Improving High School Math Education in Portugal in GLM Framework

Heqiao Ruan
Department of Statistics, University of California, Davis

Background

- Portugal, a country located in southwestern Europe
- Statistics keeps high school education quality in Portugal at tail end in Europe
- Especially serious problems in students' **failure rate and dropout rate**
- Officials want to improve math education
- Our goal: identify important factors influencing students' math grade
- Does they match common sense?
- Propose corresponding advice to help government to overcome difficulties

Dataset

- **Student Performance Dataset**
- Depicts students' achievement collected using school reports and questionnaires
- 395 observations,33 features
- Response: G1(First Stage),G2(2rd Stage), G3(Final Grade)
- 30 Predictors: Most Categorical
- Distribution of Grade Level(G3):

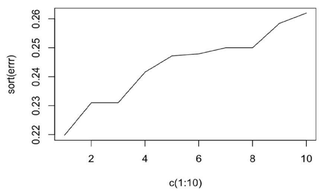
A(G3≥16)	$B(13 < G3 < 16)$	$C(11 < G3 < 14)$	$D(9 < G3 < 12)$	$F(G3 < 10)$
40	60	62	103	130

A-D Pass, F Fail

Method

- **Logistic Regression Model**: Fail(0,F)~Pass(1), 2-category classification /Probability of Passing on G3
- Prediction Error, 10-fold Cross Validation
- Model Selection by AIC

Predic/Level	Fail	Pass
Fail	60	25
Pass	70	240



Prediction Error 24.0% CV Error: 25.6%

Final Model after stepAIC:

$$Logit(E[Y|X]) \sim \beta_0+\beta_1 I(sexM)+\beta_2 age+\beta_3 I(Mjobhealth)+\beta_4 I(Mjobother)+\beta_5 I(Mjobservice)+\beta_5 I(Mjobteacher)+\beta_6 failures+\beta_7 I(schoolsupYes)+\beta_8 I(famsupYes)+\beta_9 I(higherYes)+\beta_{10} goout+\beta_{11} health+\beta_{12} failures * I(schoolsupYes)$$

Why add interaction? Schoolsupport effect contradict common sense

Method

Multinomial Model with 5-category: Proportional Odds Model, Baseline Odds Model on G3

Prediction Error:52.4% for Baseline Odds Model, 56.2% for Proportional OddsModel

Merge Categories

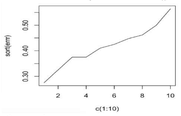
A:High, BCD:Medium, F:Low 3-category

Low(G3<10)	Medium($10 \leq G3 < 16$)	High($G3 \geq 16$)
130	225	40

Baseline Odds Model: Severe Lack of Fit. **Choose Proportional Odds Model**

Prediction Error:36.2% CV Error(10-fold):40.2% Performance Measurements:

Predic/Level	Low	Medium	High
Low	57	32	0
Medium	73	193	38
High	0	0	2



Accuracy	Precision	Recall	F-Score	F-Score($\beta = 0.5$)
0.638	0.649	0.703	0.676	0.683

Final Model after model selection :

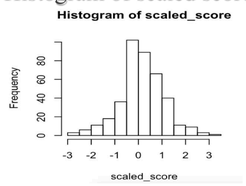
$$logit(P(Y \leq k)) = \beta_{0,k}+\beta_1 I(sexM)+\beta_2 age+\beta_3 I(PstatusT)+\beta_4 I(Mjobhealth)+\beta_5 I(Mjobother)+\beta_6 I(Mjobservice)+\beta_7 I(Mjobteacher)+\beta_8 studytime+\beta_9 failures+\beta_{10} I(schoolsupYes)+\beta_{11} I(famsupYes)+\beta_{12} I(higheryes)+\beta_{13} freetime+\beta_{14} goout+\beta_{15} health+\beta_{16} failures * I(schoolsupYes)$$

Transformed Model: Want to extract information from G1,G2 as well as G3

Perform PCA on cbind(G1,G2,G3)->Extract first principal component Y-> Scale Y to percent(0-100%) ->logit transformation on Y->Scaled Score

Scale-Score=logit(PERCENT(0.4629G1+0.5614G2+0.6859G3))

Histogram of scaled score: Approximately Normal, do **ordinary lm** on Scale-Score!



Summary of PCA: Table 10:PCA of G1,G2,G3:

Factor	PC1	PC2	PC3
Proportion of Variance	0.9095	0.06162	0.02892
Cumulative Proportion	0.9095	0.9711	1
G1	0.4629	0.8024	-0.3764
G2	0.5614	0.0632	0.8251
G3	0.6859	-0.5933	-0.4212

Final model after model selection:

$$E[Y|X] = E[ScaledScore|X] = \beta_0+\beta_1 sexM+\beta_1 I(Mjobhealth)+\beta_2 I(Mjobother)+\beta_3 I(Mjobservice)+\beta_4 I(Mjobteacher)+\beta_5 studytime+\beta_6 failures+\beta_7 I(schoolsupYes)+\beta_8 I(famsupYes)+\beta_9 I(higherYes)+\beta_{10} goout+\beta_{11} failures * schoolsupyes+\beta_{12} freetime+\beta_{13} health$$

Result(Important Predictors)

3.Logit Transformed Model after model selection

Predictors	Coefficient	Std.Error	t value	pvalue
sexM	0.251	0.092	2.735	6.53e-3
studytime	0.147	0.053	2.784	5.64e-3
failures	-0.418	0.0643	-6.506	2.5e-10
schoolsupyes	-0.465	0.137	-3.401	7.45e-4
famsupyes	-0.212	0.0872	-2.433	0.0154
romanticyes	-0.218	0.0890	-2.446	0.015
goout	-0.119	0.039	-3.098	0.002
health	-0.063	0.0299	-2.087	0.038
higher	0.750	0.632	1.186	0.236
failures:schoolsupyes	0.403	0.167	2.408	0.017

Result

1.Logistic Regression Model:

Predictors	Coefficient	Standard Error	z value	P Value
age	-0.217	0.108	-2.002	0.045
sexM	0.569	0.268	2.126	0.033
failures	-1.233	0.226	-5.460	4.76e-8
schoolsupyes	-1.334	0.385	-3.462	5.36e-4
goout	-0.346	0.114	-3.039	2.37e-3
higheryes	0.965	0.588	1.641	0.100
failures:schoolsupyes	1.412	0.475	2.982	2.87e-3

2.Proportional Odds Type Model after model selection

Predictors	Coefficient	Standard Error	z value	PVALUE
sexM	0.562	0.239	2.345	0.0195
age	-0.196	0.095	-2.065	0.396
failures	-1.278	0.221	-5.77	1.62e-8
schoolsupyes	-1.404	0.361	-3.888	1.19e-4
goout	-0.346	0.107	-2.047	0.041
higheryes	0.904	0.572	1.58	0.057
failures:schoolsupyes	1.478	0.447	3.309	1.027e-3

2(I):Intercepts

Prediction	Value	Std.Error	t value	pvalue
1|2	-4.706	1.863	-2.526	0.012
2|3	-1.163	1.842	-0.633	0.5270

Conclusion and Discussion

- Pay more attention to students who have fail before and provide extra support
- Pay more attention to Girls' study
- Arouse students' motivation to pursue higher degree, potentially help reduce dropout rate
- Extra Family support on math impose negative effect on students' grade. Avoid parents intervening
- Study more time, don't party too much

REFERENCES

[1].Paulo Cortez and Alice Silva. Using Data Mining to Predict Secondary School Student Performance. University of Minho Guimaraes, Portugal
[2].Hans-Georg Mueller. Generalized Linear Models Lecture Notes. UC Davis Winter 2018

Acknowledgement

I would like to express my thankfulness to Prof Hans Georg Mueller and TA Yaqing.

ECS 289N Project Report: Pseudo time reconstruction and evaluation in single cell RNA-seq analysis: Application to NKT cell Dataset

Heqiao Ruan SID:915490857
email:hruan@ucdavis.edu

March 18, 2018

1 Abstract

The recent technological improvement allows researchers to measure transcriptome in level of individual cells. One efficient way to gain biological insights is to quantitatively order the cells according to the transition status and relative expression along the whole process. In recent 5 years more than 10 methods are proposed to construct the trajectory while most of them are only perfectly applicable in limited cases. So new principal technique is yet to develop.

In this project we apply three popular methods in trajectory inference with some modifications in some key steps:TSCAN, Mpath and Monocle to generate the landscape of the gene expression level in the biological process and compare their performance and compare them. What's more, we will try to extract some biological insight and the key regulator controlling NKT cell differentiation in mouse.

2 Introduction

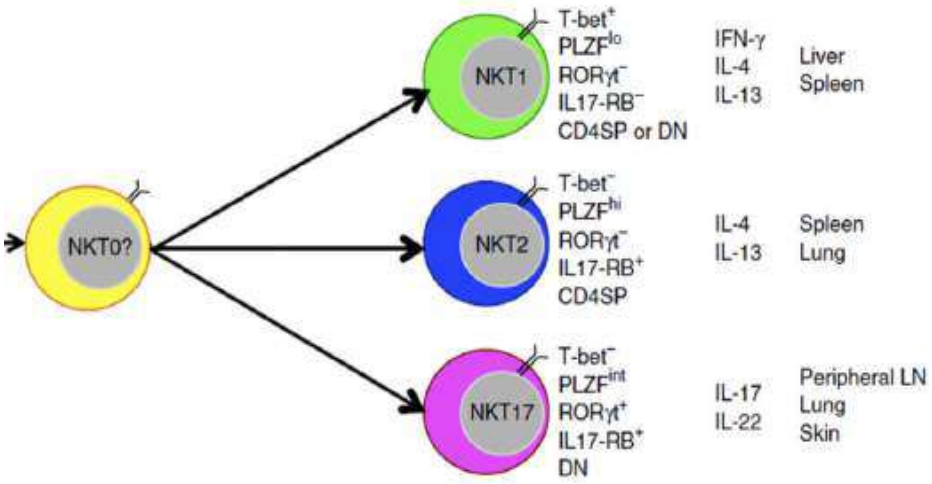
Traditionally in biological experiment we tend to measure on a bulk of. Single cell RNA-seq is a relatively new technology that allows researcher to measure the expression based on every individual cell. It has a couples of advantages compared with the traditional RNA seq techniques which are based on the average of gene expression level. First it can construct a more resolute picture to capture signals conveyed by some single cells which can be easily ignored in bulk RNA-sequence. Secondly, the single-cell RNA-seq is capable of generating a well-rounded picture of the whole gene expression landscape in a highly heterogeneous cell population.
Here we apply various trajectory inference methods including Monocle,TSCAN and Mpath

to re-order the cells according to their relative expression status and try to reconstruct the trajectory reflecting the cell differentiation process.

3 Dataset and Preprocessing

3.1 Dataset

Here we use the NKT cell dataset which was first used in [2]. The dataset contains **22694 rows and 203 columns** where each row stands for a gene while each column represent a single cell. The cells are sequences ranging from 445 to 647 and we just make the assumption that the cells with smaller number are collected prior to the cells with larger number. For example, the cell coded as 550 is collected before the cell coded as 551. We have to point out this is not a very rigorous assumption but in general the cells follow this sequence. **The dataset consists of four types of cells:NKT0,NKT1,NKT2,NKT17. We assume that the four types of cells are collected in time sequence which means NKT0 is the earliest and NKT2 is the latest collected and this is the most fundamental assumption of our downstream analysis**.
The biological path of differentiation of Natural Killer T cell in mouse is shown as below:



We can see that the common sense is NKT0 usually differentiate to NKT1, NKT2, NKT17 cells and here in this dataset the cell collection time is **NKT0− >NKT1− >NKT17− >NKT2**. However, the NKT1,NKT17 and NKT2 may have some transition relation which is yet to be identified.

3.2 Preprocessing

In the real life setting, dataset are not always in a good shape so it requires preprocessing and quality control. Especially in gene inference framework, the dataset is sparse and in general, 70% of all observations are 0. So as our ultimate goal is to explore the trend of gene expression and cell fate process based on them, it becomes particular important for us to filter out some

low or constant expressed genes as well as deleting cells that doesn't express any gene.
First, we perform log transformation on the dataset to deal with extreme value and in practice, we would like to add a pseudo smoother 1 on the original value to deal with 0 observation so that makes it still 0 after transformation. In math formulation, it's: $EXPRESSION = log_2(expressionlevel + 1)$.
After performing log transformation, we are going to perform quality control on cells and genes. Unfortunately, as we know, there's no universal method for us up to now to set the preprocessing threshold corresponding to a specific dataset. Here we just artificially specify the threshold. First we filter out the genes that are expressed in less than 5% of all cells. Secondly we filter out the genes that the coefficient of variation across all cells less than 0.5. The coefficient of variation is $v = \frac{var(Y)}{E[Y]^2}$ which represents the variance divided by the square of mean.
After performing preprocessing, the dataset are of **5138 rows(genes) and 197 columns(cells)**.
Here we have to point out that we just set one threshold for all of the methods we use but we do acknowledge there are some limitations on this setting for in practice, the threshold may makes a big difference.

3.3 Preliminary Analysis

First as we know that the gene naming system is pretty confusing and we don't have access to the specific symbol name of each gene, we have to artificially choose some "gold standard" genes which can also be named as so-called "marker genes"(it maybe a somewhat clumsy name but it does has this kind of property). Here the gold standard genes are identified as the most differential expressed genes across the time.
As we know that the most of the observation is 0, the relationship between the expression level of a single gene and the time is obviously nonlinear we can only explore the functional trend of expression across time. So here we use generalized additive model(GAM) to explore this relationship(Maybe using trend here is more appropriate).
Assuming Y as the expression level of a single gene, x is the collection order. So the generalized additive model is $g(E[Y|X = x]) = s(x, k)$, the s denotes the smoothing function with degree k. Note that normally k is no more than 3 to avoid the curse of dimensionality. Here link function g is either identity(normal additive model) or log link(Poisson additive model). After fitting, we can identify the trend of response variable. We have to point out that it can only generally check the functional relationship instead of fitting the specific regression coefficients for the observation is pretty sparse.
Then we try to identify the degree of differential expression by applying likelihood ration test.Here we use the asymptotic property: $Deviance(Nullmodel) - Deviance(fullmodel) -> \chi_1^2$ where the Nullmodel is fitted as we treat the gene expression level along time as constant. Then we can easily see that the smaller the p value, the more significant of this differential expression effect is. Then we use **holm-Bonferroni** procedure to control the familywise error rate to get a more powerful p value denoting as q value. Then we sort the genes by q value by increasing order to help us find the gold standard genes.
We artificially specify the gold standard genes as the top 100 differential genes

which has the smallest q value. These genes are particular important in our downstream analysis. What's more, the gene id of the first 20 differential expressed genes are shown in Appendix B.

4 Main Methods

4.1 TSCAN and Modifications

TSCAN(also named as Tool for Single Cell Analysis) is originally brought out by Zhicheng and Hongkai in JHU Department of Biostat in 2016 which is based on connecting the clusters after performing dimension reduction. It is a unsupervised learning technique.
This methods can be summarized into three steps. Firstly, it clusters the cells with similarly expression profiles. Then the minimal spanning tree is constructed to connect all the cluster centers as well as specifying the order of the cluster centers. Finally cells are projected to the backbone of the tree so that we can determine their pseudotime and order.
One important feature of **TSCAN** is that it cluster the similar expressed genes together to alleviate the drop out event by using the average expression profile of the clustered genes. Some research have demonstrated that clustering genes before conducting the main algorithm can improve the performance and we will validate this in the downstream analysis(potentially alleviate the dropout effect).
In the original paper, Zhicheng and Hongkai applied **PCA** in dimension reduction step and used mclust based on mixed gaussian assumptions in clustering step. PCA is among the most popular dimension reduction techniques to extract a few number of features with most of the variability of the data and in the original paper they use the LS technique to find the optimal number of principal components extracted. Mclust is a model based clustering algorithm which is optimized by EM algorithm by assuming that very cell follows a multivariate normal distribution.
Then after clustering the cells we are going to construct the trajectory and project each cell onto the trajectory. We start by constructing the minimal spanning tree which has the smallest sum of length of the edges connecting each vertex. Then while the trajectory may often be branching we find the longest path of the minimal spanning tree which has the largest numbers of clusters with the largest total numbers of cells and then in terms of choosing a origin, **we use the gene expression profile of the second marker gene(ENSMUSG00000001025.8) as it is minimal expressed at NKT0 cells compared to other three types of cells**. Then we first exhaust the main path and then add the branches onto the tree iteratively. We want to point out that usually the numbers of clusters won't excess 6 or 7 which greatly reduces the variability and complexity of the tree space comparing with Monocle and alleviate the risk of being contaminated by the various source of noise arisen from the previous analysis and even in the biological experiment. Then after ordering the clusters we project the cells onto the edges of the tree and for cell A in C_m we project it to $C_m - C_{m+1}$ is $d(A, center(C_{m+1})) < d(A, center(C_{m-1}))$ and project it to C_{m-1} vice versa. Then the cell

ordering are determined following these steps, first for cells in the same cluster projecting onto the same edge, their order is determined by the projected values on the edge(for cells in C_m,$C_{m-1} - C_m$ is negative while $C_m - C_{m+1}$ is positive). Then the order of cells in each cluster is determined by the order of edges. Finally we use the order of clusters to order them together. We can see that TSCAN greatly reduces the complexity and variability of the minimal spanning tree. **However, the good performance of TSCAN is pretty highly depend on the appropriateness of MoG clustering optimized by the EM algorithm** and it has huge potential in terms of developing even more delicate clustering techniques.
Here in actual implementation, we make several modifications and compare the new algorithm with the original one. First as some research indicate that the dimension reduction technique diffusion map which based on markovian transition matrix may performs perfect in some cases. Secondly, besides mclust, kmeans has been demonstrated a particularly efficient algorithm to combine nearby data points together(which means similarly expressed cells in gene inference our framework). What's more, we artificially set 4 clusters in mclust to perform the original version of algorithm. So we try another three techniques **Nonclu-TSCAN**,**Kmeans-TSCAN**,**Diffusionmap-TSCAN**.
The **Kmeans-TSCAN** starts from clustering similarly expressed genes following by PCA to reduce the dimension. Then we perform kmeans clustering and set the clusters as 4 after that we use TSCAN to project the cells onto the path.
The **Diffusionmap-TSCAN** starts from clustering similarly expressed genes following by dimension reduction technique Diffusion Map. Other steps are the same as the original TSCAN technique. The matrix W is given by $w_{ij} = \frac{exp(-\frac{||X_i - X_j||^2}{2*\sigma^2})}{\sum_j exp(-\frac{||X_i - X_j||^2}{2*\sigma^2})}$ and the t th step diffusion distance is given by $D_{ij}^t = \sum_k \lambda_k^{2t}(\phi^k(X_i) - \phi^k(X_j))^2) = ||\Phi(X_i) - \Phi(X_j)||^2$ where λ_k denotes the k th eigenvalue and $\Phi_k(X)$ is the k th eigenvector of the markov transition matrix. Then we can extract the first k diffusion coordinates $\P = (\lambda_1^t\phi_1(x), ..., \lambda_d^t\phi_d(x))^T$ which contain most of the information in the data. There are two main issue in the implementation,**the first is to find the tuning variance σ^2 while the second is to determine the number of diffusion components we extract**.
For the first problem, indeed there will be a range of parameter σ which the markovian transition matrix defines an ergodic diffusion process on the data as a connected graph and still the diffusion distances between the cells are informative. Here we use the **median of distance to the k th nearest neighbor of every cell** which helps us try best to maintain the local property. It indeed reaches a cost-accuracy tradeoff for the more neighbors we select, the more noise it will contain. Empirically we set **k as 10% of the total number of cells**. For the second problem, we use the rank-based criteria proposed in [12] by John A.Lee which measure the degree of consistency local property between the original data and embedding space. They use the **quality measure** $Q_{NX}(k) = \frac{1}{NK}\sum_{i=1}^{N}|v_i^k \cap n_i^k|$ **and the adjusted quality measure** $R_{NX}(k) = \frac{(N-1)Q_{NX}(k)-k}{N-1-k}$**.** We try the plot the $R_{NX}(k), Q_{NX}(k)$ versus k(search from 2 to 30 empirically) and observe them to get the best k based on the cost-accuracy tradeoff.**We implement Diffusion Map algorithm by following the above description.**
Another alternative technique here to do the dimension reduction is the **t-SNE TSCAN**.

t-SNE(also named as t-Stochastic Neighbor Embedding) is a popular dimension reduction method which has been demonstrated particular useful in visualization of high dimensional dataset. Here we aim to map the d dimension data $(X_1, ..., X_n)$ into 2 dimension in which maintain its local similarity. The similarity matrix is constructed by $p_{ij} = \frac{p_{j|i}+p_{i|j}}{2N}$ and $p_{j|i}$ is calculated by $p_{j|i} = \frac{exp(-||X_i-X_j||^2/2\sigma_i^2)}{\sum_{k\neq i} exp(-||X_i-X_j||^2/2\sigma_i^2)}$. It depicts the similarity between the two points in high dimension. We assume the transformed data points are $Y_1, ..., Y_n$ and $q_{ij} = \frac{\frac{1}{1+||Y_i-Y_j||^2}}{\sum_{k\neq i}\frac{1}{1+||Y_i-Y_k||^2}}$. Here the loss function is given by Kullback-Leibler Divergence $DL(P||Q) = \sum_{i\neq j} p_{ij} log(\frac{p_{ij}}{q_{ij}})$ which depicts the dispersion of one distribution from the other. Then we use the MoG(Mixture of Gaussian) clustering methods to cluster the similar expressed cells and use the same technique as described in TSCAN to order the cells. Then we can also apply various performance measures to evaluate the efficiency of ordering and trajectory.
Here we have to point out that actually the perplexity makes a big difference in the performance. The number of perplexity means the information they gain from the nearest neighbors so too small perplexity will bring out severe information loss while too large perplexity will contain too much noise that can't be identified by us. Typically in implementation, we use 30 and here we try three different perplexity and compare their performance.
TSCAN has its own advantage in greatly reducing the number of vertices in the minimal spanning tree so that it will be more robust than Monocle which will be demonstrated later. What's more, we use some prior information of marker gene expression and it's not completely unsupervised learning technique.

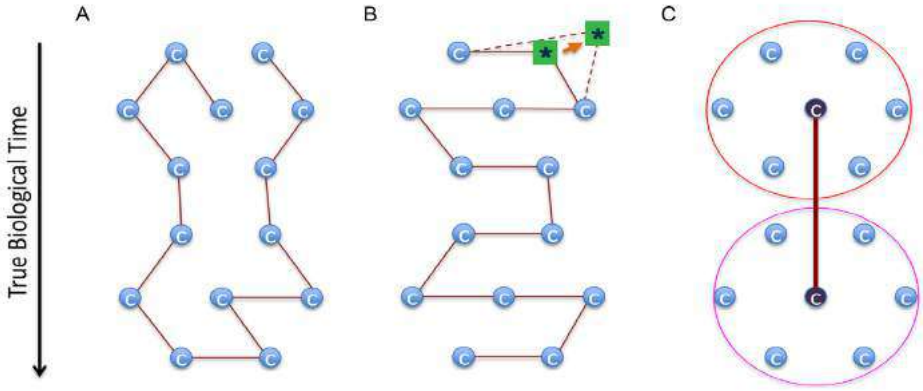
4.2 Mpath

Mpath is a relative recent algorithm that can help us construct a branching trajectory by choosing the so-called landmark cluster centers and then find the minimal spanning tree connecting them. Its idea is from the empirical assumption that the likelihood of two clusters of cells are higher if the number "between" this two clusters are large. So we construct the transition network based on this assumption.
Mpath starts by hierarchically clustering the cells after preprocessing in **ward** distance which is given by $\Delta(A, B) = \frac{n_A n_B}{n_A+n_B}||m_A - m_B||^2$ where m_A and m_B are cluster center of A,B respectively. Then the problem become how to choose the number of cut on the dendrogram which is equivalent to choose the number of clusters. Here we use two measure to perform the quality control and we denote the clusters passing this two as landmark clusters. First we use the size of 5% of total number of sizes as the threshold and the cluster sizes larger than this passing the quality control. Secondly about the purity measure, we use the shannon diversity $H = -\sum_{i=1}^{n} p_i ln(p_i)$ where p_i is the proportion of category i in the whole sample and the threshold is 0.6. **We have to point out that smaller diversity means the cluster is 'pure' which means most of cells in this cluster are from the same type which is pretty important to construct the cell fate branching network.** Then we search from 4 to 20 and plot the number of clusters versus the total number of clusters to determine the optimal number of clusters.

After that we construct the graph with directed weights on them where each vertex is a cluster center. Assume two vertices A,B The weights of $A->B$ represents the number of cells which its nearest neighbor is B and the second nearest neighbor is A while the weights of $B->A$ is calculated by the number of cells which its nearest cluster center is A and the second nearest cluster center is B. Then we use the weight to estimate the likelihood of cells transitioning between stage A and stage B.As a result, we build a complete graph(edges with weight 0 automatically) where. Then we trim the network so that the remaining tree(no circle) has the highest sum of weights(highest likelihood of state transitioning).
Finally we project every cell onto the paths, we assign cells to the cluster which center is nearest to them and project them to the edge between the nearest neighbor and second nearest neighbor. **Note that we requires some prior information about the cell collection time. What's more, we have to point out that in Mpath algorithm, we can only order the cells along several branches and observe the potential biological process along these path so it will be pretty difficult to measure the POS score and the Robustness. We measure the performance of Mpath by counting the gold standard genes among the top differential expressed genes.**
Here we implement the algorithm Mpath by ourselves and the R code for the whole algorithm and the downstream simulation can be seen in the supplementary materials.

4.3 Monocle

Monocle is a insightful method proposed by Trapnell lab which help us identify the branches based on constructing the minimal spanning tree based on all cells.
It can be easily demonstrated that Monocle is not as robust as TSCAN which is based on constructing minimal spanning tree between several clusters of cells.(Pretty sensitive to random noise which will be validated in the downstream analysis)



From the above plot we can see that both A and B are possible solution of minimal spanning tree however obviously solution B is more reasonable for it is more consistent to the true biological time.What's more, the cell ordering constructed based on every cell level is not as robust as that constructed by TSCAN given a small pertubation shown as the green node in solution B.
The Monocle algorithm first performs **ICA**(Independent Component Analysis) which aims to

extract the independent signals from the data to reduce the dimension to 2 or 3. However, ICA may be **pretty slow** in real implementation and a R package called **fastICA** demonstrate this. Then we construct the minimal spanning tree based on the reduced data points in two or three dimensions. **Although there is an alternative to reduce the time cost by reducing the number of genes using a differential expression test but it will introduce significant bias in terms of heterogeneities within subpopulations and deviation from the existing population groupings. So we won't use this trick.**
Here we replace ICA by tSNE to perform the dimension reduction the reason can be found in section(5.2.2) and compare it with that conducted by ICA.
Then we find the longest connected path in the tree after which we assign every cell to its nearest neighbor. We implement it by constructing the PQ tree and search the tree recursively along the main path.
Here we use the existing R package monocle(also needs some slight modifications in real implementation) to do this analysis.

4.4 Performance Measurement

Here we apply various techniques in measuring the efficiency of ordering and trajectory.
The first and foremost evaluation approach is to measure the to detect the differential expressed genes across the cell ordering constructed by pseudo-time. Then we can either **measure the numbers of gold standard genes detected before** in the top differential expressed genes along the pseudo-time axis(degree of deviation) or calculate **the mean rank of the gold standard genes** in the top differential expressed genes along the pseudo-time axis.
Alternatively we use the **POS score** to the efficiency of ordering. Assume the cells are collected in v time points $T_1, ..., T_v$. The POS score is given by $POS_\pi = \sum_i \sum_j g(\pi, i, j)$. If two cells are originally collected at the same time,$g(\pi, i, j) = 0$, otherwise, if the ith cell is collected from time point T_u and the jth cell is collected from time point T_v,then $g(\pi, i, j) = \frac{u-v}{D_\pi}$. The scaling factor D_π is chosen to restrict the POS score between -1 and 1. So $POS_\pi = 1$ means the order of cells produced by pseudo-time reconstruction perfectly matches the order of the cell collection time and $POS_\pi = -1$ means the cell ordering by the pseudo-time reconstruction is exactly the reverse order comparing to the original cell collection time. So in this way does POS score measure the efficiency of cell ordering.**We have to point out here that using POS score to evaluate of pseudo-time reconstruction is based on a fairly strong assumption that the cell collection time indeed reflect the true biological process.For example, we assume that the cell collected later is in a later stage of the true biological process(cell differentiation or cell apoptosis).** . We will test this on all methods.
Another insightful evaluation is the robustness of the cell ordering. We artificially add some perturbation onto the single cell RNA-seq dataset. We have two methods to perturb the dataset which denoted as cell-level and gene expression level. **For cell-level perturbation, we subsample 75%,90% and 95% of the whole sample of cells while for gene expression level perturbation, we retain all of the cells but now we add random simulated**

noise onto the original expression level of every cell. In [1],for each gene,they add the random noise as $(Y - E[Y]) * \zeta$ where ζ is chosen as 5%,10% and 25%. **However,here we use another perturbation method. We add normal distributed random noise $\epsilon_i \sim N(0, k\sigma_i^2)$ where σ_i^2 is the variance of the expression level and k can either be 5%,10% and 25%. To deal with the negative observation, we truncate it by 0. We use this because we use mclust to cluster the cells which is based on optimize of the mixture gaussian distribution so we add the normal noise.** We have to point out that in [1] they repeat each procedure 100 times but we only implement 15 times for each procedure considering the time limitation.
We have to point out that POS score and Robustness themselves are not sufficient to claim a good trajectory reconstruction, it should have some biological meaning.

5 Results

5.1 Performance Measure

Here totally we use 8 methods to do the trajectory inference and the measurements are shown in the below table.

Methods/Measure	markergeneexp	POS	Robustness	meanrank	numofmarkergene
t-SNE TSCAN	Y	Y	Y	Y	Y
Nonclu TSCAN	Y	Y	Y	Y	Y
Kmeans TSCAN	Y	Y	Y	Y	Y
Diff-map TSCAN	Y	Y	Y	Y	Y
ordinary TSCAN	Y	Y	Y	Y	Y
Monocle	Y	Y	N	Y	Y
tSNE-Monocle	Y	Y	N	Y	Y
Mpath	P	N	N	P	P

Y denotes yes and P denotes partly while N denotes not.
Here for t-SNE TSCAN we try three perplexities(30,40,50 and 30 is the default parameter in the R implementation) and choose the one with the highest POS score and the smallest mean rank of gold standard genes in the top differential expressed genes.
The choice of parameter in t-SNE is shown as below:

Method/Measure	POS score	meanrank of gold standard genes
t-SNE 30	0.5013164	532.55
t-SNE 40	0.546984	352.2
t-SNE 50	0.4664693	430.06

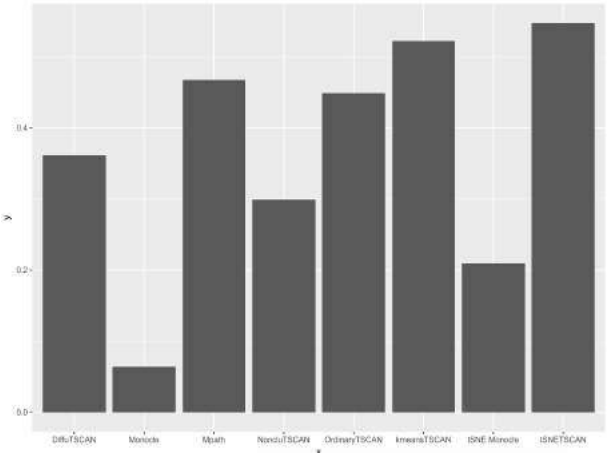
So we choose 40 as the perplexity parameter in t-SNE which indeed reaches a noise-information trade-off.

5.1.1 Marker Gene Expression

First we would like to check whether the expression trend of our artificially specified marker genes is consistent before and after cell-ordering and we only check the first 8 most differentially expressed genes. The graphs are shown as **Graph1-Graph7 in Appendix A**. We can see that in terms of consistency in trend with the original order, noncluTSCAN, kmeans TSCAN and Monocle obviously doesn't do well(the trend of the fitted values is greatly dispersed from the original one) and tSNE Monocle performs much better than the ordinary Monocle algorithm while still not that satisfactory. Diffusion map TSCAN performs decently in the first 2 marker gens while fail to maintain the trend in marker gene 3,5,6,7. Then tSNE TSCAN performs as well as the ordinary TSCAN which may not be treated as a large surprise.

5.1.2 POS Score

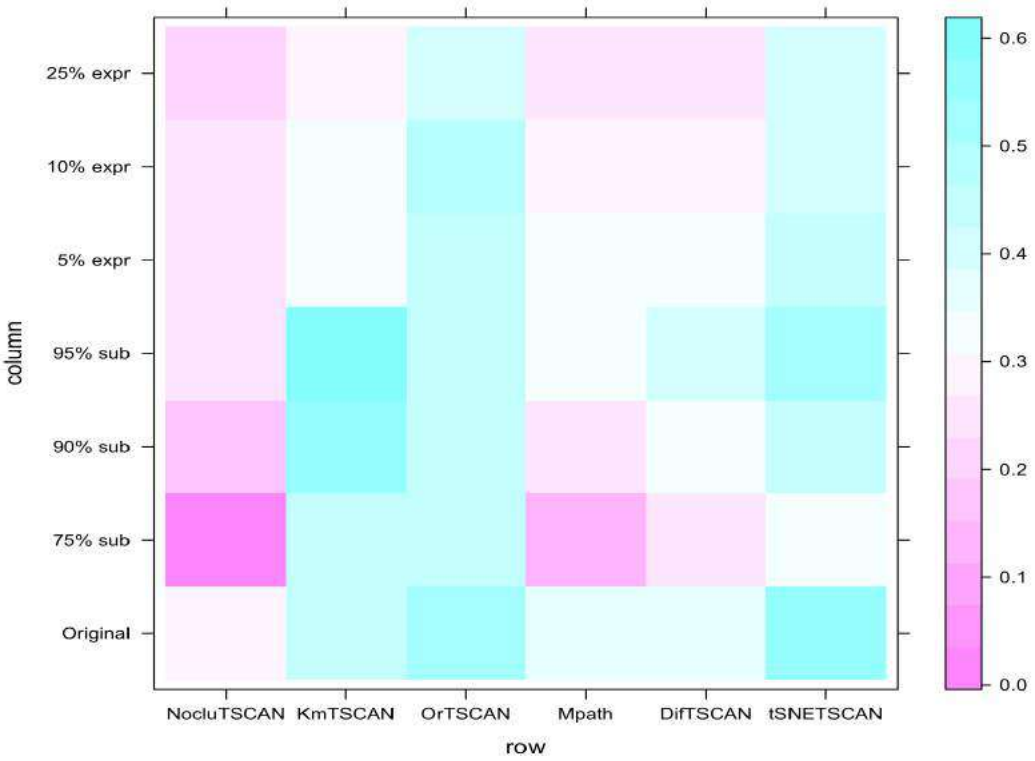
Here we compare the efficiency of ordering of the 6 trajectory inference techniques and the result is shown as below:



We can see that among the various modifications of TSCAN methods, the ordinary TSCAN(PCA+ clustering+mclust) performs the best and the diffusionmap and t-SNE also perform decently. However, Monocle is again. What's more, the ordinary Monocle doesn't perform well and even we replace ICA by tSNE in Monocle, although some improvement observed,the performance is still unsatisfactory.

5.1.3 Robustness

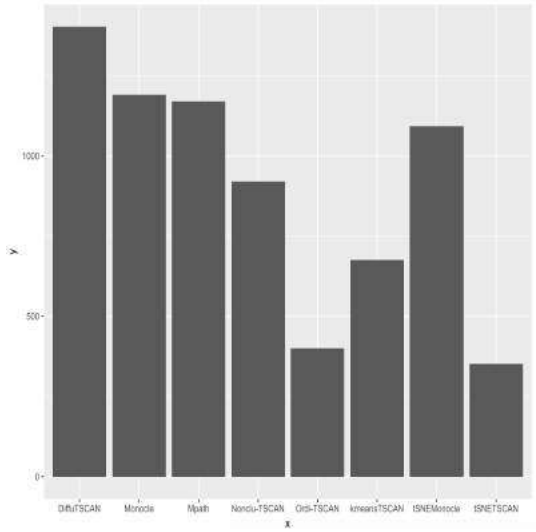
Here we compare **Ordinary TSCAN, Nonclustering TSCAN, Diffusionmap TSCAN, Kmeans TSCAN**. As we can see from the previous part, **Monocle and tSNE-Monocle** doesn't perform well in terms of POS score so we won't test its robustness **(Indeed it cost too much time to perform fastICA function in R for hundreds of times..)**. The result shown in levelplot is shown as below:



Here we can see that in terms of robustness, among the several modifications of TSCAN algorithm, **OrdinaryTSCAN and t-SNE TSCAN** does better and **kmeans-TSCAN** performs slightly worse than the previous two methods. Then we observe **Non clustering TSCAN** doesn't do well in this case. It's not surprising that non-clustering TSCAN not perform well due to the prevalent dropout effect in all kinds of biological experiments. For diffusion map TSCAN, it obviously down-performs compared with t-SNE. We guess it may arise from the choice of the ϵ and the number of diffusion components extracted and here we only use the threshold first developed in another field. Then to compare t-SNE TSCAN and the ordinary TSCAN, we conclude that the tSNE TSCAN is not that robustness comparing with the ordinary TSCAN maybe it's because we use the same perplexity for subsampling and random-noise perturbation and for each case, a specific optimum perplexity requires to be found artificially.
We have to point out a limitation because we only repeat 8-15 times(for t-SNE we only do 6 times each, for other methods we repeat for 15 times and get their average value).

5.1.4 Mean Rank of Gold Standard Genes

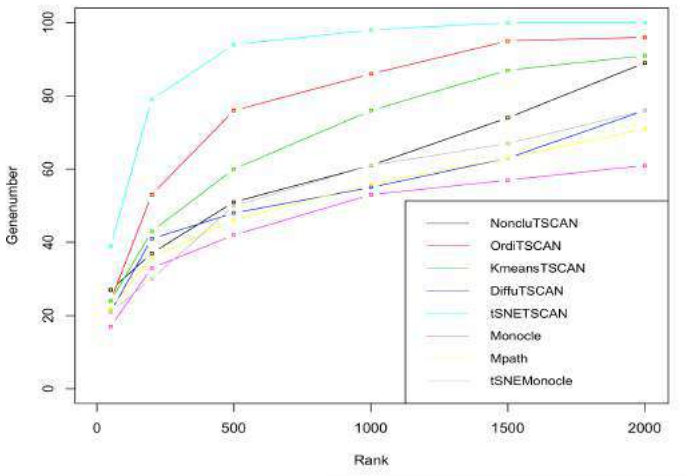
Here we compare the mean rank of the 100 marker genes across the seven trajectory inference techniques to compare their innate quality. The result is shown as below:



We can see that t-SNE are performs even a little bit better than ordinary TSCAN, however, the diffusionmap TSCAN doesn't perform well in terms of innate consistency of the cell ordering. We can also see that clustering similar expressed genes help improve the performance (ordinary TSCAN does better than non-clustering TSCAN) What's more,Monocle maintain too much noise on the individual cell level and fail to get a innate consistent ordering and for tSNE-Monocle, we still fail to observe a satisfactory performance. For Mpath, it may not fair to claim it doesn't perform well just based on one type of ordering for the branching nature of its reconstructed trajectory. We will clarify the result of Mpath and evaluate its performance in the trajectory visualization part.

5.1.5 Number of Gold Standard Genes in the top Differential Expressed Genes

Here we compare the number of gold standard genes(our pre-specified top 100 differential expressed genes as marker genes) in the top 50,100,500,1000,1500,2000 differential expressed genes to our reconstructed ordering to check the innate quality of the ordering.The result is shown as below:



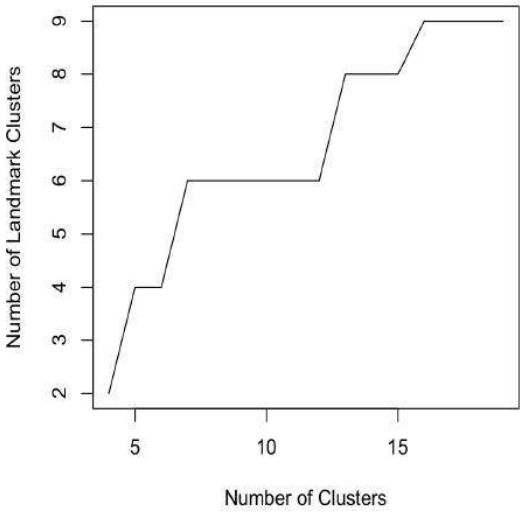
We can see that monocle doesn't perform well here because of the unidentifiable noise in constructing the trajectory based on each individual cell level and even after replacing ICA by tSNE, the performance still unsatisfactory. What's more we can see that nonclustering TSCAN and kmeans TSCAN performs slightly worse than ordinary TSCAN which is not surprising. Then what is amazing is that **tSNE-TSCAN** performs better than the ordinary TSCAN method with respect to this measure. What's more, Monocel doesn't do well and t-SNE Monocle does much better than the ordinary one although still not satisfactory. Diffusion Map TSCAN also doesn't do pretty well.

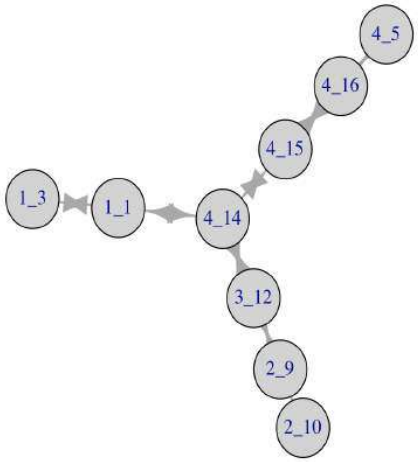
5.2 Visualization

Here we visualize the result of methods with good performance**(tSNE-TSCAN, Ordinary-TSCAN)** as well as Mpath.

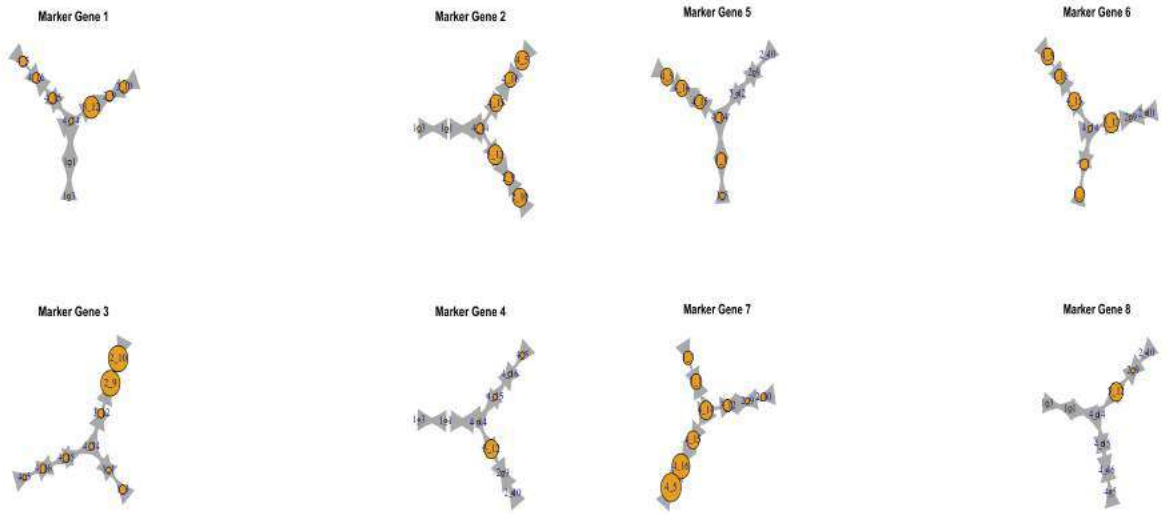
5.2.1 Mpath

Here in Mpath algorithm, most of the previous performance measure may not that adequate because in real implementation, we may need to determine the direct by ourselves because the nature of the reconstructed trajectory. Here after implementation we construct the branched path and the plot of number of landmark clusters versus the total number of clusters is shown as below:





Here the optimal number of clusters cut from the hierarchical clustering result is 16 and the branched trajectory indicate that a subpopulation of **NKT2** cell is important in controlling the further differentiation and we explore the trend of some of our pre-specified marker genes. Like **section 5.1.1** we visualize with respect to the first 8 differential expressed genes.



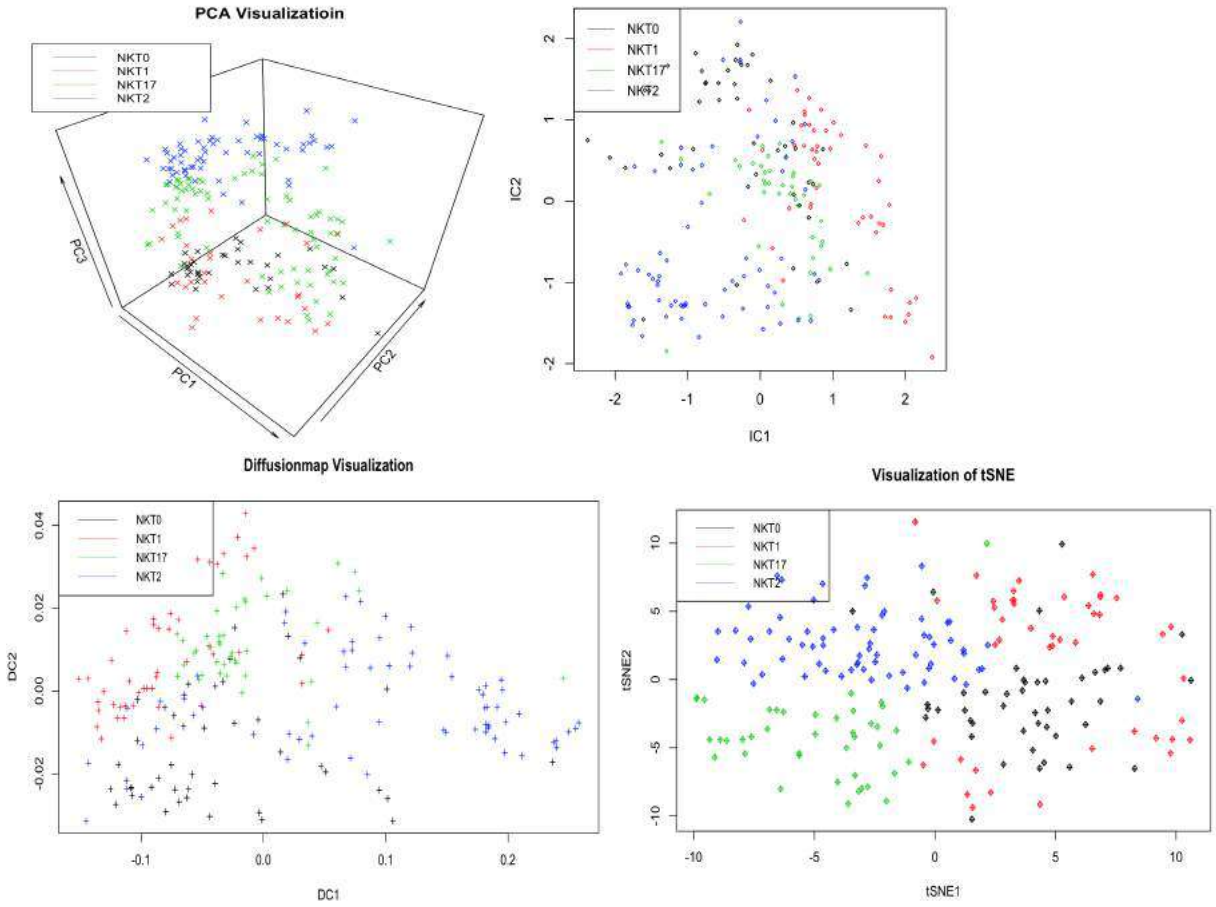
Here for marker gene 1**(ENSMUSG00000001020.8)**, the NKT 17 cell has a much higher expression profile than other types of cells; For marker gene 2**(ENSMUSG00000001025.8)**, it is higher expressed in NKT 1,2,17 cells than NKT0 cells; For marker gene 3**(ENSMUSG00000004612.9)**, it is up-expressed in NKT1 cell while for marker gene 4,8**(ENSMUSG00000014453.3, ENSMUSG00000023367.14)**, it is up-expressed in NKT 17 cell. Then for marker gene 5 and 7**(ENSMUSG00000 015314.10 and ENSMUSG00000023004.8)**, they are upregulated in NKT2 cells. For marker gene 6**(ENSMUSG00000021728.7)**, we can see that it is up-expressed in NKT 17 and NKT 2 cells.

For one branch 3-12,2-9,2-10, we can see that NKT 1 and NKT 17 cells have some transition relation. What's more, we are particularly interested to the landmark cluster **4-14**, a subpopulation of NKT 2 cells, which is the beginning of two branches and we may want to do more biological experiments to see it. NKT1 and NKT 17 may have some transitional relations which requires validation in the future. One more thing we want to point out is that in trimming the network, we ignore some of the minor interactions between different states which can also biological meaningful.

An limitation of Mpath is that when we see **Graph 8,Appendix A**,in trimming the constructed network based on the landmark clusters,we ignore the edge between cluster 3-12 and 4-15 which has 16 cells between them which may have a biological meaning. So we may loss information about the biological process we reconstruct.

5.2.2 Dimension Reduction

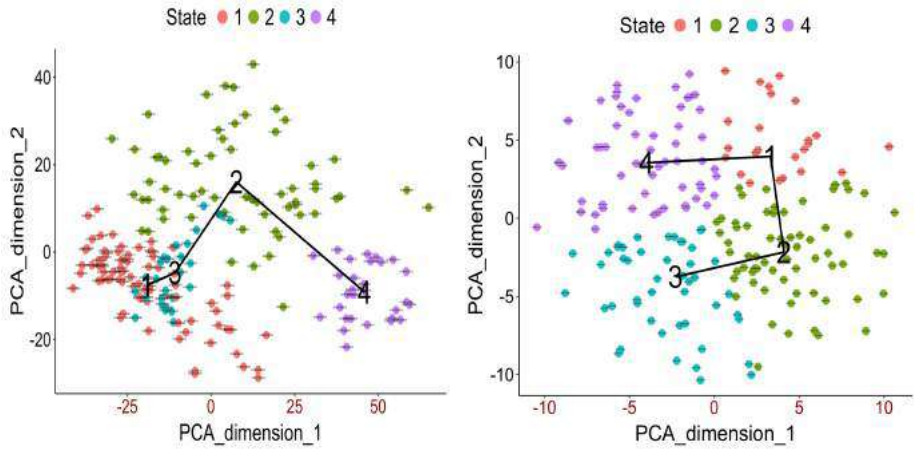
First we would like to see the result of various dimension reduction techniques(ICA,PCA,t-SNE,Diffusion Map) and they are shown as below. The right of the first row is ICA.



We can see that **PCA** and **t-SNE** performs much better than **Diffusionmap** and **ICA** here in separating different types of cells. We find that NKT 2 cell and NKT17 cell is identifiable from the plot. What's more, we find that t-SNE performs even slightly better than the ordinary PCA which is consistent to the previous result in various measurement of performance. To get a more satisfactory result of Diffusion, we may need to make some modifications to the algorithms like in [7]. Here ICA also doesn't perform well and what's more, this algorithm is too **SLOW**. **So we consider replacing ICA by t-SNE in monocle.** So here we can see that t-SNE and PCA may performs better than ICA in single cell sequencing dataset.
From the visualization we can see that the tSNE can separate NKT17,NKT2 and NKT0 while it may still fail to identify the cell population NKT 1(NKT 1 cell population pattern seems to be vague which we may be particularly interested in in the future research).

5.2.3 TSCAN

Here as demonstrated from **section 5.1**, ordinary TSCAN and t-SNE TSCAN performs among the best in various methods. So we will only visualize these two methods while the left is ordinary TSCAN and the right is tSNE-TSCAN:



Actually there are slight difference between these two trajectories and we find that indeed, the trajectory constructed by the tSNE-TSCAN is consistent to the biological nature that **the differentiation should start from NKT0 cell and may probably be a branching path** compared with that constructed via ordinary TSCAN. So here we can see that tSNE TSCAN pretty well to identify the biological path. For the ordinary TSCAN, we can identify the differentiation path $1- > 3- > 2$, however the path $2- > 4$ doesn't seem to have biological meaning(their may be interaction between the process from NKT0 to NKT1 and the process from NKT1 to NKT2). So here we can see that robustness itself doesn't ensure a good performance if from the minimal spanning tree we can't get a reasonable biological explanation we still can't claim a good result.

6 Conclusion and Discussion

We compare 8 methods of trajectory inference in our analysis and compare their performances in terms of various measurements.

First we can see that the fact that kmeans-TSCAN performs slightly worse than the ordinary TSCAN is not surprising because the kmeans can be seen as a restriction of the MoG clustering method. The euclidean distance used in calculating distance in kmeans is indeed the same as that in the log likelihood of the MoG optimized function.

What's more, we find that nonclustering TSCAN doesn't perform as well as the other methods. So clustering similarly expressed genes before analysis is really helpful to reduce the dropout effect.

Then we can see that diffusion map doesn't perform as well as the ordinary TSCAN. However, some research such as [7] has demonstrated that diffusion map may be a good choice in single cell sequencing data analysis because its ability in handling the density heterogeneity and decent robustness to noise. As [7] mentioned, they made various modifications on the original algorithm such as density normalization and consideration for missing value and uncertain observations. So we only use a developed threshold before to test and it's unfair to claim that diffusion map TSCAN is not good.

For Monocle, we can claim that it may not be a good choice compared with TSCAN because

it construct the MST based on every individual cell which may potentially introduce a lot of noise that can't be identified by researchers. So the ordering may be easily contaminated and become pretty unstable and vague. What's more, ICA is not a optimal choice of dimension reduction compared with the other dimension reduction techniques in the analysis because it does not scale well with an increasing number of genes[5]. When we replace ICA by t-SNE, Monocle's performance improve slightly while still unsatisfactory.
What's more, for t-SNE TSCAN and ordinary TSCAN, we can see that t-SNE TSCAN can perform slightly better than the ordinary TSCAN by a careful choice of the tuning parameter **perplexity** but it has lots of limitations in real application. The key is that there's no optimal technique to decide a perplexity(can be interpreted as the number of nearest neighbors).In real application, we may still prefer the ordinary TSCAN technique as we doesn't need to adjust the parameter artificially for each case, each loop. However, some research([8]) has demonstrated that t-SNE will be a good technique in single cell sequencing data analysis and it can also delicately handle density heterogeneity and random noise.
Another important issue we have to point out is that the mclust doesn't perform well here in the ordinary TSCAN and t-SNE TSCAN performs much better after mclust. We can see that the TSCAN has its own limitation in terms of difficultie in visualization.
What's more, Mpath is a really insightful method which is based on the idea that the likelihood of state is proportional to the number of cells between the two landmark clusters and actually we get some useful biological insight. What's more, the use of ward distance which depicts the cost of merging clusters in hierarchical clustering is also a highlight. For future extensions, we can extend the algorithm to identify the multi start developmental processes for more heterogeneous single-cell sequencing dataset.
We have found that there are probability that a subpopulation of NKT2 cells can potentially differentiate to NKT 17 cells and it may also control whether to differentiate(the beginning point of two branches).
We have to point out that our analysis actually have some limitations. First we artificially ignore the variability in isoform which can potentially makes somewhat a difference but in our dataset, each gene only contain one isoform.
The second limitation of our analysis lies in the simulation process, we only repeat 5 or 10 times in various measurements of robustness due to the time limitation so it may not be that accurate compared with that in [1].
The third limitation is that we have made a strong assumption prior to analysis that the cells are collected along the time sequence and the cells collected at the same time are only one type. For this dataset, however, it may not always the case because the differentiation mechanics of natural killer cells in mouse are yet to clarified.
Another limitation arises from the awful gene naming system which prevent us from getting the gene id from the gene symbol. Then we can't use any information from [2] which explored the various property of this dataset.So in our analysis we have to personally specify the marker genes by using the generalized additive model which is indeed pretty limited.

References

[1] Zhicheng Ji and Hongkai Ji. *TSCAN:Pseudo-time reconstruction and evaluation in single-cell RNA-seq analysis*. Nucleic Acids Research, May, 2016

[2] Isaac Engel, Gregory Seumois et.al *Innate-like functions of natural killer T cells subsets result from highly divergent gene programs.* Nat Immunol, June 2016.

[3] Cole Trapnell, Davide Cacchiarelli et.al *The dynamics and regulators of cell fate decisions are revealed by pseudotemporal ordering of single cells* Nature Biotechnology, 2014

[4] Laurens van der Maaten and Eric Postma. *Dimensionality Reduction:A Comparative Review* TiCC,Tilburg University,Oct 2009.

[5] Robrecht Cannoodt, Wouter Saelens and Yvan Saeys. *Computational methods for trajectory inference from single-cell transcriptomics* European Journal of Immunology, 2016.46:2496-2506

[6] Jinmiao Chen, Andreas Schlitzer, Svetoslav Chakarov, Florent Ginhoux and Micheal Poidinger. *Mpath maps multi-branching single-cell trajectories revealing progenitor cell progression* Nature Communications, Jun 2016.

[7] Laleh Haghverdi, Florian Buettner and Fabian J.Theis. *Diffusion maps for high-dimensional single-cell analysis of differentiation data Bioinformatics* 31(18),2015

[8] Laurens van der Maaten, Geoffrey Hinton. *Visualizing Data using t-SNE* Journal of Machine Learning Research 9(2008) 2579-2605

[9] Sam T.Roweis and Lawrence K.Saul. *Nonlinear Dimensionality Reduction by Locally Linear Embedding* SCIENCE,Dec 2000

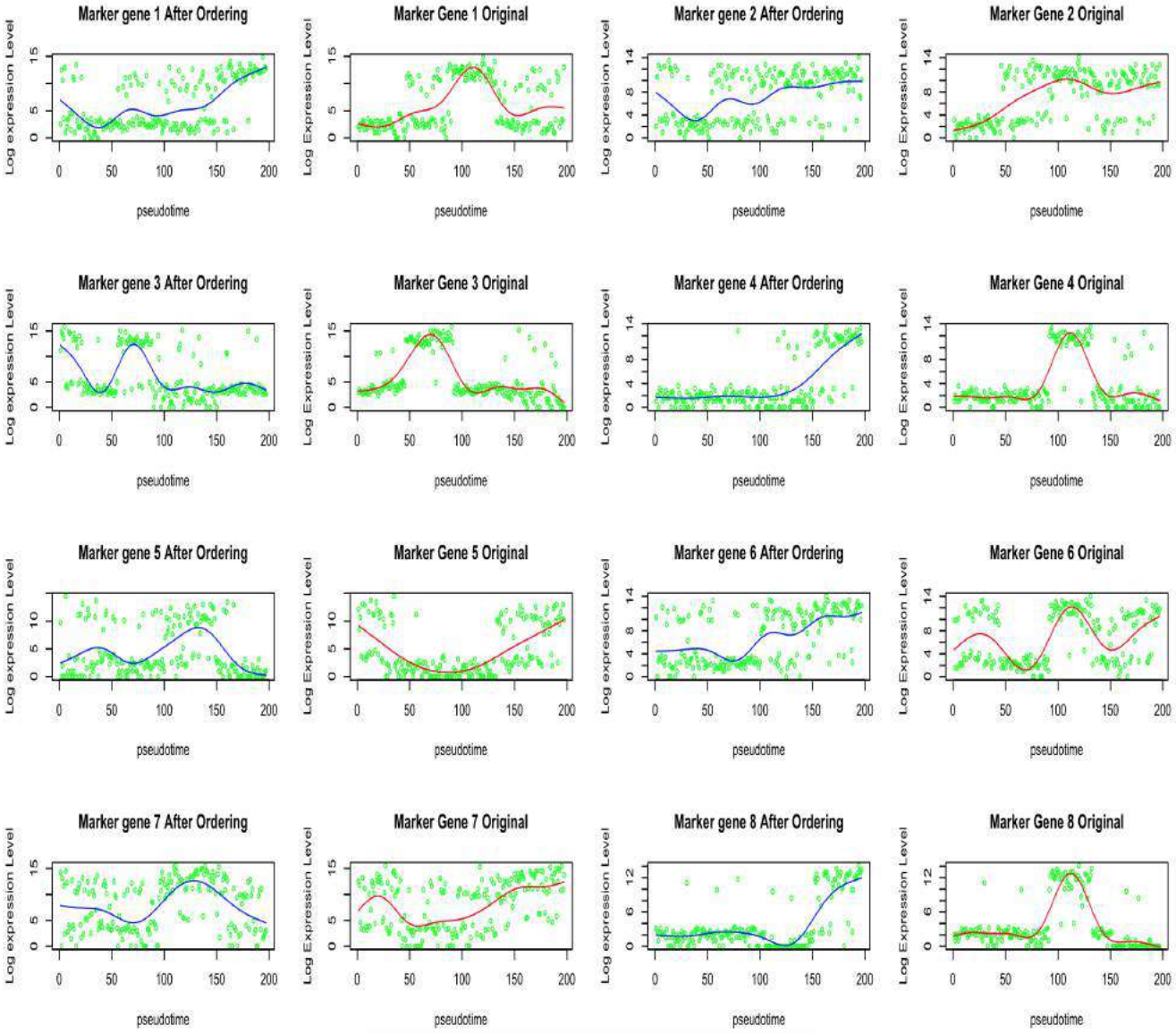
[10] Joshua B.Tenenbaum, Vin de Silva, John C.Langford *A Global Geometric Framework for Nonlinear Dimensionality Reduction* SCIENCE,Dec 2000

[11] John A.Lee,Michel Verleysen *Quality assessment of dimensionality reduction:Rank-based criteria*

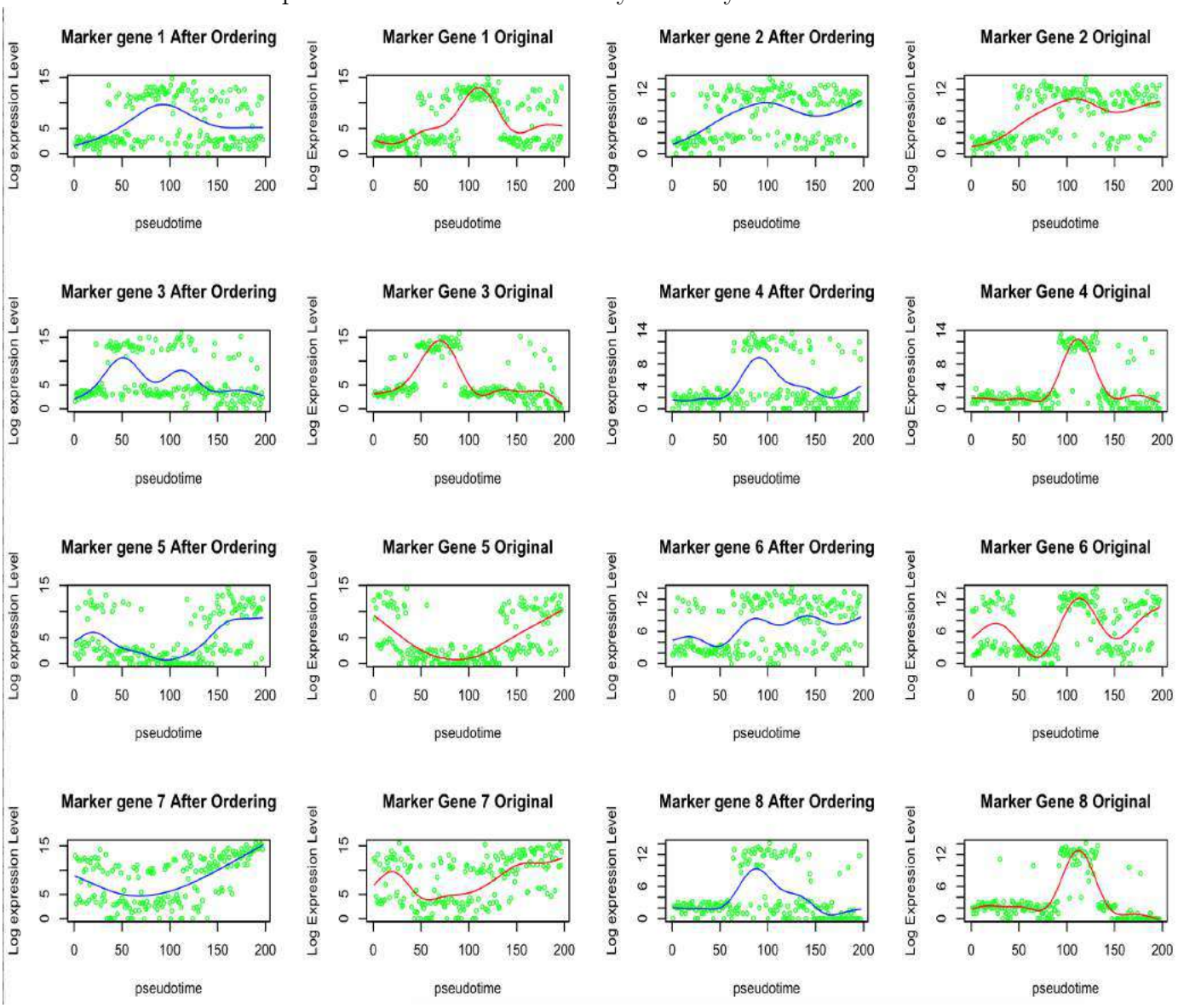
7 Appendix

7.1 Appendix A:Graphs

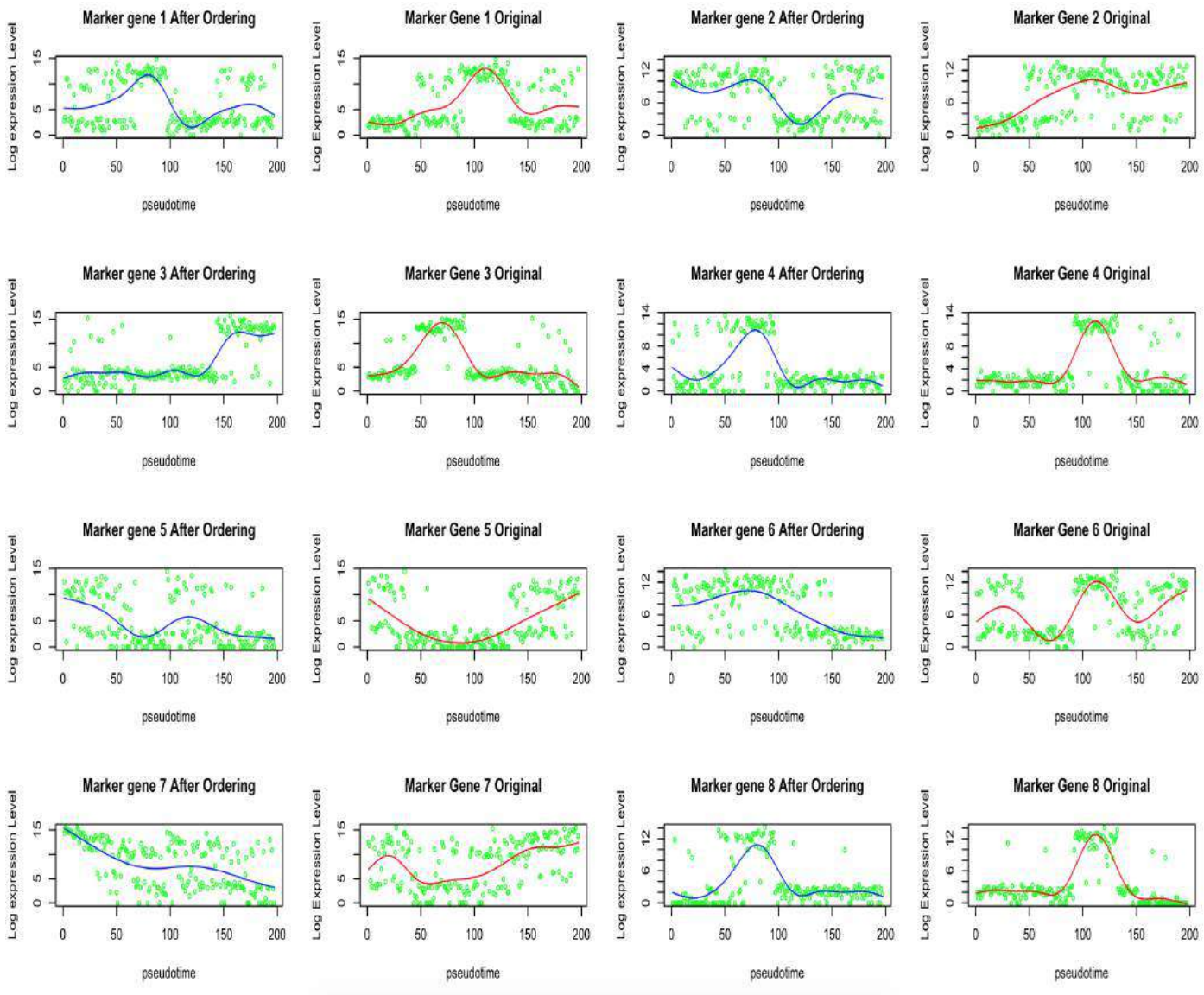
Graph1:Marker-Gene Consistency Nonclustering TSCAN:



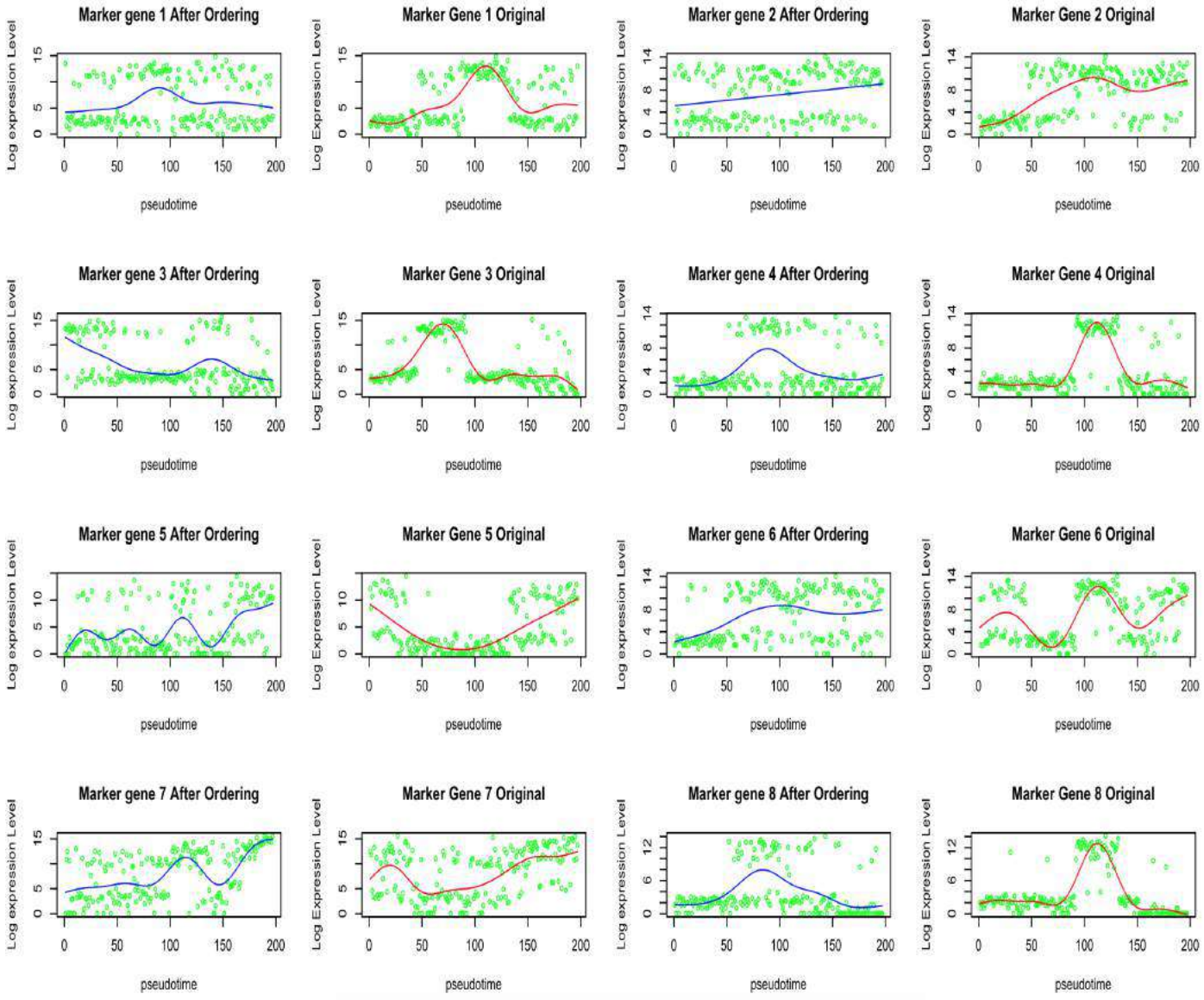
Graph2:Marker-Gene Consistency ordinary TSCAN:



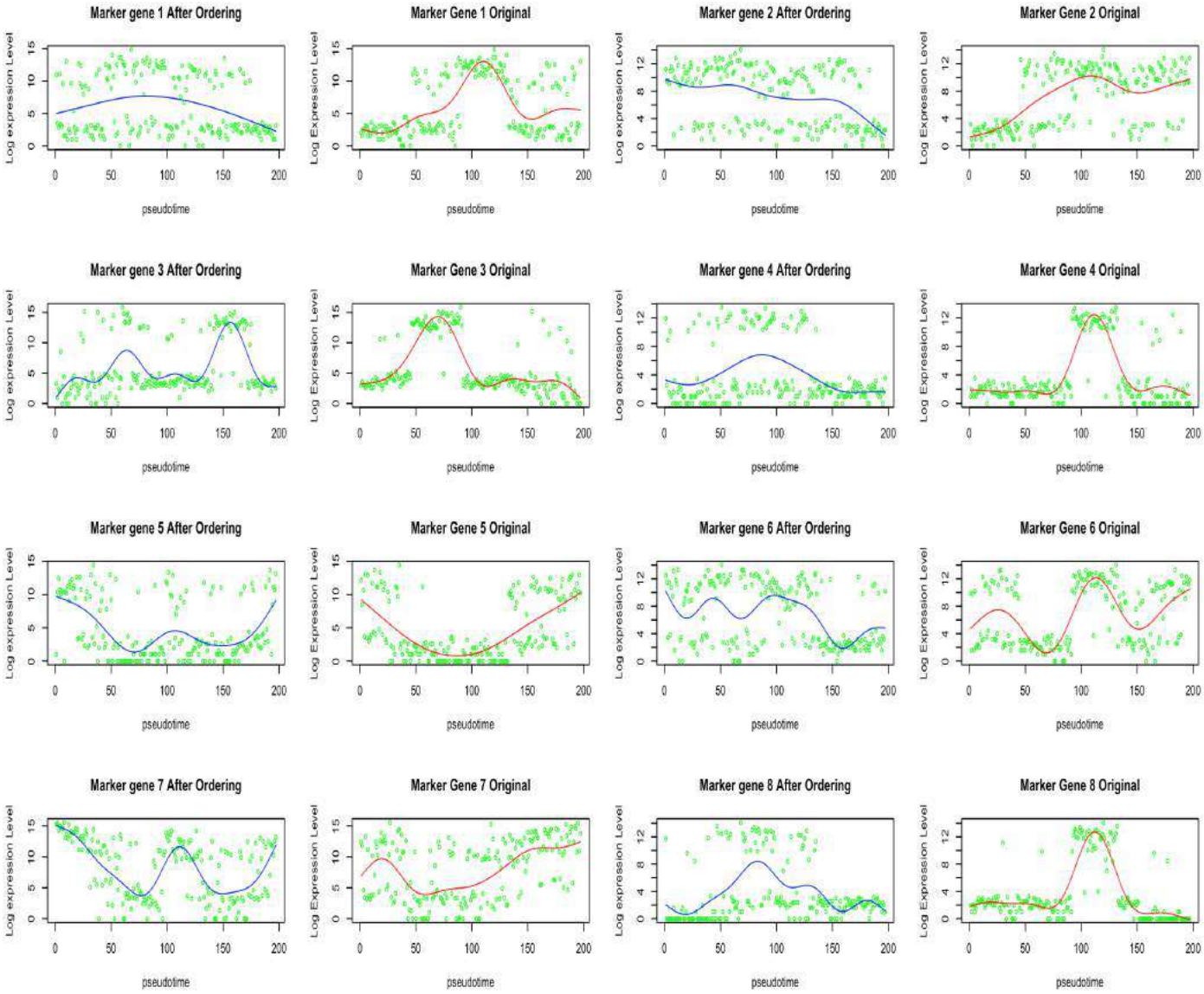
Graph3:Marker-Gene Consistency Diffusionmap TSCAN:



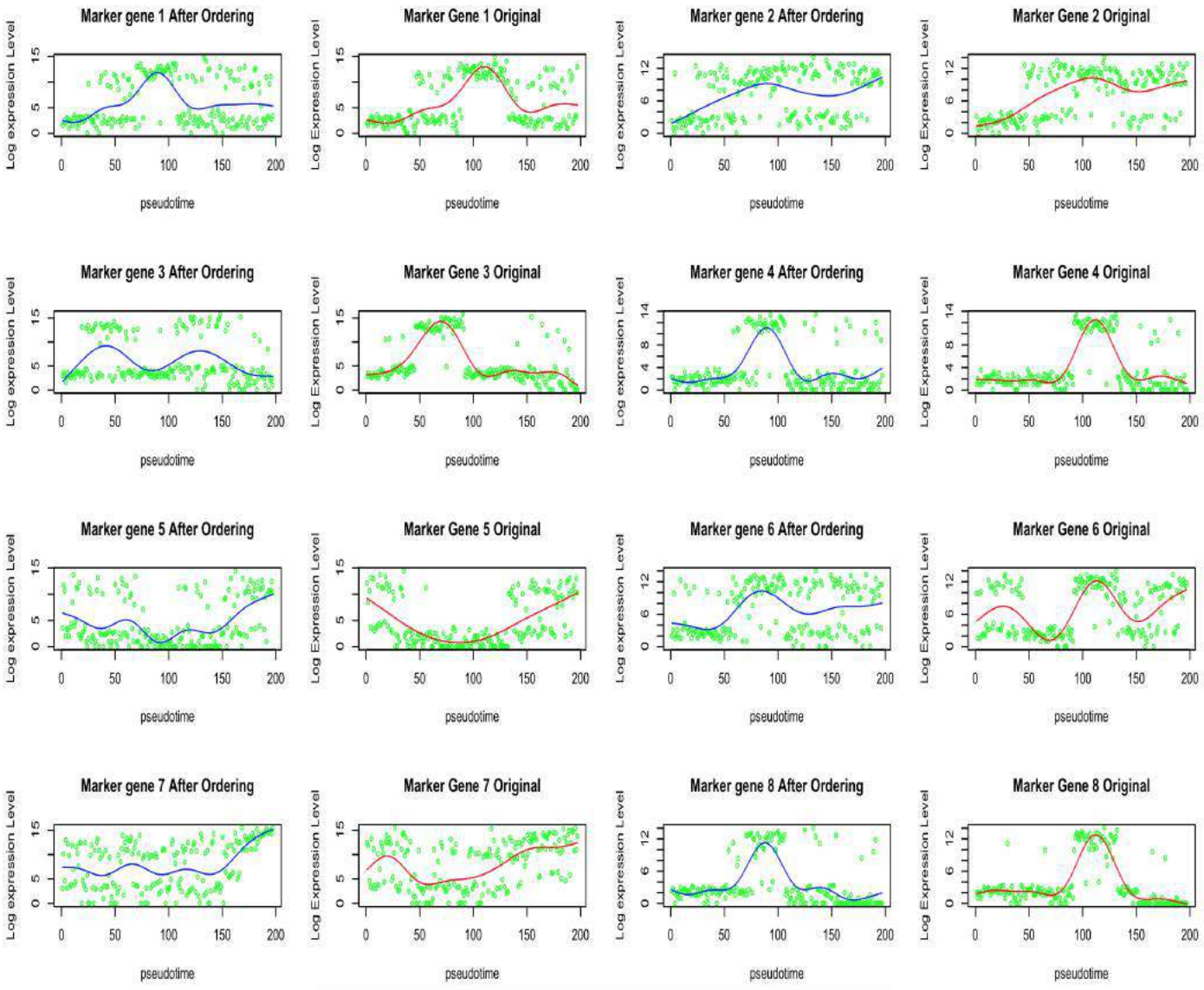
Graph4:Marker-Gene Consistency Kmeans TSCAN:



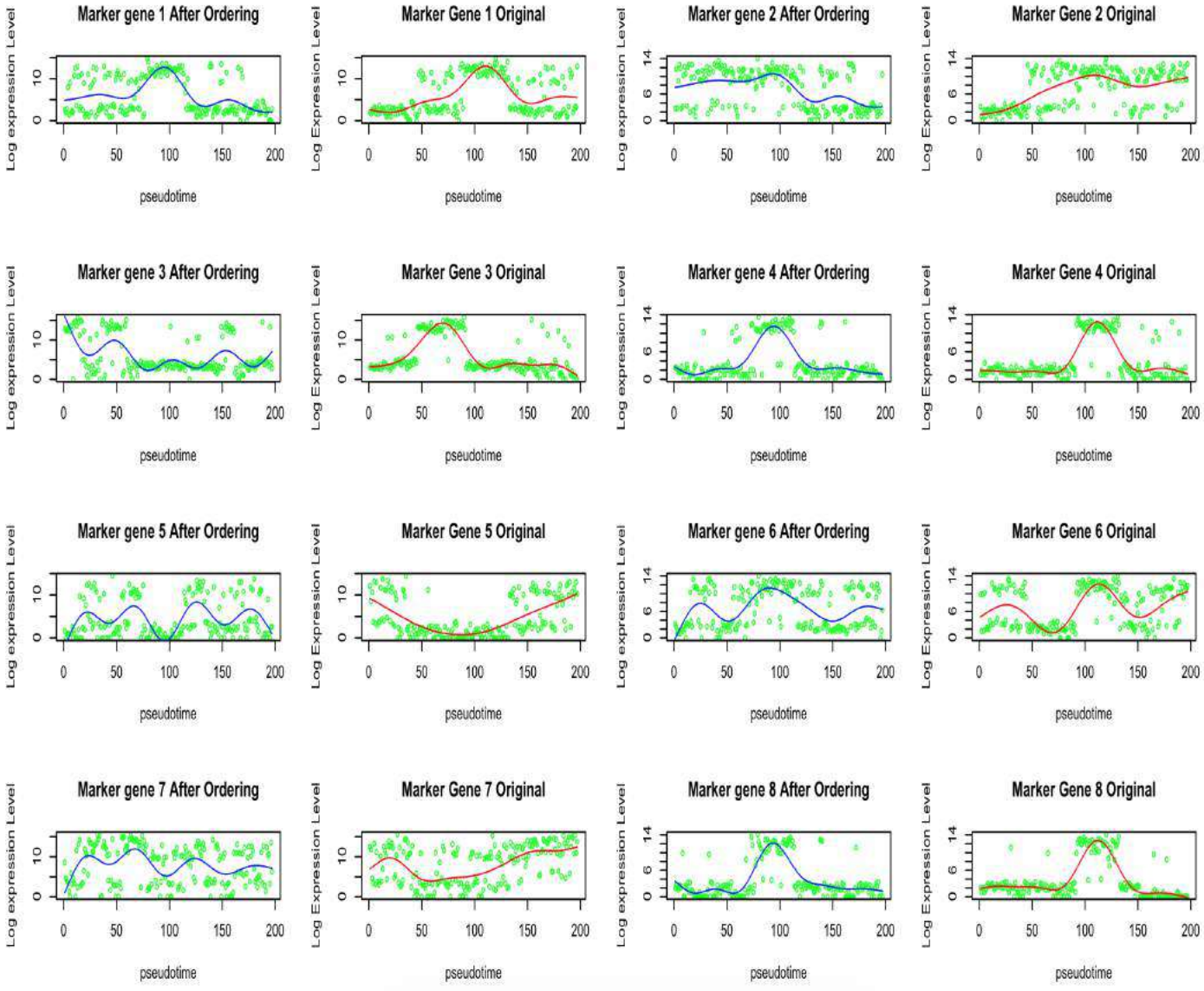
Graph5:Marker-Gene Consistency Monocle:



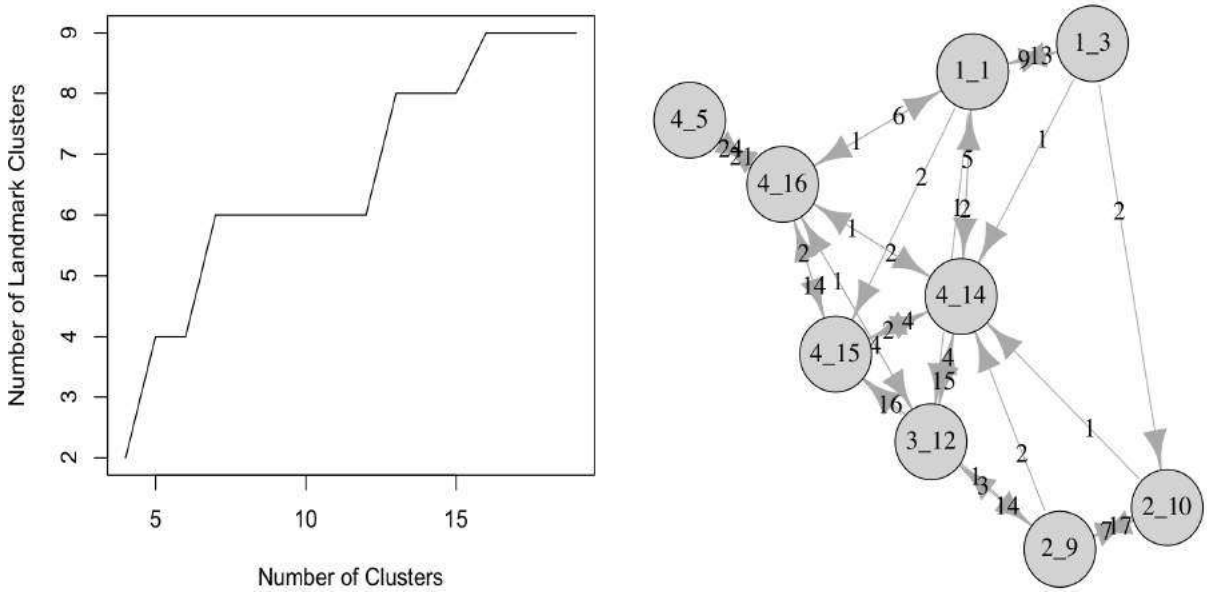
Graph6:Marker-Gene Consistency t-SNE TSCAN:



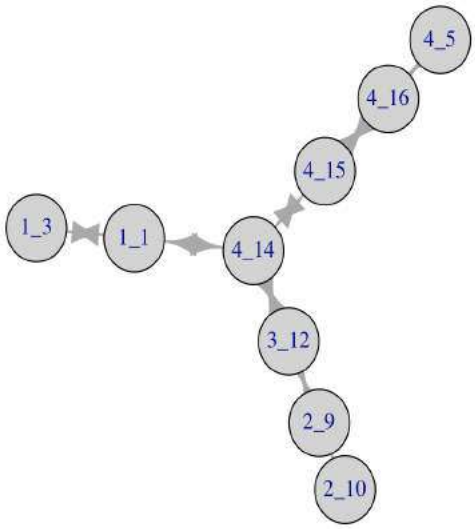
Graph7:Marker Gene Consistency t-SNE Monocle:



Graph8:Mpath clusters vs landmarker-clusters and original network construction



Graph9: Mpath Branching trajectory .



7.2 Appendix B:Artificially Specified Marker Genes

The first 20 differential expressed genes detected by us along the cell collection time are shown as below.

```
 [1] "ENSMUSG00000001020.8"  "ENSMUSG00000001025.8"  "ENSMUSG00000004612.9"
 [4] "ENSMUSG00000014453.3"  "ENSMUSG00000015314.10" "ENSMUSG00000021728.7"
 [7] "ENSMUSG00000023004.8"  "ENSMUSG00000023367.14" "ENSMUSG00000025163.6"
[10] "ENSMUSG00000026009.14" "ENSMUSG00000027863.8"  "ENSMUSG00000027985.14"
[13] "ENSMUSG00000028832.11" "ENSMUSG00000029810.15" "ENSMUSG00000030149.15"
[16] "ENSMUSG00000030165.16" "ENSMUSG00000030325.16" "ENSMUSG00000031239.5"
[19] "ENSMUSG00000031933.17" "ENSMUSG00000032026.6"
```