LRT: T Cell Trajectory Inference by Integrative Analysis of Single-Cell TCR-seq and RNA-seq data

Juan Xie[1], Gang Xin[2,3], Qin Ma[3,4] and Dongjun Chung[3,4,*]

1 The Interdisciplinary Program in Biostatistics, The Ohio State University, Columbus, Ohio, USA.

2 Department of Microbial Infection and Immunity, The Ohio State University, Columbus, Ohio, USA.

3 Pelotonia Institute for Immuno-Oncology, The James Comprehensive Cancer Center, The Ohio State University, Columbus, Ohio, USA.

4 Department of Biomedical Informatics, The Ohio State University, Columbus, Ohio, USA.

* chung.911@osu.edu

Abstract

Single-cell RNA sequencing (scRNA-seq) data has been widely used for cell trajectory inference, with the assumption that cells with similar expression profiles share the same differentiation state. However, the inferred trajectory may not reflect true clonal relationships among cells. Single-cell T cell receptor sequencing (scTCR-seq) data provides invaluable insights into the clonal relationship among cells, yet it lacks functional characteristics. Therefore, scRNA-seq and scTCR-seq data complement each other in improving trajectory inference, where a reliable computational tool is still missing. We developed LRT, a computational framework for the integrative analysis of scTCR-seq and scRNA-seq data for T cell trajectory inference. Specifically, LRT utilizes the TCR sequence information to identify clonally related cells and then uses the transcriptomics information from scRNA-seq data to construct clonotype-level cell trajectories. LRT provides a comprehensive analysis workflow, including preprocessing, cell trajectory clustering, pseudotime inference, and marker gene identification. We illustrated its utility using scRNA-

seq and scTCR-seq data of $CD4^+$ T cells with acute lymphocytic choriomeningitis virus infection, where we could identify cell trajectories that cannot be revealed solely based on scRNA-seq data. Our downstream analyses showed that (i) these trajectories are involved in distinct functional roles; (ii) the expression patterns of their marker genes over the estimated pseudotime nicely coincide with the Th1/Tfh biology that is well established for the $CD4^+$ T cell differentiation; and (iii) the higher level of TCR sequence similarities was observed within each cluster, compared to between clusters. The LRT framework was implemented as an R package 'LRT', and it is now publicly accessible at https://github.com/JuanXie19/LRT. In addition, it provides two Shiny apps 'shinyClone' and 'shinyClust' that allow users to interactively explore distributions of clonotypes, conduct repertoire analysis, implement clustering of cell trajectories, and predict cell trajectory cluster marker genes.

Author Summary

Understanding the dynamic changes behind biological processes is important for determining molecular mechanisms underlying normal tissue formulation, developmental disorders and pathologies. Usually, a biological process can be characterized by identifying a trajectory, a path that goes through the various cellular states associated with the process. Since cells in different states may express different sets of genes, researchers often infer cell trajectory via capturing transcriptomics changes. Dozens of methods have been developed for cell trajectory inference, and scRNA-seq data is predominantly utilized. However, methods based only on scRNA-seq data cannot tell us if cells from the same trajectory come from the same clone or not. T cells play a key role in the immune system, and their high antigen recognition specificity is largely determined by their TCR sequences. Thanks to the advent of scTCR-seq technology, people can identify the group of cells coming from the same clone. This paper describes our novel computational framework, namely LRT, and demonstrates that by complementing scRNA-seq data with the clonal information from

scTCR-seq data using LRT, we are able to identify cell trajectories that cannot be revealed solely based on scRNA-seq data.

Introduction

Cell trajectory inference aims to understand how cells differentiate, which is a long-standing problem in developmental biology. Understanding how progenitor cells transform into specified functional cells can provide valuable insights into the molecular mechanisms underlying normal tissue formulation, as well as developmental disorders and pathologies [1]. Single-cell RNA-seq (scRNA-seq) data has been widely used to investigate cell trajectories. Assuming that cells in different states express different sets of marker genes, we may order cells along a differentiation trajectory via capturing differentiated transcriptional activities [2]. Based on this rationale, various computational and statistical approaches have been proposed for this purpose, namely trajectory/pseudotime analysis, where well known examples include Monocle [2] and Slingshot [3] (please see Saelens et al. [4] for a comprehensive review). Specifically, Monocle builds a minimum spanning tree (MST) on cells in a low-dimensional independent component space and uses the longest path through the MST as the backbone for pseudotime ordering, while branches were handled by the PQ tree. In the case of Slingshot, in its first stage, it also uses MST to determine the global cell lineage structure. Yet instead of building MST on individual cells, Slingshot constructs an MST on cluster centers, which makes it more robust to noises. Then, in its second stage, it uses a novel method called simultaneous principal curves to obtain smooth branching trajectories. While these approaches have been shown to be useful, they still suffer from intrinsic limitations. Specifically, it assumes that cells with similar expression profiles share similar developmental stages and may come from the same lineage, which might not be always true [1, 5]. Besides, in the absence of clonal information, the inferred trajectory may not reflect the true clonal relationship between cells [1].

The recent advent of single-cell T cell receptor (TCR) sequencing (scTCR-seq) technology has enabled the use of TCR sequences as unique 'barcodes' to identify clonally related cells. This is because the cells with identical paired TCR sequences usually arise from the same T cell clone due to the high dimensionality of TCR sequence space [6]. Examining the abundance/proportion of clonal cells and their changes greatly benefits our understanding of adaptive immune response in health and disease [7, 8]; therefore, there has been a rapid accumulation of scTCR-seq data in recent years. Accordingly, several computational and statistical approaches have also been developed, where examples include Immunarch [9] and scRepertoire [10]. However, currently available approaches mostly focus on TCR repertoire analysis, such as clonality, repertoire overlap, repertoire diversity, and gene usage analyses. The utilization of scTCR-seq data for cell trajectory analysis is still limited. In addition, although TCR data provides invaluable insights into the breadth of the antigenic response and clonal relationships between cells, it does not offer information about the functional characteristics of the T cells. Hence, integrating scTCR-seq and scRNA-seq can help elucidate the interplay between TCR sequences and T cell phenotypes. For instance, Zhang et al. [11] observed that TCR similarity constrained the similarity of T cell phenotypes, and TCRs that are more similar to the center TCRs have stronger antigen binding affinity. However, despite such potential to improve T cell trajectory inference, to the best of our knowledge, a computational approach to integrate scRNA-seq and scTCR-seq data for cell trajectory inference is missing in the literature. This is partially due to the statistical and computational challenges related to the high dimensionality of the clonotype space. Specifically, for a given human subject, 10^{13} TCR sequences are typically observed while each clonotype may only span a very limited number of cells. Hence, for effective cell trajectory inference and integration of scTCR-seq and scRNA-seq data, we need a cleverly designed computational framework that can handle these challenges.

In this paper, we propose LRT (**L**ineage inference by integrative analysis of sc**R**NA-seq and sc**T**CR-seq data), a rigorous computational framework for the integrative analysis of scTCR-seq and scRNA-seq data for cell trajectory inference. LRT addresses the complexity and the high dimensionality of

scTCR-seq data through a trajectory clustering algorithm and effective information sharing between scTCR-seq and scRNA-seq data. In addition, to support researchers in utilizing the proposed LRT framework for their studies, we developed an R package 'LRT', which is now publicly accessible at https://github.com/JuanXie19/LRT. We also developed two Shiny apps, namely 'shinyClone' and 'shinyClust', and they allow researchers to interactively implement the complete LRT analysis workflow, including exploratory analysis, cell trajectory inference, and marker gene identification. These Shiny apps are provided as part of the R package 'LRT'.

Materials and Methods

The workflow of the 'LRT' framework is shown in **Fig. 1**, which starts from the integration of scTCR-seq and scRNA-seq data (**Fig. 1A**). Using this integrated dataset, LRT infers cell trajectory by implementing the following four steps. In the first step, it estimates clonotype-level trajectories by building an MST based on the cells sharing the same clonotype (**Fig. 1B**). In the second step, LRT identifies clusters of clonotype-level trajectories using the stochastic block model (SBM) based on the dynamic time warping (DTW) distances (**Fig. 1C**). In the third step, a principal curve is fitted using the cells belonging to the same trajectory cluster, to get the final trajectories and infer the pseudotime (**Fig. 1D**). Finally, differential expression (DE) analysis is implemented to identify genes characterizing each trajectory cluster, namely trajectory cluster markers (**Fig. 1E**).

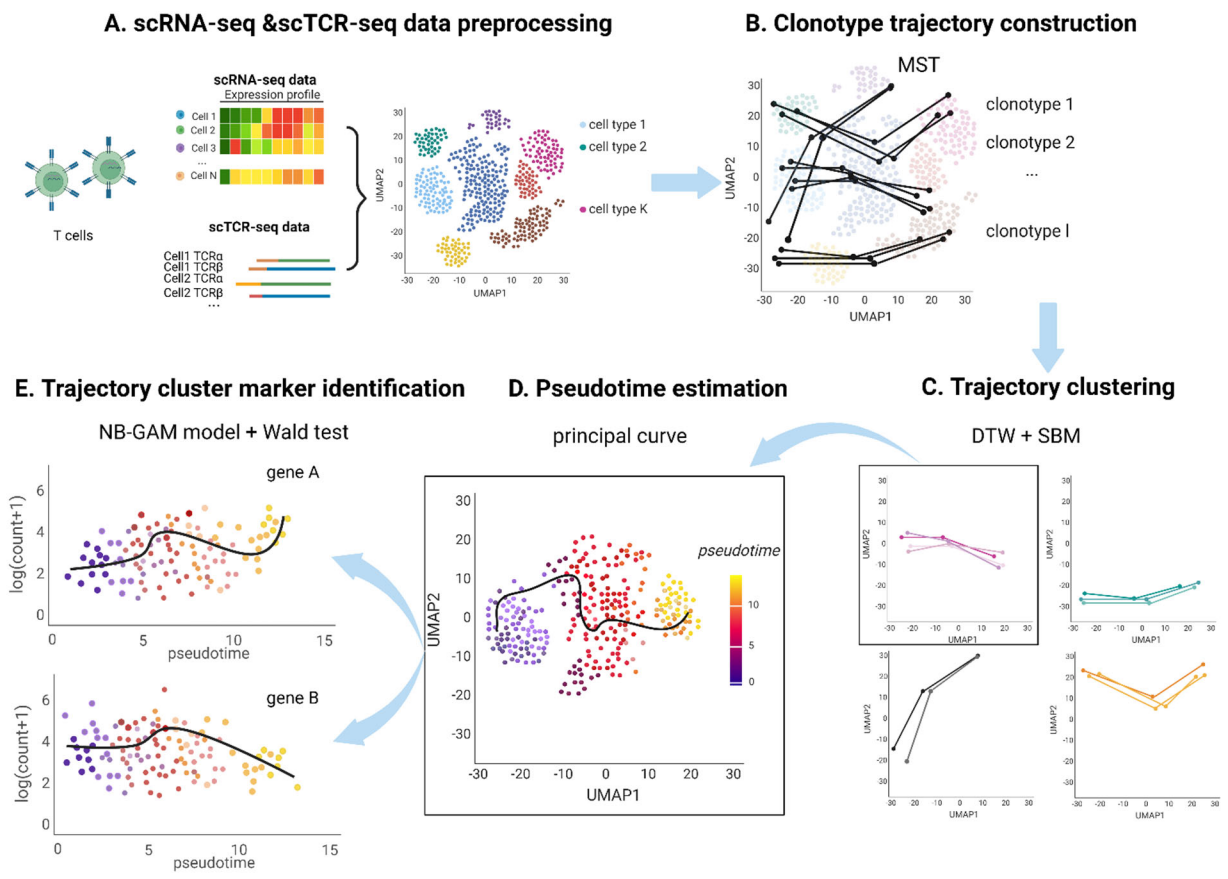


Figure 1. The framework of LRT. (A) The preprocessed scRNA-seq and scTCR-seq data are integrated based on cell barcodes. **(B)** Clonotype-level trajectories are first obtained by constructing minimum spanning trees (MST). **(C)** Similar trajectories are grouped using the stochastic block model (SBM), where similarity is evaluated based on the dynamic time warping (DTW) distance. **(D)** For each trajectory cluster, cell pseudotimes are estimated by fitting a principal curve on the Uniform Manifold Approximation and Projection (UMAP) space. **(E)** The relationship between read count and pseudotime is modeled by negative binomial generalized additive model (NB-GAM), and the Wald test is used to determine marker genes.

Trajectory inference

The core of LRT lies in its trajectory clustering and characterization through the integration of scTCR-seq and scRNA-seq data. It consists of four key steps, including clonotype-level trajectory estimation, trajectory

clustering, pseudotime inference, and marker identification (DE analysis). Each step is described in detail below.

Clonotype-level trajectory estimation

Given the integrated scRNA-seq and scTCR-seq data (see the Section 'Software implementation' for more details), LRT infers a trajectory for each clonotype. Specifically, a weighted graph is constructed for each group of cells sharing the same TCR sequence. Here nodes are cell-cluster-specific centers of these cells, while edge weights represent the Euclidean distances between the nodes on the Uniform Manifold Approximation and Projection (UMAP) space [12]. Note that both cell clusters and UMAP in this step are defined based on scRNA-seq data. An MST [13] is then constructed, and trajectories are interpreted as ordered cell clusters by tracing paths through the MST.

Trajectory clustering

After obtaining the clonotype-level trajectories, LRT conducts a clustering analysis to group similar trajectories, where similarity is defined based on DTW distances [5] between trajectories. DTW distance is a measure to reflect the similarity between two temporal sequences, which is invariant to warping and thus is particularly useful when the two sequences vary in speed. Here we treat each trajectory as time series data, with cell clusters as time points and UMAP coordinates as measurements under each time point. Let's denote two trajectories $S = (s_1, s_2, \dots, s_i, \dots, s_q)$ and $T = (t_1, t_2, \dots, t_j, \dots, t_r)$, where q and r are the numbers of "time points", respectively. We first build the distance matrix $C \in \mathbb{R}^{q \times r}$ to represent pairwise distances between S and T, where each element of C, $c(i,j) = \|s_i - t_j\|$, $i \in [1{:}\,q]$, $j \in [1{:}\,r]$. Then, we define a warping path as a sequence of points $p = (p_1, \dots, p_Q)$ with $p_l = (i_l, j_l) \in [1{:}\,q] \times [1{:}\,r]$ for $l \in [1{:}\,Q]$ to represent a possible alignment between S and T. The associated total cost/distance for p is

$$c_p(S,T) = \sum_{l=1}^{Q} c(s_{i_l}, t_{j_l}).$$

Finally, the DTW distance is defined as the cost associated with the optimal warping path, i.e.,

$$DTW(S,T) = c_{p^*}(S,T) = \min_{p \in P^{q \times r}} \{c_p(S,T)\},$$

where $P^{q \times r}$ is the set of all possible warping paths.

Based on this DTW distance matrix, LRT applies an SBM [14] to identify trajectory clusters (i.e., community detection). SBM is a generative model for random graphs, which is widely employed for recovering (latent) community structures in network/graph data [15]. Let's denote n as the total number of cells. To apply SBM in our case, we convert the DTW distance matrix to an adjacency matrix of a k-nearest neighbor graph, denoted as A, where we set $k = \sqrt{n}$ by adopting the widely used heuristics [15, 16]. Given $A = [A_{ij}] \in \mathbb{R}^{n \times n}$ denoting the binary adjacency matrix of a random graph, with elements A_{ij} indicating the presence or absence of an undirected edge between nodes i and j, we assume that the absence or presence of an edge between each pair of nodes i and j follows a Bernoulli distribution, i.e.,

$$A_{ij} \sim Bernoulli(p_{ij}),\ p_{ij} \in (0,1) \text{ for } 1 \leq i < j \leq n.$$

SBM assumes that there is a latent connectivity matrix $\Theta \in \mathbb{R}^{b \times b},\ b \ll n$, and a map $Z: \{1, \dots, n\} \rightarrow \{1, \dots, b\}$, such that

$$p_{ij} = \theta_{z(i),z(j)}.$$

SBM groups n nodes into b groups, with group labels given by Z, and we call these groups communities. A variational Expectation-Maximization algorithm [17] is used for parameter estimation. After fitting SBMs with a different number of clusters, the final number of clusters is determined using the maximum integrated completed likelihood (ICL) criterion [18].

Pseudotime estimation

As the final step of trajectory inference, LRT obtains the final trajectory by fitting a principal curve [19] on the UMAP coordinates of cells. Let us denote a random vector representing the UMAP coordinates for cells

from the same trajectory cluster as $X \in \mathbb{R}^2$. In addition, denote h as a one-dimensional smooth unit-speed curve in $\mathbb{R}^2$ parameterized by a single variable t. In the principal curve estimation, we iterate the following the projection step and the regularized expectation step:

(i) Projection step: Project all cells onto the curve $h^{(l-1)}$ and calculate the arc length from the beginning of the curve to each point's projection, $t_i^{(l)} = t_{h^{(l-1)}}(x_i)$, where the projection t_h: $\mathbb{R}^2 \to \mathbb{R}^1$ is defined as

$$t_h(x) = \sup_t \{t: \| x - h(t) \| = \inf_u \| x - h(u) \|\}$$

(ii) Regularized expectation step: Update the curve h by minimizing the distance between data points x_i and those projected onto the curve, $h(t_i^{(l)})$.

Here, we interpret the curve $h(x)$ as the final trajectory for the given cell trajectory cluster and the arc length $t_i = t_h(x_i)$ as pseudotime for cell i.

Differential expression analysis

While the algorithms described above allow us to infer cell trajectories, it is critical to identify genes characterizing these trajectories to fully understand the biological relevance of these trajectories. To facilitate the interpretation of the inferred trajectories, LRT implements DE analysis for each trajectory cluster. First, for each trajectory cluster, it infers a smooth function of the normalized gene read counts on the pseudotime, using a negative binomial generalized additive model (NB-GAM) [20, 21]. To be specific, for the ν-th trajectory cluster with n_ν cells, let Y_{gi} be the read counts for gene $g \in \{1, \dots, G\}$ and cell $i \in \{1, \dots, n_\nu\}$, the NB-GAM assumes that

$$Y_{gi} \sim NB(\mu_{gi}, \phi_g)$$

$$\log(\mu_{gi}) = f_g(t_i) + \log(N_i)$$

where μ_{gi} are cell- and gene-specific means, ϕ_g are gene-specific dispersion parameters, N_i is cell-specific offsets that account for sequencing depth, and f_g is the smoothing spline for gene g. Here we use a cubic spline function with 6 knots for f_g to balance between the goodness of fit and running time. Finally, a Wald test is used to determine DE genes [21], followed by Bonferroni correction for the multiple testing adjustment.

Software implementation

The aforementioned LRT framework and the whole analysis workflow were implemented as an R package 'LRT' and it is now publicly accessible at https://github.com/JuanXie19/LRT. The LRT software takes both scRNA-seq and scTCR-seq data as input. It accepts scRNA-seq data in the form of a Seurat object [22]. Such integration with Seurat not only allows LRT to be easily integrated into an existing scRNA-seq analysis workflow, but also utilizes Seurat's powerful features, e.g., correcting batch effects and variations due to technical factors (e.g., sequencing depth) [22, 23]. The Seurat object needs to contain clustering labels and UMAP coordinates for each cell. In the case of scTCR-seq data, it requires a data frame with at least two columns: one column with cell identifiers and another column with TCR sequence information, for which users can provide amino acid or nucleotide sequences of TCR α, β, or paired α-β chain CDR3 region.

Data preprocessing

LRT integrates scTCR-seq data with scRNA-seq data by attaching the TCR sequence information to the metadata of the Seurat object. LRT also provides some basic quality control functionalities, e.g., checking whether the cell identifiers from the Seurat object match those in the TCR data frame. After the integration, users will get an S4 object containing the updated Seurat object, which is the input for clonotype exploratory analysis and trajectory inference.

212 **Shiny apps for interactive visualization and analysis: shinyClone and shinyClust**

213 We developed two Shiny apps, *shinyClone*, and *shinyClust*, for interactive and dynamic data exploration
214 and integrative analysis of scRNA-seq and scTCR-seq data.

215 **shinyClone**

216 *shinyClone* is a Shiny app for exploratory analysis of clonotypes (**Fig. 2**). It provides commonly used
217 repertoire analysis functionalities and lets users intuitively explore how each clonotype is distributed on the
218 reduced dimensional space generated based on scRNA-seq data. *shinyClone* consists of two tabs, one for
219 exploratory analysis of clonotype distribution (tab 'Clonotype') (**Fig. 2A**) and the other for repertoire
220 analysis (tab 'Repertoire') (**Fig. 2B**). First, on the 'Clonotype' tab, users can find a table of clonotypes,
221 along with the relevant information, including (i) how many cells are associated with each clonotype; and
222 (ii) how these cells are distributed among groups (e.g., cell clusters, samples, conditions, or any other groups
223 as available in the metadata of Seurat object) (**Fig. 2A**, left). Users can sort the table by clicking each
224 column name, e.g., to check the most/least abundant clonotypes. By clicking each clonotype in this table,
225 the user can check the distribution of cells associated with each clonotype on the reduced dimensional space,
226 for which the user can choose among UMAP, t-SNE, or principal component analysis (PCA) for the
227 dimension reduction (**Fig. 2A**, middle). A cell density plot is also displayed for clones spanning more than
228 10 cells (**Fig. 2A**, right). Both plots can be downloaded in the PDF file format by clicking the "Download"
229 button. Second, the 'Repertoire' tab provides several common repertoire analysis functions using the R
230 package 'scRepertoire' [10], where examples include clonal diversity, clonal homeostasis, clonal
231 proportion, and clonal overlap analyses (**Fig. 2B**).

232

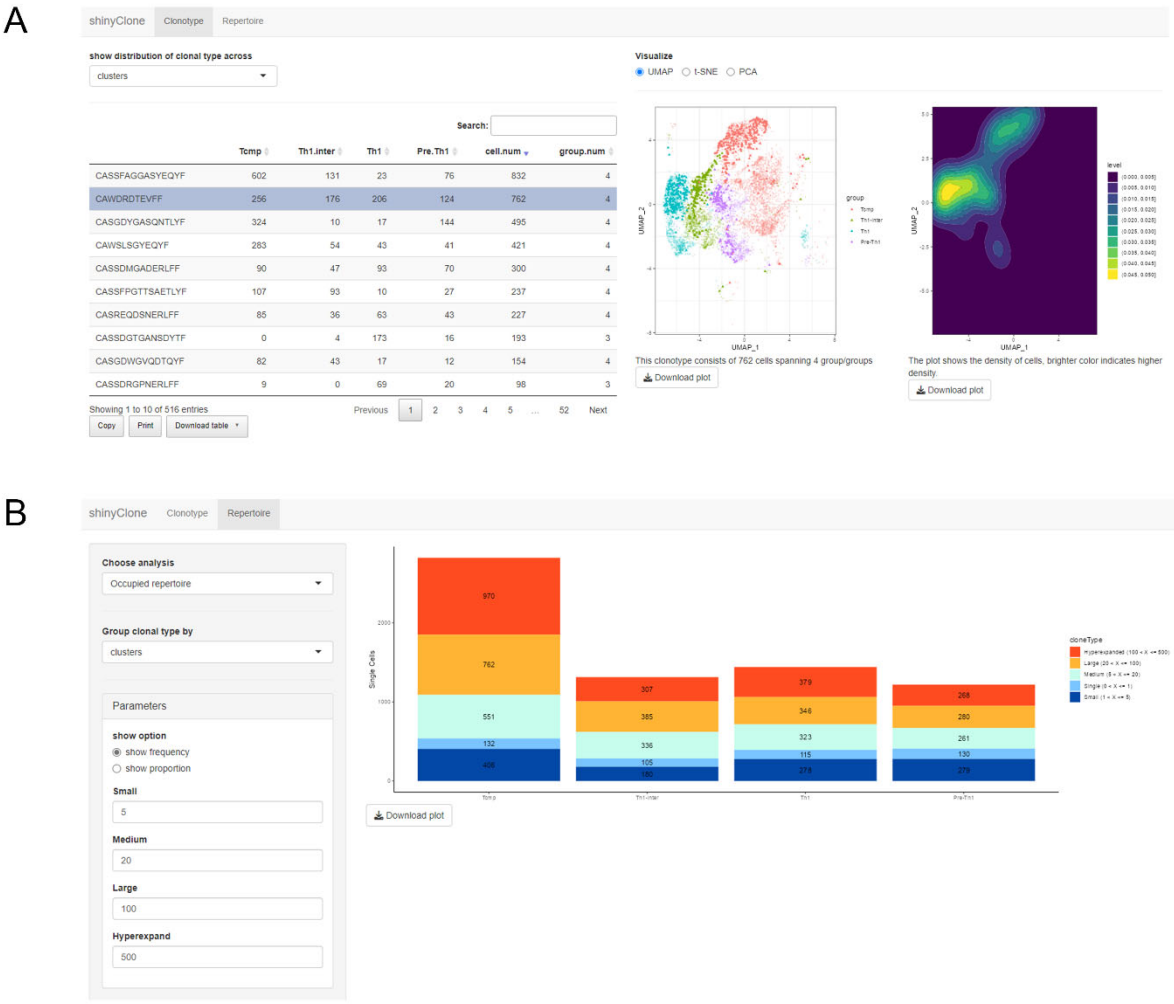


Figure 2. *shinyClone*, a Shiny app for exploratory analysis of clonotypes. (A) The 'Clonotype' tab allows users to implement various exploratory analyses of clonotype distributions. **(B)** The 'Repertoire' tab provides various TCR repertoire analysis functionalities.

shinyClust

shinyClust is another Shiny app that allows users to interactively and dynamically implement the trajectory clustering analysis and marker identification (**Fig. 3**). *shinyClust* consists of three tabs, 'Clustering exploration' (**Fig. 3A**), 'Trajectory clustering' (**Fig. 3B**), and 'DE analysis' (**Fig. 3C**). First, the 'Clustering

242 exploration' tab allows users to implement trajectory clustering interactively (**Fig. 3A**). This page shows
243 either ICL or silhouette scores under the current setting, to assist users in evaluating and determining the
244 optimal number of clusters. While SBM is set to be a default here, other clustering algorithms are also
245 provided for flexibility, including hierarchical and partitional clustering. We used the R package 'sbm' [24]
246 for the SBM implementation and the R package 'dtwclust' [25] for the hierarchical and partitional clustering
247 implementations. Users can also interactively tune parameters for each clustering algorithm. Second, on the
248 'Trajectory clustering' tab, upon choosing a particular cluster to display, *shinyClust* updates (i) a trajectory
249 plot, which shows clonotype trajectories corresponding to the trajectory cluster on the UMAP plot (**Fig. 3B**,
250 the plot on the left, titled 'Trajectories'); and (ii) a cell distribution plot, which shows the cells associated
251 with this trajectory cluster on the UMAP plot (**Fig. 3B**, the plot on the right, titled 'Cell distribution'). All
252 the plots can also be downloaded in the PDF file format by clicking the "Download" button. Finally, in the
253 'DE analysis' tab, users can implement DE analysis, which fits the NB-GAM model and conducts the Wald
254 test in the backend using the R package 'tradeSeq' [21] (**Fig. 3C**). Here, *shinyClust* shows a table of
255 predicted DE genes with both raw and Bonferroni-adjusted *p*-values, as well as a heatmap of DE gene
256 expression values. If users are interested in how the expression values of a particular gene vary along
257 pseudotime, they can click the corresponding row in the table and the page will show a scatter plot of log
258 read count versus pseudotime, along with the fitted smoothing spline curve.

259

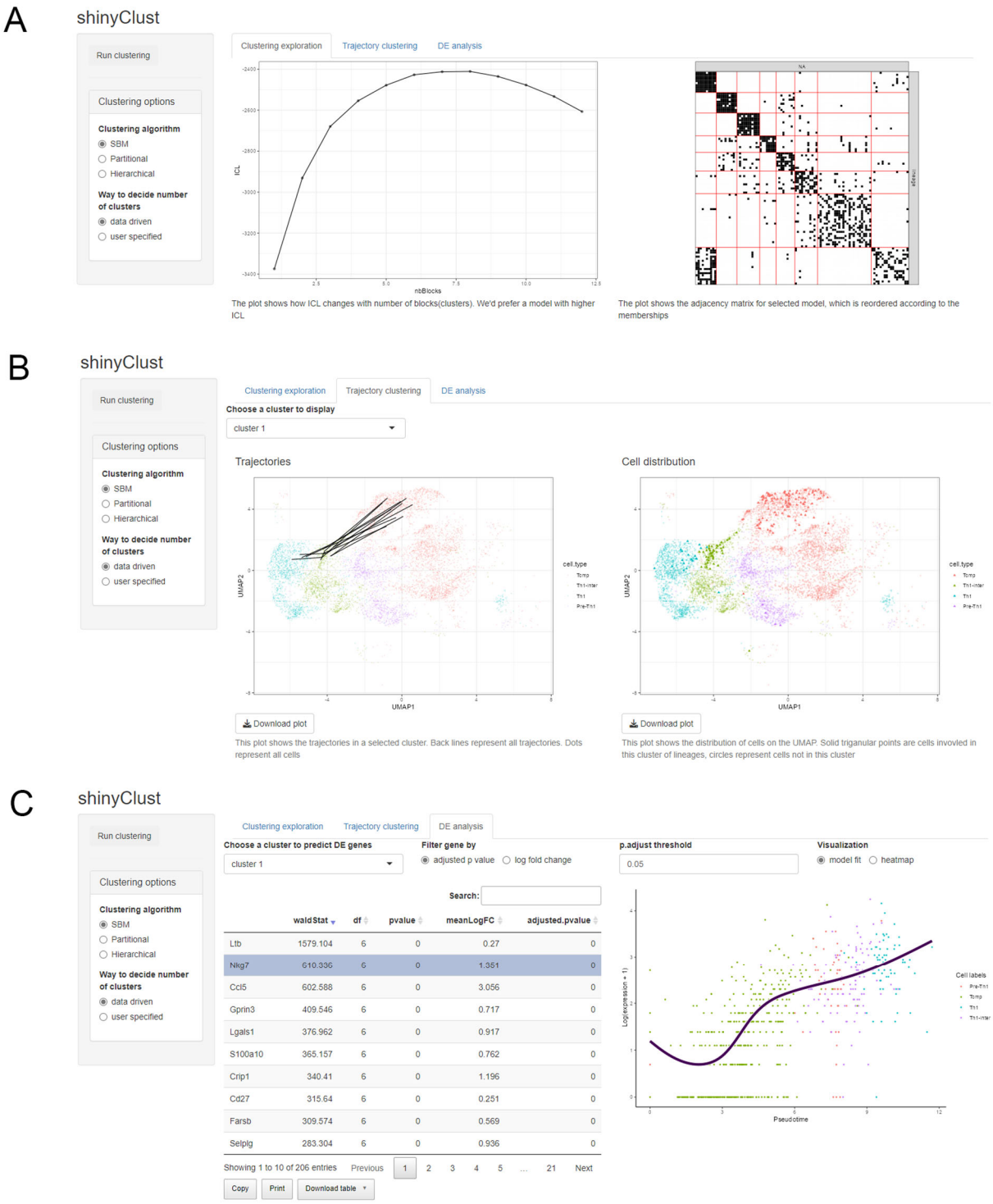


Figure3. *shinyClust*, a Shiny app for trajectory clustering and marker identification. **(A)** The 'Clustering exploration' tab shows the summary of trajectory clustering. **(B)** The 'Trajectory clustering' tab

263 shows the trajectory clustering results. **(C)** The 'DE analysis' tab allows users to identify marker genes for
264 each trajectory cluster and their changes as a function of the pseudotime.

265

266 Results

267 Exploratory analysis of scRNA-seq and scTCR-seq data

268 To demonstrate the utility of LRT, we analyzed the scTCR-seq and scRNA-seq data generated from the
269 antigen-specific $CD4^+$ T cells obtained from five mice during acute lymphocytic choriomeningitis virus
270 infection [26] (GEO accession number: GSE158896). Seurat was used to preprocess scRNA-seq data and
271 implement UMAP dimension reduction and cell clustering. Specifically, for each sample, we filtered out
272 doublets and dead cells (those with a percentage of mitochondrial genes > 10%), calculated cell cycle scores
273 for the remaining cells and regressed out, and normalized using the *SCTransform* approach [23]. Then, the
274 processed datasets from five samples were integrated using the Seurat data integration workflow. Finally,
275 PCA and UMAP dimension reduction were conducted, and then cells were clustered using the Louvain
276 algorithm [27], for which the first 20 principal components and default resolution were used. We obtained
277 11 cell clusters and annotated them based on the expert knowledge (**Fig. 4**), including T central memory
278 precursor cells (Tcmp) and the cells corresponding to different differentiation stages leading to the type 1
279 helper T cells (Th1). Specifically, based on the expression of well-known T cell subtype marker genes [26],
280 we annotated cell clusters 0, 1, 4, 9, and 10 as Tcmp, cell clusters 5 and 6 as pre-Th1, cell clusters 2 and 8
281 as Th1-intermediate (Th1-inter), and cell clusters 3 and 7 as Th1.

282 For scTCR-seq data, only the chains that were annotated as full-length and functional were retained.
283 Besides, to avoid any contamination from sorting, only clones having at least two cells from a single sample
284 were considered. After filtering, the TCR sequences from five samples were combined into a single data
285 frame using the R package 'scRepertoire' [10]. The resulting scTCR-seq data was then integrated with the
286 above-mentioned Seurat object for the remaining data analyses. Exploratory analysis for the clonotypes

287 was conducted using the *shinyClone* app. There were 516 clonotypes in this dataset and, among those, 69
288 spanned all four annotated cell types, while 182 of them spanned only one cell type. The top two most
289 abundant clonotypes (defined by CDR3 β chain amino acid sequence) were 'CASSFAGGASYEQYF' and
290 'CAWDRDTEVFF', which appeared in 863 and 762 cells, respectively. Most of the cells with the
291 clonotype 'CASSFAGGASYEQYF' were Tcmp cells and located at the bottom of the UMAP plot. The
292 cells with the clonotype 'CAWDRDTEVFF' evenly spanned all four cell types and these cells mainly
293 occupied the upper part of the UMAP plot.

294

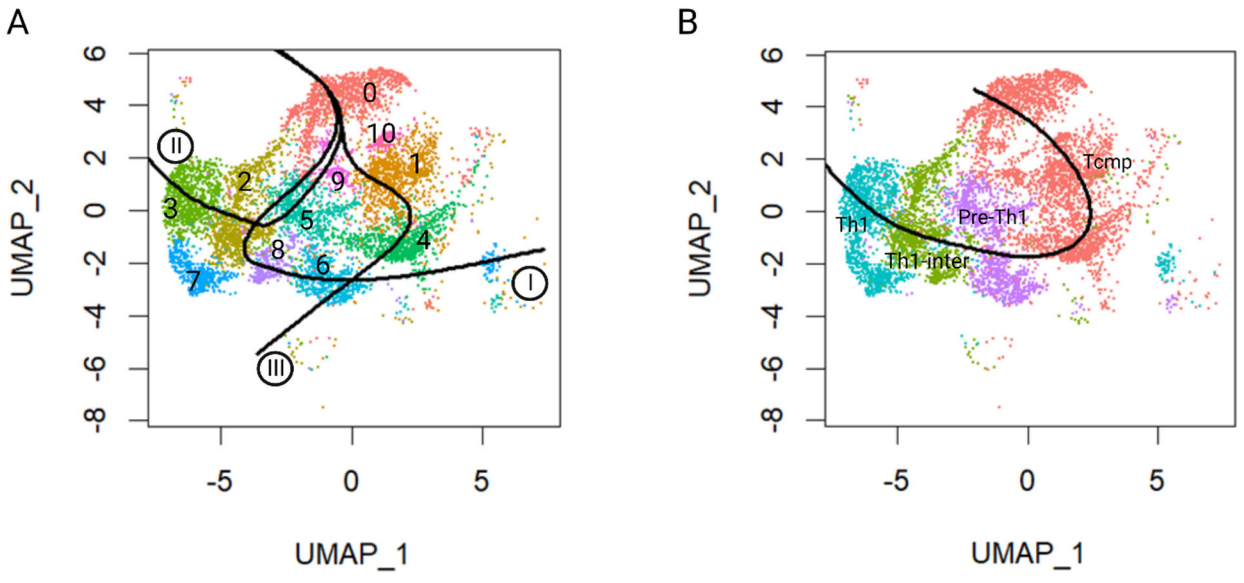
295 **Cell trajectory inference**

296 We first inferred cell trajectories based only on scRNA-seq data using the Slingshot algorithm [3] (**Fig. 4**).
297 When we used the Seurat cell clusters (using cluster 0 as the root), Slingshot predicted three trajectories
298 (**Fig. 4A**), including:

299 - Trajectory I: 0—10—9—5—2—8—7—6
300 - Trajectory II: 0—10—9—5—2—3
301 - Trajectory III: 0—10—1—4

302 When we inferred cell trajectories based only on scRNA-seq data but using four annotated cell types,
303 Slingshot identified a single trajectory of Tcmp – pre-Th1 – Th1-intermediate – Th1 (**Fig. 4B**), which
304 coincides with the known trajectory of Th1 cell differentiation.

305



306

307 **Figure 4. Trajectories inferred based only on scRNA-seq data of the antigen-specific CD4$^+$ T cells**
308 **visualized on the UMAP. (A)** Cell trajectory inferred based only on scRNA-seq data using the Slingshot
309 algorithm. Dots represent cells and curves represent the inferred trajectories. Here we considered Seurat
310 cell clusters, which are denoted by colors and numbers (0 – 10) in the plot. Roman numbers within circles
311 represent trajectory indices. **(B)** Cell trajectory inferred when we considered the expert-annotated cell types.
312 Cells are colored and labeled accordingly in the plot (salmon for Tcmp, light violet for pre-Th1, green for
313 Th1-inter, and Iris blue for Th1), and the black curve represents the inferred trajectory.

314

315 Next, we inferred cell trajectories by integrating scRNA-seq and scTCR-seq data using the LRT
316 framework (with SBM as a clustering algorithm). In this case, we discovered five sub-groups of trajectories
317 constituting this trajectory (**Fig. 5A**, **Fig. S1**). To evaluate whether this trajectory clustering makes sense
318 from the perspective of TCR sequences, we first checked similarities in TCR sequences within vs. between
319 the trajectory clusters. Specifically, we obtained numeric embeddings for TCR sequences using the tessa
320 algorithm [11] and calculated the Euclidean distances between each pair of TCR sequences. We found that
321 within-cluster distances were smaller than or comparable to between-cluster distances (**Fig. 5B**). Among
322 those, within-cluster distances are significantly smaller than between-cluster distances in the case of the

323 trajectory clusters III and IV (p-value=7.38e-14 and 4.01e-7, respectively, one-sided Welch two-sample *t*-
324 test).

325 Interestingly we found that, among the five trajectory clusters, four of them started from the Tcmp
326 cluster but three of them started from different Tcmp sub-clusters. For instance, the trajectory clusters III,
327 I, and V started from Seurat cell clusters 0, 1, and 4, respectively (**Fig. S1**). First, to understand functional
328 differences among them, we identified marker genes for each cell cluster using Seurat and conducted a gene
329 set enrichment analysis using these marker genes. Specifically, hypergeometric tests were applied based on
330 a compilation of 160 immune signatures from the Immune Landscape of Cancer [28], followed by the
331 multiple testing adjustment using the Benjamini-Hochberg procedure [29]. Results indicate that these Tcmp
332 sub-clusters play disparate functional roles (**Fig. 5C**). Specifically, cell cluster 4 was enriched for cell cycle
333 mitotic and cell cycle checkpoints genes, cell cluster 1 was enriched for IL2 activation and signaling
334 pathways, and cell cluster 0 was enriched for immune response and T cell activation. This analysis suggests
335 that integration of scTCR-seq and scRNA-seq data using LRT can reveal the complexity of trajectories that
336 can be missed by the trajectory analysis based only on scRNA-seq data. Next, to understand differences
337 among trajectory clusters from a clonal perspective, we conducted the TCR repertoire analysis based on the
338 Seurat cell clusters. Specifically, we evaluated the overlap of trajectory clusters in terms of clonotype using
339 the Morisita index (**Fig. S2**). Morisita index ranges from 0 to 1 and the larger value indicates a higher extent
340 of overlapping [30]. Repertoire analysis indicates an extremely low overlap of clonotypes among these
341 Tcmp sub-clusters, which further confirmed our clustering results.

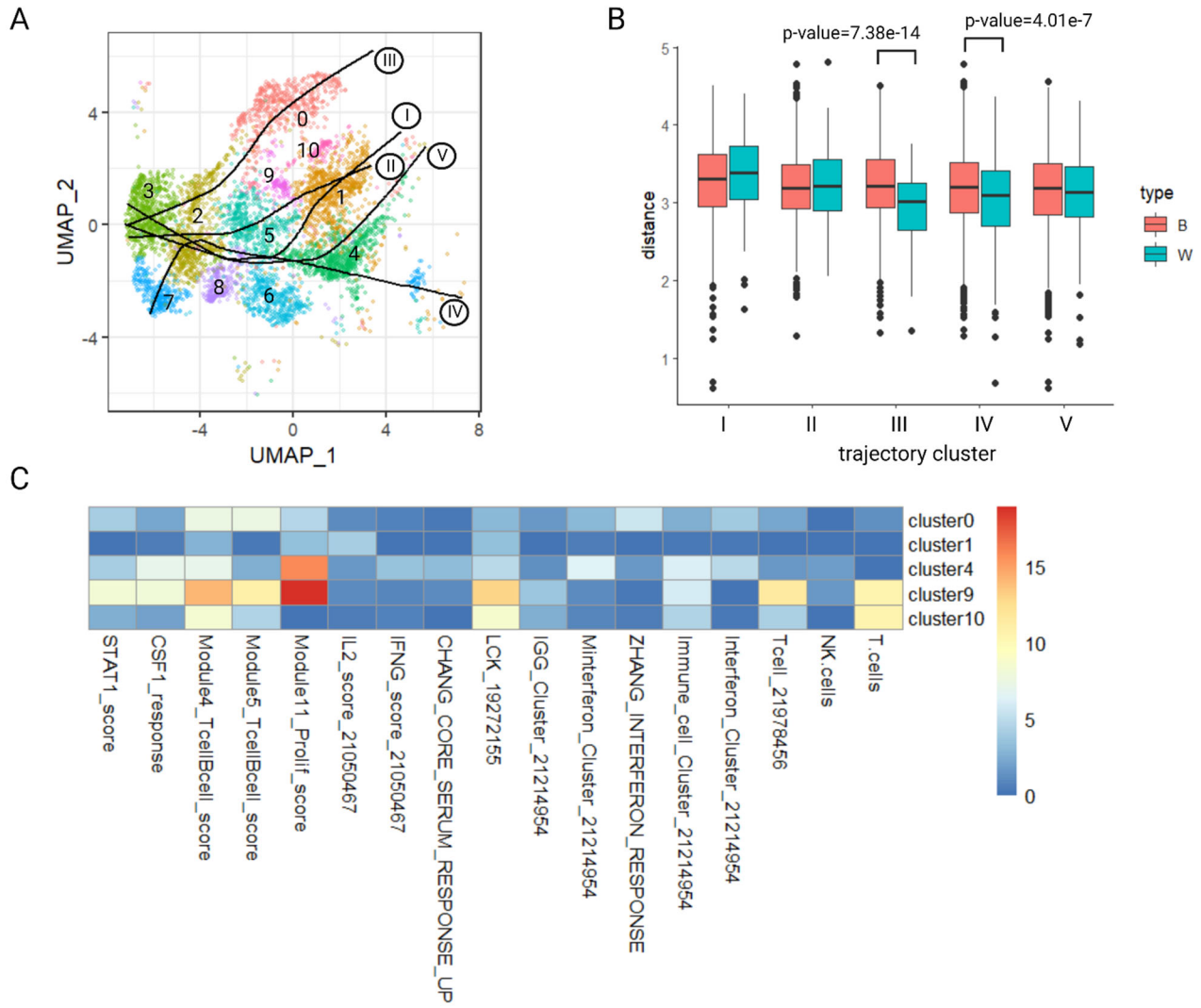


Figure 5. Trajectory inference using the LRT framework. (A) Trajectory clusters obtained using LRT. Dots represent cells, while colors and numbers (0 – 10) represent Seurat cell clusters. Each curve indicates the final trajectory for each trajectory cluster, and Roman numbers within circles (I – V) denote trajectory cluster indices. **(B)** Comparison of distances between TCR sequences, where 'W' and 'B' refer to within- and between-cluster distances, respectively (see the main text for calculation of these distances). **(C)** Enrichment analysis results for the Seurat cell clusters annotated as the Tcmp cell type. The colors represent -log10-transformed adjusted *p*-values.

351 Finally, we identified marker genes for each trajectory cluster using the NB-GAM approach (**Table**
352 **S1**) and evaluated their expressions as a function of the pseudotime. In general, we found that multiple
353 well-established Th1 markers showed increasing expressions while multiple well-known T follicular helper
354 cell (Tfh) markers showed decreasing expressions as a function of pseudotime (**Fig. 6**; **Fig. S3**). For instance,
355 it has been reported that enhanced inhibitor of DNA binding 2 (*Id2*) promotes Th1 cell differentiation while
356 suppressing *Id3* [31, 32]. Therefore, in the process of $CD4^+$ T cell differentiating towards Th1 cells, it is
357 expected that *Id2* is upregulated while *Id3* is downregulated, which coincides with our observation for the
358 identified trajectory clusters (**Fig. 6A** and **6B**; **Fig. S3**). Furthermore, chemokine receptor *Cxcr6* serves as
359 a useful marker for Th1 cells [26] while *Cxcr5* is a known marker for Tfh cells. When enhanced *Id2*
360 expression in $CD4^+$ T cell shifts the balance of Th1/Tfh towards Th1 cells, *Cxcr6* expression would be up-
361 regulated while *Cxcr5* expression would be suppressed [31]. **Fig. 6C** and **6D** showed patterns in line with
362 this point. Besides the above-mentioned markers, we also observed increasing trends for other Th1-
363 associated markers, such as the Th1 lineage-defining transcription factors *Tbx21*, *Gzmb*, *Ly6c2*, and *Cx3cr1*,
364 while observing decreasing trends for the Tfh-specific genes, such as *Ascl2*, *Il4*, *Il21*, and *Tox2* [33] (**Fig.**
365 **S3**). In summary, these expression patterns of Th1- and Tfh-associated markers nicely coincide with the
366 established Th1/Tfh biology, which further confirms the cell trajectories identified using LRT.

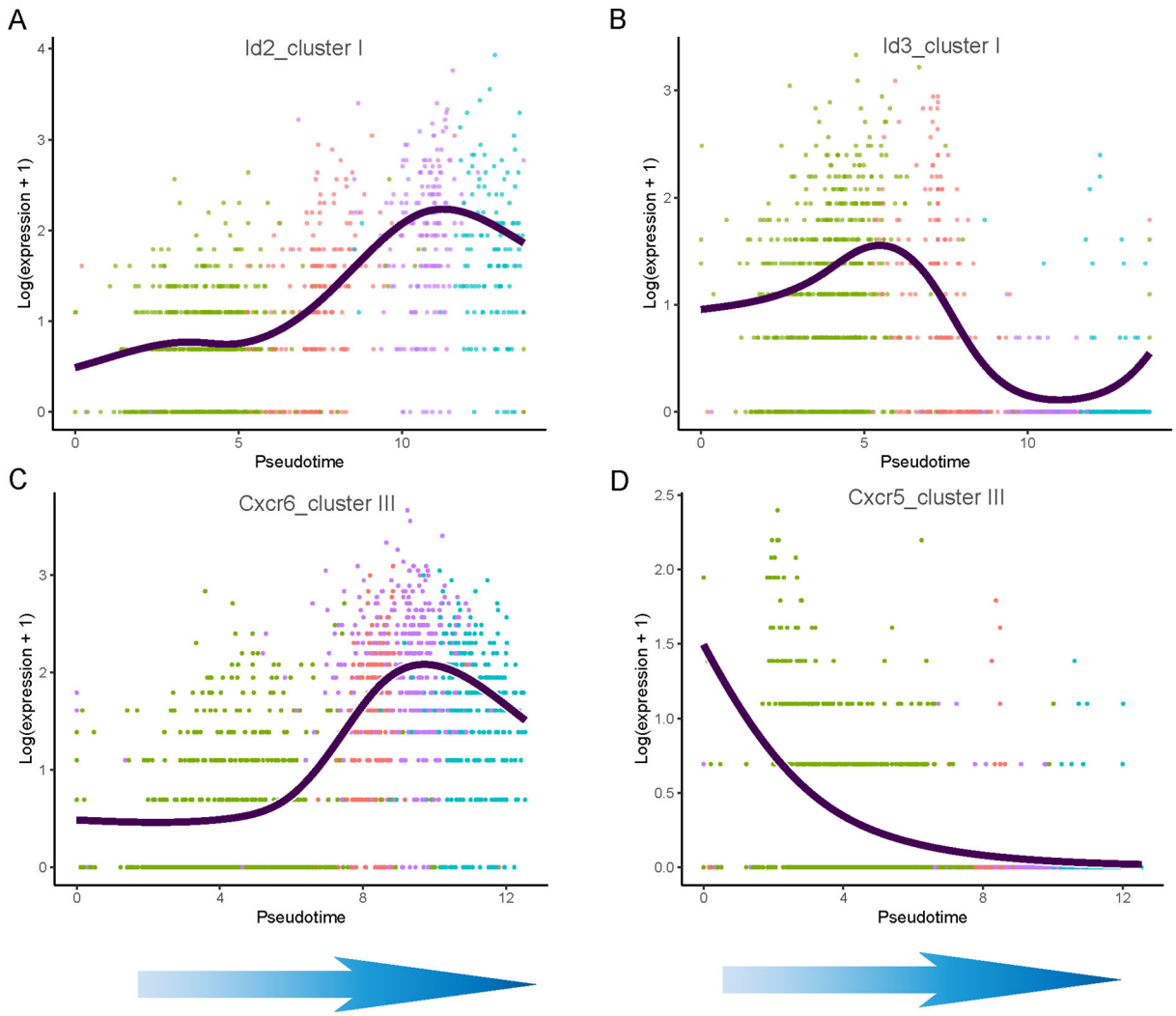


Fig. 6. Examples of the trajectory marker genes identified by LRT, which are involved in T cell differentiation. (A) and **(B)** Log-transformed read counts of *Id2* and *Id3* genes, respectively, as a function of pseudotime, for the trajectory cluster I. **(C)** and **(D)** Log-transformed read counts of *Cxcr6* and *Cxcr5* genes, respectively, as a function of pseudotime, for the trajectory cluster III. Curves represent smoothing splines fits and dots represent cells. The colors denote different cell types (green for Tcmp, salmon for pre-Th1, light violet for Th1-inter, and Iris blue for Th1). Arrows represent the direction of differentiation.

Discussion

In this paper, we proposed LRT, a novel computational framework for inferring T cell trajectory by integrating scTCR-seq and scRNA-seq data. LRT addresses the limitation of previous trajectory inference methods that solely rely on scRNA-seq data and neglects the clonal relationship between cells. Specifically, LRT utilizes both the functional information from scRNA-seq data and clonal information from scTCR-seq data. Such integrative analysis allows researchers to identify cell trajectories that cannot be revealed solely based on scRNA-seq data. In addition, LRT provides information about how clonotypes are clustered based on their DTW distances and MST, and this can help researchers decide which clonotypes need to be investigated jointly. Finally, LRT allows researchers to identify marker genes characterizing each of these trajectory clusters through DE analysis based on the principal curve and NB-GAM approaches, and this will help them understand potential drivers of cell differentiation, which is often the ultimate target.

Despite such strengths and power of LRT, it is also not free of limitations, which we plan to investigate in the future. First, while the current LRT framework assumes a single trajectory within each trajectory cluster, biologically various branching events can be possible within each trajectory cluster. We plan to address this limitation by considering more complex trajectory inference algorithms [3, 4]. Second, currently, LRT employs a simple association test evaluating which genes show significant expression changes along the pseudotime. However, researchers might also want to know whether these changes occur more in the earlier stage or in the later stage of the differentiation process. This limitation might be addressed by considering different types of the null hypothesis for the Wald test of NB-GAM. For example, we may modify the contrast matrix so that we can compare gene expression at starting points with those at ending points, as implemented in tradeSeq [21]. Third, currently, LRT implements DE analysis only to determine marker genes characterizing each trajectory cluster. However, researchers might also want to check which genes show DE between trajectory clusters. We plan to investigate approaches like those proposed by Van den Berge et al. [21] to address this. With the aforementioned strengths of the proposed

399 LRT framework and these future development plans, we believe that LRT can be a powerful tool for cell
400 trajectory inference and integrative analysis of scRNA-seq and scTCR-seq data.

401

402 Supporting information

403 **Figure S1. Clonotype trajectories belonging to each of the trajectory clusters I - V.**

404 **Figure S2. TCR repertoire overlap between cell clusters.** The numbers within cells show Morisita
405 indices. The numbers on the *x* and *y* axis denote cell cluster indices.

406 **Figure S3. Additional examples of the trajectory marker genes identified by LRT, which are**
407 **involved in T cell differentiation.** The left column shows the Th1-associated markers and the right
408 column shows the Tfh-associated markers. Curves represent smoothing splines fits and dots represent
409 cells. The colors denote different cell types (green for Tcmp, salmon for pre-Th1, light violet for Th1-
410 inter, and Iris blue for Th1).

411 **Table S1. List of DE genes identified using LRT for the antigen-specific CD4$^+$ T cell data.** Each tab
412 provides marker genes for each trajectory cluster. Wald statistics (waldStat), degree of freedom (df),
413 original *p*-value (pvalue), mean log fold change (meanLogFC), and Bonferroni-adjusted *p*-values (p.adj)
414 are reported for each gene.

415

416 Data Availability

417 The LRT framework was implemented as an R package 'LRT' and it is publicly available at
418 https://github.com/JuanXie19/LRT. The Shiny apps 'shinyClone' and 'shinyClust' are also provided as part
419 of this R package. The data used to produce the results presented in this manuscript is available at GEO
420 with accession number GSE158896.

Funding

This work has been supported through grant support from the National Human Genome Research Institute (R21 HG012482), National Institute on Aging (U54 AG075931), National Institute of General Medical Sciences (R01 GM122078 and R01 GM131399), National Institute on Drug Abuse (U01 DA045300), the National Science Foundation (NSF1945971), and the Pelotonia Institute of Immuno-Oncology (PIIO). The content is solely the responsibility of the authors and does not necessarily represent the official views of the funders.

Competing interests

The authors have declared that no competing interests exist.

References

1. Wagner DE, Klein AM. Lineage tracing meets single-cell omics: opportunities and challenges. Nature Reviews Genetics. 2020;21(7):410-27.
2. Trapnell C, Cacchiarelli D, Grimsby J, Pokharel P, Li S, Morse M, et al. The dynamics and regulators of cell fate decisions are revealed by pseudotemporal ordering of single cells. Nature biotechnology. 2014;32(4):381-6.
3. Street K, Risso D, Fletcher RB, Das D, Ngai J, Yosef N, et al. Slingshot: cell lineage and pseudotime inference for single-cell transcriptomics. BMC genomics. 2018;19(1):1-16.
4. Saelens W, Cannoodt R, Todorov H, Saeys Y. A comparison of single-cell trajectory inference methods. Nature biotechnology. 2019;37(5):547-54.
5. Zafar H, Lin C, Bar-Joseph Z. Single-cell lineage tracing by integrating CRISPR-Cas9 mutations with transcriptomic data. Nature communications. 2020;11(1):1-14.
6. Stubbington MJ, Lönnberg T, Proserpio V, Clare S, Speak AO, Dougan G, et al. T cell fate and clonality inference from single-cell transcriptomes. Nature methods. 2016;13(4):329-32.
7. Scheper W, Kelderman S, Fanchi LF, Linnemann C, Bendle G, de Rooij MA, et al. Low and variable tumor reactivity of the intratumoral TCR repertoire in human cancers. Nature medicine. 2019;25(1):89-94.
8. Wang P, Jin X, Zhou W, Luo M, Xu Z, Xu C, et al. Comprehensive analysis of TCR repertoire in COVID-19 using single cell sequencing. Genomics. 2021;113(2):456-62.
9. Team I. immunarch: an R package for painless bioinformatics analysis of T-cell and B-cell immune repertoires. Zenodo10. 2019;5281.

10. Borcherding N, Bormann NL, Kraus G. scRepertoire: An R-based toolkit for single-cell immune receptor analysis. F1000Research. 2020;9.
11. Zhang Z, Xiong D, Wang X, Liu H, Wang T. Mapping the functional landscape of T cell receptor repertoires by single-T cell transcriptomics. Nature methods. 2021;18(1):92-9.
12. McInnes L, Healy J, Melville J. Umap: Uniform manifold approximation and projection for dimension reduction. arXiv preprint arXiv:180203426. 2018.
13. Graham RL, Hell P. On the history of the minimum spanning tree problem. Annals of the History of Computing. 1985;7(1):43-57.
14. Snijders TA, Nowicki K. Estimation and prediction for stochastic blockmodels for graphs with latent block structure. Journal of classification. 1997;14(1):75-100.
15. Allen C, Jung KJ, Chang Y, Ma Q, Chung D. Analysis of community connectivity in spatial transcriptomics data. bioRxiv. 2022.
16. Hart PE, Stork DG, Duda RO. Pattern classification: Wiley Hoboken; 2000.
17. Daudin J-J, Picard F, Robin S. A mixture model for random graphs. Statistics and computing. 2008;18(2):173-83.
18. Biernacki C, Celeux G, Govaert G. Assessing a mixture model for clustering with the integrated completed likelihood. IEEE transactions on pattern analysis and machine intelligence. 2000;22(7):719-25.
19. Hastie T, Stuetzle W. Principal curves. Journal of the American Statistical Association. 1989;84(406):502-16.
20. Wood SN. Generalized additive models: an introduction with R: chapman and hall/CRC; 2006.
21. Van den Berge K, Roux de Bézieux H, Street K, Saelens W, Cannoodt R, Saeys Y, et al. Trajectory-based differential expression analysis for single-cell sequencing data. Nature communications. 2020;11(1):1-13.
22. Hao Y, Hao S, Andersen-Nissen E, Mauck III WM, Zheng S, Butler A, et al. Integrated analysis of multimodal single-cell data. Cell. 2021;184(13):3573-87. e29.
23. Hafemeister C, Satija R. Normalization and variance stabilization of single-cell RNA-seq data using regularized negative binomial regression. Genome biology. 2019;20(1):1-15.
24. Chiquet J, Donnet S, Barbillon P. sbm:Stochastic Blockmodels. 2021.
25. Sarda-Espinosa A. dtwclust: Time series clustering along with optimizations for the dynamic time warping distance. R package version. 2019;5(6).
26. Khatun A, Kasmani MY, Zander R, Schauder DM, Snook JP, Shen J, et al. Single-cell lineage mapping of a diverse virus-specific naive CD4 T cell repertoire. Journal of Experimental Medicine. 2021;218(3).
27. Blondel VD, Guillaume J-L, Lambiotte R, Lefebvre E. Fast unfolding of communities in large networks. Journal of statistical mechanics: theory and experiment. 2008;2008(10):P10008.
28. Thorsson V, Gibbs DL, Brown SD, Wolf D, Bortone DS, Yang T-HO, et al. The immune landscape of cancer. Immunity. 2018;48(4):812-30. e14.
29. Benjamini Y, Hochberg Y. Controlling the false discovery rate: a practical and powerful approach to multiple testing. Journal of the Royal statistical society: series B (Methodological). 1995;57(1):289-300.
30. Chiffelle J, Genolet R, Perez MA, Coukos G, Zoete V, Harari A. T-cell repertoire analysis and metrics of diversity and clonality. Current Opinion in Biotechnology. 2020;65:284-95.
31. Hwang S-M, Im S-H, Rudra D. Signaling networks controlling ID and E protein activity in T cell differentiation and function. Frontiers in Immunology. 2022;13.
32. Yang CY, Best JA, Knell J, Yang E, Sheridan AD, Jesionek AK, et al. The transcriptional regulators Id2 and Id3 control the formation of distinct memory CD8+ T cell subsets. Nature immunology. 2011;12(12):1221-9.
33. Ruterbusch M, Pruner KB, Shehata L, Pepper M. In vivo CD4+ T cell differentiation and function: revisiting the Th1/Th2 paradigm. Annual review of immunology. 2020;38:705-25.