Accurate Identification of Significant Aberrations in Cancer Genome: Implementation and Applications

Xuchu Hou

Thesis submitted to the faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Master of Science
in
Computer Engineering

Yue Wang, Chair
Alpay Ibrahim Özcan
Guoqiang Yu

December 6, 2012
Arlington, Virginia

Key words: Copy Number Alterations, Normal Tissue Contamination, Significant Copy number Aberrations, Concurrent Computing

Accurate Identification of Significant Aberrations in Cancer Genome: Implementation and Applications

Xuchu Hou

(ABSTRACT)

Somatic Copy Number Alterations (CNAs) are common events in human cancers. Identifying CNAs and Significant Copy number Aberrations (SCAs) in cancer genomes is a critical task in searching for cancer-associated genes. Advanced genome profiling technologies, such as SNP array technology, facilitate copy number study at a genome-wide scale with high resolution. However, due to normal tissue contamination, the observed intensity signals are actually the mixture of copy number signals contributed from both tumor and normal cells. This genetic confounding factor would significantly affect the subsequent copy number analyses.

In order to accurately identify significant aberrations in contaminated cancer genome, we develop a Java AISAIC package (Accurate Identification of Significant Aberrations in Cancer) that incorporates recent novel algorithms in the literature, BACOM (Bayesian Analysis of Copy

number Mixtures) and SAIC (Significant Aberrations in Cancer). Specifically, BACOM is used to estimate the normal tissue contamination fraction and recover the "true" copy number profiles. And SAIC is used to detect SCAs using large recovered tumor samples. Considering the popularity of modern multi-core computers and clusters, we adopt concurrent computing using Java Fork/Join API to speed up the analysis.

We evaluate the performance of the AISAIC package in both empirical family-wise type I error rate and detection power on a large number of simulation data, and get promising results. Finally, we use AISAIC to analyze real cancer data from TCGA portal and detect many SCAs that not only cover majority of reported cancer-associated genes, but also some novel genome regions that may worth further study.

Acknowledgments

I would like to take this opportunity to express my gratitude to everyone who ever helped me during my entire course of study.

First of all, I would like to thank my advisor Dr. Yue Wang for his kind help, both personally and academically. It's him who gave me the opportunity to get involved in such amazing research area and have chance to appreciate the beauty of Statistics.

Secondly, I would like to thank Dr. Alpay Özcan and Dr. Guoqiang Yu for their kindness to be my committee member and also for their forgiveness to the inconvenience I made.

Thirdly, very special thanks go to Dr. Guoqiang Yu, Dr. Bai Zhang and Dr. Xiguo Yuan for their great suggestions to my thesis writing and algorithm implementation. Whenever I turned to them for help, they always gave me illuminating explanation.

Fourthly, many thanks to my dear lab mates, Niya Wang, Ye Tian, Jinghua Gu, Fan Meng, just name a few, for their kind help and sincere suggestions whenever I needed.

Finally, deepest gratitude goes to my parents and my husband for their understanding and supports. I wish I could always let them be proud of me.

−Xuchu

December 6, 2012

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

DNA copy number (CN) is the number of copies of DNA in specific regions of the genome, the size of the regions could vary from 1kb to a complete chromosome arm [1]. In humans, the normal copy number is two for all the autosomes. However somatic copy number alterations (CNAs), both amplification and deletion are quite common in cancer [2-4] and several other complex diseases [5], such as HIV acquisition and progression [6], autoimmune diseases, and Alzheimer and Parkinson's disease [7], etc. Therefore, identification of copy number alterations (CNAs) may provide us important insight into the mechanisms of complex diseases, as well as useful information for diagnosis and treatment of the diseases, especially for cancer research. For this reason, substantial amount of efforts have been made to identify CNAs. Advanced genomic profiling technologies, such as high-throughput array comparative genomic hybridization (CGH) [8] and single nucleotide polymorphism (SNP)-based arrays [9, 10], as well as sequencing-based approaches have boosted the development of various approaches aiming to systematically identify and analyze copy number profiles in a genome-wide scale at a high resolution level.

1.1 Problems to Solve

A critical yet challenging task in the genome-wide analysis of copy number alterations in cancer is to distinguish "driver" alterations that are functionally important for tumor initiation and progression from the "passenger" alterations that are randomly acquired during the tumorigenesis [2, 11]. It is widely accepted that the regions that are likely to contain "driver" alterations are those recurrently occurring across cancer individuals. We call these regions "Significant Copy number Aberrations (SCAs)". A lot of previous study results show that these SCAs cover many well-known oncogenes and tumor suppressor genes. On the contrary, the "passenger" alterations are more individual-specific.

Another troublesome problem we are facing in copy number analysis is tissue heterogeneity. Although, molecular analysis of tumor cells in their native tissue environment provides us accurate information of their *in vivo* state, normal cell contamination is often an unavoidable confounding factor since in real sample acquisition, it's hard or even impossible to obtain pure tumor tissue without normal tissue contamination, thus the acquired tumor samples are actually the mixture of cancer and normal cells. As a result, this tissue heterogeneity could significantly affect the subsequent true copy number alteration detection, SCAs detection, and other CNAs related studies.

1.2 Survey of Existing Approaches

Identifying SCAs involves a multiple testing procedure on multiple candidate genome loci. Till now, there have been several methods available in literatures [11-17]. Although they have certain differences in algorithm details, they follow a common procedure: (1) detect copy number amplification and deletion separately; (2) design a summary statistics regarding each candidate genome loci; (3) estimate the null distribution of the summary statistics and carry out hypothesis testing to detect potential SCAs. Here we will give a brief review of some existing methods.

The most popular method is Genomic Identification of Significant Targets in Cancer (GISTIC) [11], which is first proposed by Broad Institute. The algorithm is based on log-ratio copy number data after normalization and segmentation. GISTIC assigns each genome locus a summary statistics G-Score, which is the average copy number alteration across multiple samples. Since the method assumes that the loci are independent from each other, it uses convolution to estimate the null distribution of the summary statistic. In order to control false discovery rate, GISTIC adopt a FDR correction to the estimated p-value. Finally a peeling scheme is applied to avoid less significant SCAs from being masked by strong significant SCAs. Significant Testing for Aberrant Copy number (STAC) [12] is based on a binary matrix, where zero indicating no copy number alteration while one indicating copy number alteration, either amplification or deletion. It uses two complementary statistic, footprint and frequency, to each location, and then use a multiple testing corrected permutation approach to assign P-value for each location. Discovering Copy Number Aberrations Manifested in Cancer (DiNAMIC) [13] uses segmented copy number data as input and designs a global summary statistic for testing each marker. To be specific, each time DiNAMIC only tests the significance of one locus with maximal sum of copy number across multiple samples. Another important feature of DiNAMIC is that it employs a novel cyclic permutation scheme that preserves the entire serial structure of the genome when they try to estimate the null distribution of their statistic. The input data of all the three methods are preprocessed copy number data, i.e., the data are already normalized and segmented. However, some others hold that the preprocessing may result in information loss, heavy computational burden and create unexpected artificial effect, which may degrade the performance of the SCA detection. Therefore, they prefer to use raw chip intensity as the input. Correlation Matrix Diagonal Segmentation (CMDS) [14] is one of these methods. It uses raw intensity ratio data from all the samples and explores the between-chromosomal-site correlation. It adopts a diagonal transformation strategy to construct a RCNA score, and detect the SCAs based on a standard normal null distribution where mean and variance are estimated from the input data. You can find more detailed and comprehensive comparison between different SCAs detection methods in reference [18].

1.3 Issues with Existing Approaches

Although most of the existing approaches can provide reasonably good results on copy number data, they still have some flaws that influence their performance. We will briefly describe the problems exist in these algorithms.

First of all, none of the existing methods takes the normal tissue contamination problem into consideration when they are designed. Ignoring of this problem may affect the accuracy of the copy number alteration detection and SCAs detection. Here we will discuss the ways that normal tissue contamination affects the copy number analysis.

1. Suppose we have a pure tumor sample with copy number profile after segmentation [2, 2, 5, 4, 2, 2, 2, 1, 0, 2, 2, 2]. There are four loci has copy number alterations, two amplifications and two deletions. And apparently this could be detected if we use threshold 2.5 for amplification and 1.5 for deletion. To be specific, if a probe has copy number value larger than 2.5, it is detected as copy number amplification; while if its value is less than 1.5, it is detected as copy number deletion. However if there is normal tissue contamination, e.g. 80% (in real world, the contamination can be up to this value), the obtained copy number profile will change to [2, 2, 2.6, 2.4, 2, 2, 2, 1.8, 1.6, 2, 2, 2]. Apparently, if we still use 2.5/1.5 as thresholds to detect copy number alterations, we will miss three copy number alteration loci. We may think we can change the threshold to cancel out the effect of normal tissue contamination, but no one can know the normal tissue contamination fraction in advance, and more importantly different samples usually have different normal tissue contamination degree, which makes the problem get even harder.
2. Even if we didn't miss any CNA loci using thresholds 2.5/1.5, there is still another issue, which may further influence the SCAs detection. Just as mentioned above, different samples usually have different normal tissue contamination degree, this discrepancy between the samples will also confound the SCAs detection. We will use a simple scenario in Figure 1.1 to illustrate.

Suppose we have a normalized copy number data matrix that contains 4 samples and 6 loci, and there is one and only one SCA that is located in the second locus. If all the samples are obtained from pure tumor tissues, we get the copy number data matrix as the left one, based on which, the second location could easily be identified as SCA because it has both the largest recurrent rate, ¾ and the largest average copy number alteration. However, if neither of the samples is acquired from pure tumor tissue, to be specific, the normal tissue contamination fraction for sample1, sample2, sample3 and sample4 are 60%, 10%, 30%, and 80% respectively, we will get a different matrix shown in the right. Under this circumstance, we will notice that although both the second location and third location has recurrent rate ½, but the third location has larger average copy number alteration, therefore the third one will probably be picked out as the SCA. Apparently, this is a false positive.

$$\begin{bmatrix} 2 & 2 & 2 & 2 \\ 4 & 4 & 2 & 3 \\ 2 & 4 & 6 & 2 \\ 2 & 2 & 2 & 2 \\ 2 & 0 & 2 & 2 \\ 3 & 2 & 2 & 2 \end{bmatrix} \xrightarrow[\text{sample2 10\% normal tissue contamination}]{\text{sample1 60\% normal tissue contamiantion}} \begin{bmatrix} 2 & 2 & 2 & 2 \\ 3 & 3.8 & 2 & 2.2 \\ 2 & 3.8 & 4.8 & 2 \\ 2 & 2 & 2 & 2 \\ 2 & 0.2 & 2 & 2 \\ 2.4 & 2 & 2 & 2 \end{bmatrix}$$

Figure 1.1 Example of the influence of normal tissue contamination

Except the normal tissue contamination, most existing methods also have following limitations in SCAs detection:

1. Many of those methods test the significance of individual locus assuming that different loci are independent from each other. However, it is widely accepted that consecutive CNA loci are usually highly correlated [13, 20, 21]. Thus the assumption of locus independence although simple, may result in inaccuracy in the SCAs detection.

2. The null hypothesis is the distribution of summary statistic given no SCAs existing, only sporadic CNAs. However most of existing methods use all the genome loci to estimate the null distribution. In other words, both sporadic CNAs and actual SCAs are involved in the null distribution estimation [11, 12, 13, 14], which will potentially decrease the detection power especially when there is a large number of SCAs existing in the data set.

1.4 Contributions of the Work

Considering the aforementioned issues in the existing approaches, in this thesis we introduce a two-step based statistical approach to accurately identify the significant copy number aberrations in contaminated cancer genome. We first use a recent reported approach Bayesian Analysis of Copy number Mixtures (BACOM) to estimate the normal tissue contamination fraction and extract the "true" tumor copy number profile. Then based on the multiple corrected tumor samples, we use the Genome-wide Identification of Significant Aberrations in Cancer Genome (SAIC) [16], a carefully designed statistical method to detect the SCAs that potentially contain oncogenes and/or tumor suppressor genes.

In order to better serve the research community, we implement the introduced approach in a Java software package AISAIC (Accurate Identification of Significant Aberrations in Cancer) with a friendly graphical user interface (GUI). Since modern computers and clusters often have multiple cores, we adopt concurrent computing using Java Fork/Join API to improve the efficiency of the software. For example, the new AISAIC package can complete the whole pipeline analysis of a data set with 103 samples within about 4 hours on multiple-core computer with a 24G memory and twelve 2.8GHz processors.

We evaluate the performance of the proposed approach with a large number of simulation. First we test the empirical family-wise type I error rate (FWER) of the new approach using null simulation data. And we get average FWER as 0.0516, which is within the 95% confidence interval of the theoretical value 0.05. Therefore, we can be confident about the detected SCAs using the new approach. Then we compare the detection power of the proposed approach with two peer methods GISTIC and SAIC on multiple

groups of simulation data set, and the proposed approach provides better performances on almost all the data set.

Finally, we use AISAIC package to analyze a real brain tumor data set from TCGA portal. We detect many SCAs that cover majority of cancer-associated genes that reported by other methods, and more importantly we also detect some new regions that may worth further study.

In summary:

1. We report a statistical approach based on two recent novel algorithms BACOM and SAIC to detect significant copy number aberrations in contaminated cancer genome.
2. We implement the proposed approach in a Java AISAIC package (Accurate Identification of Significant Aberrations in Cancer) with a friendly graphical user interface (GUI). One major feature in the AISAIC is that we adopt concurrent computing in the software development to make best use of the multiple cores in modern computers and clusters, and more importantly to improve the efficiency of the algorithm.
3. We evaluate the performance of the proposed approach on empirical family-wise type I error and the detection power based on a large number of realistic simulation data. Finally, we use the AISAIC software package to analyze real cancer data from TCGA portal and get promising results.

1.5 Organization of the Thesis

The whole thesis is organized as below:

In chapter 2, we will describe the theory and algorithm of our proposed approach in detail. We derive the theory of BACOM based on a realistic copy number model and some reasonable assumptions. And then we will introduce the algorithm of SAIC step by step.

In chapter 3, we discuss the implementation of our proposed approach using Java to facilitate the research community. We adopt concurrent computing using Java Fork/Join API to improve the speed of the

algorithm to a large extent, and we also design a user-friendly graphical user interface using Java Swing API for user's convenience.

In chapter 4, we evaluate our proposed approach from different aspects and compare it with GISTIC, which the most popular method for SCA detection. In addition, we apply our approach to real cancer data downloaded from TCGA portal.

In chapter 5, discussion and potential future work are given.

Chapter 2

Theory and Method

In this chapter, we will elaborate the theory and principles of BACOM and SAIC methods in detail, where BACOM is derived based on a reasonable statistic model of real copy number signal and SAIC is introduced with several novel features highlighted.

2.1 Theory of BACOM

2.1.1 Copy Number Signal Model

Taking tissue heterogeneity in to consideration, we can model the observed array intensity as the weighted sum of DNA copy number signals contributed from both normal cells and tumor cells, given mathematically by:

$$X_i = \beta \times X_{i,normal} + (1-\beta) \times X_{i,tumor}\ , \tag{1}$$

where, X_i denotes the observed DNA copy number signal at locus i, β is the normal tissue contamination fraction that needs to be estimated, $X_{i,normal}$ and $X_{i,tumor}$ stand for the unknown latent DNA copy number signals of normal cell and tumor cell at locus i, respectively. Since human somatic cells are diploid, the expected DNA copy number at a certain locus i in normal cells is 2, i.e., $E[X_{normal}] = 2$. However, when there exists a homo-deletion or hemi-deletion, the expected DNA copy number at locus i in tumor cells will change to 0 or 1, i.e., $E[X_{tumor}] = 0$ or 1 respectively. Then, by focusing only on copy number deletion and taking the expectation on both sides of Eq. (1), we have:

$$\begin{cases} E[X_i] = \beta \times 2 + (1-\beta) \times 0 = 2\beta & \text{homo-deletion} \\ E[X_i] = \beta \times 2 + (1-\beta) \times 1 = 1 + \beta & \text{hemi-deletion} \end{cases} \tag{2}$$

From Eq. (2), we can clearly see that the expectation of the observed copy number signal is the function of β. Therefore, if we could detect different copy number deletion types, theoretically speaking, we could use Eq. (2) to estimate the normal cells contamination factor β by the sample average of the deletion segments. This is exactly the idea of BACOM algorithm. In BACOM, we use a Bayesian hypothesis testing to distinguish homo-deletion and hemi-deletion segments based on the allele-specific signals, and then estimate β from Eq. (2) as below:

$$\beta = \begin{cases} E[X_i]/2 & \text{homo-deletion} \\ E[X_i] - 1 & \text{hemi-deletion} \end{cases} \tag{3}$$

2.1.2 Deletion Type Determination

From the above introduction, we know that the most critical step in BACOM is to detect the deletion type. In this section, we will discuss about how to detect the deletion type in BACOM.

Affymetrix SNP chips provide both allele-specific signals (A allele and B allele) and their summed intensity (the observed DNA copy number signal). If we denote the allele-specific signals of A allele and B allele at locus i as $X_{A,i}$ and $X_{B,i}$, then we have:

$$X_i = X_{A,i} + X_{B,i} \tag{4}$$

In BACOM algorithm, we focus on heterozygous (AB genotype) loci. This is decided by the summary statistics we designed later. In order to locate the heterozygous loci, we need the help from the paired normal samples (collected from different organs, usually from blood other than the organ where the tumor locates at) of the same subject. This is because in tumor tissue, the used to be AB genotype loci may not be AB genotype any longer because of copy number alterations.

In normal samples, the intensity of A-allele signal and B-allele signal are close to each other in AB genotype, while in AA genotype and BB genotype, they will have larger difference. Figure 2.1 shows the distribution of intensities after $\log 2$ transform of raw intensities of A-allele and B-allele of a sample from Affymetrix Genome-Wide SNP Array 6.0. We can clearly see that there are three clusters, the middle one in the diagonal direction corresponds to the AB genotype while the upper left and below right one correspond to BB genotype and AA genotype, respectively. Since false positive will affect the performance of BACOM and the subsequent analysis, we just use the probe sets that in between the green line, which includes about 20% of all the probe sets. This is a stringent criterion since the Affymetrix SNP arrays were designed to include probe sets with genotype AB on average about 25%~30%.

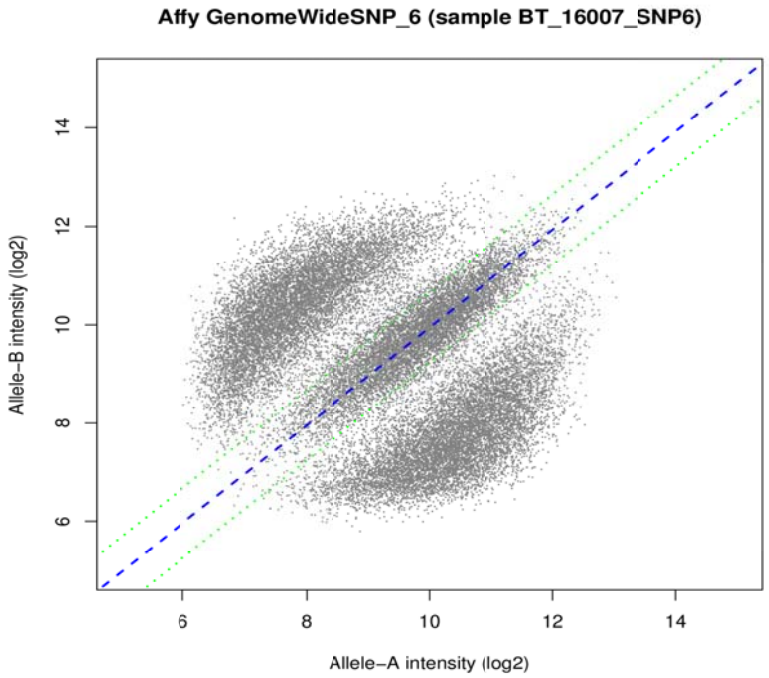

Figure 2.1 Distribution of intensities of A allele and B allele signals after log2 transform

Before deriving the algorithm of BACOM, we first make the following reasonable assumptions on the allele-specific signals:

"For a length-L homo/hemi-deletion segment $\{X_i \mid i = 1,2,3\cdots L\}$, each of the allele-specific signals $X_{A,i}$ and $X_{B,i}$ are independently distributed Gaussian random variables with different means but common variance σ^2, for $i = 1,2,3\cdots L$.

Under this assumption, the observed DNA copy number signal $\{X_i \mid i = 1,2,3\cdots L\}$ are i.i.d random variables that follow a Gaussian distribution with mean μ_{A+B} and variance σ^2_{A+B}, both of which could be estimated directly through the observed signals $\{X_i \mid i = 1,2,3\cdots L\}$. One thing should be noticed is that $X_{A,i}$ and $X_{B,i}$ are not independent from each other, instead, they are usually correlated, referred to as crosstalk between two alleles [23].

In order to distinguish between homo-deletion and hemi-deletion, we define a statistic Y as:

$$Y = \sum_{i=1}^{L} \left(\frac{X_{(A-B),i}}{\sigma_{A-B}} \right)^2 = \sum_{i=1}^{L} \left(\frac{X_{A,i} - X_{B,i}}{\sigma_{A-B}} \right)^2 \tag{5}$$

where, σ_{A-B} is the standard deviation of $X_{A,i} - X_{B,i}$. In the following section, we can derive that based on the above assumption, Y follows either a standard χ^2 distribution or a non-central χ^2 distribution depends on the copy number deletion type.

For a L-length homo-deletion segment, the expected copy number for A allele and B allele in normal cells are both 1, i.e., $E[X_{normal,A,i}] = 1$ and $E[X_{normal,B,i}] = 1$ while in tumor cells both are 0, i.e., $E[X_{tumor,A,i}] = 0$ and $E[X_{tumor,B,i}] = 0$. So for homo-deletion segment, we have:

$$\begin{aligned}
\mu_{(A-B),i} &= E[X_{A,i} - X_{B,i}] \\
&= E[(\alpha \times X_{normal,A,i} + (1-\alpha) \times X_{tumor,A,i}) \\
&\quad - (\alpha \times X_{normal,B,i} + (1-\alpha) \times X_{tumor,B,i})] \\
&= \alpha \times E[X_{normal,A,i} - X_{normal,B,i}] \\
&\quad + (1-\alpha) \times E[X_{tumor,A,i} - X_{tumor,B,i}] \\
&= \alpha \times (1-1) + (1-\alpha) \times (0-0) \\
&= 0 \quad i = 1, 2, 3 \ldots L
\end{aligned} \tag{6}$$

Therefore, the random variable $Z = \frac{X_{A,i} - X_{B,i}}{\sigma_{A-B}}$ is a standard Gaussian random variable, and Y is a standard χ^2 distribution with L degrees of freedom, whose probability density function is given by:

$$f(z; L) = \begin{cases} \frac{1}{2^{L/2}\Gamma(L/2)} z^{L/2-1} e^{-z/2} & z \geq 0 \\ 0 & \textit{otherwise} \end{cases} \tag{7}$$

where, Γ denotes the Gamma function.

For a L-length hemi-deletion segment, the expected copy number for A allele and B allele in normal cells are both 1, i.e., $E[X_{normal,A,i}]=1$ and $E[X_{normal,B,i}]=1$ while in tumor cells either A allele or B allele is 0, i.e., $E[X_{tumor,A,i}]=0$, $E[X_{tumor,B,i}]=1$ or $E[X_{tumor,A,i}]=1$, $E[X_{tumor,B,i}]=0$. So for hemi-deletion segment, we have:

$$\begin{aligned}\mu_{(A-B),i} &= E[X_{A,i}-X_{B,i}] \\ &= E[(\alpha\times X_{normal,A,i}+(1-\alpha)\times X_{tumor,A,i}) \\ &\quad -(\alpha\times X_{normal,B,i}+(1-\alpha)\times X_{tumor,B,i})] \\ &= \alpha\times E[X_{normal,A,i}-X_{normal,B,i}] \\ &\quad +(1-\alpha)\times E[X_{tumor,A,i}-X_{tumor,B,i}] \\ &= \alpha\times(1-1)\pm(1-\alpha)\times(1-0) \\ &= \pm(1-\alpha), \quad i=1,2,3,...L\end{aligned} \tag{8}$$

From Eq. (2), we have $\mu_{A+B,i}=E[X_i]=1+\alpha$ for hemi-deletion, in other words, $\alpha=\mu_{A+B,i}-1$. Thus, $\mu_{A-B,i}$ could be expressed in terms of $\mu_{A+B,i}$ as:

$$\mu_{A-B,i}=\pm(1-\alpha)=\pm(2-\mu_{A+B,i}) \tag{9}$$

Based on the above assumption, we can get $\sigma^2_{(A+B),i}$ and $\sigma^2_{(A-B),i}$ as below:

$$\begin{aligned}\sigma^2_{(A+B),i} &= E[(X_{(A+B),i}-\mu_{(A+B),i})^2] \\ &= E[(X_{A,i}-\mu_{A,i}+X_{B,i}-\mu_{B,i})^2] \\ &= E[(X_{A,i}-\mu_{A,i})^2+2(X_{A,i}-\mu_{A,i})(X_{B,i}-\mu_{B,i})+(X_{B,i}-\mu_{B,i})^2] \\ &= E[(X_{A,i}-\mu_{A,i})]^2+2E[(X_{A,i}-\mu_{A,i})(X_{B,i}-\mu_{B,i})] \\ &\quad +E[(X_{B,i}-\mu_{B,i})^2] \\ &= \sigma_A^2+2\rho\sigma_A\sigma_B+\sigma_B^2 \\ &\overset{\sigma_A=\sigma_B=\sigma}{=} 2\sigma^2(1+\rho)\end{aligned} \tag{10}$$

$$
\begin{aligned}
\sigma^2_{(A-B),i} &= E[(X_{(A-B),i} - \mu_{(A-B),i})^2] \\
&= E[(X_{A,i} - \mu_{A,i} - X_{B,i} - \mu_{B,i})^2] \\
&= E[(X_{A,i} - \mu_{A,i})^2 - 2(X_{A,i} - \mu_{A,i})(X_{B,i} - \mu_{B,i}) + (X_{B,i} - \mu_{B,i})^2] \\
&= E[(X_{A,i} - \mu_{A,i})]^2 - 2E[(X_{A,i} - \mu_{A,i})(X_{B,i} - \mu_{B,i})] \\
&\quad + E[(X_{B,i} - \mu_{B,i})^2] \\
&= \sigma_A^2 - 2\rho\sigma_A\sigma_B + \sigma_B^2 \\
&\overset{\sigma_A=\sigma_B=\sigma}{=} 2\sigma^2(1-\rho)
\end{aligned}
\tag{11}
$$

Then we have: $\sigma^2_{(A-B),i} = \sigma^2_{(A+B),i}(1-\rho)/(1+\rho)$. Here, ρ is the correlation coefficient between $X_{A,i}$ and $X_{B,i}$. From above derivation, we can see that estimating $\mu_{A-B,i}$ and $\sigma^2_{(A-B),i}$ could be easily done using only the total copy number signals X_i.

Using $\mu_{A-B,i}$ $\sigma^2_{(A-B),i}$, we can estimate the non-centrality parameter λ as:

$$
\begin{aligned}
\lambda &= \sum_{i=1}^{L} \left(\frac{\mu_{(A-B),i}}{\sigma_{(A-B),i}}\right)^2 \\
&= \sum_{i=1}^{L} \frac{(2-\mu_{(A+B),i})^2(1+\rho)}{\sigma^2_{(A+B),i}(1-\rho)} \\
&= L\frac{(2-\mu_{(A+B),i})^2(1+\rho)}{\sigma^2_{(A+B),i}(1-\rho)}
\end{aligned}
\tag{12}
$$

Finally, the pdf of statistic Y under hemi-deletion is given by:

$$
f(z;L,\lambda) = \begin{cases} \dfrac{e^{-(z+\lambda)/2}}{2^{L/2}} \displaystyle\sum_{k=0}^{\infty} \frac{z^{L/2+k-1}\lambda^k}{\Gamma(k+L/2)2^{2k}k!} & z \geq 0 \\ 0 & \textit{otherwise} \end{cases}
\tag{13}
$$

where, Γ denotes the Gamma function.

Based on the above derived deletion-type-conditioned probability density functions, we can use Bayesian hypothesis testing to determine the type of deletion for a certain segment using the copy number signals $X_{A,i}$, $X_{B,i}$ and X_i without knowing the deletion type associated with them.

Therefore, the final deletion type decision could be made as:

$$\begin{cases} \dfrac{P(\text{hemi-deletion} \mid z)}{P(\text{homo-deletion} \mid z)} \geq 1 \\ \dfrac{P(\text{hemi-deletion} \mid z)}{P(\text{homo-deletion} \mid z)} < 1 \end{cases} \Rightarrow \begin{cases} \text{hemi-deletion} \\ \\ \text{homo-deletion} \end{cases} \tag{14}$$

where, $p(\cdot \mid \cdot)$ denotes the posterior probability of the segment deletion type given the statistic Y.

2.1.3 BACOM Implementation

From the above description of BACOM algorithm, we know that in order to implement BACOM algorithm, we need to estimate the model parameters $\mu_{A+B,i}$, $\sigma^2_{(A+B),i}$ and ρ. Since different segments may have different values. Therefore each segment, $\mu_{A+B,i}$, $\sigma^2_{(A+B),i}$ can be estimated as below:

$$\mu_{A+B} = \frac{1}{L}\sum_{i=1}^{L} X_i \quad \sigma^2_{A+B} = \frac{1}{L-1}\sum_{i=1}^{L}(X_i - \mu_{A+B})^2 \tag{15}$$

In addition, we assume that ρ is identical across all the loci within one subject, hence we could use the copy number signals at the normal loci in all the normal segments to estimate, given by:

$$\mu_A = \sum_{i=1}^{N_{normal}} X_{A,i}, \mu_B = \sum_{i=1}^{N_{normal}} X_{B,i} \tag{16}$$

$$\rho = \frac{\sum_{i=1}^{N_{normal}} (X_{A,i} - \mu_A)(X_{B,i} - \mu_B)}{\sqrt{\sum_{i=1}^{N_{normal}} (X_{A,i} - \mu_A)^2}\sqrt{\sum_{i=1}^{N_{normal}} (X_{B,i} - \mu_B)^2}} \tag{17}$$

Having estimated the model parameters $\mu_{A+B,i}$, $\sigma^2_{(A+B),i}$ and ρ, we can apply Bayesian hypothesis testing based on (14) to decide the deletion type for a particular deletion segment k. Then we can estimate the normal tissue contamination fraction with Eq. (2). For homo-deletion segment, we have $\beta_k = \mu_k / 2$, while for hemi-deletion segment, we have $\beta_k = \mu_k - 1$. If there are K deletion segments in a sample, we can calculate the ultimate normal tissue contamination fraction via segment-length weighted average.

$$\overline{\beta} = \frac{\sum_{k=1}^{K} \beta_k \times L_k}{\sum_{k=1}^{K} L_k} \tag{18}$$

where, L_k is the length of the kth deletion segment.

Finally, with the estimated normal cells contamination fraction $\overline{\beta}$, we can extract the underlying "true" DNA copy number in the samples as below:

$$X_{tumor,i} = \frac{X_i - 2\overline{\beta}}{1 - \overline{\beta}} \tag{19}$$

2.1.4 Flowchart of BACOM

Finally, we give the flowchart of BACOM algorithm in Figure 2.2.

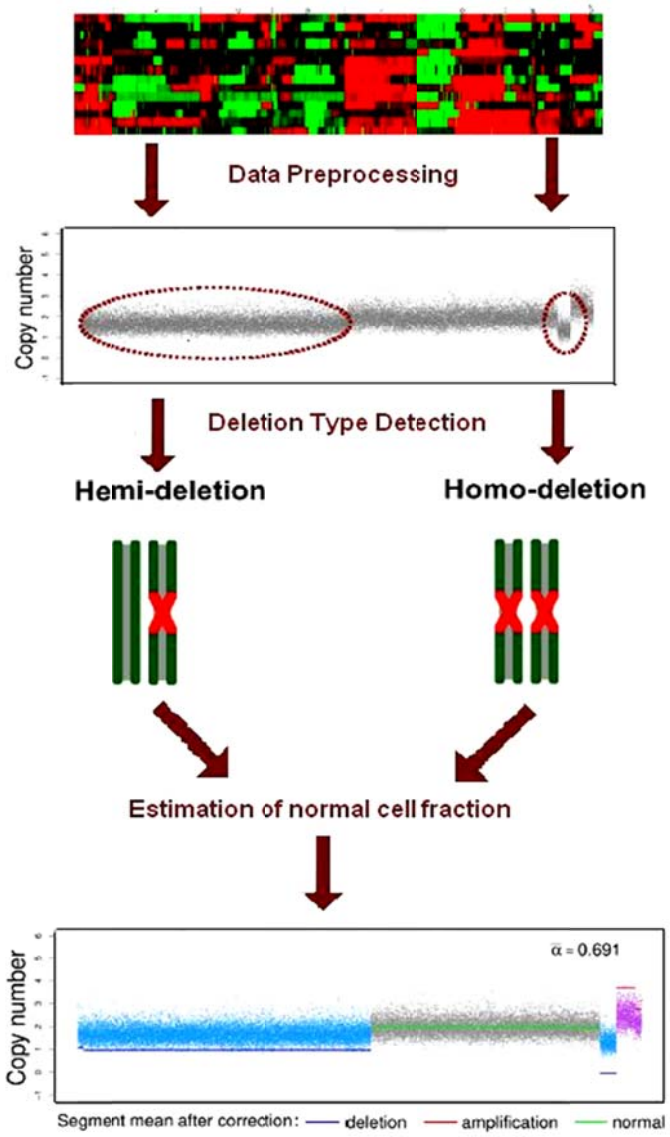

Figure 2.2 Flowchart of BACOM algorithm

2.2 Theory of SAIC Algorithm

Identification of Significant Consensus Aberrations is in essence a multiple testing problem, so it follows general hypothesis testing procedure. SAIC contains three major steps: (1) detect copy number alteration (CNAs) probes, both amplification and deletion; (2) construct CNA units that incorporate the correlation between consecutive probes, and assign a summary statistics to each CNA unit; (3) use a SCAs-excluding permutation scheme to generate a relatively more accurate null hypothesis, and assign a P-value to every CNA unit to evaluate their significance. In the following sections, we will discuss this procedure in detail.

2.2.1 CNA Probes Detection

First, we would like to introduce the data format for SAIC input. We assume that the raw copy number intensity signal has been preprocessed, and we store the preprocessed log2-ratio copy number data into a $N \times M$ matrix X , where N is the number of probes while M is the number of samples. Every entry in the matrix $x_{i,j}$, $i = 1,2,3 \cdots N$, $j = 1,2,3 \cdots M$ represents the copy number value at *i*th probe in *j*th sample. Since we analyze amplification and deletion separately, we split the input matrix into two sub-matrices that contains the same number of probes and samples. In order to do this, we use thresholds $\theta_{\text{amplification}}$ and θ_{deletion} to distinguish between altered probes with normal probes.

$$\begin{aligned} X_{\text{amplification}} &= \left\{ I\left(\left(x_{i,j} - \theta_{\text{amplification}} \right) > 0 \right) \cdot x_{i,j} \right\}, \\ X_{\text{deletion}} \quad &= \left\{ I\left(\left(x_{i,j} - \theta_{\text{deletion}} \right) < 0 \right) \cdot x_{i,j} \right\}, \end{aligned} \tag{20}$$

The probe whose associated copy number is altered, either amplified or deleted in at least one sample is defined as CNA probe.

2.2.2 CNA units and Summary Statistics

It is widely accepted that consecutive CNA probes are usually correlated, so in order to incorporate this information into SCAs detection, we combine consecutive probes that are highly and positively correlated into a CNA unit and assign a summary statistic to it.

The correlation between consecutive probes across samples is evaluated using Pearson correlation as below:

$$r_{i,j} = \frac{\sum_{k=1}^{N}(x_{k,i} - \overline{x_i})(x_{k,j} - \overline{x_j})}{(N-1)\sigma_i\sigma_j} \tag{21}$$

The following Figure 2.3 illustrates how we construct CNA units: first, we group consecutive probes into intervals, just as shown in the left figure, which contains two intervals, and the first interval covers 10 probes while the second one covers 3 probes. Then, we evaluate the Pearson correlation coefficient between consecutive probes and identify the break points where two consecutive probes are not correlated highly enough across samples, then split the interval into several CNA units, just shown in the right figure below. We denote the CNA unit using the start position and its associated length, such as $u(k, L)$ which means the CNA unit starts from position *k* with length *L*.

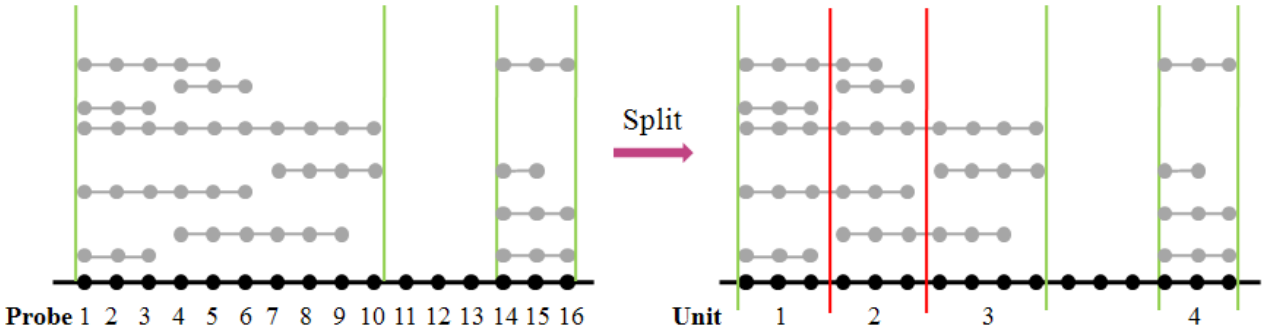

Figure 2.3 Illustration on how CNA units are constructed [Xiguo Yuan *et al*., 2012]

Each CNA unit is assigned a summary statistics, the so-called U-score, which is designed to incorporate both amplitude of CNA probes and recurrent frequency across multiple samples. So the U-score is mathematically given by:

$$U_{k,L} = \frac{1}{LM} \sum_{m=1}^{M} \sum_{l=k}^{k+L-1} x_{ml} \tag{22}$$

2.2.3 Null hypothesis Estimation and Significance Assessment

Given the observed U-score for a particular CNA unit, we want to know whether the observed U-score is significant enough that its associated CNA unit could be regarded as SCA under the null hypothesis. Random permutation is the most popular way to estimate the null hypothesis, and there are many different permutation strategies. In order to preserve the inherent correlation between adjacent CNA probes, the permutation in SAIC is based on the CNA units, not single CNA probe. It should be noted that since in SAIC there are CNA units of different length; we will generate groups of null hypothesis for CNA units of different lengths, and assess the significance of CNA unit of specific length based on the null hypothesis for the same length.

Figure 2.4 illustrates how does SAIC use CNA unit based permutation strategy to generate the null hypothesis for CNA units of different length.

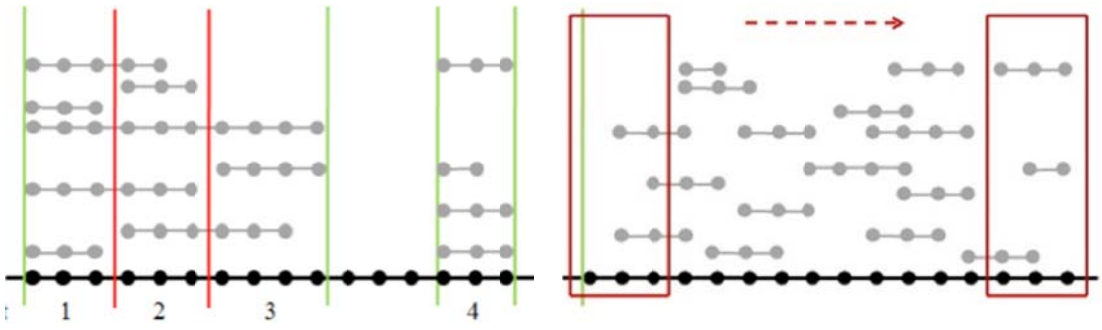

Figure 2.4 Illustration of SAIC permutation scheme and null hypothesis estimation

In Figure 2.4, the left figure shows the constructed CNA units from original copy number data matrix, after CNA unit based random permutation we get the new data matrix shown in the right figure. If we want to generate the null hypothesis for CNA unit of length 3, we will use a sliding window of length 3 to combine every 3 consecutive probes into a CNA unit and calculate the corresponding permutated U-score. The significance assessment is carried out like this:

1. Perform T times within sample permutation on CNA units, and get T permutated matrix $X_{p1,}X_{p2,}X_{p3,}\cdots X_{pT}$.
2. Construct CNA units of different length based on every permutated matrix, and calculate the associated summary statistics, $U_{k,L}\left(X_{pt}\right)$, where $t=1,2,3\cdots T$ and $k=1,2,3\cdots N-L+1$. It should be noted that $L\in\mathbb{L}$, where $\mathbb{L}$ is a set that contains all the lengths of the CNA units constructed using original data matrix.
3. Calculate P-value for every observed CNA unit $u(k,L)$ by counting the times that the maximum U-score of CNA units constructed with the same length in each permutated matrix is larger than the U-score of the observed CNA unit, given by:

$$P\left(U_{k,L}\left(X\right)\right)=\frac{1+\sum_{t=1}^{T}I\left(\max_{k'}U_{k',L}\left(X^{(t)}\right)\geq U_{k,L}\left(X\right)\right)}{T+1} \tag{23}$$

The "Max" operation in the P-value calculation is used to control the family wise error rate in multiple tests. If the calculated the P-value for U-scores associated with all the CNA units is smaller than the predefined significance level, this CNA unit will be regarded as a SCA.

However, just as we mentioned in Chapter 1, most existing algorithms use all the probes in the genome to estimate the null hypothesis. Thus both sporadic CNAs and SCAs will contribute to the estimation of null hypothesis. This will potentially reduce the detection power of the testing because the null hypothesis, in theory, should be estimated from copy number data with no SCAs, only sporadic CNAs. Having this in mind, in SAIC we introduce a SCA-excluding permutation scheme that iteratively lessens the influence of

SCAs to the null distribution estimation, and finally converge to the true null distribution. The procedure could be illustrated using the following flowchart in Figure 2.5.

1. First, all the constructed CNA units will be used to estimate the null distribution of the U-score, and based on the estimated null distribution, the significance of the CNA units will be tested according to their calculated P-values.
2. If there are any SCAs detected in the significance test, these CNA units will be excluded, and a new null distribution will be generated based on the remaining available CNA units. This process is repeated over and over again until there are no more SCAs detected, and this null distribution would be regarded as the final truth converging null distribution.
3. Finally, the P-values of the SCAs will be re-calculated based on the final truth converging null distribution.

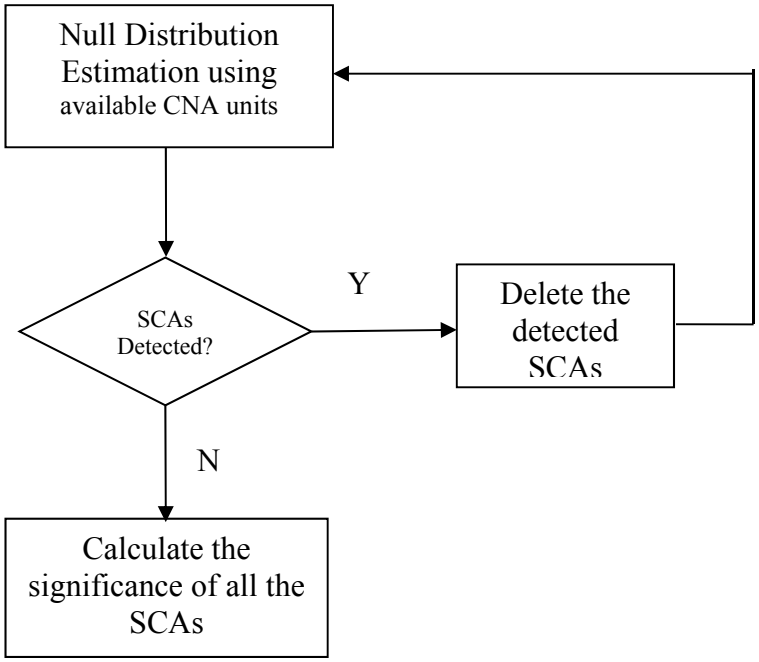

Figure 2.5 Flowchart of SCA-excluding scheme

In order to control the False Positive Rate (FPR) of the significance testing within a reasonable level α, we apply a significance level in each iteration as $\alpha' = \alpha/(1+\alpha)$, which could be proved as below:

Let α' be the significance level used in each iteration to detect SCAs in the SCA-excluding scheme. Under the truth converging null distribution, we have

$$\Pr\left(\mathrm{SCA}^{(r)}=\text{'yes'}\middle|\mathrm{SCA}^{(r-1)}=\text{'yes'}\right)=\alpha', \tag{24}$$

for iterations $r=1,2,...,\infty$ since SAIC assesses the 'new' SCAs at the *r*th iteration conditional on having found the 'existing' SCAs at the (*r*-1)th iteration.

Considering

$$\begin{aligned}&\Pr\left(\mathrm{SCA}^{(2)}=\text{'yes'}\right)\\&=\Pr\left(\mathrm{SCA}^{(2)}=\text{'yes'},\mathrm{SCA}^{(1)}=\text{'yes'}\right)\\&=\Pr\left(\mathrm{SCA}^{(2)}=\text{'yes'}\middle|\mathrm{SCA}^{(1)}=\text{'yes'}\right)\Pr\left(\mathrm{SCA}^{(1)}=\text{'yes'}\right)\\&=\alpha'\cdot\alpha'=\alpha'^{2}.\end{aligned} \tag{25}$$

Therefore for the *r*th iteration,

$$\begin{aligned}&\Pr\left(\mathrm{SCA}^{(r)}=\text{'yes'}\right)\\&=\Pr\left(\mathrm{SCA}^{(r)}=\text{'yes'},\mathrm{SCA}^{(r-1)}=\text{'yes'},...,\mathrm{SCA}^{(1)}=\text{'yes'}\right)\\&=\Pr\left(\mathrm{SCA}^{(r)}=\text{'yes'}\middle|\mathrm{SCA}^{(r-1)}=\text{'yes'},\mathrm{SCA}^{(r-2)}=\text{'yes'},...,\mathrm{SCA}^{(1)}=\text{'yes'}\right)\\&\cdot\Pr\left(\mathrm{SCA}^{(r-1)}=\text{'yes'}\middle|\mathrm{SCA}^{(r-2)}=\text{'yes'},\mathrm{SCA}^{(r-3)}=\text{'yes'},...,\mathrm{SCA}^{(1)}=\text{'yes}\right)\\&\quad\cdot...\cdot\Pr\left(\mathrm{SCA}^{(2)}=\text{'yes'}\middle|\mathrm{SCA}^{(1)}=\text{'yes'}\right)\Pr\left(\mathrm{SCA}^{(1)}=\text{'yes'}\right)\\&=\Pr\left(\mathrm{SCA}^{(r)}=\text{'yes'}\middle|\mathrm{SCA}^{(r-1)}=\text{'yes'}\right)\cdot\Pr\left(\mathrm{SCA}^{(r-1)}=\text{'yes'}\middle|\mathrm{SCA}^{(r-2)}=\text{'yes'}\right)\\&\quad\cdot...\cdot\Pr\left(\mathrm{SCA}^{(2)}=\text{'yes'}\middle|\mathrm{SCA}^{(1)}=\text{'yes'}\right)\Pr\left(\mathrm{SCA}^{(1)}=\text{'yes'}\right)\\&=\alpha'\cdot\alpha'\cdot\alpha'\cdots\alpha'=\alpha'^{r}.\end{aligned} \tag{26}$$

The rationale behind the above derivation is that $\mathrm{SCA}^{(r-1)}=\text{'yes'}$ already implies:

$\mathrm{SCA}^{(r-2)}=\text{'yes'},...,\mathrm{SCA}^{(1)}=\text{'yes'}$. In other words, we have:

$$\Pr\left(\mathrm{SCA}^{(r)}=\text{'yes'}\right)=\Pr\left(\mathrm{SCA}^{(r)}=\text{'yes'},\mathrm{SCA}^{(r-1)}=\text{'yes'},...,\mathrm{SCA}^{(1)}=\text{'yes'}\right)$$

$$\Pr\left(\mathrm{SCA}^{(r)}=\text{'yes'}\middle|\mathrm{SCA}^{(r-1)}=\text{'yes'},\mathrm{SCA}^{(r-2)}=\text{'yes'},...,\mathrm{SCA}^{(1)}=\text{'yes'}\right)=\Pr\left(\mathrm{SCA}^{(r)}=\text{'yes'}\middle|\mathrm{SCA}^{(r-1)}=\text{'yes'}\right).$$

Let α be the targeted FPR, we have

$$\begin{aligned}\alpha &= \sum_{r=1}^{\infty} \Pr\left(\text{SCA}^{(r)}=\text{'yes'}\right) \\ &= \alpha' + \alpha'^{2} + \ldots + \alpha'^{r} + \ldots = \frac{\alpha'}{1-\alpha'}, \qquad \left(\alpha' < 1\right).\end{aligned} \tag{27}$$

Accordingly, we have $\alpha' = \alpha / \left(1+\alpha\right)$.

Chapter 3

Implementation of AISAIC in Java

In order to better serve the research community, we develop a Java AISAIC package, an open-source cross-platform software package with friendly graphical user interface (GUI) to implement the whole pipeline of copy number analysis, including normal tissue contamination correction using BACOM and significant copy number aberrations detection using SAIC. However considering that both BACOM and SAIC could also be used as an independent copy number analysis tool, we provide three options “Only BACOM”, “Only SAIC” and “BACOM+SAIC” to allow users to choose the specific analysis they need. Figure 3.1 shows the overview of the AISAIC software package. Users can feed their input data along with the appropriate parameters to the AISAIC analysis module from the GUI. And based on users’ analysis option, AISAIC package will invoke corresponding analysis module and output the results.

3.1 Concurrent Computing

One major feature of AISAIC is concurrent computing, which is widely employed in the package. Considering the characteristics of copy number data and our proposed approach, and also the computing

power provided by multi-core computers and clusters, we adopt concurrent computing using Java Fork/Join API [24] to improve the efficiency of the AISAIC software package.

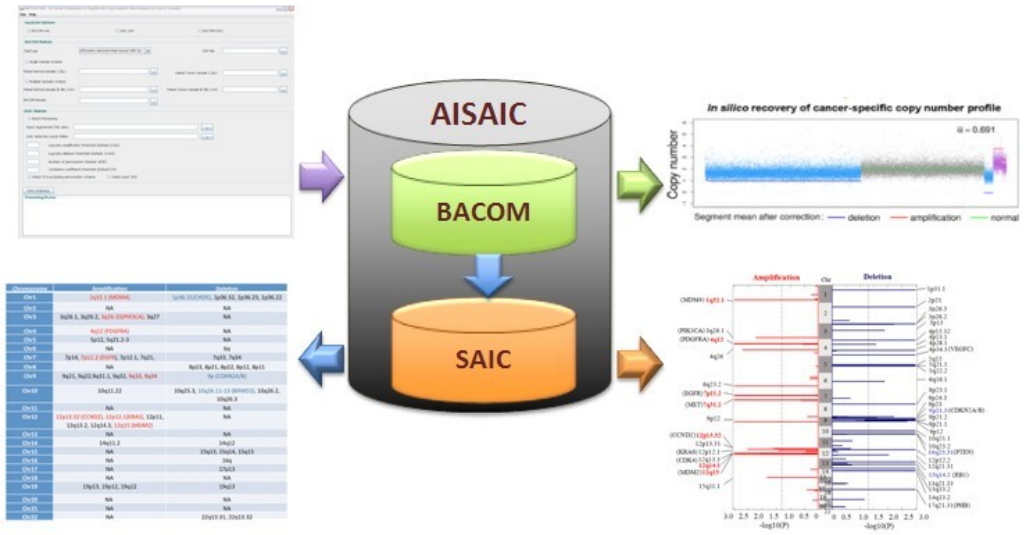

Figure 3.1 Overview of the AISAIC package

3.1.1 Basic Principle of Concurrent Computing of Java Fork/Join API

Java platform is designed to support concurrent programming. Since Java version 5.0, is has included high-level concurrency APIs in the java.util.concurrent packages. There are several ways to implement concurrent computing. However, considering the characteristics of our algorithm we specifically use Fork/Join API to implement concurrent computing.

Fork/Join API is new in Java SE 7, it is designed for work that can be divided into smaller sub-works recursively. It uses a divide-and-conquer scheme. The core of Fork/Join framework is the “ForkJoinPool” class, which uses a “work-stealing algorithm” to make best use of the computing resources. To be specific, the worker that is done with its job can “steal” tasks from other threads that are still busy. The “ForkJoinPool” object can execute a “ForkJoinTask”, which has two major methods: “fork()” and “join()”. Figure 3.2 illustrates how the “ForkJoinTask” works using “fork()” and “join()”. The original ForkJoinTask uses “Fork()” to divide the task into two sub-ForkJoinTasks, and the two sub-ForkJoinTasks

may be further divided until certain condition is satisfied. When the sub-ForkJoinTasks complete their jobs, they will join their results (for RecursivelTask) and return to the parent task.

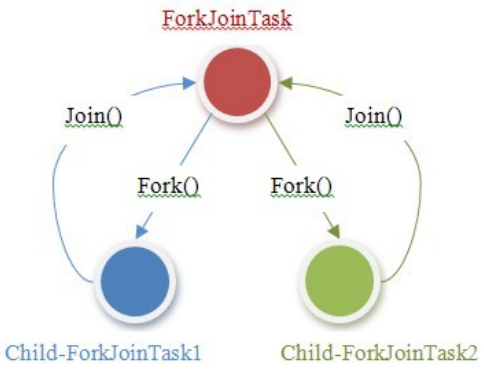

Figure 3.2 Illustration of how “ForkJoinTask” works

There are two subclasses of “ForkJoinTask”: one is “RecursiveTask”, instances of which requires every sub-workers return a result; the other one is “RecursiveAction”, instances of which requires no return value from the sub-ForkJoinTasks.

3.1.2 The Use of Fork/Join API in Software Package Design

In our algorithm, especially the SAIC part, we can adopt concurrent computing in several places.

First of all, we can analyze two consecutive chromosomes simultaneously. In SAIC we do chromosome-wise SCAs detection because different chromosomes often have different rates of background sporadic copy number alterations. Therefore we can use concurrent computing to analyze two consecutive chromosomes simultaneously because different chromosomes are independent from each other in analysis. It should be noted that, we can analyze more than 2 chromosomes at the same time, but it may make the memory a stringent resources because each time we need to load data from more chromosomes. There is another reason for not analyzing too many chromosomes together. It is known that from chromosome 1 to chromosome 23, the length decreases accordingly, thus longer chromosomes may take much more time than shorter chromosomes. If we analyze too many chromosomes simultaneously, the shorter one still needs to wait for the longer one to finish before they can move on to the next group.

Second, we detect amplified SCAs and deleted SCAs simultaneously. Just as introduced in Chapter 2, there are both copy number amplification and deletion, and we detect them separately based on different matrices derived from the original input data matrix. Therefore we can create two sub-workers that work on the two independent data matrix with no interference to each other.

Third, we also apply concurrent computing to the permutation step, which is used to generate null hypothesis. The permutation is sample-wise operation, i.e., different samples are processed independent from each other in permutation. Therefore, we could divide all the samples into several groups with less samples recursively until the sample size for every sub-work is small enough. For example, in one of our real data application, we have 485 samples, and we divide these samples into smaller chunks, and each chunk contains less than 30 samples.

3.1.3 Performance Gain from Concurrent Computing

After applying concurrent computing in our software development, the efficiency of the software improves significant. Figure 3.3 shows the resource consumption of the software package before and after concurrent computing. We can see that before using concurrent computing only one processor is used while all the remaining processors are just idle there. This is a huge waste of computing resources. However after using concurrent computing, all the processors work at their best effort, almost all around 90% usage. One thing should be noticed that the memory cost doubled after using concurrent computing, because each time, the copy number data matrix of two chromosomes are loaded into the memory. If we simultaneously do more chromosomes, it may consume more memory. In other word, this is a compromise between time and memory.

Figure 3.3 Resources consumption comparison before (up) and after (down) concurrent computing

3.2 Graphical User Interface

In our software package, we also provide a Graphical User Interface (GUI) using Java Swing API [25] to the end users for their convenience use of our software package. Figure 3.4 below shows a blank GUI of our BACOM+SAIC package. It contains four major blocks:

1. The first block is "Analysis Option", where the users can choose the analysis they want from the three options: "BACOM only", "SAIC only" and "BACOM+SAIC". Just as their names suggest, "BACOM only" will only estimate the normal tissue contamination of certain sample or group of samples as user's choice. "SAIC only" will only detect the SCAs using SAIC based on the copy number data matrix after normalization and segmentation. "BACOM+SAIC" option will work through the pipeline from the raw data (.CEL) to the detected SCAs.

2. The second block is "BACOM module", where the users need to choose the "ChipType" indicating the right chip from which the data is obtained, and the corresponding "CDF file". And in BACOM analysis, we provide two options: "Single Sample Analysis" and "Multiple Samples Analysis". For single sample analysis, the users need to choose paired normal sample file (.CEL) along with the paired tumor sample file (.CEL). However, if they choose multiple samples analysis, they need to select a normal sample ID file (.txt) that contains the normal sample file names along with their matched tumor sample ID file (.txt) that contains the corresponding paired tumor sample file names. It should be noted if "BACOM+SAIC" option is chosen in "Analysis Option", it is highly suggested to choose "Multiple Sample Analysis" for the convenience of the users. Finally, the results of BACOM analysis are stored in a user chosen directory indicated in "BACOM results".

3. The third block is "SAIC module". If the users already choose "SAIC only" in "Analysis Option", then they need to choose input segmented copy number data from "Input Segmented

CNA data". If "BACOM+SAIC" option is chosen, the SAIC module will use the extracted the "true" copy number data matrix after BACOM analysis as input, and store the results in a user-chosen folder directory from "SAIC detection result folder". Since SAIC need certain parameters in the analysis, the users could set the parameters according to their knowledge about the data.

a. "Log-ratio amplification/deletion thresholds": If the users have chosen "BACOM+SAIC", the suggested values for the two parameters are 0.322/-0.415, which corresponds to 2.5/1.5 before logarithm. The thresholds will not only influence the detection results, but also affect the time consumption of the analysis. The larger the absolute values of the thresholds are, the less the number of CNA probes and the less time needed for analysis.

b. "Number of Permutation": The default number of permutation is 1000, and the user can change this parameter as their wish. Basically, the larger the number is, the more accurate the results would be, but at the same time, the longer the analysis will take.

c. "Correlation coefficient threshold": This parameter is used in CNA unit construction based on the correlation coefficient, and the default value is 0.8. Higher value may result in more independent CNA units, and more time consumption accordingly.

d. "Adopt SCA-excluding permutation scheme": SCA-excluding permutation scheme is one of the novel features of SAIC. In case the users don't want to use this scheme, we provide this option to allow users to either choose it or not.

e. "Adopt quick SAIC": If the users want even faster speed, they can choose this option. In order to further shorten the time consumption, we apply a down sampling to the original data matrix to shrink the size of the data set. For each sample we pick one probe every three probes to represent these three probes because in most cases,

consecutive probes always have similar or even equal value. And our experiments show that this down sampling rate can provide 5-time faster speed without affect the results significantly.

4. The last block is “Processing Status”. After users clicking the “Start Analysis” button, the program will start to run, and the users could monitor the status of the analysis in this block, and we also print out the time consumption of the analysis.

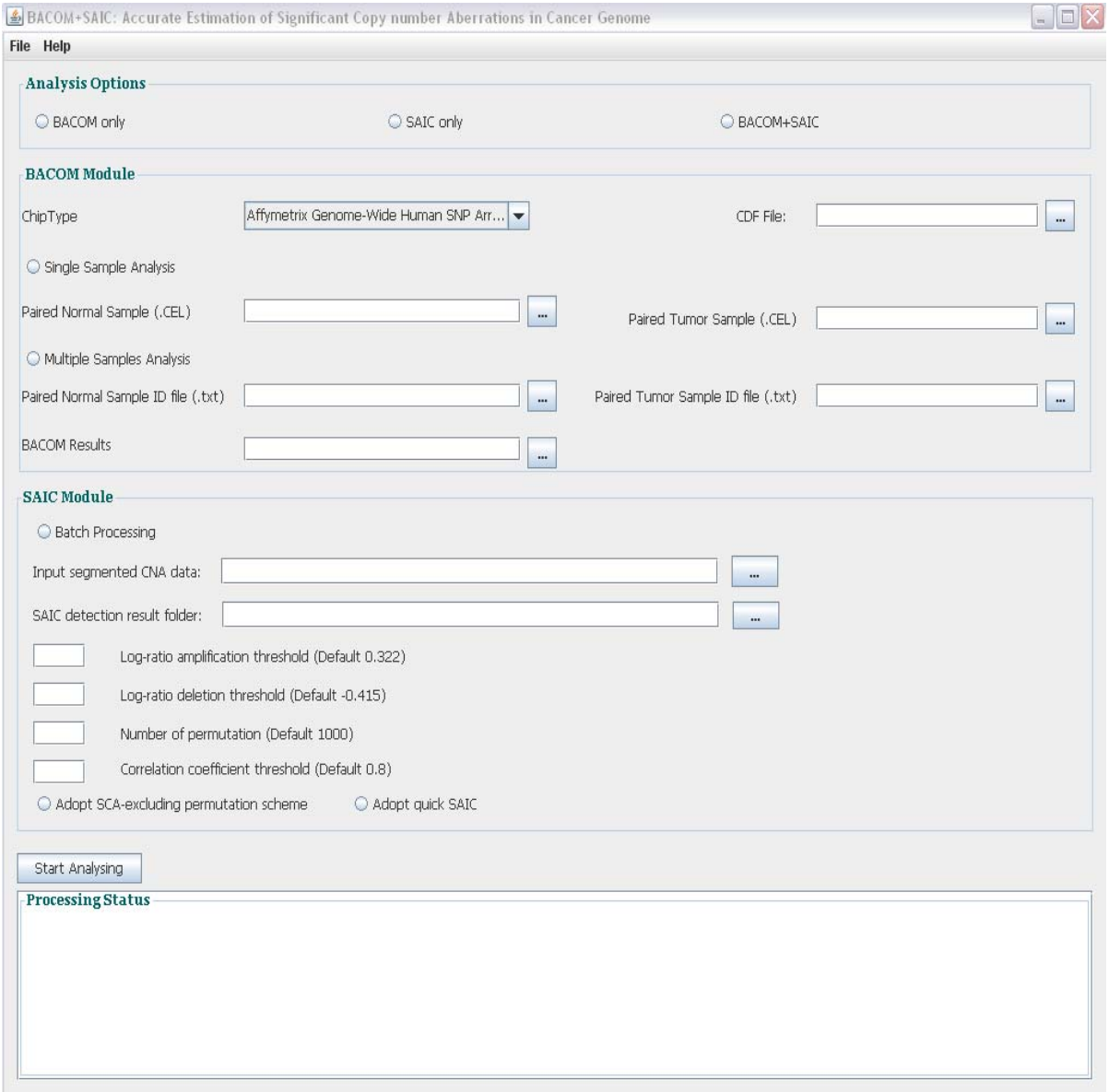

Figure 3.4 Example of a blank GUI

Chapter 4

Software Evaluation and Application

In this section, we evaluate the performance of the AISAIC software package from several aspects. Since we do not have ground truth about the recurrent CNAs in real cancer genomes, we will generate large number of realistic simulation data to evaluate the performance of the new software package.

4.1 Simulation Evaluation

4.1.1 Empirical Family-Wise Type I Error Rate

Type I error, also known as false positive rate occurs when a null hypothesis is true but is rejected in a single hypotheses test. In multiple hypotheses tests, since there is more than one hypotheses test, we usually use family-wise error rate (FWER) to evaluate the probability of making one or more false positive. It is critical in detecting SCAs based on P-value. As a matter of fact, type I error equals to significance level numerically. If the empirical FWER is smaller than the theoretical one, it may suggest that the test method may be not sensitive enough, i.e., may have lower detection power; if the empirical

FWER is larger than the theoretical one, it suggests that the method is lack of specificity, i.e., may make more false positive. Only when the empirical FWER is close to the theoretical one, we can be confident about the detected SCAs using P-value.

Therefore in this section, we will first test the empirical FWER of our proposed approach based on groups of simulated null data that no SCAs are present using approach used in [13]. The null data is generated as below:

1. Each data set contains 50 samples, and each sample contains 2000 markers. Therefore we generate a 2000*50 matrix $X = ((G1 - G2) * S) + N$, where $G1$ and $G2$ are independently generated using the instability selection model [26] with different parameters, and the markers are equally spaced on the interval (0, 1). Note that $G1 - G2$ represents a matrix of idealized copy number data whose entries in a given row corresponds to a Markov chain with three states: copy number = -1, 0, 1 to simulate the copy number deletion, normal and amplification, and the transition probability for the Markov chain are derived from the instability selection model. In order to simulate the normal tissue contamination in the tumor samples, a multiplication by column vector S is used. Since different samples may have different normal tissue contamination, the entries in *S* are randomly generated from a random number generation ranging from 0.1 to 0.8. Finally, we add some random noise *N* following i.i.d normal distribution with mean 0 and standard deviation 0.25 to mimic the noise in the real data set.
2. However, considering the fact that in real data the markers may not distributed evenly and the correlation between the consecutive markers may be different because of the distance, we also generate a variant of matrix *X* where the matrix $G1$ and $G2$ are generated using a common set of unequally spaced markers. A certain fraction of the markers, ranges from 25% to 100% are contained in the eight equally spaced clumps of size 0.25. The remaining markers are just uniformly distributed in the remaining intervals in (0, 1). We call this kind of data "Clumped copy number data" with a certain fraction.

Based on the approach introduced above, we generate 4 groups of copy number data: "Copy number data" with evenly placed markers, and "Clumped copy number data" with different clump fraction. For each group, we generate 10,000 data sets. The empirical FWER was calculated as the proportion of 10,000 null datasets for which there are SCAs detected with significance level $\alpha = 0.05$. The result is shown in Table 4.1. The average type I error of our method is 0.0516, which is within the 95% confidence interval of the theoretical value. Therefore, we can be confident about the SCAs detected using our approach.

Table 4.1 Empirical type I error rate for simulated data sets under the null hypothesis.

Null simulation model	Empirical FWER at $\alpha = 0.05$ level
Copy number data	0.0516
Clumped copy number data (25%)	0.0490
Clumped copy number data (50%)	0.0520
Clumped copy number data (75%)	0.0540
Average	0.0516

4.1.2 Detection Power Evaluation

We then investigate the detection power of our method by comparing it with GISTIC, the most popular approach for SCAs detection. Using the simulation model proposed in [22], we generate data set with sample size from *N*=40~80, and each sample with *M*=5000 markers. To simulate the normal tissue contamination, we generate copy number signal for each probe as the weighted sum of normal and tumor copy number value, where the normal tissue contamination fraction β is simulated by randomly drawing from a normal distribution $N(\mu_\beta, \sigma_\beta)$ with mean μ_β and standard deviation σ_β. In each sample, we randomly insert two sporadic CNA regions and two SCAs (one deletion and one amplification events). The recurrence of SCAs across multiple samples is determined by a predefined parameter, frequency *f*. The baseline parameter setting in our simulation is: *N*=60, *M*=5000, f=0.2, μ_β=0.6, σ_β=0.25.We define a successful detection as the one that detects both amplified and deleted SCAs. Therefore, the detection power is calculated as the proportion of successful detection in 100 data sets.

In order to comprehensively compare the detection power of our method with GISTIC, we carried out 4 groups experiments. For each one, we change one of the parameters while fixing the remaining, and compare the detection power of our method with GISTIC and SAIC. The results are shown below in Figure 4.1, Figure 4.2, Figure 4.3 and Figure 4.4.In each figure, there are four lines that correspond to GISTIC, SAIC, BACOM+GISTIC and BACOM+SAIC respectively. Figure 4.1 shows the detection powers of the four methods when we increase standard deviation from 0.15 to 0.35 while fixing the other parameters. Figure 4.2 presents the detection power comparison when we increase the mean from 0.4 to 0.8 while fixing the other parameters. From Figure 4.1 and Figure 4.2 we can clearly see that as the standard deviation and mean of normal tissue contamination fraction increases, the detection power of both GISTIC and SAIC decreases, but SAIC is superior to GISTIC in most cases. More importantly, with normal tissue contamination correction, both GISTIC and SAIC become immune to the change of normal tissue contamination, and both have good and stable performance. Figure 4.3 plots the detection powers when we increase the frequency from 0.1 to 0.25 while fixing the other parameters, and Figure 4.4 plots the detection power comparison when we increase the sample size from 40 to 80 while fixing the other parameters. From Figure 4.3 and Figure 4.4 we can see that as the frequency and sample size increase, the detection power of all four increase accordingly. In addition, with BACOM help, the performance of both GISTIC and SAIC improves significantly. In all of the situations, our proposed approach BACOM+SAIC always has the best performance.

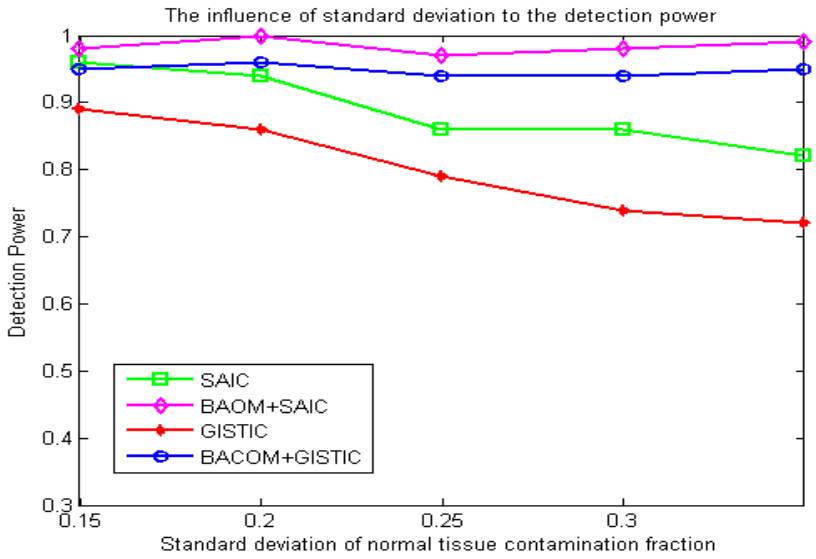

Figure 4.1 The comparison of detection power between proposed method and GISTIC when we change standard deviation of normal tissue contamination fraction from 0.15 to 0.35.The pink line with diamond shows the detection power of BACOM+SAIC. The blue line with circle on it shows the detection power of BACOM+GISTIC. The green line with square shows the detection power of SAIC. The red line with star on it shows the detection power of GISTIC.

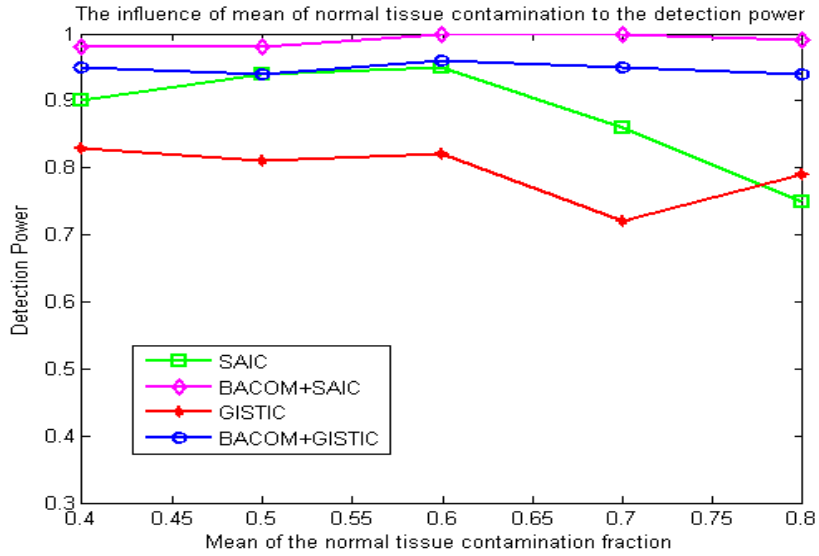

Figure 4.2 Comparison of detection power between proposed method and GISTIC when we change mean of normal tissue contamination fraction from 0.4 to 0.8.The pink line with diamond shows the detection power of BACOM+SAIC. The blue line with circle on it shows the detection power of BACOM+GISTIC. The green line with square shows the detection power of SAIC. The red line with star on it shows the detection power of GISTIC.

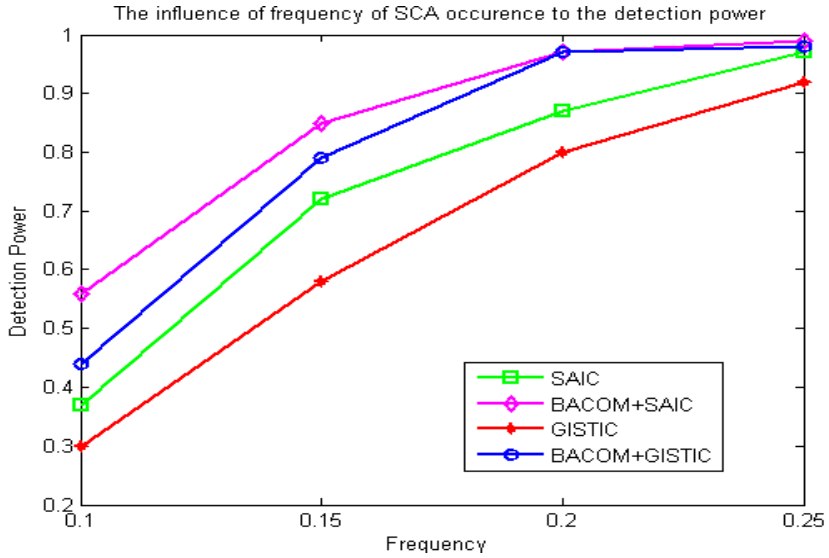

Figure 4.3 Comparison of detection power between proposed method and GISTIC when we change recurrent frequency from 0.1 to 0.25.The pink line with diamond shows the detection power of BACOM+SAIC. The blue line with circle on it shows the detection power of BACOM+GISTIC. The green line with square shows the detection power of SAIC. The red line with star on it shows the detection power of GISTIC.

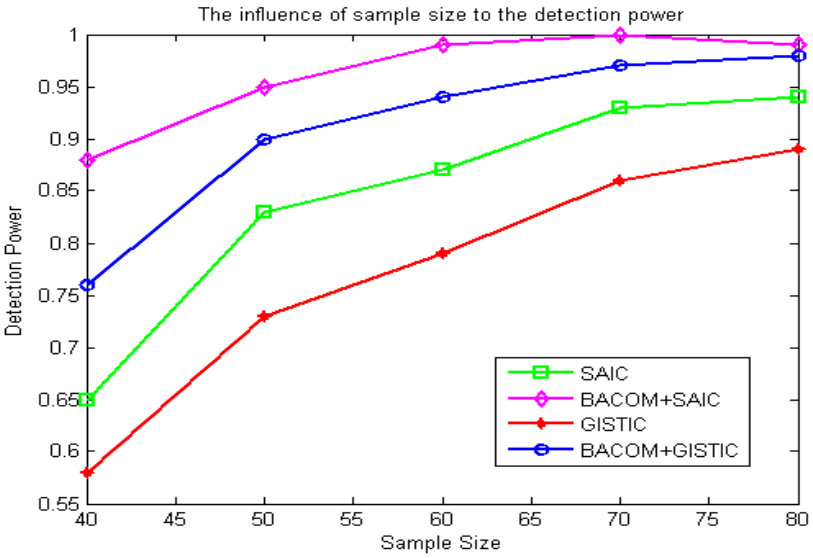

Figure 4.4 Comparison of detection power between proposed method and GISTIC when we change sample size from 40 to 80.The pink line with diamond shows the detection power of BACOM+SAIC. The blue line with circle on it shows the detection power of BACOM+GISTIC. The green line with square shows the detection power of SAIC. The red line with star on it shows the detection power of GISTIC.

4.2 Application to Real Data Set

We apply our method to real copy number data of Glioblastoma Multiforme (GBM), the most common and most aggressive malignant primary brain tumor in humans. The data set we use contains N=103 tumor samples along with their corresponding paired normal samples, which are downloaded from TCGA portal. The samples are obtained by Affymetrix Genome-wide SNP array 6.0, which contains about 1.8 million probes, including 906,600 SNPs and 946,000 copy number probes.

We first estimate the normal tissue contamination fraction of the 103 tumor samples one by one, and extract their "true" tumor copy number profile using BACOM. The result shows that the normal tissue contamination fraction of these GBM samples ranges from 0.33 to 0.96, with sample mean of 0.61 and sample standard deviation of 0.13. The detail of the normal tissue contamination analysis is shown below in Table 4.2.

Table 4.2 The statistic summary of BACOM analysis on GBM data

Statistic Summary	Mean	Median	Max	Min	Std
GBM	0.60905	0.69683	0.95995	0.32759	0.12562

Then we use SAIC to detect the SCAs with the corrected tumor samples. Since we have recovered the "true" copy number profile, we can just set the log-ratio thresholds for copy number amplification and deletion detection as $\theta_{amp} = 0.322$ and $\theta_{del} = -0.415$ respectively.

In total, our method detected 34 amplified SCAs and 31 deleted SCAs (after combining some SCAs within the same cytobands), listed in Table 4.3. These detected SCAs encompass majority of reported oncogenes and tumor suppressor genes (TSG) in GISTIC, such as EGFR(7p11.2), PDGFR(4q12), MDM4(1q32.1), MDM2(12q15), KRAS(12p12.1), PIK3CA(3q26.33), CDKN2A/B (9p21.3), RB1(13q14.2), etc. In addition, we also detect tumor-associated genes that are not reported by GISITC, such as amplification of AKT3(1q43), which is involved in a wide variety of biological processes, including cell proliferation, differentiation, apoptosis and tumorigenesis etc., [27] and deletion of BRWD2(10q26.12), which is involved in cell cycle progression, apoptosis and gene regulation [28].

Besides, many additional newly detected SCAs (e.g., 9q33-34, 8p23, etc.) may also worth further studying.

Table 4.3 The detected SCAs in GBM data using proposed approach

Chromosome	Amplification	Deletion
Chr1	1q32.1 (MDM4)	1p36.31(CHD5), 1p36.32, 1p36.23, 1p36.22
Chr2	NA	NA
Chr3	3q26.1, 3q26.2, 3q26.33(PIK3CA), 3q27	NA
Chr4	4q12 (FDGFRA)	NA
Chr5	5p12, 5q21.2-3	NA
Chr6	NA	6q
Chr7	7p14, 7p11.2 (EGFR), 7p12.1, 7q21,	7q33, 7q34
Chr8	NA	8p23, 8p21, 8p22, 8p12, 8p11
Chr9	9q21, 9q22,9q31.1, 9q32, 9q33, 9q34	9p (CDKN2A/B)
Chr10	10q11.22	10q25.3, 10q26.11-13 (BRWD2), 10q26.2, 10q26.3
Chr11	NA	NA
Chr12	12p13.32 (CCND2), 12p12.1(KRAS), 12p11, 12q13.2, 12q14.3, 12q15 (MDM2)	NA
Chr13	NA	NA
Chr14	14q11.2	14q12
Chr15	NA	15q13, 15q14, 15q15
Chr16	NA	16q
Chr17	NA	17p13
Chr18	NA	NA
Chr19	19p13, 19p12, 19q12	19q13
Chr20	NA	NA
Chr21	NA	NA
Chr22	NA	22q13.31, 22q13.32

Chapter 5

Discussion and Future Work

5.1 Discussion

Significant Copy number Aberrations (SCAs) detection is a critical yet challenging task in cancer research; it tries to distinguish the "driver" copy number aberrations from background sporadic "passenger" alterations that are randomly acquired during the tumor progression. However tissue heterogeneity, which is widely existed in real data set, introduces a new confounding factor to the SCAs detection. Because in most real data sets, different samples always have different degree of normal cell contaminations, and this discrepancy adds extra randomness to the data sets in addition to the randomness of background sporadic copy number alterations. This will potentially influence the detection power of SCAs detection. Therefore in this thesis we introduce a two-step based statistical approach that incorporates two recent novel algorithms in the literature to accurately identify the significant copy number aberrations in contaminated cancer genome. First, in order to minimize the adverse influence of normal tissue contamination, the Bayesian Analysis of Copy number Mixtures (BACOM) is used to extract the "true" tumor copy number profile from the raw copy number intensity. Then based on the

multiple corrected tumor samples, we use the Genome-wide Identification of Significant Aberrations in Cancer Genome (SAIC) [16], a carefully designed statistical method to detect the SCAs that potentially contain oncogenes and/or tumor suppressor genes. The SAIC algorithm has three novel features that tackle the limitations in other similar algorithms. First of all, considering the correlation between adjacent probes, SAIC constructs CNA units based on the correlation between consecutive probes, and assign a summary statistic to every CNA unit, not single probe; secondly in order to preserve the inherent correlation between neighboring probes, SAIC perform a random positional permutations on CNA units; thirdly a carefully designed SCA-excluding permutation scheme is adopted to exclude the effect of SCAs to the estimation of null distribution.

In order to better serve research community, we implement the introduced approach in a Java software package AISAIC (Accurate Identification of Significant Aberrations in Cancer) with a friendly graphical user interface (GUI). In order to improve the efficiency of the software, we adopt concurrent computing using Java Fork/Join API in the development. For example, the AISAIC package can complete the whole pipeline analysis of a data set with 103 samples within about 4 hours on a 12-core computer.

We evaluate the performance of the proposed approach with a large number of simulation data from two aspects. First we test the empirical family-wise type I error rate (FWER) of the new approach using null simulation data. And we get average FWER as 0.0516, which is within the 95% confidence interval of the theoretical value 0.05. Then we compare the detection power of the proposed approach with two peer methods GISTIC and SAIC on multiple groups of simulation data set, and the new approach provides better performances on almost all the data set.

Finally, we use AISAIC package to analyze a real brain tumor data set from TCGA portal. We detect many SCAs that cover majority of cancer-associated genes that reported by other methods, and more importantly we also detect some new regions that may worth further study.

5.2 Future Work

In the entire genome, the Significant Copy number Aberrations (SCAs) could either be very short or very long regions that may cover the chromosome arm or whole chromosome. The former type is focal SCAs, while the latter one is called arm-level SCAs. It is reported that in a typical tumor sample, 25% of the genome is affected by arm-level SCAs while 10% by focal SCAs [2]. However the arm-level SCAs often cover too many genes, and this makes it too difficult to determine the specific target genes. On the contrary, focal SCAs can pinpoint more accurate target genes. Therefore one of the future work may be how to filtering out the arm-level SCAs leaving only focal SCAs. Since focal SCAs often have higher amplitude than arm-level SCAs, we may set higher thresholds to exclude arm-level copy number alterations. However this approach may also exclude some focal SCAs that are significantly consensus but has relatively lower amplitude. Another approach to distinguish between arm-level SCAs and focal SCAs may be purely based on their length. We may incorporate this arm-level SCAs excluding module in future AISAIC software package.

Significant Copy numbers Aberrations (SCAs) are supposed to occur in a large portion of the tumor samples of the same type. However it is observed that some detected SCAs appear only in about 10%~20% of the total samples, and one possible explanation may be that these SCAs are only recurrent across a certain subgroup of all the tumor samples, i.e., tumor has subtypes. As a matter of fact, this intra-tumor heterogeneity has long been noticed substantial effort has been devoted to this research.

It is regarded that “Carcinogenesis is initiated by the accumulation of genetic mutations in a single cell and is driven by the emergence of further genetic and epigenetic alterations that confer more aggressive, invasive, and drug-resistant phenotypes” [29]. During tumor progression, the emergence of multiple new and malignant genetic mutations gives rise to remarkable variability in almost all the distinguishable phenotypic traits, such as cellular morphology, cell surface receptors, metabolism, proliferative, metastatic potential, angiogenic, metastatic potential and response to therapy etc. [30-32]. Copy number alteration, as a major genetic structural variation in cancer initiation and progression could be used as a

feature to cluster the tumor samples into different subtypes. However the high dimension (1.8 million probes) of copy number data makes this effort almost infeasible. As an alternative, significant copy number aberrations (SCAs) may serve as a potential feature to cluster cancer samples since different subgroups of the same cancer type may foster different SCAs that confer different phenotypes. The benefits of using SCAs as clustering features are at least two folds: first SCAs has much smaller dimension comparing to pure copy number data, which make this problem approachable; second since SCAs are supposed to contain "driver" mutations, they may offer higher distinguishing power. Actually some initial effort has been put into this endeavor and in depth exploration is still in urgent need.

REFERENCES

[1] Stephen W Scherer, Charles Lee, Ewan Birney, et al., "Challenges and standards in integrating surveys of structural variation". *Nature Genetics.* vol 39, June 2007

[2] Beroukhim R, Mermel CH, Porter D, Wei G, Raychaudhuri S, Donovan J, Barretina J, Boehm JS, Dobson J, Urashima M et al,"The landscape of somatic copy-number alteration across human cancers". *Nature* 2010, 463(7283): 899-905.

[3] Leary RJ, Lin JC, Cummins J, Boca S, Wood LD, Parsons DW, Jones S, Sjoblom T, Park BH, Parsons R et al,"Integrated analysis of homozygous deletions, focal amplifications, and sequence alterations in breast and colorectal cancers". *Proc Natl Acad Sci USA* 2008, 105(42): 16224-16229.

[4] Wood LD, Parsons DW, Jones S, Lin J, Sjoblom T, Leary RJ, Shen D, Boca SM, Barber T, Ptak J et al,"The genomic landscapes of human breast and colorectal cancers". *Science* 2007, 318(5853): 1108-1113.

[5] McCarroll SA, Altshuler DM. "Copy-number variation and association studies of human disease". *Nature Genetics*, vol 39, July 2007

[6] Ludmila Shostakovich-Koretskaya, et al., "Combinatorial content of CCL3L and CCL4L gene copy number influence HIV-AIDS susceptibility in Ukrainian children". *PMC,* 2010

[7] Lupski JR. "Genomic rearrangements and sporadic disease". *Nature Genetics*. vol 39, July 2007

[8] Davies JJ, Wilson IM, Lam WL, "Array CGH technologies and their applications to cancer genomes", *Chromosome Research* 2005; 13(4): 423

[9] Zhao X, Li C, Paez JG, Chin K, Janne PA, Chen TH, Girard L, Minna J, Christiani D, Leo C et al,"An integrated view of copy number and allelic alterations in the cancer genome using single nucleotide polymorphism arrays". *Cancer Res* 2004, 64(9): 3060-3071.

[10] Pinkel D,Albertson DG,"Array comparative genomic hybridization and its applications in cancer" . *Nat Genet* 2005, 37 Suppl: S11-17.

[11] Beroukhim R, Getz G, Nghiemphu L, Barretina J, Hsueh T, Linhart D, Vivanco I, Lee JC, Huang JH, Alexander S et al,"Assessing the significance of chromosomal aberrations in cancer: methodology and application to glioma". *Proc Natl Acad Sci U S A* 2007, 104(50): 20007-20012.

[12] Diskin SJ, Eck T, Greshock J, Mosse YP, Naylor T, Stoeckert CJ, Jr., Weber BL, Maris JM,Grant GR,"STAC: A method for testing the significance of DNA copy number aberrations across multiple array-CGH experiments". *Genome Res* 2006, 16(9): 1149-1158.

[13] Walter V, Nobel AB,Wright FA,"DiNAMIC: a method to identify recurrent DNA copy number aberrations in tumors". *Bioinformatics* 2011, 27(5): 678-685

[14] Zhang Q, Ding L, Larson DE, Koboldt DC, McLellan MD, Chen K, Shi X, Kraja A, Mardis ER, Wilson RK et al,"CMDS: a population-based method for identifying recurrent DNA copy number aberrations in cancer from high-resolution data". *Bioinformatics* 2010, 26(4): 464-469.

[15] Mermel CH, Schumacher SE, Hill B, Meyerson ML, Beroukhim R,Getz G,"GISTIC2.0 facilitates sensitive and confident localization of the targets of focal somatic copy-number alteration in human cancers". *Genome Biol* 2011, 12(4): R41.

[16] Xiguo Yuan, Guoqiang Yu, Xuchu Hou, Le-Ming Shih, Yue Wang, et al., "Genome-wide Identification of Significant Abberrations in Cancer Genome", *BMC Genomics*. July 27, 2012

[17] C. Rouveirol, et al., "Computation of recurrent minimal genomicalterations from array-CGH data," *Bioinformatics,* vol. 22, pp. 849-56, Apr 1, 2006

[18] Rueda OM,Diaz-Uriarte R,"Finding Recurrent Copy Number Alteration Regions: A Review of Methods"*Current Bioinformatics* 2010, 5: 17.

[19] Clark R., Ressom, H.W., Wang, A., et al. "The properties of high-dimensional data spaces: implications for exploring gene and protein expression data*. Nat Rev Cancer*, 8(1), 37–49.

[20] Ivakhno S,Tavare S, "CNAnova: a new approach for finding recurrent copy number abnormalities in cancer SNP microarray data". *Bioinformatics* 2010, 26(11): 1395-1402.

[21] Klijn C, Holstege H, de Ridder J, Liu X, Reinders M, Jonkers J,Wessels L,"Identification of cancer genes using a statistical framework for multiexperiment analysis of nondiscretized array CGH data". *Nucleic Acids Res* 2008, 36(2): e13.

[22] Yu G, Zhang B, Bova GS, Xu J, Shih IM,Wang Y, "BACOMin silico detection of genomic deletion types and correction of normal cell contamination in copy number data". *Bioinformatics* 2011, 27(11): 1473-1480.

[23] H. Bengtsson, R. Irzarry, B. Carvalho, T.P.Speed, "Estimation and assessment of raw copy numbers at the single locus level", *Bioinformatics* 2008, vol. 24(6): 759-767.

[24] Java Fork/Join API,http://docs.oracle.com/javase/tutorial/essential/concurrency/forkjoin.html

[25] The Java Swing API tutorial: http://docs.oracle.com/javase/6/docs/technotes/guides/swing/

[26] Michael A. Newton, Michael N. Gould, Catherine A. Reznikoff, and Jill D. Haag, "On The Statistical Analysis of Allelic-loss Data", *Statistics in Medicine*, 17, pp: 1425-1445, 1998

[27] AKT3 in Wikipedia: http://en.wikipedia.org/wiki/AKT3

[28] BRWD2 in Wikipedia: http://en.wikipedia.org/wiki/BRWD2

[29] Franziska Michor, Kornelia Rolyak, "The origins and Implications of Intratumor Heterogeneity", *Cancer Prevention Research,* 3(11) November 2010

[30] Gloria H. Heppner, "Tumor Heterogeneity", *Cancer Research* 44, 2259-2265, June 1984

[31] Andriy Marusyk, Kornelia Polyak, "Tumor heterogeneity: cause and consequences", *Biochim Biophys Acta*. 2010 January; 1805 (1):105

[32] Andriy Marusyk, Vanessa Almendro, Kornelia Polyak, "Intro-tumor heterogeneity: a looking glass for cancer?", *Nature Reviews Cancer* 12, 323-334, May 2012

[33] Dan L. Longo, "Tumor Heterogeneity and Personalized Medicine", *The new England Journal of Medicine*, 956-957, March 8, 2012