Distribution Agreement

In presenting this thesis or dissertation as a partial fulfillment of the requirements for an advanced degree from Emory University, I hereby grant to Emory University and its agents the non-exclusive license to archive, make accessible, and display my thesis or dissertation in whole or in part in all forms of media, now or hereafter known, including display on the world wide web. I understand that I may select some access restrictions as part of the online submission of this thesis or dissertation. I retain all ownership rights to the copyright of the thesis or dissertation. I also retain the right to use in future works (such as articles or books) all or part of this thesis or dissertation.

Signature:

_____ _____

Hanyi Zheng Date

Consensus clustering of subclone structure for multi-sample sequencing data

By

Hanyi Zheng

Master of Science in Public Health

Biostatistics and Bioinformatics Department

Hao Wu

Thesis Advisor

Wenyi Wang

Reader

Consensus clustering of subclone structure for multi-sample sequencing data

By

Hanyi Zheng

Bachelor of Science

Zhejiang University

2017

Thesis Committee Chair: Hao Wu

Reader: Wenyi Wang

An abstract of

A thesis submitted to the Faculty of the

Rollins School of Public Health of Emory University

in partial fulfillment of the requirements for the degree of

Master of Science in Public Health

in Biostatistics and Bioinformatics

2020

Abstract

Consensus clustering of subclone structure for multi-sample sequencing data

By Hanyi Zheng

Background: The tumor heterogeneity describes the heterogeneity in morphology and phenotype in tumor cells and is related to cancer therapeutics. The accurate assessment of tumor heterogeneity is an essential step for understanding how a tumor evolves and the determination of tumor subpopulation is a challenge. In this work, we present a combinatorial algorithm that can exploit samples from multiple time points over the development of the tumor within a single patient to determine the subclone cluster.

Methods: We firstly estimated CCF (cancer cell fraction) and cluster information for each time point by implementing a hierarchical Bayes statistical model and MCMC process (Pyclone). After the imputation of co-clustering matrix, we used non-negative sparse coding to determine consensus cluster across all time points to avoid trivial cluster. Finally, we made adjustment to the covariance matrix and used BIC to decide the optimal number of clusters.

Results: We use weighted CCF as the CCF for the cluster and observe the trend of each cluster. For PR42, k=5 is the optimal cluster number and every cluster has a unique trend. For PR44, the whole trend for mutations goes down then goes up, which implies that the therapy does well at first but then lost its effect. For PR240 we found that the therapy is ineffective for this patient at all since the trend of CCF for all clusters across all K increases with time.

Conclusions: This study presents a combinatorial algorithm to decide the subclone cluster of multi-timepoints tumor gene data. The model works well when data does not have a high percentage of missing mutations. Besides, the purity of the sample and the trivial clusters generated by Pyclone can affect the results. We also found that missing mutations directly impact the co-clustering matrix and covariance matrix in the BIC step.

Consensus clustering of subclone structure for multi-sample sequencing data

By

Hanyi Zheng

Bachelor of Science

Zhejiang University

2017

Thesis Committee Chair: Hao Wu

Reader: Wenyi Wang

An abstract of

A thesis submitted to the Faculty of the

Rollins School of Public Health of Emory University

in partial fulfillment of the requirements for the degree of

Master of Science in Public Health

in Biostatistics and Bioinformatics

2020

Table of Contents

1.Introduction

Cancer is a dynamic disease that results from gene mutations. These mutations are caused by environmental factors like tobacco and accumulate during an individual's lifetime[1]. The accumulation of mutations leads to heterogeneity among tumor cells. The tumor heterogeneity describes the heterogeneity in morphology and phenotype in tumor cells, such as variations in cellular morphology, gene expression, copy number, and other aberrations[2]. Tumor heterogeneity is related to cancer therapeutics, and causes difficulties in developing effective therapies[3, 4].

The accurate assessment of tumor heterogeneity is an essential step for understanding how a tumor evolves[5]. The clonal evolution model of the tumor was first postulated by Nowell in the 1970s. It points out that all tumor cells descend from a single mutated cell, accumulating other mutations in the progress. Some cells are more competitive than other cells for clonal expansion and therefore, can proliferate quicker to become dominant subclones[6]. This process is called "clonal expansion". Tumor evolves into multiple subpopulations during the subsequent clonal expansion, with each subpopulation harboring specific mutations.

Determine tumor subpopulation is a challenge. Nowadays, sequencing technology enables performing large-scale molecular profiling of tumors to comprehend cancer development and determine disease progression[1]. Theoretically, single-cell sequencing gives access to a more fine-scaled level of tumor heterogeneity than bulk sequencing data, however, this method is not widely used due to high cost[7]. Inferring the tumor's subclone composition by analyzing

massively parallel DNA sequencing data is the most popular way[8]. In this way, Cancer Cell Fraction (CCF) of each subclone cluster can be calculated from read depth and variat information.

Single-nucleotide variants (SNVs) and copy number aberrations (CNAs) are widely used variat types for the determination of tumor subpopulation. Recent studies reconstructing the subclone architecture of tumor focusing on either SNVs or CNAs or both. SNVs based method firstly estimates CCF from variant allele frequency (VAF) of SNPs, then clusters SNVs with similar CCF[9]. CNA-based get allelic imbalances from B-allele frequency (BAF).

Computational tools are developed according to different data types. Clonality inference in tumors using phylogeny (CITUP) is a typical SNVs based method. It is based on an exact Quadratic Integer Programming (QIP) formulation[10], with a model assuming that the copy number is two. Many methods such as PhyloSub and PyClone add CNAs into the model to relax this assumption. Monte Carlo Markov chain (MCMC) sampling is widely used in this category of methods[11]. PhyloSub is based on Bayesian inference and MCMC sampling paradigm to infer a distribution over all possible phylogenies. PyClone is another clonal inference approach that is also based on MCMC and Bayes statistics, but this method generates many trivial clusters (cluster that only contain 1 mutation)[12].

In this article, we present a combinatorial algorithm that can exploit samples from multiple time points over the development of the tumor within a single patient to determine the subclone cluster. Our framework also includes MCMC to estimate CCF and cluster information for each time point, but unlike the previous approaches mentioned earlier, non-negative sparse coding and BIC test are

used to determine consensus cluster across all time points to avoid trivial cluster. In the event that one mutation is often missing at more than one timepoints, our method considers the linear relationship between each time point and takes low read depth data into consideration, treat those mutations differently according to the different scenarios. The method performs well for the sequencing data from the biopsy sample, which has lower purity than the tumor sample.

2 Methods

2.1 Data Collection and Cleaning

Prostate cancer is the most commonly diagnosed non-cutaneous cancer and the second leading cause of cancer death is in the United State[13]. The data are plasma genome sequencing data of prostate cancer from The University of Texas MD Anderson Cancer Center. The dataset includes 10 patients with prostate cancer, among those patients, 1 have data recorded at 5 time points, 2 have data at 4 time points, 1 have 3 time points' data and 6 of them have recorded at 2 time points. We focus on 4 patients that have data collected at more than 2 time points since this project aims to determine multiple time subclone clustering.

For each mutation, the related information was recorded: Mutation position, gene name, read depth for reference, read depth for variation and copy number. Some mutations only had read depth = 1 or 2 at some time points, and they were too small to be considered as reliable. We treated the mutation that with low read depth as missing:

$$x = \begin{cases} 0, & x < 3 \\ x, & x \geq 3 \end{cases} \quad , \textit{x: read depth}$$

And for mutations that only occurred at first time point but missing in all time points followed, we treated them as outliers and delete them.

2.2 Pyclone

Pyclone is a statistical method for inferring cancer clonal population structures. Based on reading depth (reference allele and variation allele) and copy number (normal, minor and major copy number) information, Pyclone can estimate CCF of each mutation by implementing a hierarchical Bayes statistical model, also clustering structure can be imputed at the same time. The framework of Pyclone includes four parts:

(1) Beta-binomial emission densities.

Pyclone uses beta-binomial emission densities instead of binomial models that previous methods use. It is shown that the beta-binomial emission densities is more effective and accurately because it models data sets with more variance in allelic prevalence measurements. The clustering result by Pyclone is more credible according to the account of higher accuracy of allele prevalence variance modeling. The overdispersion problem can be addressed by the beta-binomial model.

(2) Flexible prior.

Sometimes the available information is not enough and causes low confidence in reconstruction. Pyclone takes this situation into consideration and uses flexible prior probability to estimate possible mutant genotypes, reflecting how the allele prevalence measure is linked to zygosity and copy number variants.

(3) Bayesian nonparametric clustering.

Pyclone uses Bayesian nonparametric clustering to find the mutation groups and group numbers at the same time. This avoids determining the number of groups a priori and allows estimation of cell prevalence to reflect the uncertainty of this parameter[13].

(4) Section sequencing.

Pyclone can perform joint analysis on multiple samples at the same time to take advantage of the scenario where cloned populations are shared between samples.

Based on the framework and MCMC process, Pyclone outputs the clustering and CCF (or cellular frequency) with high confidence but remains shortcomings since many trivial clusters are generated. For example, for a sample with 76 mutations, Pyclone divides those mutations into 47 clusters, with 44 clusters only have 1 mutation.

We use the CCF and cluster information that Pyclone inferred for further analysis. However some mutations may miss at some time points, this leads to the mutation number differ at each time point for the same patient and causes difficulties in building the consensus co-clustering matrix for deciding the clustering structure.

2.3 Co-Clustering matrix imputation

In order to fix the problem of inconsistent mutation data at different time point, we find ways to deal with the imputation work. Based on the result from Pyclone, we build X^TX co-clustering matrix at each time point, where X is the mutation number across all time points. The below is an example of the co-clustering matrix.

	mtt_1	**mtt_2**	**mtt_3**	**mtt_4**	**mtt_5**	**mtt_6**
mtt_1	1	1	0.5	0.7889	0	0
mtt_2	1	1	0	0	0	0
mtt_3	0.5	0	1	0.5	0	0
mtt_4	0.7889	0	0.5	1	0	0
mtt_5	0	0	0	0	1	1
mtt_6	0	0	0	0	1	1
mtt_5	0	0	0	0	1	1
mtt_6	0	0	0	0	1	1

For each time point:

(1) If two mutations both exist at this time point: 1 means two mutations are in the same cluster, like mutation_1 and mutation_2 in the example, and 0 means they are in different clusters such mutation_1 and mutation_5.

(2) If at least one of the pair is missing, which means we don't have the clustering information between these two mutations, we would refer to the matrix at the nearest neighboring time that both mutations exist. There are two cases here:

a. This time point is the edge; For example, if this is the first time point, then we would use the value at the second time point in the same cell of matrix (if the cell of second time point is still missing then use the third).

b. This time point is in the middle, the mean of the same cells for neighboring matrix would be taken. Such as mutation_1 and mutation_3, one of them is missing at this time point, but they are in the same cluster at one of the neighboring time points and not cluster together at the other.

(3) If two mutations never exist together across all time points. In the situation that always at least one of the mutations is missing at all time points, which means we cannot take advantage of the neighboring matrix. The off-diagnose mean of the matrix would be used, like what happened to mutation_1 and mutation_4 in the example.

We have K equal rank matrixes after all imputation work are finished, then non-negative sparse coding is used to determine the consensus clustering structure.

2.4 Non-negative sparse coding

Non-negative sparse coding is a method to decompose multivariate data into non-negative sparse components. It is a combination of sparse coding and non-negative matrix factorization[14].

2.5 Deciding optimal number of clusters

Bayesian information criterion (BIC) is originally used in model selection, the smaller the BIC the better the model. In this project, we use BIC in deciding the optimal cluster number.

Assume in each cluster, the CCF of mutation follows normal distribution:

$$\mathbf{x_{ik}} \overset{iid}{\sim} N(\mu_{\mathbf{k}}, \mathbf{\Sigma_k})$$

$$\mu_{\mathbf{k}} = (\mu_{k.1}, \mu_{k.2} \ldots \mu_{k.t})$$

$$\mathbf{\Sigma_k} = \begin{bmatrix} \sigma_{k.1}^2 & \rho_{12}\sigma_{k.1}\sigma_{k.2} & \cdots & \rho_{1k}\sigma_{k.1}\sigma_{k.t} \\ \rho_{12}\sigma_{k.1}\sigma_{k.2} & \sigma_{k.2}^2 & \cdots & \rho_{2k}\sigma_{k.2}\sigma_{k.t} \\ \cdots & \cdots & \cdots & \cdots \\ \rho_{1k}\sigma_{k.1}\sigma_{k.t} & \rho_{2k}\sigma_{k.2}\sigma_{k.t} & \cdots & \sigma_{k.t}^2 \end{bmatrix}$$

Where i is mutation, k is cluster and t is time point. x_{ik} denotes to the CCF of i^{th} mutation in cluster k. μ_k means the mean of CCF for cluster k and $\Sigma_k$is the covariance matrix for cluster k However, for some clusters, all the mutations inside are missing at some timepoints and causes $\sigma_{kt}^2 = 0$ for cluster k and time t. In this regard, Σ_i becomes rank-deficient and makes the trouble in subsequent calculations for BIC. Therefore, we find a way to fix this problem by adjust the covariance matrix.

2.5.1 Covariance matrix adjustment

For clusters that all mutations inside are missing in at least one time point, which means for cluster k at time t, μ_{kt}=0 and $\sigma_{kt}^2 = 0$, we use the variance of the neighboring time point which has the most represented mutations for that cluster as the reference. The way for determining the reference time point t^* for variance is shown below:

(1) If $t = 1$ or $t = T$, then the reference time point should be the nearest time point that has p_{kt^*}>0.5, where T is the number of time point for data, p_{kt^*} is the frequency of the existing mutations in cluster k at time t, $p_{kt} = n_{exist}/n_k$.

(2) If $1 < t < T$, same as the previous case, reference time point is the nearest time point that has p_{kt^*}>0.5. However, if $p_{k(t+1)} > 0.5$ and $p_{k(t-1)} > 0.5$, we are going to compare $p_{k(t+1)}$ $p_{k(t-1)}$ and use the time that has larger p as the reference.

(3) If for all time points the $p_{kt} \leq 0.5$, then the t^* should be the time point that has the largest p_{kt}

Once the reference time point t^*, the modification of the covariance matrix would be straight forward. $\sigma_{kt}^2 = \sigma_{kt^*}^2$ and $\sigma_{ktt^*} = 0.95\ \sigma_{kt^*}$, $\sigma_{ktt_1} = \sigma_{kt^*t_1}$, where t_1 is the time point other than t and t^*. For example, below is a covariance matrix where $t = Time3$ and $t^* = Time1$

	Time1	Time2	Time3
Time1	0.014688	0.001307	0.013954
Time2	0.001307	0.003847	0.001307
Time3	0.013954	0.001307	0.014688

2.5.2 likelihood function

Based on this, the likelihood function for cluster k is:

$$N(\mathbf{x_i}|\mu_\mathbf{k}, \mathbf{\Sigma_k}) = \frac{1}{\sqrt{(2\pi)^T|\mathbf{\Sigma_k}|}} \times exp(-\frac{(\mu_\mathbf{k} - \mathbf{x_i})^T \mathbf{\Sigma_k}^{-1}(\mu_\mathbf{k} - \mathbf{x_i})}{2})$$

$$P_i(\mathbf{x_i}, Z_i|\mu_\mathbf{k}, \mathbf{\Sigma_k}) = \sum_{k=1}^{K} Z_i \times N(\mathbf{x_i}|\mu_\mathbf{k}, \mathbf{\Sigma_k})$$

$$\text{Where } z_i = \begin{cases} 1, & \text{assigned cluster} \\ 0, & \text{otherwise} \end{cases}$$

Then the log-likelihood can be calculated as below:

$$ll = log(\prod_{i=1}^{n} P_i) = \sum_{i=1}^{n} log(P_i)$$

BIC for cluster number = K is:

$$BIC = -2 \times ll + T(K-1)log(n)$$

Where K: cluster number; T: number of time points for the sample; n: mutation number for the sample. $T(K-1)log(n)$ is the penalty for BIC

By comparing BIC for each K, the optimal number of clusters can be decided

3. Result

3.1 Data summary

For 4 samples that have more than 2 time points: PR42, PR44, PR240 and PR246, after the data cleaning step in 2.1, the data summary is shown below:

Table 1 Data summary

Sample	Type	Time point	Purity	Mutation num	Total mutation num
PR42	Castration resistant	Time1 (therapy)	0.185	43	86
		Time2	0.737	76	
		Time3 (12 months after T1)	0.705	83	
PR44	Castration resistant	Time1 (therapy)	0.338	111	131
		Time2 (20 days after T1)	0.131	85	
		Time3 (13 months after T1)	0.349	125	
		Time4 (14 months after T1)	0.236	129	
PR240	Castration sensitive	Time1 (therapy)	0.461	7	83
		Time2 (3 weeks after T1)	0.091	7	
		Time3 (6 weeks after T1)	0.039	19	
		Time4 (18 weeks after T1)	0.058	41	
	Castration resistant	Time5 (24 weeks after T1)	0.139	72	
PR246	Castration sensitive	Time1 (therapy)	0.049	73	142
		Time2 (3 weeks after T1)	0.051	80	
		Time3 (12 weeks after T1)	0.147	129	
	Castration resistant	Time4 (24 weeks after T1)	0.247	141	

There are two sample types: castration sensitive and castration resistant. Castration sensitive/ resistant is a sign of treatment effectiveness. PR42 has 3 time points, PR44 and PR246 have 4 and PR240 has 5 time points. Time intervals between each time point are described in brackets. Purity denotes to the cancer cell fraction of the plasma sample. Mutation num is the number of mutations that exist at that time point and the total mutation num is the total mutation number of the sample.

For example, PR42 includes 86 mutations, but at time 1 half of them are missing, and 10 of these 86 mutations are missing at time2.

From the summary table above, for the same patient the purity can be much different between each time point. Take PR240 as an example, the minimum purity is 0.039 and the maximum is 0.461, also the missing mutation numbers across time points are unbalancing for PR240.

3.2 Result of Pyclone

For 4 samples (PR42, PR44, PR240, PR246), the output of Pyclone of each time point are all have many trivial clusters (cluster size >1), in fact, most of clusters are trivial clusters (Table 2).

Table 2 Pyclone output summary

Timepoint		K	Number of trivial clusters	Nontrivial cluster size	Purity	Number of mutations	Number of missing
pr42	T1	21	17	11; 8; 5; 2	0.186	43	43
	T2	47	44	16; 10; 6	0.737	76	10
	T3	30	27	29; 21; 6	0.705	83	3
pr44	T1	7	6	105	0.13	111	20
	T2	7	4	77; 2; 2	0.35	85	46
	T3	4	1	115; 5; 4	0.24	125	6
	T4	16	15	114	0.32	129	2
pr240	T1	6	5	2	0.461	7	76
	T2	5	4	3	0.091	7	76
	T3	5	4	15	0.039	19	64
	T4	4	3	38	0.058	41	42
	T5	2	1	71	0.139	72	11
pr246	T1	9	8	65	0.049	73	69
	T2	15	14	67	0.051	80	62
	T3	24	22	105; 2	0.147	129	13
	T4	5	4	137	0.247	141	1

For each sample in every time point, K is the number of clusters. At time 1 PR42 has 21 clusters but 17 of them only have 1 mutation inside, and 4 nontrivial clusters include 11/8/5/2 mutations respectively. Trivial cluster indicates that the mutation in that cluster has CCF that much different from others. However, if most of the clusters are trivial cluster then little information can be gathered from them since if every mutation assigned to different then it doesn't imply anything.

3.3 Consensus cluster

3.3.1 weighted CCF

We use weighted CCF as the CCF for the cluster, where the CCF weighted by the total read depth of the mutation is used as the CCF of the cluster, the formula is shown below:

$$CCF_{kt} = \sum_{i=1}^{I} \frac{R_{ikt}}{R_{kt}} \times CCF_{ikt}$$

Where i is the mutation, t is time point and k is the cluster. CCF_{kt} is the weighted CCF for cluster k at time t. R_{ikt} denotes to Total Reads of mutation i; $R_{kt} = \sum_{i=1}^{I} R_{ikt}$

3.3.2 PR42 result

After co-clustering matrix imputation, non-negative sparse coding and covariance matrix adjustment step, BIC is calculated for different cluster numbers (Table 3). Log-likelihood and penalty are all increases with K, which is consistent with the theoretical definition of Log-likelihood.

Table 3 PR42 BIC info

K	Log-likelihood	Penalty	BIC
2	76.45191	13.36304	-139.54077
3	95.61794	26.72608	-164.50979
4	134.25475	40.08913	-228.42037
5	167.43204	53.45217	-281.4119
6	165.79089	66.81521	-264.76657

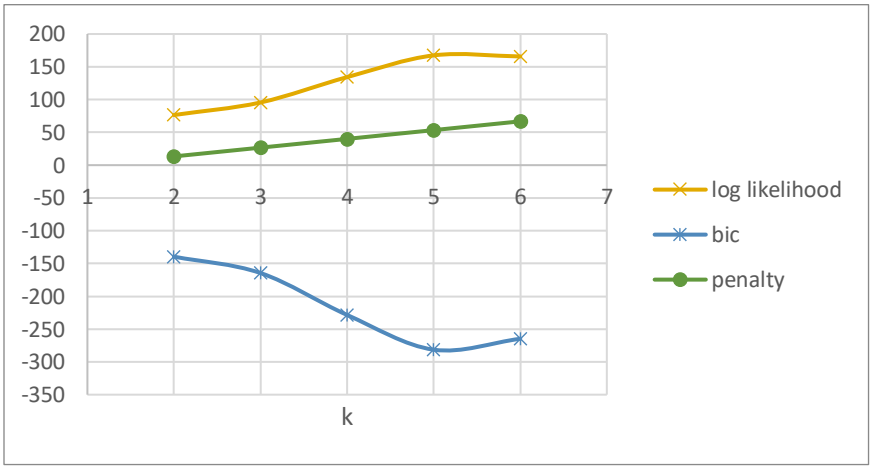

Figure 1 PR42 BIC for different clusters

BIC has the smallest value when K=5 (Figure 1), thus we choose 5 as the optimal number of clusters for PR42

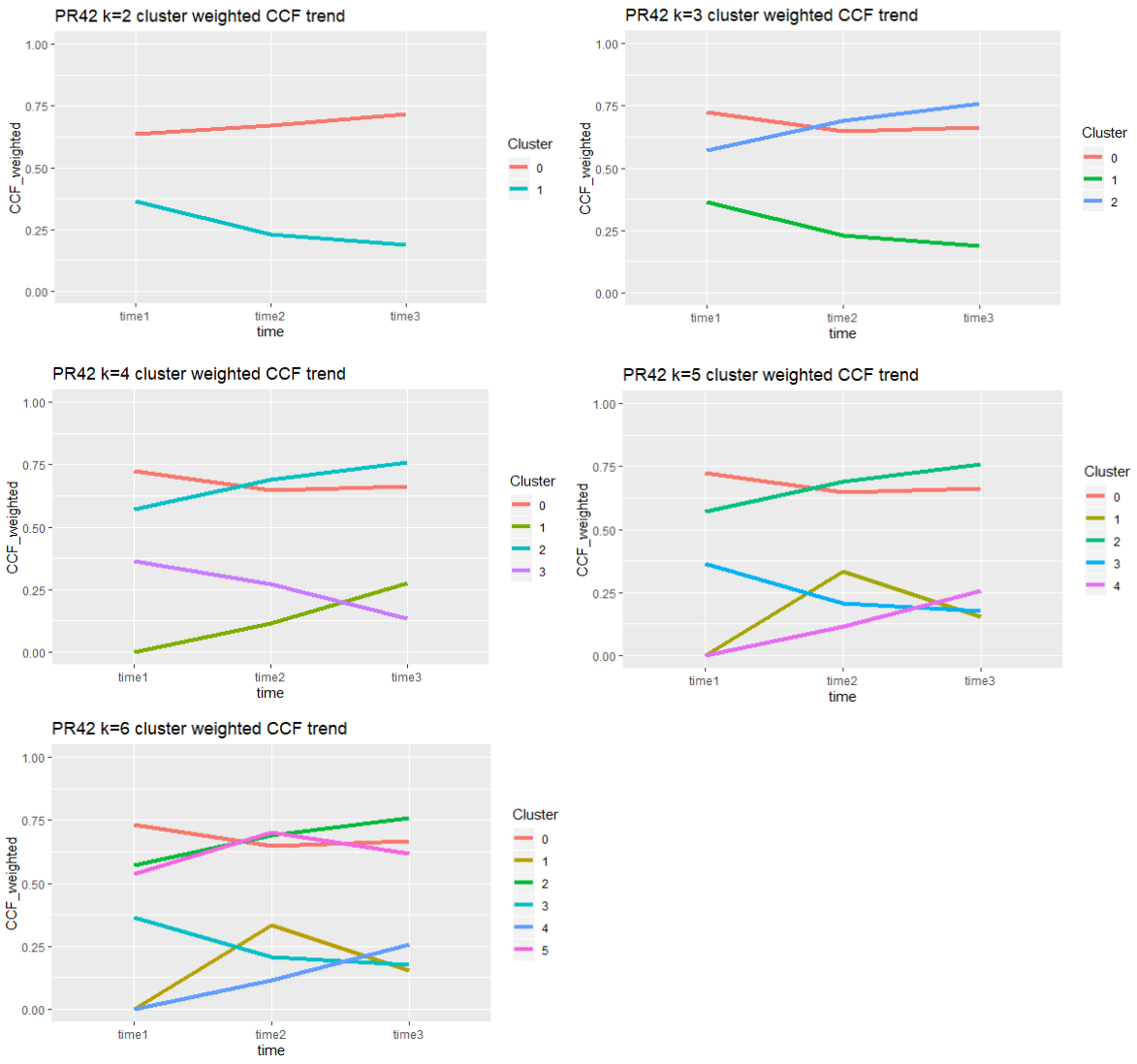

Figure 2 PR42 CCF trend

Mutations in this sample are split into two at CCF=0.5 (Figure 2. PR42 k=2). The two clusters in k=2 divided into 2 or 3 as k increase and BIC has the largest value at cluster number equal to 5. When k=5 every cluster has a unique trend. Cluster 0 (red line) and Cluster 2 (green line) have high frequency (CCF>0.5) but one goes up and one goes down. The rest three cluster have lower CCF and among them, Cluster 3 (blue line) goes down over time. This means the therapy at time 1 is effective for cluster 0 and cluster 3 but does not work well for the other 3 clusters.

3.3.3 PR44 result

PR44 is the one with the least missing mutations. The BIC is small at k=2, k=3 and k=5

Table 4 PR44 BIC info

K	Log likelihood	Penalty	BIC
2	750.61491	19.50079	-1481.73
3	630.08244	39.00158	-1221.16
4	619.1102	58.50237	-1179.72
5	630.98579	78.00316	-1183.97
6	509.92691	97.50395	-922.35

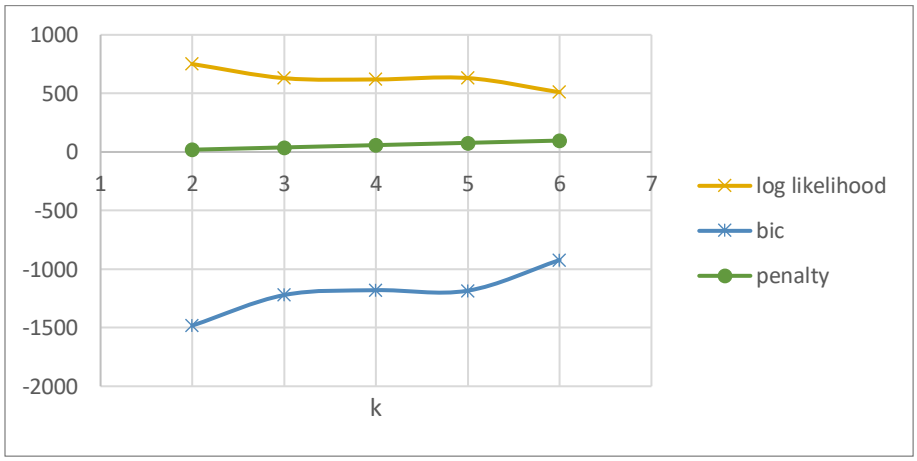

Figure 3 PR44 BIC for different clusters

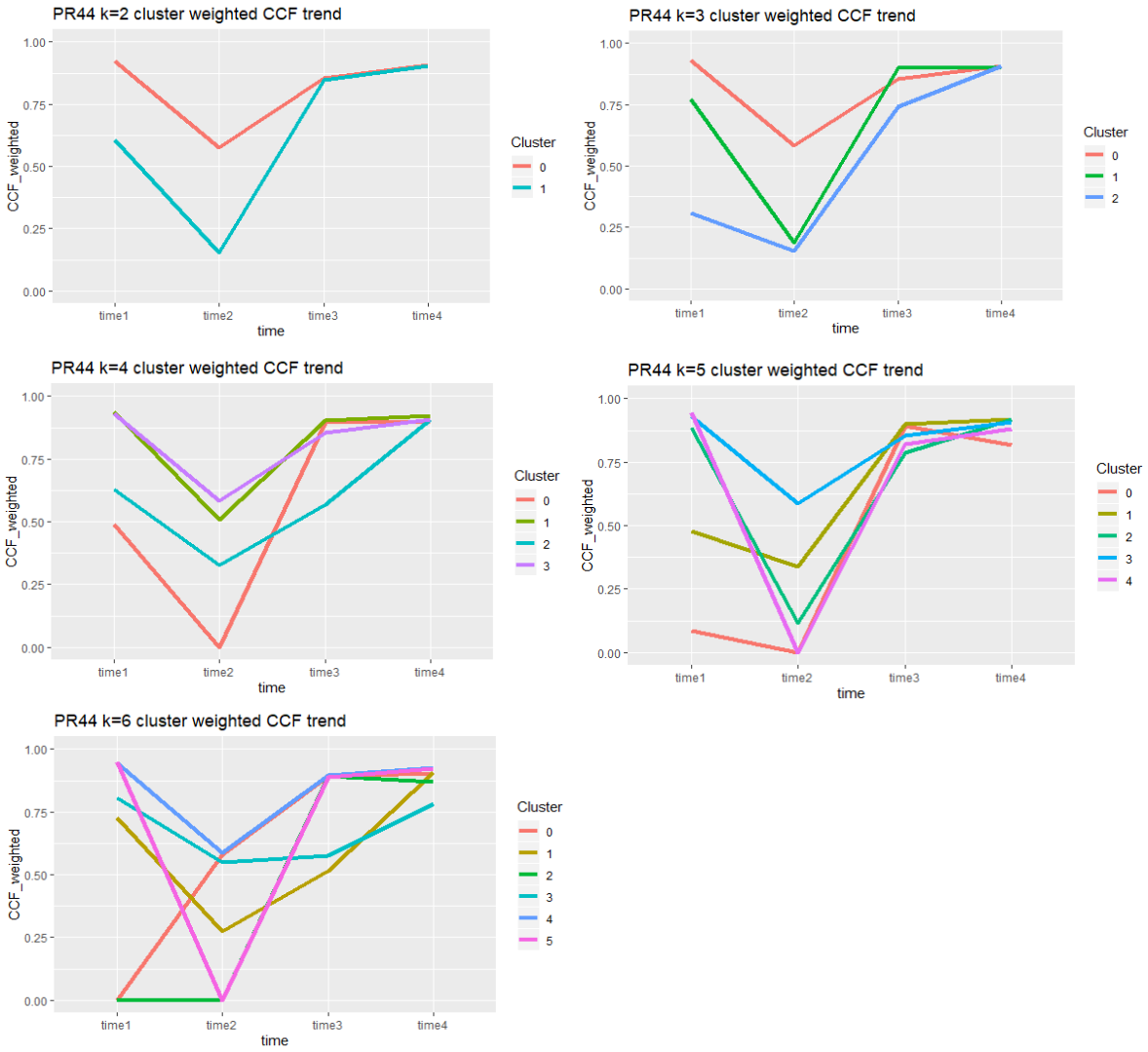

Figure 4 PR44 CCF trend

The whole trend for mutations in PR44 goes down then goes up, which implies that the therapy does well at first but then lost its effect. When k=2, two clusters have a similar trend that CCF decrease during Time1 and Time2 then goes up after that. Cluster 0 and Cluster 1 almost have the same CCF at Time1/3/4 and only differ at Time2. When k=3/5 the situation is similar to k=2 with the same trend.

3.3.4 PR240 result

PR240 is a special case that only has 7 mutations at time 1 and time 2, 19 mutations at time 3 (Table 1), with a total number of mutations = 83, which means most of the mutations come up after the therapy. This indicates that the therapy at Time1 is ineffective for this patient. A large amount of missing data also causes trouble in clustering. The co-clustering matrix at Time1 and Time2 are not reliable with more than 90% of mutation are missing since the imputation of the missing would highly depend on the other time point, and the trivial cluster problem increases the uncertainty of the imputation.

Table 5 PR240 BIC info

K	Log likelihood	Penalty	BIC
2	214.5095	22.0942	-406.9247
3		Special case	
4	182.18561	66.28261	-298.08861
5	162.31066	66.28261	-258.33871
6	201.03878	66.28261	-335.79495

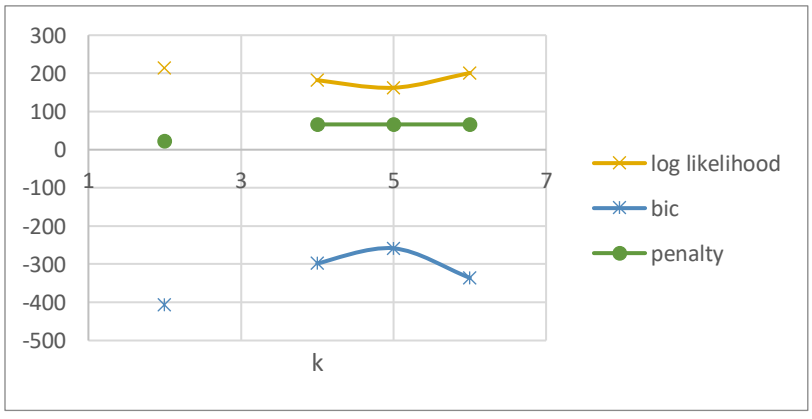

Figure 5 PR240 BIC for different clusters

BIC cannot be calculated at k=3 because of the big percentage of missing data. We got small BIC at k=2.

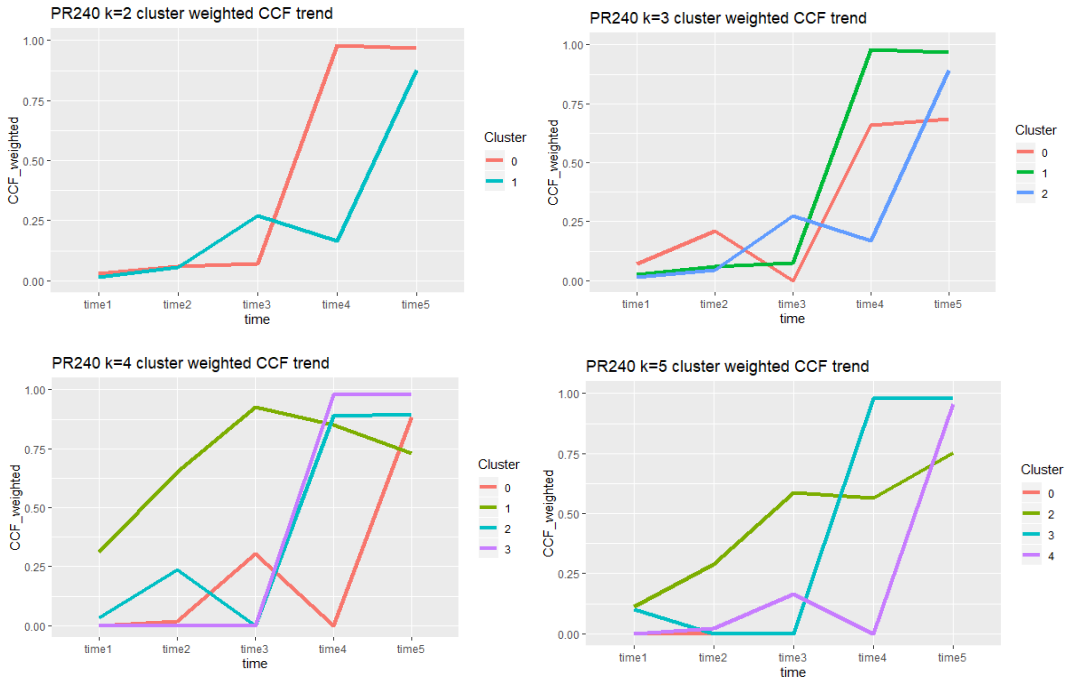

Figure 6 PR240 CCF trend

The trend of CCF for all clusters across all K increases with time. When k=2, two clusters differ in the time that CCF goes up.

3.3.5 PR246

PR246 is castration sensitive at the beginning but becomes resistant later. It is abnormal that the log-likelihood decreased as K increases. in this regard, it seems 2 is the best selection of cluster numbers.

Table 6 PR246 BIC info

K	Log likelihood	Penalty	BIC
2	640.99834	19.82331	-1262.17338
3	540.71928	39.64662	-1041.79195
4	500.80353	59.46992	-942.13713
5	432.8657	79.29323	-786.43817
6	478.16194	99.11654	-857.20734

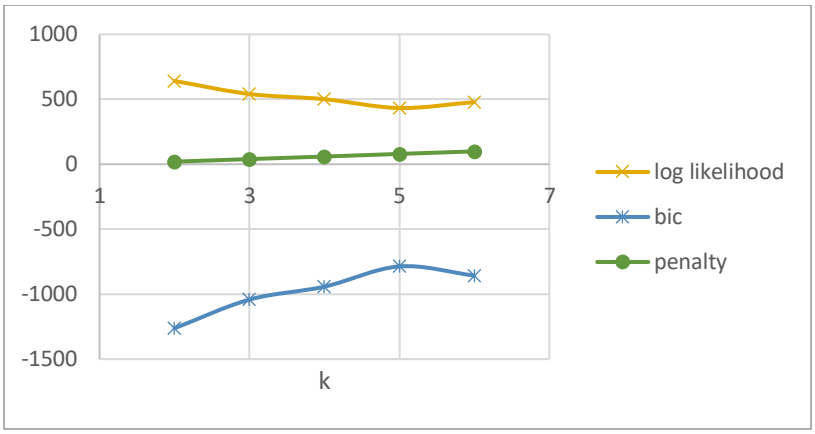

Figure 7 PR246 BIC for different clusters

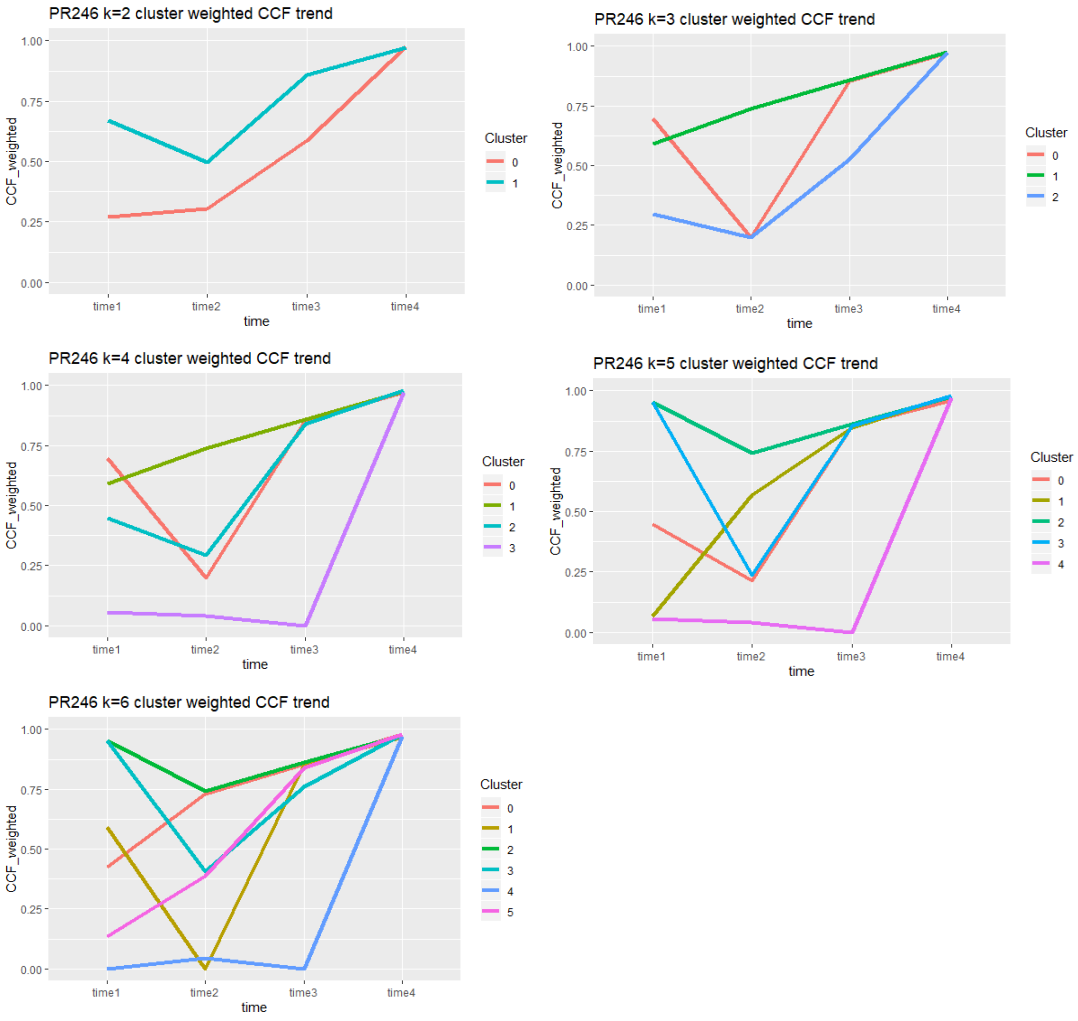

Figure 8 PR246 CCF trend

4 Discussion

This article aims to present a combinatorial algorithm to decide the subclone cluster of multi-timepoints tumor gene data. This model works well when data does not have a high percentage of missing mutations. Besides, the purity of the sample and the trivial clusters generated by Pyclone can affect the results.

Missing mutations directly impact the co-clustering matrix and covariance matrix in the BIC step. For the co-clustering matrix part, cells for missing mutations borrows information from neighboring time points and the off-diagonal mean of the matrix. In this circumstance, the information that the original matrix has may not enough for deducing the value of "missing cell" with more missing mutations. For example, PR240 only has 7 mutations at both Time1 and Time2, 19 at Time3, thus the co-clustering matrixes for Time1/2/3 refer to Time4 and Time5. However, in Time5 almost all mutations belong to one cluster (Time5 have 2 clusters and one of them is a trivial cluster), so the information provided is insufficient to infer the missing cell in Time1/2/3. Similarly, the missing mutations also influence the covariance matrix.

The method is based on the Pyclone process, therefore the trivial clusters from Pyclone could affect the consensus cluster determination and all processes followed. All samples have trivial clusters generated by Pyclone and lead to the inaccuracy of the co-clustering matrix.

Reference

1. Tai, A.-S., et al., *Decomposing the subclonal structure of tumors with two-way mixture models on copy number aberrations.* PLOS ONE, 2018. **13**: p. e0206579.
2. Beerenwinkel, N., et al., *Cancer Evolution: Mathematical Models and Computational Inference.* Systematic biology, 2014. **64**.
3. Dagogo-Jack, I. and A.T. Shaw, *Tumour heterogeneity and resistance to cancer therapies.* Nature Reviews Clinical Oncology, 2018. **15**(2): p. 81-94.
4. Fittall, M.W. and P. Van Loo, *Translating insights into tumor evolution to clinical practice: promises and challenges.* Genome Medicine, 2019. **11**(1): p. 20.
5. Deshwar, A.G., et al., *PhyloWGS: Reconstructing subclonal composition and evolution from whole-genome sequencing of tumors.* Genome Biology, 2015. **16**(1): p. 35.
6. Bolli, N., et al., *Heterogeneity of genomic evolution and mutational profiles in multiple myeloma.* Nature Communications, 2014. **5**(1): p. 2997.
7. Dentro, S., D. Wedge, and P. Loo, *Principles of Reconstructing the Subclonal Architecture of Cancers.* Cold Spring Harbor Perspectives in Medicine, 2017. **7**: p. a026625.
8. McGranahan, N. and C. Swanton, *Clonal Heterogeneity and Tumor Evolution: Past, Present, and the Future.* Cell, 2017. **168**: p. 613-628.
9. Jiao, W., et al., *Inferring clonal evolution of tumors from single nucleotide somatic mutations.* BMC Bioinformatics, 2014. **15**(1): p. 35.
10. Malikic, S., et al., *Clonality inference in multiple tumor samples using phylogeny.* Bioinformatics, 2015. **31**(9): p. 1349-1356.
11. Qiao, Y., et al., *SubcloneSeeker: a computational framework for reconstructing tumor clone structure for cancer variant interpretation and prioritization.* Genome Biology, 2014. **15**(8): p. 443.
12. Roth, A., et al., *PyClone: statistical inference of clonal population structure in cancer.* Nature Methods, 2014. **11**(4): p. 396-398.
13. Fraser, M., et al., *Genomic hallmarks of localized, non-indolent prostate cancer.* Nature, 2017. **541**(7637): p. 359-364.
14. Yuan, L., W. Liu, and Y. Li, *Non-negative dictionary based sparse representation classification for ear recognition with occlusion.* Neurocomputing, 2015. **171**.