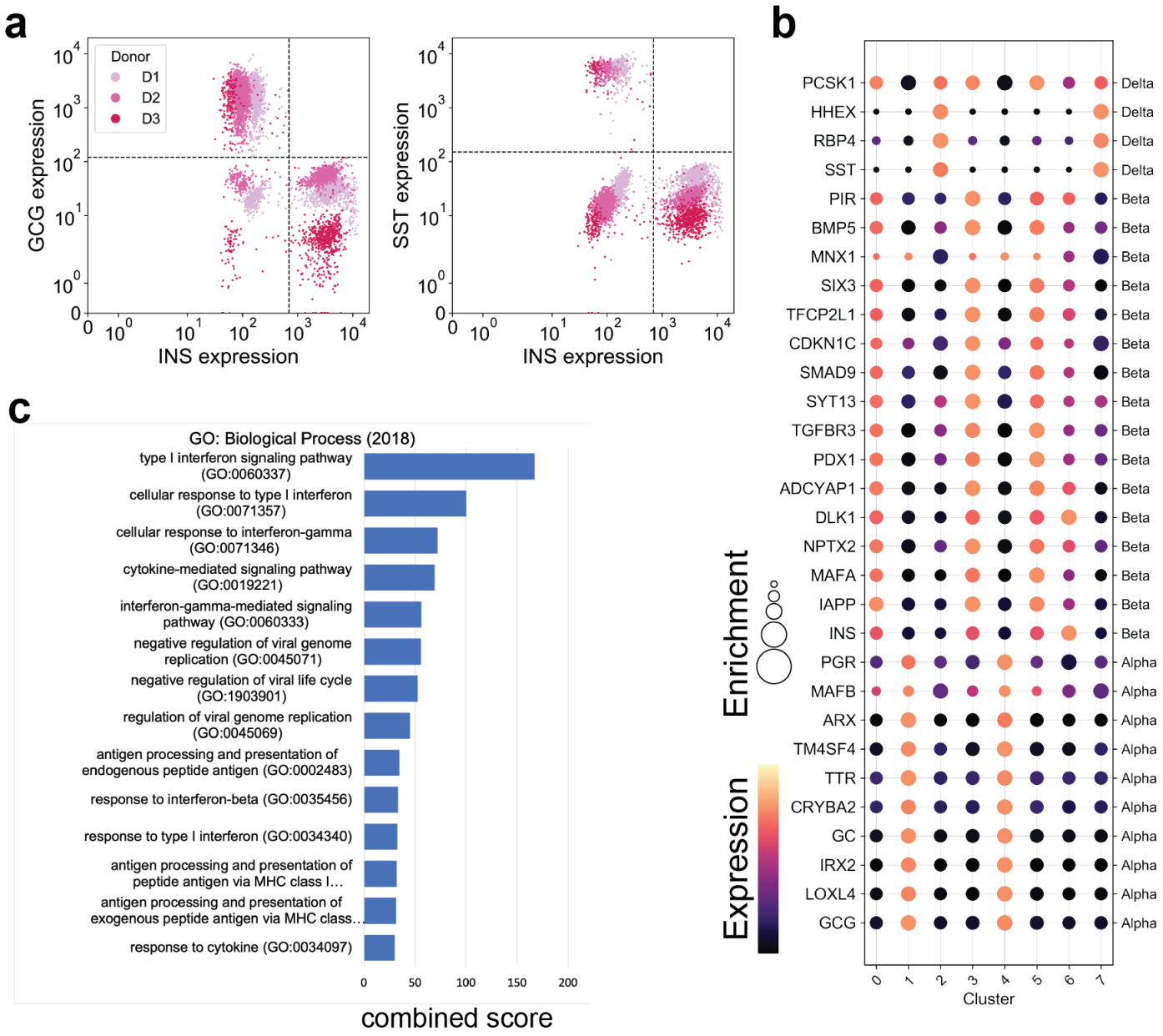

Figure S6: Analysis of pancreatic islet cells from three donors. (**a**) Library-size normalized expression of insulin (INS), glucagon (GCG), and somatostatin (SST) shows donor-specific batch effect across islet cells. (**b**) Normalized expression of previously identified marker genes of alpha, beta, and delta cells[52] in each cluster. The color of the dot for each gene in each cluster indicates the expression level after MAGIC and the size of the dot corresponds to the normalized Wasserstein distance between expression within cluster to all other clusters. (**c**) Results of Enrichr[37] gene set enrichment of 491 signature genes identified in at least one cell-type shows strong enrichment for genes in the interferon signalling pathways.