CALISTA: Clustering And Lineage Inference in Single-Cell Transcriptional Analysis

Nan Papili Gao[1,2], Thomas Hartmann[1], Rudiyanto Gunawan[1,2]
[1]Institute for Chemical and Bioengineering, ETH Zurich, 8093 Zurich, Switzerland
[2]Swiss Institute of Bioinformatics, 1015 Lausanne, Switzerland

Abstract

Recent advances in single-cell transcriptional profiling technologies provide a means to shed new light on the dynamic regulation of physiological processes at the single-cell level. Single-cell gene expression data taken during cell differentiation have led to novel insights on the functional role of cell-to-cell variability. Here, the clustering of cells, the reconstruction of cell lineage progression and developmental trajectory, and the identification of key genes involved in the cell fate decision making, are among the most common-yet-challenging data analytical tasks. In this work, we developed CALISTA (Clustering And Lineage Inference in Single-Cell Transcriptional Analysis) for clustering and lineage inference analysis using single-cell transcriptional profiles. CALISTA adopts a likelihood-based approach using the two-state model of stochastic gene transcriptional bursts to capture the heterogeneity of single-cell gene expression. We demonstrated the efficacy of CALISTA by applying the method to single-cell gene expression datasets (RT-qPCR and RNA-sequencing) from four cell differentiation systems without any lineage bifurcation and with one or multiple lineage bifurcations. CALISTA is freely available on https://github.com/CABSEL/CALISTA.

1. Introduction

The differentiation of stem cells into multiple cell types relies on the dynamic regulation of gene expression [1]. In this regard, advances in single-cell gene transcriptional profiling techniques have given a tremendous boost in elucidating the decision making process governing stem cell commitment to different cell fates (see a recent review on single-cell transcriptomics technologies[2]). The applications of single-cell transcriptional analysis have led to new insights on the functional role of cell-to-cell gene expression heterogeneity in the physiological cell differentiation process [3–7]. Along with the surge in single-cell transcriptional profiling studies, algorithms for analyzing single-cell transcriptomics data have received much attention due to new challenges in the data interpretation. In comparison to measurements from aggregate samples of cell population, single-cell gene expression profiles display much higher variability, not only due to technical reasons, but also because of the intrinsic stochastic (bursty) dynamics of the gene transcriptional process [8]. In particular, the stochastic gene transcription has been shown to generate highly non-Gaussian distributed expression data [9], which further complicate data analysis using standard methods that rely on a standard noise distribution (e.g. Gaussian).

Numerous algorithms have recently been developed specifically for the analysis of single-cell gene expression data. A class of these computational algorithms is geared toward characterizing cell types or states (clusters) within a heterogeneous cell population based on single-cell expression data. Traditional clustering algorithms such as *k*-means and hierarchical clustering have been applied for such a purpose [10–12]. Several other single-cell clustering strategies, such as CIDR [13], pcaREDUCE [14], scLVM [15] and SNN-cliq [16], adapt more advanced algorithms such as principal component analysis (PCA) or nearest neighbors search. Time-variant clustering strategies have also been implemented to elucidate the appearance of cell types in lineage progression [17,18]. Consensus clustering methods, such as SC3 [19] and SIMLR [20], have also received much interest thanks to their superior stability and robustness. Finally, a recent likelihood-based method called SABEC (Simulated Annealing for Bursty Expression Clustering) [21] employs a model of the bursty stochastic dynamics of gene transcriptional process to cluster cells, such that the gene expression distribution of each cell cluster conforms with the analytical distribution governed by the model.

Another important class of algorithms for single-cell gene expression data analysis deals with the reconstruction of lineage progression during cell differentiation process and the pseudotemporal ordering of single cells along the cell developmental path(s). The lineage progression graph describes the transition of stem cells through developmental stages during the cell differentiation. The lineage progression may comprise a single developmental path into one cell fate or bifurcating paths into multiple cell fates. In these algorithms, the reconstruction of the lineage progression and developmental paths is commonly implemented for the purpose of pseudotemporal cell ordering. The pseudotime of a cell represents the relative position of the cell along the developmental path, and is typically normalized to be between 0 and 1. By plotting the gene expression against the pseudotime of the cells, one obtains the trajectory of the gene expression which reflects the gene regulation driving the cell fate decision making. Numerous algorithms, such as DPT [22], SCUBA [17], and MONOCLE [23,24], have been developed for single-cell transcriptional data analysis for cell lineage inference and cell ordering (for further details see a recent review by Canoodt et al. [25]).

In this work, we developed CALISTA (Clustering And Lineage Inference in Single Cell Transcriptional Analysis). CALISTA provides three types of analysis of single-cell gene transcriptional profiles, namely cell clustering, lineage progression inference, and pseudotemporal cell ordering. CALISTA adopts a likelihood-based strategy using the two-state model of the stochastic gene transcription process [26]. Like a previous clustering method SABEC, the single-cell clustering in CALISTA is based on the assumption that cells belonging to a cluster have the same cell state. More specifically, the gene expression distribution of each cell cluster (state) follows an analytical distribution derived from the two-state model with a particular set of parameters. In comparison to SABEC[21], CALISTA offers several orders of magnitude of computational speed-up through the implementation of a greedy algorithm. Based on the cell clustering result, CALISTA generates the cell lineage progression graph using cluster distances, a likelihood-based measure of dissimilarity between cell clusters. Moreover, for any two connected cell clusters (states) in the lineage progression graph, CALISTA provides the set of transition genes, which represent highly informative gene markers and potential regulators of the cell state transition. Given a lineage progression graph, CALISTA subsequently assigns cells to state transition edges and evaluates the cell pseudotimes. Finally, CALISTA generates gene expression profiles along a developmental path – a sequence of state transition edges in the lineage progression graph – in the lineage progression by ordering of the cells belonging to the path according to their pseudotimes.

In the following, we described the application of CALISTA to single-cell transcriptional profiles taken during the differentiation of human induced pluripotent stem cells (iPSCs) into either a desired mesodermal (M) or an undesired endodermal (En) fate [27]. In the supplementary material, we further included the results of CALISTA application to the analysis of four single-cell transcriptional datasets from single-cell RTqPCR and single-cell RNA-sequencing (scRNA-seq) experiments. In these case studies, CALISTA was able to reconstruct cell lineage progressions with multiple lineage bifurcations, a particularly challenging task in single-cell gene expression analysis that often requires specialized algorithms [17,22,28]. We further compared the performance of CALISTA to frequently used algorithms for clustering and pseudotemporal cell ordering, including SIMLR [20], SC3 [19], DPT [22], SCUBA [17], and MONOCLE2 [23,24].

2. Results

2.1 General framework of CALISTA

Figure 1 summarizes the analysis of single-cell transcriptional profiles in CALISTA, which comprises three main elements: (1) single-cell clustering, (2) reconstruction of lineage progression, and (3) pseudotemporal ordering of cells. In analyzing single-cell expression data, CALISTA adopts a likelihood-based approach, where the likelihood of a cell is defined by the joint probability of the measured gene expression according to the steady-state probability distribution derived from the two-state model [26] (see Methods). As illustrated in Figure 1b, the single-cell clustering in CALISTA involves two steps: (1) independent runs of maximum likelihood clustering using a numerically efficient greedy algorithm, and (2) consensus clustering using *k*-medoids [29]. The maximum likelihood clustering in the first step starts with a random assignment of cells into a desired number of clusters, followed by iterations of (a) maximum likelihood estimation of two-state model parameters for each gene and each cluster and (b) reassignments of each cell to the cluster that give the largest cell likelihood value, until no more cell reassignments occur or until a prescribed maximum number of iterations is reached. When the number of clusters is not known *a priori*, CALISTA provides the eigengap plot based on which users can choose the appropriate number of clusters in their dataset (see Methods). The likelihood-based single-cell clustering in CALISTA parallels the clustering algorithm SABEC [21], but offers a substantial improvement in computational speed due the difference in the implementation of the maximum likelihood clustering (see Supplementary Table 1).

By viewing the single-cell clusters as cell states through which stem cells transition during the cell differentiation, CALISTA uses distances among the cell clusters to reconstruct a lineage progression graph. The cluster distance between any two clusters – defined as the maximum difference in the cumulative likelihood value upon reassigning the cells from the original cluster to the other cluster (see Methods) – gives a measure of dissimilarity in their gene expression distributions. CALISTA generates the lineage progression graph by sequentially adding state transition edges connecting closely distanced clusters until every cell cluster is connected to at least another cluster (Figure 1c). Users can further curate the lineage progression graph by manually adding or removing state transition edges based on the cluster distances and available information about the cell differentiation system. For assigning directionality to the edges, CALISTA relies on user-provided information, for example information on the cell stage or sampling time, the starter/progenitor cells or the expected temporal profiles of the gene expression. Furthermore, for any two connected clusters in the lineage progression, CALISTA provides the set of transition genes based on gene-wise likelihood differences between having the cells in separate cluster and having them together in a single cluster (see Methods). Here, the gene-wise likelihood difference reflects the informative power of a gene to discriminate cells between two clusters. The transition genes represent candidates of gene markers and may also point to the genes regulating for the state transition.

The final element of CALISTA concerns the pseudotemporal ordering of cells along a user-defined developmental path, i.e. a sequence of connected clusters and state transition edges, in the lineage progression graph (Figure 1d). For this purpose, we first set the pseudotime for each cluster, which is normalized such that the starting cluster in the lineage progression graph has a value of 0 and the final cell cluster (or clusters) has a value of 1 (see Methods). Subsequently, we assign each cell to a transition edge pointing to or emanating from the cluster to which the cell belongs, again by adopting a maximum likelihood principle. More specifically, we assume that the distribution of the single-cell gene expression varies monotonically between cell states (clusters) in the lineage progression. The likelihood of a cell along a transition edge is computed using a linear interpolation of the cell likelihood values for the connected clusters. Each cell is then assigned to the transition edge that maximizes its likelihood value. Anologously, the cell pseudotime is computed by a linear interpolation of the cluster pseudotimes, and set to the corresponding maximum point of the cell likelihood value. Finally, for any user-defined developmental path, CALISTA identifies the cells belonging to the transition edges in the path and orders the cells in increasing pseudotime.

2.2 Application of CALISTA to iPSC to cardiomyocyte differentiation

In the following, we applied CALISTA to analyze single-cell gene expression dataset describing the differentiation of human induced pluripotent stem cells (iPSCs) into cardiomyocytes [27]. In the original study, Bargaje et al. measured the expression level of 96 genes by RT-qPCR for 1896 cells collected across 8 time points (day 0, 1, 1.5, 2, 2.5, 3, 4, 5) after induction to differentiate. Figure 2a shows the four previously reported developmental states in the differentiation system: epiblast cells (E) in the early stage (day 0, 1, 1.5), primitive streak (PS)-like progenitor cells in the intermediate stage (day 2, 2,5), and a lineage bifurcation into either a desired mesodermal (M) or an undesired endodermal (En) fate in the late stage (day 3, 4, 5).

First, we clustered the cells by using *k*-means algorithm and CALISTA without using the time information. The optimal number of clusters was chosen to be five based on the eigengap plot (see Methods and Supplementary Figure 1). As depicted in Figure 2b, the single-cell clustering by *k*-means is incongruent with the known developmental states and with the capture times. One of the cell clusters (cluster 3 in Figure 2b, green dots) includes cells from both early (E) and late stage (En and M) cells. On the other hand, the single-cell clustering by CALISTA as shown in Figure 2c, recapitulates the previously identified developmental states. Here, cluster 1, 2, and 5 contain mostly E, PS, and En cells, respectively, while the M cells are split between two clusters (cluster 3 and 4), demarcated by different capture times (see cluster composition in Supplementary Figure 2).

After single-cell clustering, we employed CALISTA to infer the lineage progression graph by calculating the cluster distances and by adding state transition edges in decreasing magnitude of the distances until each cluster was connected by at least one other cluster. An edge connecting cluster 1 and 5 was manually removed as the edge was inconsistent with the cell capture time information (i.e., bypassing intermediate capture time points). The cluster pseudotime was set to the mode (most frequent value) of the cell capture times in the clusters normalized by the maximum capture time. The directionality of the state transition edges was set according to the cluster pseudotimes (also see Supplementary Figure 3). As illustrated in Figure 2d, the lineage progression graph reconstructed by CALISTA accurately replicates the lineage bifurcation event as the cells transition from PS-like cells to take on either M or En cell fates [27]. Subsequently, for each state transition edge, we identified the set of transition genes (see Supplementary Figure 4). Among the identified transition genes in the lineage progression graph (25 genes in total), many are known lineage specific transcriptional regulators involved in the iPSC cell differentiation, such as EOMES, GATA4, GSC, HAND1, KIT, MESP1, SOX17 and T [27].

In addition, as a measure of cell-to-cell variability, we evaluated the Shannon entropy of the gene expression distribution in each cluster, referred to as cluster entropy (see Methods). A higher cluster entropy indicates that the cells in the cluster display broader and more uncertain gene expression distribution [7]. As depicted in Figure 2e, the cluster entropy rises at the beginning of the lineage progression during the transition from E to PS-like cells, and then decreases as the cells progress into more differentiated M and En states. The cluster entropy reaches a peak value for PS-like progenitor cells (~ day 2), which coincides with the lineage bifurcation event [27]. The rise-and-fall of the cluster entropy agrees well with published reports from other cell differentiation systems, suggesting that stem cells go through a transition state of high uncertainty before committing to a particular cell fate [7,12].

Finally, we employed CALISTA to generate the pseudotemporal ordering of cells along the developmental paths in the lineage progression. We identified two cell developmental trajectories in the lineage progression: (1) the M path forming mesodermal cells (cluster 1 – 2 – 3 – 4, red path in Figure 2d) and (2) the En path forming endodermal cells (cluster 1 – 2 – 5, green path in Figure 2d). After (re)assigning cells to the state transition edges and prescribing cell pseudotimes, we ordered cells belonging to each developmental trajectory in increasing pseudotimes, for a total of 1408 cells in the M path and 1215 cells in the En path. Furthermore, to visualize the trajectories of the gene expression along the two cell differentiation paths, we calculated the moving average expression values of the transition genes for the pseudotemporally ordered cells using a moving window of 190 cells (see Figure 2g-i and Supplementary Figure 5).

A number of transition genes follow the same dynamic expression profile along the M and En paths, with an upregulation from E to PS-like transition, followed by a downregulation from PS-like to M or En transition (see Figure 2g). The majority of genes with the aforementioned trajectory are known PS-like markers (for example EOMES, GSC, MESP1, and MIXL1 [27,30,31]), and thus, we refer to these genes as PS-genes (Supplementary Table 2). Another group of transition genes show differential profiles after the bifurcation in the lineage progression (i.e., after PS-like state). Here, we define M-genes as the genes with higher expressions along the M path than the En path (see Figure 2h). Correspondingly, we define En-genes as genes with higher expressions along the En path than the M path (see Figure 2i). Notably, many of the known M marker genes (e.g., BMP4, HAND1, MYL4 and TNNT2 [27,32]) are among the M-genes (see Supplementary Table 2), and several of the known En marker genes (e.g. HNFA4, KIT, and SOX17 [27,33,34]) are among the En-genes (see Supplementary Table 2). In general, after the lineage bifurcation, the expressions of M genes are upregulated along the M path, but are either downregulated (BMP4, HAND1, KDR) or unchanged (MYL4, TGFB2, TNNT2) along the En path. Among the En genes, the expression of HNFA4 increases along the En path after the lineage bifurcation, but remains relatively unchanged along the M path. The expression of KIT follows the opposite profile, where the gene expression is downregulated along the M path and is relatively unchanged along the En path. While the expression of SOX17, like KIT, decreases along the M path, the gene is upregulated along the En path immediately after the lineage bifurcation before being downregulated toward the end of the developmental trajectory.

We subsequently constructed gene co-expression (correlation) networks based on the moving average gene expression profiles for the M and En developmental paths (pairwise Pearson correlation, p-value ≤ 0.01 and correlation value ≥ 0.8, see Supplementary Figure 6). We further identified cliques in the M and En gene co-expression networks, i.e. a subset of genes (at least 5) that are connected to each other, using a maximal clique analysis by the Bron-Kerbosch algorithm [35]. Figure 2j depicts the cliques from the M and En gene co-expression networks, showing three distinct gene regulatory modules: one module specific to the M path (red), another specific to the En path (green), and finally a shared module (grey). Here, most of the PS genes, including known PS marker genes, belong to the shared module. Besides, the shared module includes DKK1, FZD1 and T, regulators of Wnt signaling pathway, which have been shown to promote cell differentiation [36]. Meanwhile, GATA6, which plays an important role in the endoderm commitment [30], is in the En module. Finally, the M module shows activating relationships among mesoderm markers (e.g. BMP4, MYL4 and HAND1), and antagonist interactions between several M genes and KIT, an En gene and marker.

We further evaluated the performance of CALISTA using three publicly available RT-qPCR and RNA-seq datasets [11,37,38] (see Supplementary Note 1). Notably, for each of the datasets, CALISTA is able to produce accurate clustering results that are consistent with the known population substructures, and to infer single[11] or multiple [37,38] bifurcation events in the lineage progression.

2.3 Comparison to existing methods

In comparison to single-cell clustering algorithm SABEC [21], the clustering procedure in CALISTA converges substantially faster, while producing comparable clustering results (see Supplementary Fig. 10). For example, in the application to single-cell expression data from hematopoietic stem cell (HSC) differentiation [38], 50 repeats of SABEC took roughly 28 hours to complete, while our greedy algorithm converged to the final consensus-based solution in seconds on the same computer workstation (see Table 1).

We also compared the performance of CALISTA to three additional clustering algorithms, namely SIMLR[20], SC3[19] and MONOCLE 2[28]. SIMLR and SC are consensus-clustering methods that learn cell-cell similarities by using multiple kernels and Laplacian transformations, respectively. On the other hand, MONOCLE 2 is a software tool specifically designed for single-cell gene expression data analysis that also includes, among other features, a single-cell clustering algorithm using a *t*-distributed method. For the comparison of performance, we considered four publicly available scRNA-seq datasets[39–42], including Kolodziejczyk et al. study on the differentiation of pluripotent stem cells under different conditions [39]; Pollen et al. study on the identification of different cell types within a heterogeneous human fetal population [43]; Usoskin et al. study using single mouse neural cells [41]; and Villani et al. study on the detection of human blood dendritic cell subpopulations [42]. Here, the reference cell identities were taken from single-cell labeling in the original studies [39,41–43]. To assess the clustering accuracy of each method, we calculated the Adjusted Rand Index (ARI)[19,44], which ranges between 0 (fully incorrect labels) and 1 (fully correct labels).

Table 1A compares the ARIs of SIMLR, SC3, MONOCLE 2, and CALISTA for the four datasets mentioned above (see Supplementary Table S3 for cell clustering results). The results show comparable clustering performance among all of the algorithms for two of the studies (Kolodziejczyk et al.[39] and Pollen et al.[40] datasets), with high accuracy (ARI> 0.85). Meanwhile, for the remaining two studies (Usoskin[41] and Villani[42] datasets), CALISTA and SC3 outperform SIMLR and MONOCLE 2.

Next, we compared the performance of CALISTA in reconstructing the lineage progression with three existing algorithms, namely Diffusion Pseudotime (DPT) [22], SCUBA [17], and MONOCLE 2 [28], that are able to infer linear and branched trajectories. SCUBA and MONOCLE 2 are able to detect multiple bifurcation events, while DPT is designed for the identification of a single branching point [22]. Of note, like CALISTA, MONOCLE 2 reconstructs the lineage progression without any user inputs. In contrast, users have to specify information on the topology of the lineage progression (linear or branched) in DPT and to provide the cell stage information in SCUBA. We applied the algorithms to single-cell gene expression datasets from four studies characterized a variety of lineage topologies, including Bargaje et al. study above [11]; Chu et al. study [37] on the differentiation of human embryonic stem cells into endodermal cells; Moignard et al. study [38] on hematopoietic stem cell differentiation; and Treutlein et al. study [11] on mouse embryonic fibroblast differentiation into neurons. We excluded SCUBA for one of the datasets (from Moignard et al. study [38]) since the capture time information was not available. Table 1B summarizes the results of the reconstruction of lineage progression. For each of the datasets, CALISTA was able to accurately recover the lineage progression with single or multiple bifurcations (see Figure 2 and Supplementary Figure 7). Conversely, DPT, SCUBA and MONOCLE 2 had difficulties in reconstructing lineage progression without bifurcation (e.g., inconsistency with cell capture times or identification of spurious bifurcation) as well as that with multiple bifurcations (see Supplementary Figures 11-13).

2.4 Algorithmic stability

In order to assess the robustness of CALISTA results, we performed repeated independent runs of CALISTA using a random subsample of cells from Bargaje et al. [27], Chu et al. [37], Moignard et al. [38], and Treutlein et al. [11] datasets, using the same settings used in the analysis of the full dataset. We then evaluated the similarity between the outcome of CALISTA single-cell clustering of the subsamples with that of the original full dataset, by computing the figure of merit (FOM)[45] and the area under the Receiver Operating Characteristic (AUROC) and the Precision-Recall curve (AUPR)[46]. Notably, the performance of CALISTA decreased only marginally with decreasing cell population in the random subsample between 60% and 90% of the original dataset (see Supplementary Figure 14 and Supplementary Table 4).

3. Methods

3.1 Input data

The single-cell gene expression matrix should be formatted into an $N\times G$ matrix, where G denotes the number of genes, N denotes the number of cells, and the matrix element $m_{n,g}$ is the transcriptional expression value of gene g in the n-th cell. CALISTA accepts expression values from RT-qPCR (2^{C_t} value) and RNA-sequencing (i.e. log(RPKM) or log(TPM)). In preparing the input matrix, the single-cell gene expression data should first be normalized [3,47] and transformed into the appropriate expression values [48,49]. When the cells come from different stages or time points, the stage or time information can also be passed into CALISTA, which will be used for determining lineage progression.

Before performing single-cell analysis, CALISTA first preprocesses the single-cell expression matrix by removing the genes and cells (i.e., rows and columns of the expression matrix, respectively) with a large fraction of zero expression values, exceeding a user-defined threshold (default threshold: 90% for genes and 100% for cells). When the number of genes G is much higher than the number of cells N (for example, in the case of RNA-seq datasets), CALISTA further selects a number of informative genes Y (default $Y = G/2$ or 200, whichever is smaller) for the single-cell analysis following a previously described procedure [50]. Briefly, CALISTA first calculates gene-wise mean (average) expressions and coefficients of variation (CV) across the cells. The genes are partitioned into 20 bins based on their mean expressions. The CV values of the genes within each bin are subsequently transformed into z-scores. Finally, the genes from the Y largest CV z-scores among all bins are selected for the single-cell analysis.

In the final preprocessing step, CALISTA scales the single-cell expression values such that the maximum value of any gene is 200. The scaling is carried out as follows:

$$\widehat{m}_{n,g} = 2^{\frac{log_2(m_{n,g}+1)}{log_2(m_{max,g})}log_2(200)} \qquad (1)$$

where $m_{n,g}$ and $\widehat{m}_{n,g}$ represent the original and scaled expression value of gene g in the n-th cell respectively, and $m_{max,g}$ is the maximum expression value in gene g (i.e. the maximum of $m_{n,g}$ over all cells). The scaling above allows CALISTA to use a pre-computed table for the maximum likelihood in the clustering analysis, thereby reducing the computational cost significantly (see **Cell Clustering** below). We tested using *in silico* datasets and confirmed that the scaling of the expression values does not affect the clustering accuracy (see Supplementary Note 2).

3.2 Cell Clustering

As illustrated in Figure 1b, CALISTA combines maximum likelihood and consensus clustering algorithms for single-cell clustering analysis. Following a previous method SABEC, CALISTA uses the stochastic two-state model of gene transcription to derive the steady-state probability distribution of gene expression values. The probability distribution is used to prescribe a likelihood value to a cell based on the measured single-cell expression values of its genes [21]. In contrast to SABEC that employs a simulated annealing algorithm, CALISTA implements a numerically efficient greedy algorithm to solve the likelihood maximization problem. By repeatedly running the greedy algorithm, each starting from a different random initial cell assignment, CALISTA generates a consensus matrix containing the number of times that any two cells are assigned to a cluster. Finally, CALISTA uses the consensus matrix to cluster cells by k-medoids.

Stochastic two-state gene transcriptional model

The two-state model of stochastic gene transcriptional process in CALISTA follows the model developed by Peccoud and Ycart [26]. The model describes a promoter that switches stochastically between an OFF (inactive or non-permissive) state, where the gene transcription cannot start, and an

ON (active or permissive) state, where the gene transcription can proceed. The set of reactions describing the stochastic gene transcription in the two-state model are as follows:

$$OFF \xrightarrow{\theta_{on}} ON$$

$$ON \xrightarrow{\theta_{off}} OFF$$

$$ON \xrightarrow{\theta_t} ON + mRNA$$

$$mRNA \xrightarrow{\theta_{dm}} \emptyset$$

where θ_{on} is the rate of the promoter activation, θ_{off} is the rate of the promoter inactivation, θ_t is the rate of mRNA production when the promoter is active, θ_d is the rate constant of mRNA degradation, and m denotes the number of mRNA molecules. At steady state, the probability distribution of mRNA count m can be approximated by the following density function:[9]

$$P(m;\ \theta_{on},\theta_{off},\theta_t) = \frac{\Gamma(\theta_{on}+m)\Gamma(\theta_{on}+\theta_{off})\theta_t^m}{\Gamma(m+1)\Gamma(\theta_{on}+\theta_{off}+m)\Gamma(\theta_{on})}\ {}_1F_1(\theta_{on}+m,\theta_{on}+\theta_{off}+m,-\theta_t) \quad (2)$$

where ${}_1F_1$ represents the confluent hypergeometric function of the first kind.

Maximum likelihood clustering by greedy algorithm

As illustrated in Figure 1b, the greedy algorithm for the maximum likelihood single-cell clustering in CALISTA comprises three basic steps:

1. Initialization. Cells are randomly assigned into K clusters with equal probabilities (i.e., uniform random cell assignments).

2. Parameter estimation. For each cluster k (between 1 and K) and each gene g (between 1 and G), CALISTA obtains the parameter vector of the two-state gene transcription model $\theta^*(g,k) = \{\theta_{on}^*,\theta_{off}^*,\theta_t^*\}_g^k$ that maximizes the joint probability of obtaining the gene expression values of the cells in the cluster, as follows:

$$\theta^*(g,k) = arg\max_{\theta} \sum_{n\in N_k} \ln P(\widehat{m}_{ng};\theta) \quad (3)$$

where N_k is the set of cell indices for the cells in cluster k. The probability $P(m;\theta)$ is computed using the steady state probability distribution from the two-state gene transcription model, as given in Eq. (2) [9]. For computational efficiency, as previously done in SABEC [21], the log-probability $\ln P(m|\theta)$ values are pre-computed for m between 0 and 200 and for a set of parameter vectors $\{\theta_{on}^w,\theta_{off}^w,\theta_t^w,\}_{w=1}^W$, producing a $W\times 201$ matrix L. The parameter set consists of all combinations from the following parameter discretization: θ_n=[0.5, 1, 2, 3, 4, 5], θ_{off}=[0.25, 0.5, 1, 2, 3, …, 19], and θ_t = [1, 51, 101, …, 451]. The parameter ranges in CALISTA are adapted from SABEC and selected to cover the parameter combinations for which $P(m|\theta)$ spans a family of steady state distributions of mRNA counts, including negative binomial, Poisson, and bimodal distributions. While the number of parameter combinations (W = 945) in CALISTA is much lower than that in SABEC (W = 39500), our extensive testing indicated that the single-cell clustering in CALISTA performs as well as SABEC at much lower computational requirements.

By pre-computing the log-probability values, the solution to the parameter estimation problem above reduces to finding the parameter combination corresponding to the maximum element of the following vector:

$$L\times r^{k,g} \qquad (4)$$

where $r^{k,g}$ is a 201×1 vector with its i-th element being the number of cells in cluster k with its expression value of gene g rounded to the nearest integer equaling to $\widehat{m}_{n_k,g}$.

3. Cell re-assignment. Once the optimal parameter values for all G genes and all K clusters have been obtained, CALISTA reassigns each cell to its new cluster following a greedy algorithm. More specifically, CALISTA evaluates the likelihood for each cell to belong to a cluster as follows:

$$\Lambda_k(n) = \sum_{g=1}^{G} \ln P\left(\widehat{m}_{n,g}; \theta^*(g,k)\right) \qquad (5)$$

where $\Lambda_k(n)$ denotes the likelihood value of the n-th cell to be in cluster k. We reassign each cell to the cluster that gives the maximum likelihood.

Steps 2 and 3 from the algorithm outlined above are repeated iteratively until convergence, i.e. until there are no new cell reassignments.

Single-cell clustering using consensus matrix

To improve both the robustness of the single-cell clustering, CALISTA executes the iterative greedy clustering algorithm above until convergence for multiple times (by default 50), each starting with a different random initial cell assignment. The results from the independent clustering runs are combined to produce a $N\times N$ symmetric consensus matrix C. The (i,j)-th element of the matrix C gives the number of times that the i-th and j-th cells are clustered together, as illustrated in Figure 1b. CALISTA subsequently applies k-medoids algorithm using the consensus matrix to obtain the final cell clustering assignment [29].

Choosing the number of clusters

As an aid for choosing the appropriate number of clusters K – when the number is not known *a priori* – CALISTA provides the eigengap plot based on the normalized graph Laplacian matrix $A_{N\times N}$ of the consensus matrix C, defined as follows [51]:

$$A = I - B^{-1/2}CB^{-1/2} \qquad (6)$$

where I is the $N\times N$ identity matrix and B is the $N\times N$ degree matrix of C, i.e. the diagonal matrix whose elements represent the row-wise sum of the consensus matrix (i.e., $b_{i,i}=\sum_{j=1}^{N}c_{i,j}$). The eigengap plot is the plot of the eigenvalues λ of the matrix A in an ascending order. The number of clusters K can be heuristically chosen based on a "gap" in the eigenvalues of A, where the lowest K eigenvalues $\lambda_{1\ldots}\ \lambda_K$ values are much smaller than λ_{K+1}.

3.3 Lineage inference

The second main component of CALISTA is the reconstruction of cell lineage graph, which reflects the lineage progression in the differentiation process. Based on the view that cell clusters represent cell states, the nodes of the lineage graph comprise cell clusters, while the edges represent state transitions in the lineage progression. For inferring the lineage graph, CALISTA computes cluster distances based on dissimilarities in the gene expressions among cells from two clusters. Again, CALISTA adopts a likelihood-based strategy using the probability distribution of mRNA from the stochastic two-state gene transcription model to define the cluster distances. Finally, users can use CALISTA to extract the set of transition genes between any two connected clusters in the lineage graph.

Cluster distance

Given K clusters from the single-cell clustering analysis above (or provided by the user), CALISTA evaluates a $K\times K$ dissimilarity matrix S where the element s_{kj} gives the likelihood of cells from cluster k to be assigned to cluster j, computed as follows:

$$s_{kj}=\frac{1}{N_k}\frac{1}{G}\left(\sum_{n_k=1}^{N_k}\sum_{g=1}^{G}\ln P\left(\widehat{m}_{n_k,g};\theta^*(g,j)\right)\right)\quad(7)$$

Note that the diagonal element s_{kk} is the sum of the cell likelihood in the original cluster assignment, i.e. $s_{kk}=\sum_{n\in N_k}\Lambda_k(n)$, and should be larger than other elements s_{kj}, $j\neq k$. The dissimilarity coefficients in the matrix S is subsequently normalized by subtracting each element with the diagonal element of the corresponding rows, as follows:

$$\hat{s}_{kj}=s_{kk}-s_{kj}\quad(8)$$

Since the likelihood always takes on negative values, the normalized dissimilarity coefficient assumes a positive value. A larger $\hat{s}_{kj}$ reflects a higher degree of dissimilarity between the two clusters. The distance between the j-th and k-th cluster, denoted by d_{kj}, is defined as:

$$d_{kj}=max\left(\hat{s}_{kj},\hat{s}_{jk}\right)\quad(9)$$

Lineage graph construction

CALISTA generates a lineage graph by connecting single-cell clusters (states) based on the cluster distances. The lineage graph describes the state transition of stem cells during the cell differentiation process, under the assumption that the transitions occur between closely related cell states (i.e. between clusters with low distances). Briefly, CALISTA starts with a fully disconnected graph of single cell clusters, and recursively adds one transition edge at a time in increasing magnitude of cluster distance until each cluster is connected by at least one edge. Users can manually add or remove transition edges between pairs of clusters/nodes based on the cluster distance ranking, time/cell stage information or biological knowledge.

Once the lineage graph has been established, CALISTA assigns directionalities to the edges in the lineage graph according to user-provided information (e.g. stage or time information, starting cells, or expected gene expression profiles). When the stage or time information is available for the cells, the clusters are first labeled using the most frequent value (mode) of cells' stage or time. CALISTA then assigns the edge directionalities according to the causal direction of the cluster stage or time. Alternatively, CALISTA can also employ users provided information on the start cells or a marker gene with a known expression profile during cell differentiation (e.g., in in Figure 1d, gene Nfe2 is expected to be downregulated during the differentiation process) to determine the starting cell cluster.

Transition genes

Given an edge between cluster *k* and *j* in the lineage graph, CALISTA provides the set of transition genes whose expressions are differentially distributed between the two clusters. More specifically, CALISTA evaluates the likelihood difference between having cells in clusters *j* and *k* separately and having the cells together in one cluster, again using the steady-state distribution of mRNA from the two-state gene transcription model, as follows:

$$v_{jk}^{g} = v_{j}^{g} + v_{k}^{g} - v_{j+k}^{g} \qquad (10)$$

with

$$v_{j}^{g} = \sum_{n \in N_j} \ln P\left(\widehat{m}_{n,g}; \theta^{*}(g,j)\right) \qquad (11)$$

$$v_{k}^{g} = \sum_{n \in N_k} \ln P\left(\widehat{m}_{n,g}; \theta^{*}(g,k)\right) \qquad (12)$$

$$v_{j+k}^{g} = \sum_{n \in N_k \cup N_j} \ln P\left(\widehat{m}_{n,g}; \theta^{*}(g,j+k)\right) \quad (13)$$

where the optimal parameter vector $\theta^{*}(g,j+k)$ is obtained according to Equation (3) for the combined cells in clusters *j* and *k*. A large v_{jk}^{g} reflects that gene *g* is differentially expressed between clusters *j* and *k*, according to the stochastic two-state gene transcriptional model. For each edge in the lineage graph, CALISTA generates a rank list of genes in decreasing values of v_{jk}^{g}. The transition genes correspond to the set of top genes in the list such that the ratio between the sum of v_{jk}^{g} among these genes and the total sum of v_{jk}^{g} among all genes exceeds a given threshold (default threshold: 50%).

3.4 Pseudotemporal cell ordering

Given a lineage progression graph among cell clusters, the third and last main component of CALISTA concerns the pseudotemporal ordering of cells. For this purpose, we first set pseudotimes to the clusters. If the time or stage information of the cells is provided, then the pseudotime of each cluster is set to the mode of the time/stage of the cells in the cluster. Subsequently, the pseudotimes are normalized to have values between 0 and 1 by dividing with the maximum possible value among the clusters. The user can manually assign the pseudotimes of the clusters based on some prior knowledge. Subsequently, each cell is assign to one of the state transition edges that are incident to the cluster containing the cell using the maximum likelihood principle, and then given a pseudotime that corresponds to the maximum point. Finally, given a developmental path in the linage progression, CALISTA can generates a pseudotemporal ordering of cells belonging to the state transition edges in the defined path.

Cell assignment to transition edges

For pseudotemporal ordering of the cells, CALISTA first assigns cells to the edges in the lineage graph. In the following illustration, let us consider a cell *n* in cluster *k*. CALISTA allocates the cell *n* to one of the edges incident to cluster *k* according to the maximum likelihood principle. For this purpose, we define the likelihood value of a cell *n* to belong to an edge pointing from any cluster *j* to cluster *k* as follows:

$$\Lambda_{j\to k}(n) = \max_t \Lambda_j(n) + \frac{(t - t_j)}{(t_k - t_j)}\big(\Lambda_k(n) - \Lambda_j(n)\big) \quad (14)$$

where t_k denotes the cluster pseudotime label. Similarly, we define the likelihood value of the same cell to belong to an edge pointing from cluster *k* to any cluster *l* by the following:

$$\Lambda_{k\to l}(n) = \max_t \Lambda_k(n) + \frac{(t - t_k)}{(t_l - t_k)}\big(\Lambda_l(n) - \Lambda_k(n)\big) \quad (15)$$

CALISTA computes all possible $\Lambda_{j\to k}(n)$ and $\Lambda_{k\to l}(n)$ lineage graph, and assigns the cell to the edge that gives the maximum of all $\Lambda_{j\to k}(n)$ and $\Lambda_{k\to l}(n)$ values. The pseudotime of the cell $t(n)$ is set to t that gives the maximum likelihood value, as follows:

$$\begin{gathered} t(n) = \arg\max_t \Lambda_j(n) + \frac{(t - t_j)}{(t_k - t_j)}\big(\Lambda_k(n) - \Lambda_j(n)\big) \\ or \\ t(n) = \arg\max_t \Lambda_k(n) + \frac{(t - t_k)}{(t_l - t_k)}\big(\Lambda_l(n) - \Lambda_k(n)\big) \end{gathered} \quad (16)$$

depending on the cell assignment to edges above.

Cell ordering along a developmental path

Given a lineage progression graph, users can identify one or several developmental paths. A developmental path is defined as the sequence of connected clusters in the lineage progression graph with transition edges pointing from one cluster to the next in the sequence. CALISTA generates a pseudotemporal ordering along a given developmental path by first identifying cells belonging to the state transition edges in the path and order these cells according to their pseudotimes. Note that in defining the likelihood function for assigning cells to edges, we have made the assumption that the steady state probability distributions of gene expressions vary linearly between two connected clusters or states. But the result of the cell ordering does not change if we replace the linear varying function in the likelihood definition with any monotonic function.

3.4 Cluster entropy

CALISTA provides the entropy of gene expression for clusters based on the definition of Shannon entropy, as follows:

$$H_k = -\frac{1}{G}\sum_{g=1}^{G}\sum_{b_k^g=1}^{B_k^g} p_{b_k}^g log_2\left(\frac{p_{b_k}^g}{dim\left(b_k^g\right)}\right) \quad (17)$$

where $p_{b_k}^g = \frac{z_{b_k}^g}{\dim(N_k)}$, B_k^g is the number of bins in the histogram of expression values of gene g in cluster k, $z_{b_k}^g$is the number of cells in the b_k^g-th bin, $dim\left(b_k^g\right)$ is the size of the b_k^g-th bin and $\dim(N_k)$ is the dimensionality of the set N_k. When binning the expression values of each gene g, we ensure that each bin contains at least 10 cells.

Code availability

A MATLAB and R version of CALISTA used in this study is freely available from the following website: https://github.com/CABSEL/CALISTA.

Data availability

All the public single cell data sets analyzed in this study are available from the original publications. *In silico* datasets for evaluating the effects of gene expression scaling are available on CALISTA website: https://github.com/CABSEL/CALISTA.

Competing interests

The authors declare that they have no competing interests.

Funding

This work was supported by the Swiss National Science Foundation (grant number 157154).

References

1. Ralston, A. & Shaw, K. Gene Expression Regulates Cell Differentiation. *Nat. Educ.* **1(1),** 127 (2008).

2. Kalisky, T. *et al.* A brief review of single-cell transcriptomic technologies. *Brief. Funct. Genomics,* **17**, 64–76 (2018).

3. Guo, G. *et al.* Resolution of cell fate decisions revealed by single-cell gene expression analysis from zygote to blastocyst. *Dev. Cell* **18,** 675–85 (2010).

4. Kumar, R. M. *et al.* Deconstructing transcriptional heterogeneity in pluripotent stem cells. *Nature* **516,** 56–61 (2014).

5. Cacchiarelli, D. *et al.* Integrative Analyses of Human Reprogramming Reveal Dynamic Nature of Induced Pluripotency. *Cell* **162,** 412–424 (2015).

6. Petropoulos, S. *et al.* Single-Cell RNA-Seq Reveals Lineage and X Chromosome Dynamics in Human Preimplantation Embryos. *Cell* **165,** (2016).

7. Richard, A. *et al.* Single-Cell-Based Analysis Highlights a Surge in Cell-to-Cell Molecular Variability Preceding Irreversible Commitment in a Differentiation Process. *PLOS Biol.* **14,** e1002585 (2016).

8. Kærn, M., Elston, T. C., Blake, W. J. & Collins, J. J. Stochasticity in gene expression: from theories to phenotypes. *Nat. Rev. Genet.* **6,** 451–464 (2005).

9. Raj, A. *et al.* Stochastic mRNA Synthesis in Mammalian Cells. *PLoS Biol.* **4,** e309 (2006).

10. Grün, D. *et al.* Single-cell messenger RNA sequencing reveals rare intestinal cell types. *Nature* **525**, 251–255 (2015).

11. Treutlein, B. *et al.* Dissecting direct reprogramming from fibroblast to neuron using single-cell RNA-seq. *Nature* **534,** 391–395 (2016).

12. Stumpf, P. S. *et al.* Stem Cell Differentiation as a Non-Markov Stochastic Process. *Cell Syst.* **5,** 268–282 (2017).

13. Lin, P., Troup, M. & Ho, J. W. K. CIDR: Ultrafast and accurate clustering through imputation for single-cell RNA-seq data. *Genome Biol.* **18,** 59 (2017).

14. žurauskienė, J. & Yau, C. pcaReduce: hierarchical clustering of single cell transcriptional profiles. *BMC Bioinformatics* **17,** 140 (2016).

15. Buettner, F. *et al.* Computational analysis of cell-to-cell heterogeneity in single-cell RNA-sequencing data reveals hidden subpopulations of cells. *Nat. Biotechnol.* **33,** 155–160 (2015).

16. Xu, C. & Su, Z. Identification of cell types from single-cell transcriptomes using a novel clustering method. *Bioinformatics* **31,** 1974–80 (2015).

17. Marco, E. *et al.* Bifurcation analysis of single-cell gene expression data reveals epigenetic landscape. *Proc. Natl. Acad. Sci. U. S. A.* **111,** 5643–50 (2014).

18. Huang, W., Cao, X., Biase, F. H., Yu, P. & Zhong, S. Time-variant clustering model for understanding cell fate decisions. *Proc. Natl. Acad. Sci. U. S. A.* **111,** E4797-806 (2014).

19. Kiselev, V. Y. *et al.* SC3: consensus clustering of single-cell RNA-seq data. *Nat. Methods* **14,** 483–486 (2017).

20. Wang, B., Zhu, J., Pierson, E., Ramazzotti, D. & Batzoglou, S. Visualization and analysis of

single-cell RNA-seq data by kernel-based similarity learning. *Nat. Methods* **14,** 414–416 (2017).

21. Ezer, D. *et al.* Determining Physical Mechanisms of Gene Expression Regulation from Single Cell Gene Expression Data. *PLOS Comput. Biol.* **12,** e1005072 (2016).
22. Haghverdi, L., Büttner, M., Wolf, F. A., Buettner, F. & Theis, F. J. Diffusion pseudotime robustly reconstructs lineage branching. *Nat. Methods* **13,** 845–848 (2016).
23. Trapnell, C. *et al.* The dynamics and regulators of cell fate decisions are revealed by pseudotemporal ordering of single cells. *Nat. Biotechnol.* **32,** 381–386 (2014).
24. Qiu, X. *et al.* Reversed graph embedding resolves complex single-cell trajectories. *Nat. Methods* **14,** 979–982 (2017).
25. Cannoodt, R., Saelens, W. & Saeys, Y. Computational methods for trajectory inference from single-cell transcriptomics. *Eur. J. Immunol.* **46,** 2496–2506 (2016).
26. Peccoud, J. & Ycart, B. Markovian Modeling of Gene-Product Synthesis. *Theor. Popul. Biol.* **48,** 222–234 (1995).
27. Bargaje, R. *et al.* Cell population structure prior to bifurcation predicts efficiency of directed differentiation in human induced pluripotent cells. *Proc. Natl. Acad. Sci. U. S. A.* **114,** 2271–2276 (2017).
28. Qiu, X. *et al.* Reversed graph embedding resolves complex single-cell trajectories. *Nat. Methods* (2017). doi:10.1038/nmeth.4402
29. Kaufman, L. & Rousseeuw, P. J. *Finding Groups in Data: An introduction to cluster analysis.* John Wiley & Sons, Hoboken, NJ (2009).
30. Tiyaboonchai, A. *et al.* GATA6 Plays an Important Role in the Induction of Human Definitive Endoderm, Development of the Pancreas, and Functionality of Pancreatic β Cells. *Stem cell reports* **8,** 589–604 (2017).
31. Ng, E. S. *et al.* The primitive streak gene Mixl1 is required for efficient haematopoiesis and BMP4-induced ventral mesoderm patterning in differentiating ES cells. *Development* **132,** 873–884 (2005).
32. Jagtap, S. *et al.* Cytosine arabinoside induces ectoderm and inhibits mesoderm expression in human embryonic stem cells during multilineage differentiation. *Br. J. Pharmacol.* **162,** 1743–56 (2011).
33. Ng, V. Y., Ang, S. N., Chan, J. X. & Choo, A. B. H. Characterization of Epithelial Cell Adhesion Molecule as a Surface Marker on Undifferentiated Human Embryonic Stem Cells. *Stem Cells* **28,** 29–35 (2010).
34. Thomas, T., Nowka, K., Lan, L. & Derwahl, M. Expression of Endoderm Stem Cell Markers: Evidence for the Presence of Adult Stem Cells in Human Thyroid Glands. *Thyroid* **16,** 537–544 (2006).
35. Bron, C. & Kerbosch, J. Algorithm 457: finding all cliques of an undirected graph. *Commun. ACM* **16,** 575–577 (1973).
36. Davidson, K. C. *et al.* Wnt/β-catenin signaling promotes differentiation, not self-renewal, of human embryonic stem cells and is repressed by Oct4. *Proc. Natl. Acad. Sci. U. S. A.* **109,**

4485–90 (2012).

37. Chu, L.-F. *et al.* Single-cell RNA-seq reveals novel regulators of human embryonic stem cell differentiation to definitive endoderm. *Genome Biol.* **17,** 173 (2016).

38. Moignard, V. *et al.* Characterization of transcriptional networks in blood stem and progenitor cells using high-throughput single-cell gene expression analysis. *Nat. Cell Biol.* **15,** 363–72 (2013).

39. Kolodziejczyk, A. A. *et al.* Single Cell RNA-Sequencing of Pluripotent States Unlocks Modular Transcriptional Variation. *Cell Stem Cell* **17,** 471–485 (2015).

40. Pollen, A. A. *et al.* Low-coverage single-cell mRNA sequencing reveals cellular heterogeneity and activated signaling pathways in developing cerebral cortex. *Nat. Biotechnol.* **32,** 1053–1058 (2014).

41. Usoskin, D. *et al.* Unbiased classification of sensory neuron types by large-scale single-cell RNA sequencing. *Nat. Neurosci.* **18,** 145–153 (2014).

42. Villani, A.-C. *et al.* Single-cell RNA-seq reveals new types of human blood dendritic cells, monocytes, and progenitors. *Science* **356,** 283 (2017).

43. Pollen, A. A. *et al.* Low-coverage single-cell mRNA sequencing reveals cellular heterogeneity and activated signaling pathways in developing cerebral cortex. *Nat. Biotechnol.* **32,** 1053–1058 (2014).

44. Yeung, K. Y. & Ruzzo, W. L. Details of the Adjusted Rand index and Clustering algorithms Supplement to the paper &quot; An empirical study on Principal Component Analysis for clustering gene expression data &quot; (to appear in Bioinformatics). (2001).

45. Levine, E. & Domany, E. Resampling Method for Unsupervised Estimation of Cluster Validity. *Neural Comput.* **13,** 2573–2593 (2001).

46. Papili Gao, N., Ud-Dean, S. M. M., Gandrillon, O. & Gunawan, R. SINCERITIES: Inferring gene regulatory networks from time-stamped single cell transcriptional expression profiles. *Bioinformatics* **34**, 258–266 (2018).

47. Vallejos, C. A., Risso, D., Scialdone, A., Dudoit, S. & Marioni, J. C. Normalizing single-cell RNA sequencing data: challenges and opportunities. *Nat. Methods* **14,** 565–571 (2017).

48. Ståhlberg, A., Rusnakova, V., Forootan, A., Anderova, M. & Kubista, M. RT-qPCR work-flow for single-cell data analysis. *Methods* **59,** 80–88 (2013).

49. Qiu, X. *et al.* Single-cell mRNA quantification and differential analysis with Census. *Nat. Methods* **14,** 309–315 (2017).

50. Macosko, E. Z. *et al.* Highly Parallel Genome-wide Expression Profiling of Individual Cells Using Nanoliter Droplets. *Cell* **161,** 1202–1214 (2015).

51. von Luxburg, U. A tutorial on spectral clustering. *Stat. Comput.* **17,** 395–416 (2007).

52. Smoot, M. E., Ono, K., Ruscheinski, J., Wang, P.-L. & Ideker, T. Cytoscape 2.8: new features for data integration and network visualization. *Bioinformatics* **27,** 431–2 (2011).

Figures

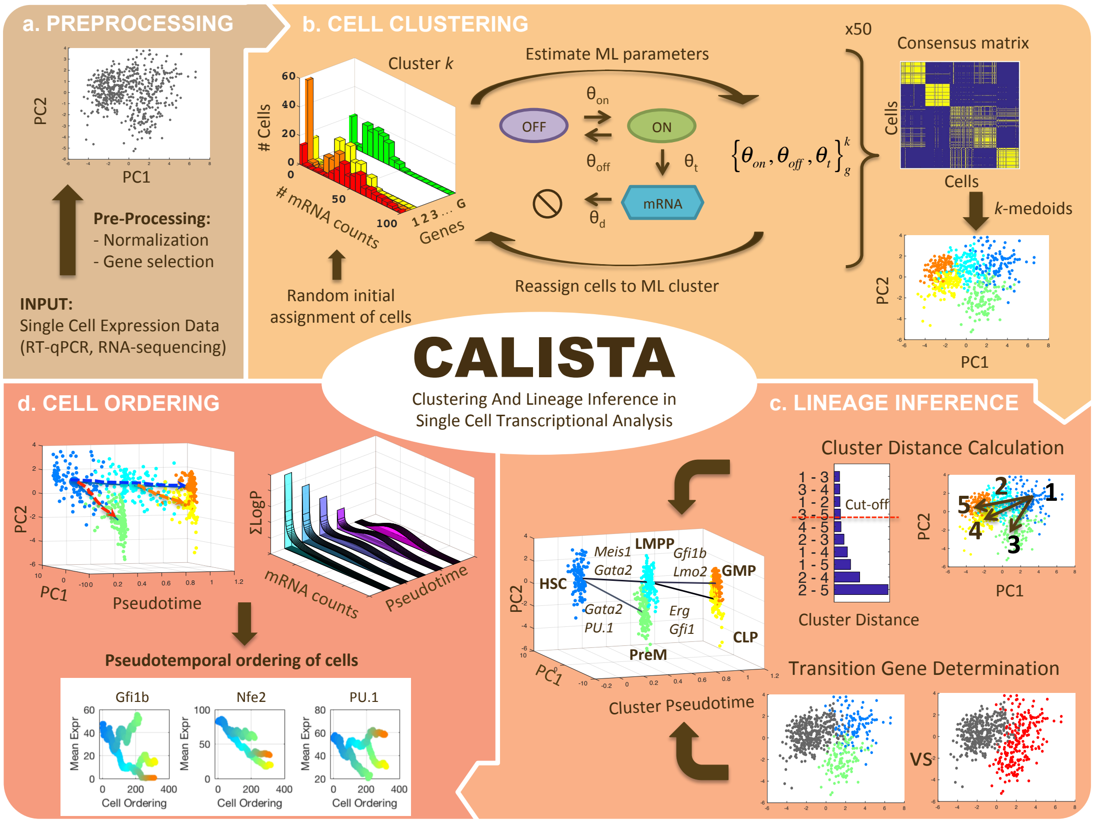

Figure 1: The workflow of CALISTA. (a) Preprocessing of single-cell expression input data. CALISTA accepts both single-cell RT-qPCR (C_T values) or RNA-seq (RPKM or TPM). (b) Single-cell clustering in CALISTA combines maximum likelihood and consensus clustering approaches. In the maximum likelihood step, CALISTA relies on the two-state model of the gene transcription process to describe the distribution of single-cell gene expression. The maximum likelihood clustering is implemented multiple times, each starting from random assignment of cells into clusters, using a greedy algorithm and producing a consensus matrix. The final step implements *k*-medoids clustering algorithm using the consensus matrix. (c) CALISTA evaluates cluster distances - a measure of dissimilarity in the gene expression distribution between any two clusters, to reconstruct the lineage progression graph. The state transition edges are added in increasing magnitude of cluster distances. In addition, CALISTA provides transition genes between any two connected clusters in the lineage graph based on the gene-wise likelihood differences of having the cells separately and having them in one cluster. (d) For pseudotemporal ordering of the cells, CALISTA reassigns each cell to a transition edge and computes its pseudotime by maximizing the cell likelihood. CALISTA uses a linear interpolation for evaluating the cell likelihood between two connected clusters. Given a developmental path in the lineage progression, CALISTA generates moving averaged gene expression trajectory for the pseudotemporally ordered cells belonging to the path.

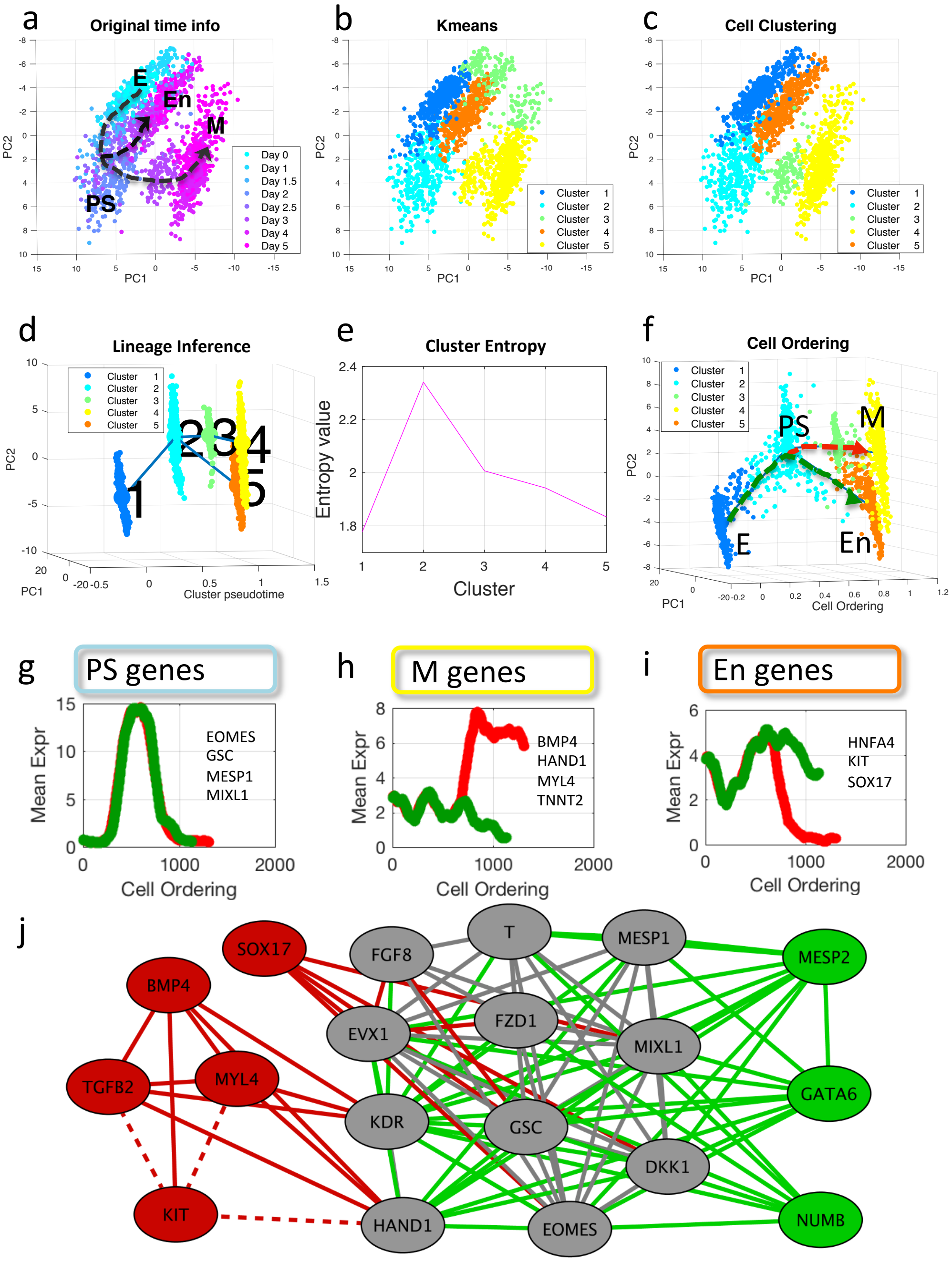

Figure 2: Analysis of lineage relationships during iPSC to cardiomyocyte differentiation using CALISTA. The single-cell gene expression dataset was taken from the study of Bargaje et al. (a-c) Plots of single-cell dataset along the first two principal component. Each symbol represents a cell, while the colors represent (a) the capture time info, (b) *k*-means single-cell clusters, and (c) CALISTA single-cell clusters. (d) Lineage progression graph. The graph was generated by adding transition edges in increasing order of cluster distances until each cluster is connected by an edge. An edge connecting cluster 1 and 5 was originally included in the graph, but manually removed based on the capture time information. (e) Cluster entropies. The entropy is the highest for the cluster harboring primitive streak (PS)-like progenitor cells, i.e. the cluster at the bifurcation in the lineage progression. (f) Pseudotemporal ordering of cells. Cells were first assigned to the state transition edges in the lineage progression, and pseudotemporally ordered along two developmental paths – mesodermal (M) path (red) and endodermal (En) path (green). (g-i) Representative moving average gene expression profiles for (g) PS genes, (h) M genes, and (i) En genes. (j) Gene modules in M path (red and grey) and En path (green and grey) (drawn using Cytoscape [52]). The grey nodes and edges are shared between the M and En modules.

Tables

Table 1. Comparison of CALISTA with Existing Clustering and Lineage Inference Methods. (A) Adjusted Rand Index (ARI) of SIMLR, SC3, MONOCLE 2 and CALISTA. (B) Detection of lineage bifurcations by DPT, SCUBA, MONOCLE 2 and CALISTA.

A	Clustering Accuracy (Adjusted Rand Index, ARI)			
	SIMLR	**SC3**	**MONOCLE 2**	**CALISTA**
Kolodziejczyk et al. (704 cells, 3 clusters)	1	1	0.97	0.96
Pollen et al. (249 cells, 11 clusters)	0.94	0.95	0.86	0.89
Usoskin et al. (622 cells, 4 clusters)	0.68	0.88	0.57	0.91
Villani et al. (751 cells, 6 clusters)	0.37	0.88	0.58	0.86
B	**Lineage Inference**			
	DPT	**SCUBA**	**MONOCLE 2**	**CALISTA**
Inferrable bifurcation event/-s	Single	Multiple	Multiple	Multiple
Prior Information requirement	Linear/Branching	Time Points	None	None
Bargaje et al. (1896 cells, 1 bifurcation event)	✗	✓	✓	✓
Chu et al. (728 cells, no bifurcation event)	✗	✗	✗	✓
Moignard et al. (597 cells, 2 bifurcation events)	✓ Partial	–	✓ Partial	✓
Treutlein et al. (405 cells, 1 bifurcation event)	✓	✓	✓	✓