Analysing Tumour Heterogeneity with Advanced Statistical Models

Mohammad Shamsur Rahman

Submitted for the fulfillment of the requirements for the degree of

Doctor of Philosophy

MONASH
University

Faculty of Information Technology

Clayton campus

Notice

Declaration

Analysing Tumour Heterogeneity with Advanced Statistical Models

This thesis contains no material which has been accepted for the award of any other degree or diploma at any university or equivalent institution and that, to the best of my knowledge and belief, this thesis contains no material previously published or written by another person, except where due reference is made in the text of the thesis.

Mohammad Shamsur Rahman

July 3, 2018

Acknowledgement

First of all, I would like to thank Allah for keep loving me, fulfilling all my needs and guiding my life. Without His blessing, I will not be able to finish this thesis.

I will forever be thankful to my supervisors Dr. Gholamreza Haffari, and Prof. Ann E. Nicholson. I consider myself extraordinarily lucky to have been given a chance to learn and develop under their careful and patient guidance. They always encourage me to continue and complete my Ph.D. research in all possible ways. They consistently give a solution for every problem during my study. Both Dr. Reza and Prof Ann always support me whenever I need to be away from the research because of family matters.

I thank my panel members Prof. Balasubramaniam Srinivasan, Prof. David Green, and Dr. Aldeida Aleti for their feedbacks on this thesis.

A special feeling of gratitude goes to my wife Nafisa Chowdhury, who has been supporting me sice our marriage. Despite her serious physical illness (Thyroid carcinoma), she always encourages me to complete this thesis.

I am thankful to my parents, Mohammad Tayabur Rahman and Shamsun Nahar Begum, who have always been supportive and encouraging over many years, I cannot thank you enough. I am also thankful to my mother-in-law and my sisters.

Of course no acknowledgments would be complete without giving thanks to wonderful colleagues and friends. I would like to thank my friends in Monash, Dr. Ehsan Shareghi Nojehdeh, Komal Singh Komal, Yang (Kelvin) Li, Dr. Md. Mamunur Rashid, Mohaimen Muhin and Md. Ashraful Islam. I give thanks to my colleague Bo Chen, Sameen Maruf, Parthan Kasarapu, Yathindu Rangana Hettiarachchige, Jishan Giti, Mohammad Adalah, Bahman Sari Yari.

Abstract

Cancer arises from successive rounds of mutations, resulting in tumour cells with different somatic mutations known as clones; this phenomenon is called tumour heterogeneity. Drug responsiveness and therapeutics of cancer depend on the accurate detection of clones in a tumour sample. Recent research has considered inferring the clonal composition of a tumour sample using computational models based on the short reads (segmented DNA parts of different tumour cells) data of the sample, generated using the next generation sequencing (NGS) technology. Short reads are noisy; therefore, inferring clones and their mutations from the data is a difficult and complex problem. Methods to infer tumour heterogeneity have three main drawbacks: they are (1) unable to discover different clones having the same clonal frequency, (2) unable to detect clone-specific allelic composition of mutations of a genomic location, and (3) do not consider the inter-dependency between mutations found in three dimensional (3D) structure of DNA, called long-range mutational influences. In the thesis, we address these drawbacks and develop computational methods to infer tumour heterogeneity more precisely.

We address the first two drawbacks by developing a computational method based on Factorial Hidden Markov Model to infer clones and their proportions from the noisy and mixed short reads. However, this method works on single tumour sample data. Therefore, we extend this method for multiple tumour samples and long-ranges mutational influences, captured from the known gene-gene interaction networks, and gene and mutation locations. Moreover, the cancer data consists of known and unknown interactions between genes. We then focus on to predict the unknown gene-gene interactions from high-dimensional data. We propose a new graphical model structure discovery method which predicts the known and unknown gene-gene interactions from the high dimensional data. This method is developed based on optimising a minimum message length (MML) based objective function. In this approach, we assume that all observations are generated from the same underlying multivariate distribution i.e. single sub-type of cancer. However, the real-life datasets exhibit heterogeneity, which means the data is mixture of normal cells and different cancer sub-types. Therefore, we extend our graphical model structure discovery method for heterogeneous data to capture this, using mixtures of graphical models, instead of single graphical model.

Empirical results confirm that the first computational method and its extension infer clonal composition more accurately than previous works. In the experiments of graphical model discovery on both the synthetic and the real-life data, our designed graphical model discovery methods detect gene-gene interactions with low false discovery rates. Therefore, the thesis makes three major research contributions: it (1) infers clone specific allelic composition of mutations and clonal frequencies together, using both adjacent and known long-range mutational influences, (2) discovers the graphical model to represent the known and unknown interactions between genes to find the long-range mutational influences, and (3) predicts the structures of graphical models to discover shared and context-specific graphical models from the high dimensional heterogeneous data. Hence, these three major contributions to computational tumour heterogeneity research give more precise prediction of tumour heterogeneity.

Contents

List of Figures

List of Tables

List of Algorithms

List of Symbols

Symbols	Description
T	A collection of mutations or mutant locations
X	A collection of samples
K	The number of clones
n	The number of samples
d	The number of random variables
$\boldsymbol{O}$	List of the observed data
$\boldsymbol{O}_t$	List of the observed data at location t
$\boldsymbol{O}_{t,x}$	List of the observed data of sample x at location t
a_t	Reference read counts at location t
$a_{t,x}$	Reference read counts of sample x at location t
$\bar{a}_t$	Mismatched read counts at location t
$\bar{a}_{t,x}$	Mismatched read counts of sample x at location t
N_t	Total read count at location t
$N_{t,x}$	Total read count of sample x at location t
l_t	Log ratio of normal-tumour content at t
$l_{t,x}$	Log ratio of normal-tumour content of sample x at t
$\boldsymbol{\mathcal{G}}$	Vector of clone-specific genotypes of all mutations
$\mathcal{G}_{t,k}$	Genotype at the location t of clone k
$\vec{\mathcal{G}}_{-t,k}$	A vector of the genotype of mutations having long range influences over the mutation at location t of clone k
$\mathcal{G}^j_{-t,k}$	Genotype of jth mutations having long range dependencies with mutation at tth location of clone k
$\boldsymbol{\mathcal{G}}_t$	Vector of all clone specific genotypes at location t
$\boldsymbol{\phi}$	A vector of clonal frequencies

Symbols	Description
ϕ_i	Clonal frequency of clone i
ϕ_0	Clonal frequency of normal clone
$\mathbf{\Phi}$	A matrix of sample-specific clonal frequencies or cellular prevalences
$\mathbf{\Phi_x}$	A vector of sample-specific clonal frequencies of sample x
$\phi_{i,x}$	Sample specific (sample x) clonal frequency of clone i
φ_k	Cellular frequency of a mutation k
BAF_t	B allele frequency at a location t.
$A_{t,k}(q,r) = P(\mathcal{G}_{t,k} = q \| \mathcal{G}_{t-1,k} = r)$	Transition probability between genotypes $\mathcal{G}_{t,k}$ and $\mathcal{G}_{t-1,k}$
$E_{t,k}(q)$	Transition probability to capture interdependencies between the genotypes of mutations having long range mutations
ρ_t	Prior probability that some genetic event occur between two mutations having adjacent dependencies at location t
τ	Prior probability that some genetic event occur between two mutations having long range influences dependencies
L	Average sequence read's length
L_G	Average gene's length
d_t	Geometric distance between two locations t and $t-1$
d_t^j	One dimensional (1D) distance between jth and tth mutations of same pathway
$d_{g_t^j}$	Geometric distance between genes of jth and tth mutations of same pathway
D_k	Dimension of genotype state (i.e. 21)
$\tilde{\mu}_t$	Mean of log-ratio of tumour normal content at location t
$\tilde{\mu}_{t,x}$	Mean of log-ratio of tumour normal content at location t of sample x
σ	Standard deviation
P_{b_t}	Probability of a_t with respect to N_t at location t
$P_{b_{t,x}}$	Probability of $a_{t,x}$ with respect to $N_{t,x}$ of sample x at location t
$r_{g_t^k}$	The number of reference allele of genotype $\mathcal{G}_{t,k}$ of clone k at location t
$c_{g_t^k}$	Copy number of genotype $\mathcal{G}_{t,k}$ of clone k at location t
$c_{g_t^0}$	Copy number of genotype $\mathcal{G}_{t,0}$ of normal clone at location t
ψ	Tumour ploidy parameter. By default it is 3.
η	Learning parameter of EG algorithm
Δ	Simplex containing all probability vectors

Symbols	Description
h	Homogeneity: It computes the proportion of the members of a cluster contains only members of a single gold standard clone
c	Completeness: It computes the proportion of the members of a gold standard clone contains only members of a predicted cluster
$V\text{-}Measure$	V-Measure an external entropy based cluster evaluation metrics based on completeness and homogeneity
Cls^g	Set of gold standard clones
Cls^p	Set of predicted clusters
Cls^s	Set of predicted significant clusters
$\overline{Cls}^s$	Set of predicted insignificant clusters
b_{ij}	The number of data points that are the members of gold standard clone i and predicted cluster j
m	The number of data points
$RMSD$	Root Mean Square Distance which is used to compute the distance or error between the cellular prevalence of predicted clusters and the gold standard
$\mathcal{D}$	Dataset having n samples and d random variables
$\mathcal{D}_i$	Data points of cluster i
$\mathcal{D}^C$	Data points of a clique (or separator) C
$\mathcal{D}_i^C$	Data points of a clique (or separator) C of cluster i
D_{ij}	jth data point of ith cluster
D_i^C	ith data point of clique (or separator) C
D_{ij}^C	jth data point of clique (or separator) C of cluster i
X_i	Data points of a random variable i
θ_i	Parameters of cluster i (i.e. $\theta_i = \{\mu_i, \Sigma_i\}$)
θ^C	Parameters of a clique (or separator) C
θ_i^C	Parameters of a clique (or separator) C of cluster i
θ'	Parameters of candidate model
μ_i	Mean vector of cluster i
μ^C	Mean vector of a clique (or separator) C
μ_i^C	Mean vector of a clique (or separator) C of cluster i
$\hat{\mu}_i$	Empirical mean vector of cluster i
$\hat{\mu}^C$	Empirical mean vector of a clique (or separator) C

Symbols	Description
Σ_i	Covariance of cluster i
Σ^C	Covariance of a clique (or separator) C
Σ_i^C	Covariance of a clique (or separator) C of cluster i
$\hat{\Sigma}_i$	Empirical covariance of cluster i
$\hat{\Sigma}^C$	Empirical covariance of a clique (or separator) C
G	A graph
$\boldsymbol{G}$	A set of graphs
G'	Candidate graphical model
G_i	Context-specific graphical models
G'_i	Context-specific candidate graphical models
G^*_i	Context-specific candidate graphical models without shared edges
G_0	Graphical model contains only shared edges
G'_0	Graphical model contains only shared edges
V	Vertex list
E	Edge list
E^c	Candidate edge list
E_i	Context-specific edge list including shared edges
E_0	Shared edge list
$E_{complete}$	Edge list of a complete graph
$N(a)$	Vector of neighbours of a vertex a
$\mathcal{C}$	Clique set
$\mathcal{S}$	Separator set
C	A maximal clique
S	A minimal separator
$pdf(D)$	Probability density function
$pdf^C(\mathcal{D}^C)$	Clique (separator) specific probability density function
$pdf_i(\mathcal{D}_i)$	Context specific probability density function
$\mathcal{M}$	Decomposable model
$\mathcal{M}'$	Candidate decomposable model
$\mathcal{S}^+(G)$	All positive definite matrices whose zero patterns are consistant with the graph G
$\mathcal{K}$	Precision matrix
$\hat{\mathcal{K}}$	Empirical precision matrix of reference model
$\hat{\mathcal{K}}'$	Empirical precision matrix of candidate model
ssd	Sum squared distance
β	ContChordalysis's layered p-value threshold
m_c	The number of free parameters
q	Lattice quantisation constant

Symbols	Description
MML_e	Minimum message length to encode an edge e with its parameters and data; and graph structures
MML_e^*	Minimum message length to encode an edge e with its parameters and data without graph structures
$\mathcal{F}(\mu^C, \Sigma^C)$	Fisher information of μ and Σ of a maximal clique (or minimal separator) C
$\mathcal{F}(\mu^C)$	Fisher information of μ of a maximal clique (or minimal separator) C
$\mathcal{F}(\Sigma^C)$	Fisher information of Σ of a maximal clique (or minimal separator) C
$\mathcal{F}(\mu_i, \Sigma_i)$	Fisher information of μ and Σ of cluster i
$\mathcal{F}(\mu_i)$	Fisher information of μ of cluster i
$\mathcal{F}(\Sigma_i)$	Fisher information of Σ of cluster i
θ	List of parameters
$\hat{z}$	cluster indicator vector
n_i	The number of data points of cluster i
ϵ	Noise vector
$\boldsymbol{G}_{Chordal}$	Vector of chordal graphs that can be formed by adding an edge to candidate model at t iteration
$\lvert \cdot \rvert$	Determinant
$I(a)$	Optimal code length to convey some event a whose probability is $p(a)$
γ_i	Mixing coefficient of ith cluster
α	Parameter that control the spread of sampled mixing coefficients in the mixture model
adj_j	Adjacency matrix of cluster j
BIC	Bayesian information criterion
AIC	Akaike information criterion
λ	Regularization parameters of Lasso
$\mathcal{B}in$	Binomial distribution
$\mathcal{N}$	Normal or Gaussian distribution
$\mathcal{U}$	Uniform distribution
$\mathcal{B}$	Beta distribution

List of Acronyms

Acronyms	De-abreviation
DNA	Deoxyribonucleic acid
RNA	Ribonucleic acid
NGS	Next Generation Sequencing
SNP	Single nucleotide polymorphism
CNV	Copy number variation
GGM	Gaussian graphical model
BAF	B-allele frequency
VBMM	Variational Bayesian mixture model
SNV	Single nucleotide variation
TSSB	Tree structured stick breaking process
AML	Acute myeloid leukemia
CLL	Chronic lymphocytic leukemia
TNBC	Triple negative breast cancer
HGSOC	High grade serous ovarian cancer
LOH	Loss of heterozygosity
HMM	Hidden Markov model
LRR	Log ratio of normal-tumour content
MAP	Maximum a posterior
EM	Expectation maximization
BIC	Bayesian information criterion
FHMM	Factorial hidden Markov model
MLE	Maximum likelihood estimate
LASSO	Least absolute shrinkage and selection operator
RMSD	Root mean square distance
MCMC	Markov chain Monte Carlo
EG	Exponentiated gradient descent
BRCA	Breast invasive cancer
TCGA	The cancer genome atlas
MML	Minimum message length
MDL	Minimum descriptor length
RD	Residual disease
pCR	Pathological complete response
ER	Estrogen-receptor
AIC	Akaike information criterion

Publications during enrolment

1. Mohammad S Rahman, Ann E. Nicholson and Gholamreza Haffari "HetFHMM: A novel approach to infer tumour heterogeneity using factorial hidden Markov model", *Journal of Computational Biology*, 25(2): 182-193, 2018, https://doi.org/10.1089/cmb.2017.0101

2. Mohammad S Rahman and Gholamreza Haffari, "A statistically efficient and scalable method for exploratory analysis of high-dimensional data", Revised version submitted to *Journal of Data Mining and Knowledge Discovery.*

Chapter 1

Introduction

1.1 Introduction

Cancer is a disease, caused by variations accumulated in the genome, called the genomic variations, during the lifetime of a human (Stratton et al., 2009). The genomic variations create different cancer cells, known as clones, leading to a phenomenon called *tumour heterogeneity* (Ha et al., 2012). According to Sharma et al. (2015), the genomic variations result from (a) errors during DNA replication or (b) other types of damage to DNA (e.g. caused by exposure to radiation or carcinogens), which then may undergo error-prone repair or (c) alterations in the gene after it has come in contact with mutagens[1] and environmental causes. Based on a recent research survey from Harris (2017), 66% of cancer-causing *mutations* are due to errors during DNA replications, error-prone repairs, and mutagens, 29% are due to the environment, and 5% are inherited.

Recent cancer therapeutics are developed to target cancer cells, based on the genomic variations that they harbour (Caraco, 1998). From the treatment perspective, therefore, it is both critical and difficult to identify the genomic variations[2] in cancer cells (i.e. tumour heterogeneity). Most of the researches focus on identifying the clones with the genomic variations of DNA replication error, error-prone repairs and mutagens. Next

[1] In genetics, a mutagen is a physical or chemical agent that changes the genetic material, usually DNA, of an organism and thus increases the frequency of the genomic variations/mutations above the natural background level.

[2] Note that, in this thesis, we use the term mutations and genomic variations interchangeably.

Generation Sequencing (NGS) offers a great opportunity to identify clones in a tumour sample based on the generated *short reads*[3]. NGS produces billions of short reads by chopping off the genome of cells into small segments, and then conveniently reading these short sequences. It is challenging, however, to reconstruct the genome of the sequenced cells since the short reads are not tagged with the *ordering* information. The difficulty of the problem is further exacerbated due to the *mixing* of short reads from cells belonging to different clones.

In this thesis, we present computational methods to identify the tumour heterogeneity (i.e. cancer clones and their genomic variations) from mixed and noisy short reads of a tumour sample(s). In this chapter, we first focus on the cancer biology, and the researches and their drawbacks to predict cancer clones and their genomic variations in the fields of computer science and machine learning. We then discuss our research objectives to resolve the drawbacks of the current methods and our research contributions. Finally, we outline the organization of the thesis.

1.2 Cancer

Cancers are a large family of diseases that involve abnormal cell growth with the potential to invade or spread to other parts of the body (Anand et al., 2008). They form a subset of neoplasms, called a tumour. A tumour is a group of cells that have undergone unregulated growth and will often form a mass or lump, but may be distributed diffusely (Anand et al., 2008). All tumour cells show the five hallmarks of cancer as follows (Hanahan & Weinberg, 2011):

- Erroneous cell division during mitosis.
- Avoidance of programmed cell death
- Limitless number of cell divisions
- Promoting blood vessel construction

[3] The fragmented DNA whose length is in between 100 and 600 base-pairs (Caraco, 1998).

- Invasion of tissue and formation of metastases

Therefore, cancer is fundamentally a disease of tissue growth regulation. In order for a normal cell to transform into a cancer cell, genetic changes can occur in an entire chromosome through errors in mitosis[4]. The most common genetic changes are mutations, which change the nucleotide sequence of a DNA strand.

1.2.1 Mutations

In biology, a mutation is a permanent alteration of the nucleotide sequence of the genome of an organism. Mutations can be classified into two categories: *somatic* and *germline*. Somatic mutation is a genetic alteration acquired by a cell that can be passed to the progeny of the mutated cell in the course of cell division. Whereas, germline mutations are inherited genetic alterations that occur in the germ cells, i.e. sperm and eggs. Somatic mutations are frequently caused by DNA replication error, error-prone repairs, mutagens, and oncogenes[5]. According to cancer biology, various types of somatic mutations can appear in human cells (Freese, 1959):

1. Point mutation: It changes a single nucleotide to another nucleotide, and is caused by malfunctioning of DNA replication[6] (Freese, 1959).
2. Insertion mutation: It adds one or more extra nucleotides into the DNA (Freese, 1959).
3. Deletion mutation: It removes one or more nucleotides from the DNA (Freese, 1959).
4. Chromosome structural mutation: It amplifies, deletes or translocates parts of a chromosome structure (Freese, 1959). Fig-1.1 shows an example of chromosome structural mutations.

[4] In cell biology, mitosis is a part of the cell cycle when replicated chromosomes are separated into two new nuclei to produce two new cells.

[5] An oncogene is a gene that has the potential to cause cancer. In tumour cells, they are often mutated and/or expressed at high levels.

[6] In molecular biology, DNA replication is the biological process of producing two identical replicas of DNA from one original DNA molecule.

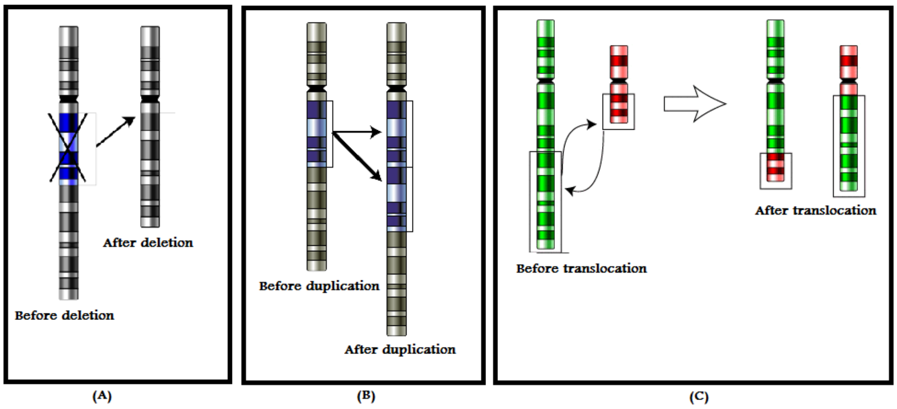


Figure 1.1: Chromosome structural mutations. (A) Deletion of a part of chromosome which is marked by the blue region, (B) Amplification of a part of chromosome, where the blue region is amplified by two i.e. duplication, and (C) Translocation of two parts of two chromosomes (Freese, 1959).

Among the above stated types of mutations, the first three mutation types are *Single nucleotide polymorphisms (SNPs)* and the last one includes *Copy number variations (CNVs)*. In a normal human genome, each chromosome has two copies. When a structural variation mutation apperas in a tumour genome, the number of copies of the structurally mutated part of the genome can change, (i.e. copy number variation). In the human body, these somatic mutations create cells with different genomic variations, which are known as *clones*.

1.2.2 Clones

Somatic mutations create cancer cells with different genomic variations, where a set of cancerous cells show distinct morphological[7] and phenotypic profiles[8], including gene expression and metabolism. A set of cells with identical genomic variations (i.e. mutations) are known as *tumour clones*. The observation of different cancer cells in a tumour is known as *tumour heterogeneity*.

According to Nowell (1976), tumours arise from a single mutated cell, accumulating additional mutations as it progresses. These changes give rise to additional subpopu-

[7] Morphology is a branch of biology dealing with the study of the form and structure of organisms and their specific structural features.

[8] A phenotype (from Greek phainein , meaning "to show ", and typos , meaning "type") is the composite of an organism's observable characteristics or traits

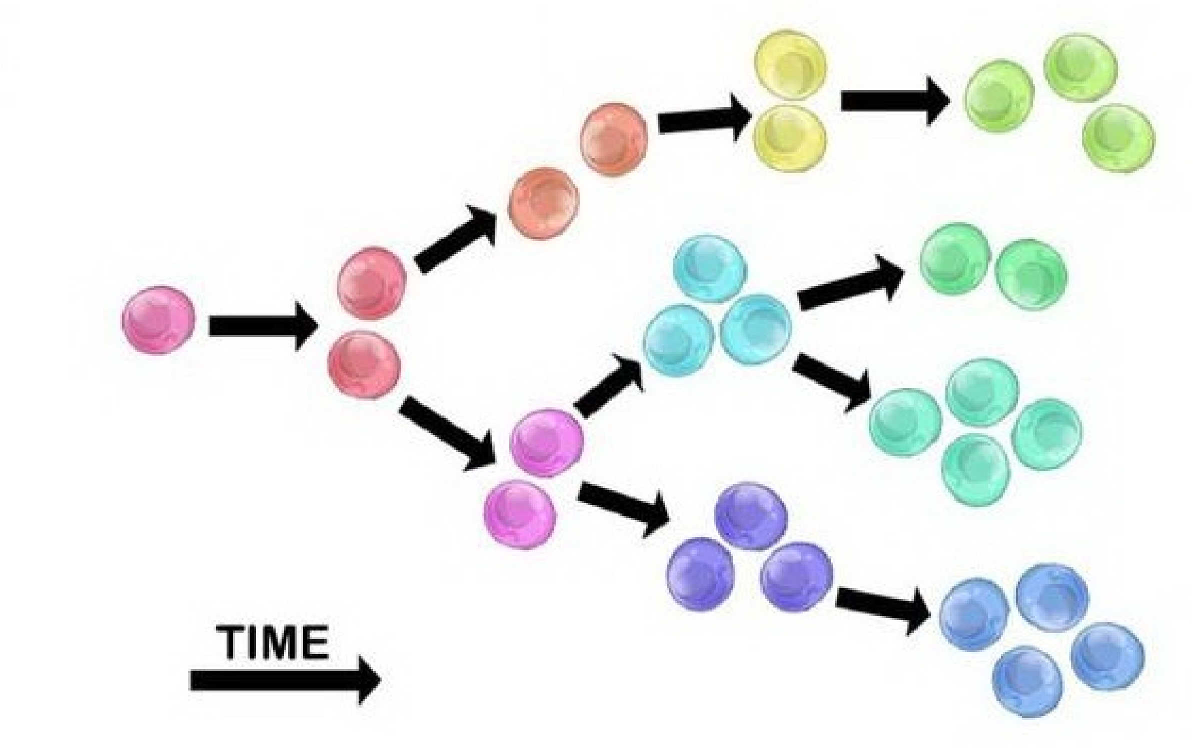


Figure 1.2: Progression of clones from a normal cell.

lations (i.e. tumour clones), and each of these subpopulations has the ability to divide and mutate further (Merlo, Pepper, Reid, & Maley, 2006). This heterogeneity gives rise to more clones that possess an evolutionary advantage over the others within the tumour environment, and these clones may become dominant in the tumour over time (Merlo et al., 2006).

Recent cancer therapeutics have been developed to target cancer clones based on the mutations that they harbour (Caraco, 1998). From the cancer treatment perspective, therefore, it is critical to identify the genome of cancerous cells (i.e. the composition of mutations) forming a patient's tumour clones.

1.2.3 DNA sequencing

The composition of somatic mutations of a clone can be inferred based on DNA sequencing (Olsvik et al., 1993): a process for determining the sequences of nucleotides within a DNA molecule. DNA sequencing is used to determine the order of the four bases: adenine, guanine, cytosine, and thymine, in a DNA strand. There are different types of DNA sequencing technologies, which can be classified according to their cost, time and accuracy. Sanger and Coulson (1975) proposed a DNA sequencing technique called *Sanger sequencing.* It is considered the first generation DNA sequencing technique.

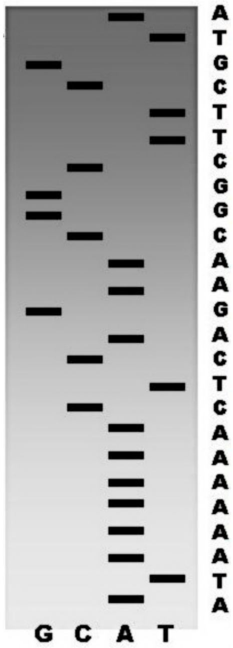


Figure 1.3: Sanger sequencing

In Sanger sequencing, the sequence of the nucleotides of a DNA strand is identified by using DNA polymerase[9] and four independent sequencing reactions for four nucleotide of the DNA strand. At a genomic location, a nucleotide reacts with a DNA polymerase, and is added to the corresponding column of the sequencing ladder. Finally, using gel electrophoresis, ddNTP[10] and Ultra-violet light, the columns of the sequencing ladder are visualized. From the visualized sequencing ladders, the sequence of the DNA strand is read.

This technique suffers from several drawbacks: (a) the quality is deteriorated after approximately 700-900 nucleotides[11], (b) it cannot work on very long DNA pieces, e.g.

[9] In molecular biology, DNA polymerases are enzymes that synthesize DNA molecules from deoxyribonucleotides – the building blocks of DNA.

[10] A dideoxynucleotide (ddNTP) is an artifical molecule that lacks a hydroxyl group at both the 2' and 3' carbons of the sugar moiety. There are four types of ddNTPs: ddATP (for adenine) ddGTP (for guanine), ddCTP (for cytosine) and ddTTP (for thymine). Each of ddNTP is used to determine nucleotide of a DNA strand

[11] ddNTP can work fine for 700-900 nucleotides in each reaction. For this reason, Sanger sequencing can detect 700-900 nucleotides at a time.

a full chromosome, and (c) it is very costly (approximately several million US dollars per human genome).

1.2.4 Next Generation Sequencing

In contrast, *Next Generation Sequencing* (NGS) offers better and large-scale sequencing that often aims at sequencing very long DNA pieces, such as whole chromosomes. It allows the sequencing of a DNA much more quickly and cheaply than the previously used Sanger sequencing. NGS consists of cutting (with restriction enzymes) or shearing (with mechanical forces) large DNA fragments into shorter DNA fragments. The fragmented DNA pieces are then cloned into a DNA vector and amplified in a bacterial host, such as *Escherichia coli*. Finally, short DNA fragments purified from individual bacterial colonies are individually sequenced and assembled electronically into a long contiguous sequence. The sequenced short DNA fragments are called *short reads*.

Sequence assembly refers to putting together the short reads in order to reconstruct the original sequence. There are two different approaches to sequence assembly:

De-novo: Assembling short reads to create full-length (sometimes novel) sequences, and

Mapping: Assembling short reads against an existing reference sequence, and building a sequence that is similar but not necessarily identical to the reference sequence.

In terms of the complexity and time requirements, de-novo assembly methods are orders of magnitude slower and more memory intensive than mapping assembly methods. In this thesis, we adopt the second approach and reconstruct genomes by mapping their short reads to the reference genome.

It is challenging to reconstruct the genome of the sequenced cells since the short reads are not tagged with the *ordering* information. The difficulty of the problem is further exacerbated due to the *mixing* of short reads generated from cells belonging to different clones. Various computational methods have been proposed to identify the clones and their genetic make-up[12] from mixed and noise short reads.

1.3 Computational methods to predict tumour heterogeneity from short reads

The problem of identifying clones and their mutations has recently attracted attention due to the viability of NGS technology. GPHMM (A. Li et al., 2011) (*Global Parameter Hidden Markov Model*) is a pioneering work to discover mutations involving CNVs and SNPs. It assumes the tumour sample is a mixture of the normal tissue and a tumour clone possessing all cancer-causing mutations. However, as discovered in cancer biology (Stratton et al., 2009), a tumour sample often has multiple clones and each clone harbours a subset of mutations in the tumour sample. This is the basis of Apolloh (Ha et al., 2012), Onco-SNP (Yau, 2013) and TH-HMM (Xia et al., 2013) in inferring CNVs and SNPs. To infer mutations, the aforementioned models assume the presence of three clones in a sample, where two of them are cancerous, and a mutation can appear in either of these cancer clones or both. TITAN (Ha et al., 2012) and CLImAT-HET (Yu, Li, & Wang, 2017) allows a tumour sample to contain more than two cancer clones; however, it makes the assumption that a mutation cannot belong to more than one clone. This limitation has been remedied in the follow up works TrAp (Strino, Parisi, Micsinai, & Kluger, 2013) PyClone (Roth et al., 2014), PhyloSub (Jiao et al., 2014), Rec-BTP (Hajirasouliha et al., 2014), PhyloWGS (Deshwar, Vembu, Yung, et al., 2015), CITUP (Malikic et al., 2015), BitPhylogeny (Jianga, Qiu, Minn, , & Zhang, 2016), Canopy (K. Yuan, Sakoparnig, Markowetz, & Beerenwinkel, 2015) and PASTRI (Satas & Raphael, 2017), where they output the list of mutations characterising each clone as well as the *clonal frequency* i.e. the percentage of tumour cells belonging to the clone. However, these methods to predict tumour heterogeneity suffer from many drawbacks which we discuss in the next section.

[12] The genetic make-up of a clone is the set of mutations harboured by the clone, and the genotype of those mutations.

1.4 Research challenges

Current methods to detect tumour heterogeneity from the mixed and noisy short reads, have three main limitations:

L1: Inability to discover the clones with same frequency. All tumour heterogeneity inferring methods discover the clones by assuming they have different clonal frequencies. However, in the real cancer data, some of the clones have the same frequency but with different allelic compositions. Thus, the existing tumour heterogeneity inferring methods cannot infer clones with the same frequency. These methods consider clones with the same frequency as a single clone.

L2: Incapable to detect fine-grain composition of clones. Methods to infer tumour heterogeneity assume that a single type of mutation appears at a location. However, Kandoth et al. (2013) and Vogelstein et al. (2013) observe that different types of mutations can appear at a location and that these are harboured by each clone independently. This indicates that the type of mutation and its appearance is clone dependant or specific. Therefore, the existing methods are unable to infer the exact type of the mutations (allelic composition) at a location.

L3: Omission of the long range influences among mutations. Ji et al. (2016) observed the relationships between the mutations of cancer arise from the three-dimensional DNA structure of the human cells. They found that due to the three-dimensional structure of DNA, cancer-causing genes with *driver* mutations mask the genomic functionalities of their nearest proteins and genes, which in turn cause the appearance of other mutations. However, the one dimensional genomic distance between these driver mutations and the newly appeared passanger mutations are not close to each other. This type of influences among the mutations is known as *long-range influences.* Existing methods do not use this important relationship between the mutations in their methods.

In this thesis, we address the above mentioned limitations to infer more accurately cancer clones and their genomic composition from the NGS data.

1.5 Objectives of the thesis

We set four research goals to improve tumour heterogeneity inference:

O1: Inferring clonal composition and frequencies from single sample data. To achieve the first objective, we propose a computational method to detect the clonal frequencies and their allelic compositions of tumour clones from a single tumour sample to resolve the first two limitations (L1 and L2). In this objective, we also focus on predicting the clones with same frequency from the given data-set.

O2: Extending the first objective to multiple-sample data and long-range mutational influences. Ellenbroek and v. Rheenen (2014) found that each clone of a tumour is formed within a particular area of the tumour bulk. Tumour samples of a patient can be taken from different parts of the tumour bulk. Thus, the inferred clonal frequencies should be different, depending on the area from which the sample has been taken. Whereas, the genomic composition of clones should be the same across the samples.

Moreover, mutations have long-range influences among themselves, which is not captured by their 1-D genomic distance. Therefore, to achieve the second objective, we extend the tumour heterogeneity detection method to multiple samples and long-range inter-dependencies among mutations to resolve the last shortcoming (L3). Ji et al. (2016), observe that mutations with long-range influences appear in genes of the same pathways. Therefore, addressing the second objective, we use the known gene-gene interaction networks to identify long-range mutational influences.

O3: Discovering the structure of the gene-gene interaction networks to find out long-range mutational influences. Although we use known gene-gene interaction in the second objective to find out the long-ranges mutational influences, some of the gene interactions are not found in the known networks. By good fortune, the gene expression data can help us to find these missing interactions. To achieve our third objective, we develop a data based method to discover gene-gene interaction from gene expression data. We cast this problem as discovery of the structure of a probabilistic graphical model (PGM).

O4: Extending the graphical model structure discovery to heterogeneous data. Recent studies on cancer genome atlas network have found that gene expression data can be described as a mixture of a small number of components harbouring different expression pathways (Mukherjee & Roriguez, 2016). Thus, real-life datasets are heterogeneous, which can be accommodated through the use of mixtures of graphical models. This allows each component exhibit different conditional dependencies among variables, a.k.a context-specific-dependencies (Meilă & Jordan, 2000; Rodriguez, , Lenkoski, & Dobra, 2011). To achieve our final research objective, we propose a method to discover context specific graphical models.

1.6 Research Contribution

The research area in tumour heterogeneity prediction is rich with many open questions and challenges. We address and solve the aforementioned limitations (stated in Section 1.4) through our objectives (given in Section 1.5). The major contributions of this thesis are as follows:

C1: Prediction of clonal composition and clonal frequencies. One of the major contributions of this research is to discover the clones with equal frequencies and their mutational architecture. We develop a statistical model based on Factorial Hidden Markov Model to infer the clonal compositions from noisy and mixed short reads. First, we predict the clonal composition from single sample tumour data (O1). Later we incorporate long-range influences among the mutations in addition to that of adjacent mutations (O2). Then we incorporate multiple samples in our extended statistical model (O2) for more accurate tumour heterogeneity inference.

PUBLICATION: Mohammad S Rahman, Ann E. Nicholson and Gholamreza Haffari “HetFHMM: A novel approach to infer tumour heterogeneity using factorial hidden Markov model", *Journal of Computational Biology*, 25(2): 182-193, 2018, `https://doi.org/10.1089/cmb.2017.0101`

C2: Discovery of graphical models with low false discovery rate. Another major contribution of our research is to discover the structure of probabilistic graphical models with low false discovery rate. In C1, we only use the long-range mutational influences available in the known gene-gene interaction networks. Whereas, the real cancer data contains many unknown gene-gene interaction as well. To capture the unknown gene-gene interaction, we have proposed a scalable and statistically efficient approach for graphical model structure discovery involving continuous variables (O3). We introduce a novel method based on the minimum message length (MML) (Wallace & Boulton, 1968) for statistical inference to improve the structure discovery of Gaussian graphical models.

PUBLICATION: Mohammad S Rahman and Gholamreza Haffari, “A statistically efficient and scalable method for exploratory analysis of high-dimensional data", Revised version submitted to *Journal of Data Mining and Knowledge DIscovery.*

C3: Structure learning with mixture of Gaussian graphical models. Our last contribution to tumour heterogeneity research is to discover the context-specific graphical models from heterogeneous data, considered as a mixture of graphical models. We propose a statistically efficient method to discover context-specific Gaussian graphical models structure, along with their shared edges, from high-dimensional data(O4). We introduce PaGIAM (***P****artition* ***a****nd* ***G****raphical model discovery* ***I****terative* ***A****lgorithm based on* ***MML***) for clustering the data and tGDM (***t****he context-specific* ***G****aussian graphical models* ***D****iscovery using* ***MML***) for discovering the context-specific graphical models with shared edges based on the minimum message length. Most of the existing methods discovering the mixture of GGMs use the hard EM algorithm to cluster the heterogeneous data and find the GGM components in the mixture. However, they do not consider the existence of context specific graphical structures. In contrust, PaGIAM is an iterative algorithm based on EM which considers the existence of context-specific graphical structures. The tGDM algorithm is a step-wise greedy algorithm to find the context-specific GGMs and their shared edges. By combining the ideas of these two algorithms, we discover the context-specific graphical structures from the heterogeneous data more accurately than the existing methods.

1.7 Thesis organization

The thesis is organised into six chapters.

Chapter 2 gives an overview of the related work in the areas of both inferring the clonal composition and structures of graphical models. First, we review different approaches to inferring the clonal architecture and clonal frequencies. Then we review different Gaussian graphical model structure learning techniques to discover gene-gene interaction networks, which helps to capture long-range inter-dependencies among mutations. In this review, we focus on the methods that discover Gaussian graphical models from homogeneous as well as heterogeneous data.

Chapter 3 introduces our novel framework for inferring the clonal composition for single-sample tumour data (O1). This framework is based on Factorial Hidden Markov Models (Ghahramani & Jordan, 1997) to infer clones and their proportions from the noisy NGS data, called ***Het****erogeneity prediction using* ***F****actorial* ***H****idden* ***M****arkov* ***M****odels* or HetFHMM. We then present an extensive evaluation for HetFHMM with various synthetic and publicly-available real cancer datasets. We compare the performance of HetFHMM with other state-of-the-art techniques in tumour heterogeneity prediction- PyClone and PhyloSub- based on accuracy and robustness for tumour heterogeneity prediction.

Chapter 4 presents the extension of HetFHMM to discover clones and their genetic make-up from multiple samples and leveraging long-range mutational influences (O2). To capture long-range influences among mutation, we use known gene-gene interaction network. We call this extension of HetFHMM *extended multisample HetFHMM* or emHetFHMM. We present extensive evaluation of emHetFHMM and compare with PyClone, PhyloSub and HetFHMM based on using both synthetic and real cancer data.

Many gene-gene interactions are hidden and not found in the existing gene-gene interaction networks. Therefore, we aim to discover the gene-gene interaction networks from real-life cancer data. In **Chapter 5**, we introduce a new method to discover gene-gene interaction network, formulated as Gaussian graphical model structure discovery (O3). To select the best suitable edges for the graphical model, we use minimum message length (MML), which makes it possible to learn the structure from high-dimensional data, where the number of random variables is larger than samples. We call our method *ContChordalysis-MML*. We present an evaluation framework and compare ContChordalysis-MML with strong baselines: TIGER (H. Liu, 2017), r-GLasso (Avagyan, Alonso, & Nogales, 2017), FoBa-gdt (J. Liu, Fujimaki, & Ye, 2014), CLIME (Cai, Liu, & Luo, 2011) and GLasso (J. Friedman et al., 2008).

Real cancer datasets exhibit heterogeneity, i.e. the mixture of different cancer subtypes and normal tissue. These cancer subtypes have different gene-gene interaction networks with a set of common gene-gene interactions. **Chapter 6** presents the extension of our graphical model discovery method for heterogeneous data (O4). To discover context-specific graphical models and their shared structure, our method incrementally adds best edges minimising an MML-based scoring function in the forward selection algorithm (Deshpande et al., 2001). Following the previous chapters, we present extensive evaluation results to compare our method along with two strong baselines: New-SP (Gao, Zhu, Shen, & Pan, 2016) and JSEM (Ma & Michailidis, 2016).

Finally, **Chapter 7** summarizes the work of the thesis, draws conclusions, and outlines the possible future directions.

Chapter 2

Background and literature review

2.1 Introduction

In the Introduction chapter, we highlighted the drawbacks of the existing methods to infer tumour heterogeneity from the tumour samples. Based on drawbacks, we set four research objectives for the thesis. The first two objectives aim to predict clones and their genetic make-up along with clonal frequencies. To predict clones, their genetic make-up, and frequencies, some methods have been developed. Among these methods, a group of methods infer clones by clustering mutations and predict the tumour phylogeny to detect the tumour clones and their frequencies. Another group of methods predict clones by inferring the genotypes[1] of mutations. All of these methods suffer from drawbacks as mentioned in the previous chapter. In this chapter, we discuss all of the methods that are close to our methods.

Moreover, as described in the Introduction chapter, we set the third objective to develop a method to discover the gene-gene interaction networks using Gaussian Graphical Model (GGM) from the high-dimensional data. In this chapter, we also discuss the existing methods which discover the GGM from the high-dimensional data and their drawbacks.

The final objective is to discover context-specific GGMs from heterogeneous data

[1] Genetic make-up of a mutation.

containing mixture of GGMs. Existing methods to discover context-specific GGMs from heterogeneous data are discussed in this chapter. Therefore, the whole chapter is divided into four sections as below:

- Methods that leverage the tumour phylogeny to predict the tumour clones and their frequencies.
- Methods that leverage genotypes of the mutations to infer the tumour clones.
- Methods that discover Gaussian graphical models (GGMs).
- Methods that learn context-specific GGMs from the heterogeneous data.

2.2 Inferring tumour clones with phylogeny

A group of tumour heterogeneity inference methods detect the tumour clones by inferring clusters of mutations and clonal frequency of clones on the phylogeny property[2] of tumour. According to the phylogeny property of the tumour, each clone can contain two types of mutations:

1. Old mutation: Mutations that appear in a clone and its ancestor clones.
2. New mutation: Mutations that appear in a clone only.

The cellular frequency of new mutations of a clone would be same, and equal to the clonal frequency[3] of that clone. This can be expressed as follows:

$$\varphi_i = \phi_j \tag{2.1}$$

where i is a new mutation that appears in the clone j only. φ and ϕ are the cellular frequency of a mutation and clonal frequency of a clone, respectively. Moreover, according to the phylogeny property, old mutations that appear inside an ancestor of some or all

[2] All clones follow a perfect persistent phylogeny, where a mutation cannot be reverted back to its original state

[3] The fraction of cells that are genetically similar to each other.

of the existing clones should be present in all of its descendant clones. Therefore, the cellular frequency of an old mutation follows the following mathematical relation:

$$\varphi_k \;=\; \phi_{ancestor} \;+ \sum_{\forall\ j\in\ D(ancestor)} \phi_j \tag{2.2}$$

where k is an old mutation which appears in the clone *ancestor* and its descendent clones j. $D(ancestor)$ is a function which finds the descendants of the clone *ancestor*. According to the equation (2.2), the cellular frequency of an old mutation would be higher than a new mutation. Moreover, the cellular frequency of old mutations of an ancestor clone would be different from the other ancestors. So, it is possible to predict clones and their mutations by clustering mutations according to their cellular frequencies.

In reality, it is not easy to compute cellular frequency of mutations from DNA sequences of a tumour sample. DNA sequencing is a technique which is used to determine the precise order of nucleotides of DNA sequences. But DNA sequencing cannot detect the complete DNA sequences. It produces fragmented pieces of DNA sequences which are called **short reads**. Mutations are identified by comparing the nucleotides of short reads with the reference genome[4]. Mismatched nucleotides of short reads are considered as mutations. The fraction of short reads containing a mutation at any genomic location t may be considered as the cellular frequency of mutation t, which is called BAF[5] ((B allele[6] frequency)).

Strino et al. (2013), Zare et al. (2014), Hajirasouliha et al. (2014), Miller et al. (2014), Malikic et al. (2015), Popic et al. (2015), El-Kebir, Oesper, Acheson-Field, and Raphael (2015), and Satas and Raphael (2017) consider BAF as the cellular frequency

[4] A complete human DNA sequence.

[5] $BAF_t = \frac{\bar{a}_t}{a+\bar{a}_t}$, where a_t and $\bar{a}_t$ are the number of the short reads of matched and B allele at a location t, respectively. Some of the researchers refer to it as the variant allele frequency or VAF

[6] In the context of short read, the "B" allele is the non-reference allele observed in a heterozygous SNP, i.e. in the normal or tumour sample. Since the tumour cells' DNA are originally derived from the normal cells' DNA, most of these SNPs will also be present in the tumour sample. But due to allele-specific copy number alterations, loss of heterozygosity or allelic imbalance, the count of B allele of these SNPs may be different in the tumour, and that is the evidence that one (or both) of the copies was gained or lost during tumour evolution.

of a mutation in their methods. They use different clustering approaches to cluster the cellular frequencies of mutations to infer the clones. TrAp (Strino et al., 2013), Clomial (Zare et al., 2014) and AncesTree (El-Kebir et al., 2015) use the matrix factorization method to infer clonal frequencies and clones. SciClone (Miller et al., 2014), Rec-BTP (Hajirasouliha et al., 2014) and PASTRI (Satas & Raphael, 2017) use variational Bayesian mixture model (VBMM) to cluster BAF. CITUP (Malikic et al., 2015) uses the standard K-mean clustering approach. In LICHeE, Popic et al. (2015) compute the similarity score between the BAFs of two mutations as follows:

$$similarity\ score_{i,j} = \sum_{\forall \text{ samples } i} \frac{min(BAF_i, BAF_j)}{max(BAF_i, BAF_j)} \tag{2.3}$$

where i and j are two mutations. If the *similarity score*$_{i,j}$ is higher than a given threshold, the mutation i is assigned to the cluster having the mutation j. All of the above mentioned methods assume that all mutations are single nucleotide variation (SNV)[7]. These are not designed for copy number variation (CNV).

Interestingly, during DNA sequencing, some fragments of a DNA sequence are amplified by some enzymes[8] and *E. coli*[9], which increases the number of short reads. Moreover, some fragments are harmful to *E. coli* and cause the death of *E. coli* bacteria. Therefore, it reduces the number of short reads. On the other hand, different types of mutations especially CNVs also affect the short reads counts. Tumour sample is the admixture of different tumour clones and normal cells. Hence, the number of short reads is not able to express the exact number of cells that contain a mutation. That is BAF should not be considered as the cellular frequency of a mutation.

Jiao et al. (2014), Roth et al. (2014), Deshwar, Vembu, Yung, et al. (2015), K. Yuan et al. (2015) and Jianga et al. (2016) cluster mutations using the Bayesian approach (posterior probabilities) from the short read counts of B allele.

Roth et al. (2014) proposes a method called PyClone, which uses Hierarchical Dirich-

[7] Some researchers also call it single nucleotide polymorphism (SNP).

[8] Enzymes are used to cut the DNA sequence into multiple fragments

[9] In DNA sequencing, *E. coli* is used to isolate the each fragments from others.

let process as the prior. To compute the likelihood, they consider three types of subpopulations to be present in tumour samples:

(a) Normal subpopulation, which contains the normal cells,

(b) Reference subpopulation, which contains the tumour cells, but do not exhibit mutation i,

(c) Variant subpopulation, which contains tumor cells and exhibit mutation i.

Moreover, they assume that the number of B-allele short reads $\bar{a}$ follows the Binomial distribution, and computes as follows:

$$P(\bar{a}_i|N_i, \mathcal{G}_i, \phi, \phi_0) = \mathcal{B}in(N_i, P_b(\mathcal{G}_i, \phi, \phi_0)) \tag{2.4}$$

where the expected probability of B allele reads P_b is

$$P_b = \frac{\phi_0 C(\mathcal{G}_i^{(n)})B(\mathcal{G}_i) + (1-\phi_0)(1-\phi_t)C(\mathcal{G}_i^{(r)})B(\mathcal{G}_i) + (1-\phi_0)\phi_t C(\mathcal{G}_i^{(v)})B(\mathcal{G}_i)}{\phi_0 C(\mathcal{G}_i) + (1-\phi_0)(1-\phi_t)C(\mathcal{G}_i) + (1-\phi_0)\phi_t C(\mathcal{G}_i)}. \tag{2.5}$$

ϕ_0 and ϕ_t are frequencies of the normal and variant populations, respectively. $\mathcal{G}_i^{(n)}$, $\mathcal{G}_i^{(r)}$ and $\mathcal{G}_i^{(v)}$ are the genotypes of the normal, reference and variant subpopulations of the mutation i, respectively. $C(\mathcal{G})$ and $B(\mathcal{G})$ are functions which give the total and B allele copy number from genotype $\mathcal{G}$, respectively. Interestingly, this method is dependent on the inference methods for predicting genotypes.

Jiao et al. (2014) also infer clusters of mutations by posterior probability. They use a tree structured stick breaking process prior (Adams, Ghahramani, & M.I.Jordan, 2010) to cluster and to compute the prior probabilities. Similar to PyClone, they assume that B-allele short reads follow the binomial distribution, but the expected probability function P_b is different from PyClone:

$$P_b = \phi_t P_b^{(t)} + (1-\phi_t)P_b^{(n)} \tag{2.6}$$

where $P_b^{(t)}$ and $P_b^{(n)}$ are the probabilities of sampling the B allele from the tumour and normal populations, respectively. Their proposed method is known as PhyloSub. The advantage of PhyloSub over PyClone is that PhyloSub is not dependent on the output of genotype prediction methods.

Deshwar, Vembu, and Morris (2015) improve PhyloSub by incorporating treeCRP (tree structured Chinese Restaurant Problem) to cluster and compute the posterior probabilities. This modified method is known as treeCRP (Deshwar, Vembu, & Morris, 2015). PhyloSub, PyClone and treeCRP work on only the single nucleotide variations (SNVs), and not on the copy number variations (CNVs). Later, Deshwar, Vembu, Yung, et al. (2015) improve PhyloSub further by considering the presence of CNVs and SNVs within the tumour population to cluster mutations. They consider any of the following four scenarios of a mutation:

One. Only SNV appears,

Two. Only CNV appears,

Three. SNV appears after CNV,

Four. SNV appears before CNV,

For each scenario, they assume the value of B-allele short reads counts would be different. Therefore, Deshwar, Vembu, Yung, et al. (2015) modify the computation of the expected probability function P_b. They introduce two terms: The number of copies of the reference allele $C_i^{(r)}$ and B allele $C_i^{(v)}$ at i. The copy numbers of reference allele and B allele would be different in each scenario. In scenarios one and two: $C_i^{(r)} = C_i^{(r)} + \phi_u \times 2$ (u is the reference subpopulation) and $C_i^{(v)} = C_i^{(v)}$. In scenarios three and four: $C_i^{(r)} = C_i^{(r)} + \phi_u \times 2$ (u is the reference subpopulation) and $C_i^{(v)} = C_i^{(v)} + \phi_u$ (u is the variant subpopulation). Finally, the expected probability function P_b is changed as below:

$$P_b = \frac{C_i^{(r)}(1-\epsilon) + C_i^{(v)}\epsilon}{C_i^{(r)} + C_i^{(v)}} \tag{2.7}$$

where ϵ is the probability of reading the reference allele when the location contains the variant allele. This method is known as *PhyloWGS*.

K. Yuan et al. (2015) also use posterior probability to cluster mutations, which is exactly the same as the PhyloSub. But they use different technique to discover the tumour phylogeny, which we discuss later in this section. This method is known as *BitPhylogeny*. Similar to PhyloWGS, Jianga et al. (2016) infer clusters by assuming that both SNVs and CNVs are present in the tumour samples. But they develop a single equation to capture the expected probability function P_b for the four scenarios (mentioned earlier), which is as follows:

$$P_b = \frac{C_{mut} \cdot \phi_{cancer}}{2 \times (1 - \phi_{cancer}) + C_{total} \times \phi_{cancer}} \tag{2.8}$$

where, ϕ_{cancer} is the purity of the cancer cell, which is user defined. C_{mut} and C_{total} are the mutant-allele copy number and total copy number, respectively. This method is named *Canopy*.

After clustering mutations, most of the methods discover the phylogeny of the clusters to find tumour clones and their frequencies. PyClone, Clomial and SciClone do not predict the tumour phylogeny from the mutations clusters. These three methods assume the predicted clusters to be the tumour clones and the frequencies of the clusters as the clonal frequencies. But in the real world, an old clone may change to a set of new clones by harbouring a set of new mutations and become extinct in the tumour sample. Hence, PyClone, Clomial and SciClone predict more clones than the real clones.

Other methods infer tumour phylogeny to predict tumour clones and their clonal frequencies. TrAp (Strino et al., 2013), AncesTree (El-Kebir et al., 2015), LICHeE (Popic et al., 2015), Canopy (Jianga et al., 2016) and PASTRI (Satas & Raphael, 2017) create a $k \times m$ binary matrix B, where the entry B_{ij} is one if mutation j is present in the clone i, and zero otherwise. k and m are the number of clusters and mutations, respectively. From this binary matrix the tumour phylogeny is created in following fashions:

Ancestor clone and descendent clone prediction: From the binary matrix, these methods select a cluster as the root node of the tree, which contains all of the mutations and these mutations are also present in all other clusters. These methods select a cluster as an internal node if it contains the mutations of its parent nodes and some new mutations which are also present in its descendant nodes. Finally, these methods select leaf nodes clusters, which contain all of the old mutations of their ancestor nodes and some new mutations.

Maintaining the children sum to parent condition: According to the *children sum to parent condition*, the clustering frequency of an internal node would be equal or greater than the sum of the clustering frequencies of its children nodes. These methods maintain this relationship to infer the internal nodes of the tree.

Existing clone prediction: These methods assume the leaf node clusters as the existing clones. Moreover, these methods select an internal node as existing clones if the clustering frequency of this internal is greater than that of its children.

As mentioned earlier, TreeCRP (Deshwar, Vembu, & Morris, 2015), PhyloSub (Jiao et al., 2014), PhyloWGS (Deshwar, Vembu, Yung, et al., 2015) cluster mutations into several clusters either by tree structured Chinese restaurant process or by tree structured stick breaking process (TSSB), respectively. In both processes, two more hyperparameters γ and λ are used to control the tree topologies with concentration parameter α. The hyperparameters α and λ determine the number of nodes (subclones) in the tree, λ also affects the height of the tree and γ affects the width of the tree. These three hyperparameters are sampled as part of MCMC sampling e.g. Gibbs sampling. After defining the tree topologies, Deshwar, Vembu, Yung, et al. (2015) Deshwar, Vembu, and Morris (2015) and Jiao et al. (2014) place clusters with lower frequencies into the leaf nodes of the tree, and then place the rest of the clusters in the tree by maintaining the following frequency relationships between the internal nodes and their children:

$$\phi_{internal} = freq_{internal} - \sum_{\forall\ u \in\ Child(internal)} \phi_u \tag{2.9}$$

where, $freq_{internal}$ is the cluster frequency of the cluster *internal*, and $Child(internal)$ is a function denoting the children of the node *internal*.

Hajirasouliha et al. (2014) predicts tumour phylogeny from clustered BAFs in a different way. In Rec-BTP, they modify the children sum to parent condition as follows:

$$freq_{internal} = \sum_{\forall u \in Child(C)} \phi_u \tag{2.10}$$

Rec-BTP starts with a cluster with the highest frequency i.e. 1.00 and selects as the root. Then it selects two clusters whose frequencies satisfies the equation (2.10). If the sum of the two next higher clusters are not equal to the frequency of the root node, Rec-BTP selects the next highest cluster and introduce a dummy cluster called *auxiliary node*, and makes them the children of root node. The frequency of the auxiliary node is as followed:

$$\phi_{auxiliary} \;=\; \phi_{root} \;-\; \phi_{(next\ highest\ frequncy\ cluster)} \tag{2.11}$$

For the next highest node, Rec-BTP uses the same technique to find its children. Rec-BTP continues until all clusters are placed in the tumour evolutionary tree. Therefore, Rec-BTP produces many auxiliary nodes inside the tumour phylogeny, which is one of the major drawbacks of Rec-BTP.

All of the methods mentioned above (PhyloSub, PhyloWGS, TreeCRP, Rec-BTP, TrAp, AncesTree, LICHeE, Canopy and PASTRI) infer the tumour phylogeny assuming that the phylogenetic tree is a binary tree. Malikic et al. (2015) use Beyer-Hedetniemi algorithm to predict the n-ary tumour phylogeny.

BitPhylogeny (K. Yuan et al., 2015) use a mixture of two Laplace distributions to model the parent–child relation,

$$P(\phi_i|\phi_{parent(i)}, \Lambda, \vartheta, w) = \prod_{i=1}^{K} w_i Laplace(\vartheta, \Lambda) + (1 - w_i) Laplace(-\vartheta, \Lambda) \tag{2.12}$$

where ϑ defines the location of a positive and a negative mode. Intuitively, the positive mode generates parameters that give a high probability of observing mutation events, whereas the negative mode has the opposite effect. The hyperparameter Λ models variation within the modes. The weights w_i and $(1 - w_i)$ of the two Laplace densities specify the probabilities of either mode being selected for sampling the child parameter.

The performance of all of the above discussed methods were tested on the synthetic and real cancer data e.g. AML (Acute Myeloid Leukemia), CLL (Chronic Lymphocytic Leukemia), TNBC (Triple Negative Breast Cancer), HGSOC (High Grade Serous Ovarian Cancer) etc. Each method shows its efficient performance on predicting tumour clones and the clonal frequencies.

Despite the performance, the above mentioned methods have the following serious drawbacks:

(a) They assume that all mutations appear independently. But in reality, a mutation appears under the effect of other mutations. Modeling this important feature is missing in these methods.

(b) Except PhyloWGS and Canopy, they consider mutations are SNVs.

(c) They assume that only a single type of mutation can appear at a location. But in the real world, mutations are different in clones. None of the above discussed methods consider that mutations are clone specific.

In this thesis, we develop a method to infer clones with their mutations, and frequencies to resolve the above mentioned drawbacks.

2.3 Inferring tumour clones with genotype

From the cancer research literature (Freese, 1959; Caraco, 1998; Anand et al., 2008; Hanahan & Weinberg, 2011), mutations appear under the influence of other previous mutations. This influence is known as the position specific effect, and it is a stochastic process.

Moreover, genotype is a representation of a mutation along with its types. In genotype, ‘A’ and ‘B’ are used to represent the two SNP alleles inherited from parents. Different kinds of chromosomal abnormalities such as copy number gain/loss and LOH[10] are modeled by genotypes. Genotype also represent the zygosity[11] of a mutation. Table 2.1 lists all mutation genotypes that are used in different methods. Genotypes not only represent a point mutation but also the copy number of the mutated part of the genome.

Table 2.1: The genotype variable space

copy number	Genotype state	Genotype	Description
0	0	∅	Nullizygous[12]
1	1	A	Hemizygous[13]
	2	B	
2	3	AA	Copy neutral with LOH
	4	AB	Normal copy
	5	BB	Copy neutral with LOH
3	6	AAA	Three copies with LOH
	7	AAB	Three copies with duplication of **A** allele
	8	ABB	Three copies with duplication of **B** allele
	9	BBB	Three copies with LOH
4	10	AAAA	Four copies with LOH
	11	AAAB	Four copies with duplication of **A** allele
	12	AABB	Four copies with duplication of both alleles
	13	ABBB	Four copies with duplication of **B** allele
	14	BBBB	Four copies with LOH
4	15	AAAAA	Five copies with LOH
	16	AAAAB	Five copies with duplication of **A** allele
	17	AAABB	Five copies with duplication of both alleles
	18	AABBB	
	19	ABBBB	Five copies with duplication of **B** allele
	20	BBBBB	Five copies with LOH

However, genotype of a mutation can be predicted by capturing the position specific effects. Therefore, the dependency among the genotypes of the mutations can be modelled by a Markov model.

From DNA sequencing, it is not possible to detect the genotypes of mutations directly. As mentioned earlier, DNA sequencing produces short reads of a DNA sequence.

[10] Loss of heterozygosity (LOH) is a cross chromosomal event that results in loss of the entire gene and the surrounding chromosomal region (Joseph et al., 2014). Human cells contain two copies of the genome, one from each parent. Each human copy contains approximately 3 billion bases (adenine (A), guanine (G), cytosine (C) or thymine (T)). For the majority of positions in the genome the base present is consistent between individuals, however a small percentage may contain different bases (usually one of two; for instance, 'A' or 'G') and these positions are called 'single nucleotide polymorphisms' or 'SNPs'. When the genomic copies derived from each parent have different bases for these polymorphic regions (SNPs) the region is said to be heterozygous. Most of the chromosomes within somatic cells of individuals are paired, allowing for SNP locations to be potentially heterozygous. However, one parental copy of a region can sometimes be lost, which results in the region having just one copy. The single copy cannot be heterozygous at SNP locations and therefore the region shows loss of heterozygosity (LOH). Loss of heterozygosity due to loss of one parental copy in a region is also called hemizygosity in that region.

[11] Zygosity is the degree of similarity of the alleles for a trait in an organism.

[12] Both alleles are missing at genomic location.

[13] One allele is missing at genomic location.

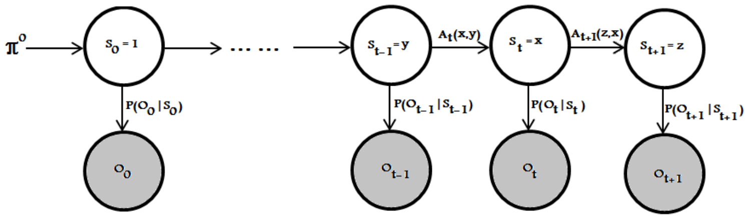


Figure 2.1: Probabilistic graphical model of an ordinary HMM

Therefore, it is important to predict the genotype of a mutation from the short reads. A special Markov chain model is required to infer genotypes from short reads, where genotypes are hidden states and short reads are observations. Hidden Markov Model (HMM) is a suitable Markov model to infer genotypes from short reads.

In HMM, there are three types of probabilities required to infer hidden variables (or hidden states):

- Transition probabilities $P(S_t = x|S_{t-1} = y) = A_t(x, y) \;\; \forall t > 0$;
- Emission probabilities $P(\boldsymbol{O_t}|S_t, \theta)$;
- Initial probabilities $P(S_0) = \pi(S_0) = \pi^0$;

where S_t and $\boldsymbol{O_t}$ are the hidden states, and observations at t location, respectively. θ denotes the parameters of HMM.

Several HMM based methods are developed to infer genotypes of a sequence of mutations appear in tumour genome. GPHMM (A. Li et al., 2011) (*Global Parameter Hidden Markov Model*) is one of the pioneer HMM-based method to predict genotypes of mutations. In GPHMM, it is assumed that only a single type of mutation can appear at one location. Each hidden state is expressed as $S_t = (\mathcal{G}^{(t)}, \mathcal{G}^{(n)})$ where $\mathcal{G}^{(t)}$ and $\mathcal{G}^{(n)}$ are the genotype of the tumour and normal populations at location t, respectively. As inputs, it takes the number of B-allele short reads $\bar{a}_t$, and LRR[14] l_t at a location t. A. Li et al. (2011) assume that $\bar{a}_t$ and LRR are Gaussian distributed. The emission

[14] Log ratio of normal-tumour contents. $l_t = \log \frac{a_t^{(n)}}{a_t^{(v)}+\bar{a}_t^{(v)}} = \log \frac{a_t^{(v)}+\bar{a}_t^{(v)}}{\psi_t}$. where $a_t^{(v)}$ and $\bar{a}_t^{(v)}$ are the short read counts of matched and B-allele at a location t of tumour, respectively. $a_t^{(n)}$ is the short read count of normal cells at a location t. Together LRR and BAF express the zygosity of a mutation along with the copy number (A. Li et al., 2011).

probabilities of $\bar{a}_t$ is defined as:

$$P(\bar{a}_t|\mathcal{G}_t, \theta) = \mathcal{N}(\bar{a}_t, \mu_{\bar{a}_t}, \sigma_b) \tag{2.13}$$

where $\mathcal{G}_t$ is the genotype of the tth location: $\mathcal{G}_t = [\mathcal{G}_t^t, \mathcal{G}_t^n]$, μ_t is the mean of B-allele reads, which is as follows

$$\mu_{\bar{a}_t} = \phi_t \frac{B(\mathcal{G}_t^t)}{C(\mathcal{G}_t^{(t)})} + (1 - \phi_t) \frac{B(\mathcal{G}_t^{(n)})}{C(\mathcal{G}_t^{(n)})}$$

where $B(\mathcal{G}_t)$ and $C(\mathcal{G}_t)$ are the functions to get the B-allele copy number and total copy number for any genotype $\mathcal{G}_t$ at t location, respectively. ϕ_t is the pre-assumed frequency of the tumour population at tth location. σ_b is the variance of the B-allele short reads.

The emission probabilities of LRR l_t is defined as

$$P(l_t|\mathcal{G}_t, \theta) = \mathcal{N}(l_t, \mu_{l_t} + o_t, \sigma_l) \tag{2.14}$$

where o_t and σ_l are the LRR base-line shift, which depend on the ploidy of the location t and the variance of LRR, respectively. The mean μ_{l_t} is as follows

$$\mu_{l_t} = \phi_t C(\mathcal{G}_t^{(t)}) + (1 - \phi_t) C(\mathcal{G}_t^{(n)})$$

Initial probabilities are defined as follows

$$\pi^0 = \begin{cases} \frac{1-P(0)}{K-1} & \text{If } \mathcal{G}_0 > 0 \\ P(0) & \text{If } \mathcal{G}_0 = 0 \end{cases}$$

where $P(0)$ is the initial probability of LRR base-line shift, which is set to 10^{-4}. K is the number of hidden states. The transition probabilities of GPHMM is as follows

$$A_t(\mathcal{G}_t, \mathcal{G}_{t-1}) = \begin{cases} 1 - P(0) - P(f) & \mathcal{G}_t = \mathcal{G}_{t-1}, \mathcal{G}_t > 0, \mathcal{G}_{t-1} > 0 \\ \frac{P(f)}{N-2} & \mathcal{G}_t \neq \mathcal{G}_{t-1}, \mathcal{G}_t > 0, \mathcal{G}_{t-1} > 0 \\ P(f) & \mathcal{G}_{t-1} = 0 \\ \frac{1-P(f)}{-1} & \mathcal{G}_t = 0, \mathcal{G}_{t-1} > 0 \end{cases} \tag{2.15}$$

where $P(f)$ is the initial transition probability between two different non-fluctuation[15] states, which is set to be 10^{-5}. Initial values of the global parameters o, σ_b and σ_l are set to 0, 0.2 and 0.03. The authors pre-assume the value of ϕ_t for each t location.

A. Li et al. (2011) employ the EM algorithm to update parameters $\theta = (\sigma_b, \sigma_l, o)$ and transition probabilities. In the E step, they use partial log-likelihood functions for $\bar{a}_t$ and l_t. In the M step, the Baum-Welch algorithm and Newton-Raphson method are used to estimate the transition probabilities.

In GPHMM, it is assumed that the number of clones at a mutant location t is one. But in the real world, multiple clones can harbour the same mutant location. This problem is improved by Xia et al. (2013) in their method TH-HMM (*Tumor Heterogeneity-Hidden Markov Model*). In TH-HMM, Xia et al. (2013) assume that the presence of two clones, instead of one clone, at any location. This leads to three combinational cases for any location:

One. Mutation appears at clone C_1 only.

Two. Mutation appears at clone C_2 only.

Three. Mutation appears at clones C_1 and C_2 both.

Except the concept of the presence of two clones at any location, there is no difference between TH-HMM and GPHMM.

Similar to GPHMM, in TH-HMM, it is assumed that all data are Guassian distributed. Yu et al. (2014) observe that, the number of total short reads N follows an

[15] Fluctuation state is the nullizygous state of a location. In GPHMM, [N/A,AA], [N/A,AB] and [N/A,BB] are fluctuation states

over-dispersed distribution. Several effects (e.g. deletion or amplification of short reads) cause on large difference between the total short reads covering all locations. Yu et al. (2014) also found that the number of B-allele short reads $\bar{a}$ follows the Binomial distribution. Therefore, Yu et al. (2014) proposed new emission probabilities of B allele short reads $\bar{a}_t$ in their method *CLImAT* (CNV and LOH Assessment in Impure and Aneuploid Tumors), which is as follows

$$P(\bar{a}_t | N_t, \mathcal{G}_t, \theta) = \mathcal{B}in(N_t, P_b(\mathcal{G}_t, \phi_t)) \tag{2.16}$$

where P_b is the expected probability of the number of B-allele reads, which is as follows

$$P_b(\mathcal{G}_t, \phi_t) = \phi_t \frac{B(\mathcal{G}_t^{(t)})}{C(\mathcal{G}_t^{(t)})} + (1 - \phi_t) \frac{B(\mathcal{G}_t^{(n)})}{C(\mathcal{G}_t^{(n)})}$$

In CLImAT, over-dispersed total reads are modeled by the negative binomial distribution. The emission probabilities of the total short reads at the location t, is as follows

$$\begin{aligned} P(N_t | \mathcal{G}_t, \theta) &= \mathcal{NB}(N_t, \lambda_t, P_b) \\ &= \tfrac{\Gamma(N_t + \lambda_t)}{\Gamma(N_t + 1)\Gamma(\lambda_t)} P_b^{N_t} (1 - P_b)^{\lambda_t} \end{aligned} \tag{2.17}$$

where λ_t is the expected number of read counts, which is as follows

$$\lambda_t = \frac{P_N}{2} \lambda + o$$

λ is the mean value of copy neutral read count. The expected probability of total reads P_N is as follows

$$P_N = \phi_t C(\mathcal{G}_t^{(t)}) + (1 - \phi_t) C(\mathcal{G}_t^{(n)})$$

GPHMM, TH-HMM, CLImAT assume that only one type of mutation appears at a location. For this reason, these methods cannot detect genetic make-up of a mutation which is clone specific, i.e. clone specific genotypes.

In TH-HMM and CLImAT, the frequencies of two clones C_1 and C_2 are user defined. Yau (2013) improve these methods using single variant population frequency ϕ_t. They assume that at a location t, there are three types of populations:

- The normal cell population with frequency ϕ_0,
- The tumour cell populations which do not harbour t mutation, with frequency $(1-\phi_t)(1-\phi_0)$, and
- The tumour cell populations which harbour the mutation t, with frequency ϕ_t.

It takes the number of B-allele and total short reads as inputs. Unlike CLImAT, it assumes that the number of total reads follows the student t-distribution. The exact number of total reads are unknown due to the loss or amplification of DNA fragments or change of copy numbers. Student t-distribution is a distribution whose mean is known, but standard deviation is unknown. Due to the unknown number of exact total reads, Yau (2013) cannot compute its standard deviation. So they assume that the number of the total short reads follows the student t-distribution. The emission probabilities of the total short reads are as follows

$$P(N_t|\mathcal{G}_t, \phi_0, \phi_t, \sigma_n, \nu) = \mathcal{S}tudent(\mu_t, \sigma_n, \nu) \tag{2.18}$$

where ν is the degree of freedom, which is set to 4. μ_t is the mean of total reads, which is as follows

$$\mu_t = (\phi_0 + (1-\phi_t)(1-\phi_0))C(\mathcal{G}_t^{(n)}) + (1-\phi_0)\phi_t C(\mathcal{G}_t^{(t)}) \tag{2.19}$$

Similar to CLImAT, Yau (2013) assumes that the number of B-allele short reads follows the Binomial distribution. The emission probabilities of the number of B allele short reads are as follows

$$P(\bar{a}_t|N_t, \mathcal{G}_t, \theta) = \mathcal{B}in(N_t, P_b) \tag{2.20}$$

Where

$$P_b = \frac{(\phi_0 + (1-\phi_t)(1-\phi_0))B(\mathcal{G}_t^{(n)}) + (1-\phi_0)\phi_t B(\mathcal{G}_t^{(t)})}{(\phi_0 + (1-\phi_t)(1-\phi_0))C(\mathcal{G}_t^{(n)}) + (1-\phi_0)\phi_t C(\mathcal{G}_t^{(t)})} \tag{2.21}$$

P_b is the expected probability of the B-allele reads at a location t. Yau (2013) also use the improved transition probabilities, which capture better position specific effect of mutations (proposed by Colella et al. (2007)), as follows

$$A_t(\mathcal{G}_t|\mathcal{G}_{t-1}) = \begin{cases} \rho_t & \mathcal{G}_t = \mathcal{G}_{t-1} \\ \frac{1-\rho_t}{K-1} & \text{Otherwise} \end{cases} \tag{2.22}$$

where $\rho_t = 1 - \frac{1}{2}[1 - e^{-(\frac{d_t}{2L})}]$. d_t and L are the distance between two genotypes ($\mathcal{G}_t$ and $\mathcal{G}_{t-1}$) and average length of short reads, respectively.

Unlike previous methods, Yau (2013) does not employ any algorithm to learn the parameters of OncoSNP-SEQ. Rather, they consider the pre-assumed range of values of the parameters $\theta = (\nu, \phi_0, \phi_t)$. Without requiring a matched normal sample, OncoSNP-SEQ fully explores genotypes of each location. It captures the position specific effect successfully. Similar to CLImAT, GPHMM and TH-HMM, OncoSNP-SEQ also made assumption on the appearance of single type of mutation at any location t. Sometimes pre-assumed parameters cannot predict accurate genotype, specially when the normal-tumour cells ratio is 55:45 or more (Ha et al., 2014).

In parallel to GPHMM, Ha et al. (2012) developed a method to predict genotypes, named Apolloh. It takes the number of total reads, B-allele reads and LRR as inputs. They have assumed that the number of B-allele reads follows the Binomial distribution, and LRR follows the normal distribution. Emission probabilities of the B-allele reads and LRR are as follows

$$P(\bar{a}_t|\mathcal{G}_t, \theta) = \mathcal{B}in(N_t, P_b) \tag{2.23}$$

where $P_b = \phi_t \frac{B(\mathcal{G}_t^{(t)})}{C(\mathcal{G}_t^{(t)})} + (1 - \phi_t)\frac{B(\mathcal{G}_t^{(n)})}{C(\mathcal{G}_t^{(n)})}$ is the expected probability of B-allele reads at a location t. and

$$P(l_t|\mathcal{G}_t, \theta) = \mathcal{N}(l_t, \mu_t, \sigma_l) \tag{2.24}$$

where $\mu_t = \phi_t C(\mathcal{G}_t^{(t)}) + (1 - \phi_t)C(\mathcal{G}_t^{(n)})$ is the mean of LRR at a location t. σ_l is the variance, which is set to 10^{-3}. It uses the same transition probabilities of OncoSNP-SEQ

(Yau, 2013). Initial probabilities are computed using Dirichlet process prior. Ha et al. (2012) employs the EM algorithm to estimate the parameters. In the E step, Apolloh computes the expectation of the complete-data likelihood. In the M step, it updates its parameters $\theta = (\boldsymbol{\phi_t}, \alpha)$ using maximum a posteriori (MAP) technique. It captures position specific effect successfully to predict the genotype. But similar to TH-HMM, CLImAT and OncoSNP-SEQ, it cannot infer clone specific genotypes.

TH-HMM, CLImAT, OncoSNP-SEQ and Apolloh assume that a tumour sample can have only two tumour clones. But in the real world, each of the tumour sample may have more than two clones and each clone contains a set of mutations. Therefore, it is also important to cluster mutations to detect clones. But none of the above methods predict clusters of mutations. Ha et al. (2014) revises their method by incorporating the concept of Factorial Hidden Markov Model[16] to predict genotype and the cluster id of a mutation t. Ha et al. (2014) rename their method as TITAN. In TITAN, they build two chains FHMM: One chain for the genotype G and another for the cluster id Z. The emission probabilities are exactly the same as Apolloh, but the expected probability of B allele reads P_b at location t and μ_t the mean of LRR follow the equations (2.21) and (2.19) of OncoCNP-SEQ respectively. The transition probabilities for genotypes are the same as equation (2.22). The transition probabilities for the cluster id are as follows

$$T_t(Z_t|Z_{t-1}) = \begin{cases} \rho_Z & Z_t = Z_{t-1} \\ 1 - \rho_Z & \text{Otherwise} \end{cases} \tag{2.25}$$

where $\rho_Z = 1 - \frac{1}{2}[1 - e^{-(\frac{d_t}{2L_Z})}]$. L_Z is the average distance between two mutations of same cluster. TITAN does not require any external clustering algorithm to find the clones of the tumour sample. Similar to the previous methods, it assumes only one type of mutation appear at any location t.

TITAN and previous all methods use log-likelihood as the objective function. According to Giraud (2014), log-likelihood has the issue of over-fitting model. For this

[16] Factorial hidden Markov model or FHMM is an HMM which contains n number of hidden chains instead of one chain.

reason, Yu et al. (2017) improve their method CLImAT to predict the clusters of the mutations along with the genotypes by using BIC as the objective function. Following TITAN, Yu et al. (2017) use Factorial HMM, where one chain is depicting aberration state sequence, and other is delineating corresponding clonal clusters. Moreover, they extend emission probabilities of read counts and B-allele read depth for capturing the cluster id. Given the aberration state G and the kth clonal cluster, they assume B-allele read depth is the binomial distributed with the conditional probability defined as follows:

$$P(\bar{a}_i|N_i, \boldsymbol{\mathcal{G}}, k) = \binom{b_i}{N_i}(\frac{B(\mathcal{G}_{t,k})}{C(\mathcal{G}_{t,k})})^{\bar{a}_i}(1 - \frac{B(\mathcal{G}_{t,k})}{C(\mathcal{G}_{t,k})})^{N_i - \bar{a}_i} \tag{2.26}$$

where $B(\mathcal{G}_{t,k})$ and $C(\mathcal{G}_{t,k})$ are functions to get B allele and total copy number for the genotype $\mathcal{G}_{t,k}$, respectively. These functions are defined as below

$$B(\mathcal{G}_{t,k}) = C_t^{(n)} BAF_t^{(n)}(1-\phi_k) + C_t^{(v)} BAF_t^{(v)}\phi_k \quad C(\mathcal{G}_{t,k}) = C_t^{(v)} BAF_t^{(v)}(1-\phi_k) + C_t^{(v)}\phi_k \tag{2.27}$$

where $C_t^{(n)}$ and $BAF_t^{(n)}$ denote the copy number and expected B allele frequency (BAF) of the normal cells, respectively, $C_t^{(v)}$ and $BAF_t^{(v)}$ represent the copy number and BAF of tumour cells, respectively. In addition, they assumed that read counts is negative binomial (NB) distributed, which is as follows:

$$p(N_t|\lambda, P_b, \boldsymbol{\mathcal{G}}, k) = \frac{\Gamma(N_t + \lambda_{\mathcal{G},k}\frac{1-P_b}{P_b})}{\Gamma(N_t+1)\Gamma(\lambda_{\mathcal{G},k}\frac{1-P_b}{P_b})}(1-P_b)^{\lambda_{\mathcal{G},k}\frac{1-P_b}{P_b}} P_b^{N_t} \tag{2.28}$$

where, λ is the mean read counts associated with normal copy and $\lambda_{G,k}$ is formulated as

$$\lambda_{\mathcal{G},k} = \frac{C(\mathcal{G}_{t,k})}{2}\lambda \tag{2.29}$$

The rest of all probabilities and parameter estimations are the same as CLImAT. They also use the transition probabilities of cluster id from TITAN. To capture better genotype and clusters of the mutation, they use BIC as the objective function:

$$BIC_{CLImAT-HET} = -\mathcal{L} + \frac{\lambda}{2} m_C \cdot ln(T) \tag{2.30}$$

where λ and m_C the regularization parameter and the number of free parameters.

Except OncoSNP-SEQ, all of the above discussed methods are evaluated on the synthetic and the real data e.g. triple negative breast cancer. But these methods have a number of serious drawbacks:

1. They consider the effect of adjacent previous mutation on the current mutation. Whereas Ellenbroek and v. Rheenen (2014) observe that mutations in genes of the same pathway have strong dependencies, known as long-range mutational influences. This important type of dependency is ignored in all of the methods

2. To infer genotypes, they assume the clonal frequencies are pre-defined.

3. All method specially, TITAN and CLImAT-HET, assume that a mutation appears only in one clone. But in the real world, genomic aberration can appear in different clones in different way. Therefore, no method predict the clone specific genemic aberrations.

According to above discussed drawbacks, our research objective is to develop an Factorial HMM which considers the presence of multiple mutations effect on one mutation including the types of effect.

2.4 Gaussian graphical model structure discovery

In Chapter 1, we mentioned that we set a research objective (second objective) to develop a method which works on the long-range mutational influences to predict statistically better tumour heterogeneity. According to Ji et al. (2016), since the existence of three dimensional structure of DNA, mutations with long-range influences are located close to each other and genes of these mutations form a pathway. Therefore, genes of same pathway can skeleton the list of mutations having the long-range influences. We, therefore, use the known gene-gene interaction networks to find genes and their pathways. From genes, their pathways, and their gene locations, we identify the mutations with the long-range mutational inter-dependencies.

All gene-gene interactions can not be found in the known gene-gene interaction networks. For this reason, we set another objective (third objective) for the thesis to discover the gene-gene interaction networks from the cancer data. It is well known that, gene expression data is an important source to discover gene-gene interaction networks. Moreover, most of the gene expression data is Gaussian distributed (Shokirov, 2013). Therefore, we plan to develop a method to discover gene-gene interaction networks as the structure of Gaussian Graphical Model (GGM). In this section, we discuss existing methods to discover the structure of GGM from the high dimensional data.

Let $\mathcal{D} = \{X_1, \ldots, X_n\}$ be a training set consisting of n data points where $X_i \in \mathbb{R}^d$ and d is the number of dimensions (equivalently attributes, or random variables). It is assumed that the observed input vectors have been generated from a multivariate Gaussian distribution $\mathcal{D} \sim \mathcal{N}_d(\mu, \Sigma)$. The aim is to discover the unobserved undirected Gaussian graphical structure $G = (V, E)$, where V is the set of vertices, each of them corresponds to a random variable (or a dimension of the input vectors), and E is the set of edges capturing the interaction (statistical associations between) random variables, from the observed/sampled vectors in $\mathcal{D}$. Interestingly, the dependency structure among the variables corresponds to non-zero entries of the precision matrix $\mathcal{K} = \Sigma^{-1}$ in Gaussian graphical models (GGM) (Koller & Friedman, 2009). Here, Σ is the covariance matrix. Hence, methods to discover the GGM from the high-dimensional data focus on inferring the precision matrix from the data.

Initially, Gaussian graphical models (GGMs) are discovered based on the exhaustive search algorithms. These algorithms are classified into three categories:

Forward selection: This algorithm starts with the simplest model with no edge (i.e. $E = \emptyset$). Edges are added incrementally, as long as the objective function finds an optimal GGM.

Backward elimination: This algorithm starts with the complete graph over the $|V|$ vertices, and edges are deleted incrementally as long as the new hypothesised models are not rejected according to the objective function.

Forward-backward combination: This algorithm is the combination of forward selection and backward elimination algorithms. In forward step, the algorithm initially starts from an empty edge list E and then adds the edges incrementally as long as it improves the objective function at least by ϵ_E, otherwise the algorithm terminates. Then, in the backward step, the algorithm checks one or more of the previously added edges that do not contribute at least ν_E to the objective function, then the algorithm removes them from the edge list E. This procedure ensures that at each round, the objective function is improved by at least $(1 - \nu_E)\epsilon_E$.

Deshpande et al. (2001) proposed a forward selection based greedy algorithm to discover GGMs from the high-dimensional data. They maintain the chordality[17] in GGM, which gives the statistical guarantee to compute the objective function. They use Maximum Likelihood Estimate (MLE) as the objective function. Due to the use of MLE based objective function, their method faces the over-fitting problem and not efficient when the number of samples is far less than the number of variables.

Noting on the drawback of Deshpande et al. (2001)'s method, Jalali, Johnson, and Ravikumar (2011) propose an algorithm to discover the structure of a graphical model based on the forward-backward algorithm, called *FoBa*. They use penalized likelihood as the objective function. Moreover, they define $\epsilon_E = \frac{c_n \cdot \log d}{n}$, where c_n is a constant between 0 and 1, and $\nu_E = 0.5$. However, FoBa predicts only 45% of true edges from the data.

To improve the performance of FoBa, Johnson, Jalali, and Ravikumar (2012) modify FoBa by introducing a new objective function and ϵ_E. The updated objective function is as follows:

$$\begin{aligned}\mathcal{L}(\mathcal{D}|\mathcal{K})_{Johnson} &= -\frac{d}{2}\log 2\pi - \frac{1}{2}\log|\Sigma| exp\Big\{-\frac{1}{2}\sum_{i=1}^{n}(\mathcal{D}_i - \mu)\Sigma^{-1}(\mathcal{D}_i - \mu)^T\Big\} \\ &\quad + tr\{\Sigma^{-1}(\mathcal{K} - \hat{\mathcal{K}})\} - tr\{\mathcal{K}_0(\mathcal{K} - \hat{\mathcal{K}})\} + (\lambda - \log(1+\lambda)) \quad (2.31)\end{aligned}$$

[17] In graph theory, a "chordal graph" is one in which all cycles of four or more vertices have a chord, which is an edge that is not part of the cycle but connects two vertices of the cycle. Equivalently, every induced cycle in the graph should have exactly three vertices (West, 2001).

where $\lambda = (\sqrt{c\frac{\log d}{n}})$. c is the constant tuning parameter, set to 0.5. Johnson et al. (2012) modified ϵ_E to $\frac{deg_{max}(G)\cdot\log d}{n}$. $deg_{max}(G)$ is the function to find the maximum degree of the graph G. Their method is known as *FoBa-Johnson.* FoBa-Johnson showed it good performance on random network, but not well on star, hub and small world networks.

J. Liu et al. (2014) observed that neither FoBa nor FoBa-Johnson estimate ϵ_E from the data, and it affects the performance. Therefore, they estimate ϵ_E using the precision matrix $\mathcal{K}$, as below:

$$\epsilon_{FoBa\text{-}gdt} = |\mathcal{K}|\sqrt{\frac{\log d}{n}} \tag{2.32}$$

J. Liu et al. (2014) call their algorithm *FoBa-gdt.* However, greedy based GGM discovery algorithms suffer from the poor performance when the number of sample is far less than the number of variables.

It is well know that the dependency structure among variables corresponds to non-zero entries of the precision matrix in Gaussian graphical models (GGM) (Koller & Friedman, 2009). To predict the precision matrix, Tibshirani (1996) presents a relation between the precision matrix and regression coefficient. More specifically, Tibshirani (1996) describes that the neighbours of each node can be found by regressing that variable against the remaining variables. Therefore, Lasso (***L**east **A**bsolute **S**hrinkage and **S**election **O**perator*) can be used to infer the precision matrix efficiently. Therefore, some methods have been developed using Lasso to predict the precision matrix from the high dimensional data.

For the first time, Meinshausen and Buhlmann (2006) propose a method using Lasso to estimate the precision matrix, named *NLasso.* Using Lasso, the method updates the precision matrix in row-by-row fashion. NLasso estimates of ith row of precision matrix $\hat{\mathcal{K}}_i$ is as follows:

$$\hat{\mathcal{K}}_i = \underset{\hat{\mathcal{K}}_i:\hat{\mathcal{K}}_{ii}=0}{\arg\min}\left\{\frac{\|D_i - \mathcal{D}\mathcal{K}\|_2^2}{n} + \lambda\|\mathcal{K}\|_1\right\} \tag{2.33}$$

where λ is the regularization parameter. During the computation of the precision matrix, Meinshausen and Buhlmann (2006) assign zeros to the diagonal entries $\mathcal{K}_{ii}$. The method keeps updating the rows of the precision matrix until the pseudo maximum likelihood finds the optimal solution. Due to zeros in the diagonal entries of the precision matrix, NLasso does not produce a positive definite precision matrix when the size of the sample is far less than the number of variables. Therefore, the precision matrix would not be invertible and likelihood can not be estimated. Moreover, in the real world, GGMs are not densely connected graph, rather these are sparse networks. The level of sparsity is not maintained in NLasso.

M. Yuan and Lin (2007) propose a new Lasso based method by assuming that observations are suitably centred and scaled; and the diagonal elements of the sample precision matrix equal to one to maintain the positive definiteness. They estimate the ith row of the precision matrix as follows

$$\hat{\mathcal{K}}_i = \arg\min_{\hat{\mathcal{K}}} \left(2 \sum_{i<j} \Sigma_{ij} \hat{\mathcal{K}}_{ij} + \sum_{i} \Sigma_{ii} \hat{\mathcal{K}}_{ii} - \log \left(| \sum_{i} \hat{\mathcal{K}}_{ii} I^i + \sum_{i<j} \hat{\mathcal{K}}_{ij} I^{ij} | \right) \right) \tag{2.34}$$

subject to $\sum_i \hat{\mathcal{K}}_{ii} I^i + \sum_{i<j} \hat{\mathcal{K}}_{ij} I^{ij}$ being positive definite, where I^i is an $n \times n$ matrix with the (i,i)th entry being 1 and all other entries being 0. I^{ij} is an $n \times n$ matrix with the (i,j)th and the (j,i)th entries being 1 and all other entries being 0. M. Yuan and Lin (2007) update each of the rows until the following penalized likelihood based objective function finds the optimal solution:

$$\mathcal{L}(\mathcal{D}|\mathcal{K})_{Yuan} = -\log |\hat{\mathcal{K}}| + tr\{\hat{\mathcal{K}}\Sigma\} + \lambda \sum_{i \leq j} \mathcal{K}_{ij} \tag{2.35}$$

where $\lambda = \frac{\log n}{n}$ is the regularization parameter to control the sparseness of the graph. Interestingly, a larger value of λ corresponds to a sparser solution that fits the data less well. A smaller λ corresponds to a solution that fits the data well but is less sparse. Therefore, the choice of λ is an important issue and it should be data dependent. Whereas, the regularization parameter proposed by M. Yuan and Lin (2007) is not well defined.

In parallel to M. Yuan and Lin (2007), Banerjee, Ghaoui, and d'Aspremont (2007) used the Block Coordinate Descent algorithm to estimate the precision matrix. They also define a new penalized objective function to estimate GGMs better, which is as follows:

$$\mathcal{L}(\mathcal{D}|\mathcal{K})_{COVSEL} = -\log|\hat{\mathcal{K}}| + tr\{\hat{\mathcal{K}}\Sigma\} + \lambda||\mathcal{K}|| \tag{2.36}$$

They estimate λ in the following manner:

$$\lambda = \arg\max_{\mathcal{K}} \frac{\mathcal{N}(\mathcal{D}, 0, \Sigma)}{\sqrt{n - 2 + \mathcal{N}(\mathcal{D}, 0, \Sigma)^2}} \tag{2.37}$$

Banerjee et al. (2007) call their method *COVSEL*. COVSEL is not scalable method to discover GGMs. Therefore, J. Friedman et al. (2008) improve COVSEL with respect to faster computation and call it ***Graphical Lasso*** or *GLasso*. Later, T. Wang et al. (2016) improve GLasso in terms of the computational time by using cyclical coordinate descent algorithm. This improved GLasso is known as FastGGM. COVSEL, GLasso, FastGMM use the same objective function to estimate the optimal graphical structure.

(Ledoit & Wolf, 2004) showed that the largest and the smallest sample eigenvalues tend to increase with d/n and affect the stability of GLasso. To improve the stability of Glasso, Avagyan et al. (2017) propose using the k-root of the sample covariance matrix, with $k \geq 1$, to attain less spread eigenvalues, and therefore, obtain a more stable estimation of $\hat{\mathcal{K}}^{-\frac{1}{k}}$ and also $\hat{\mathcal{K}}$. The proposed k-root Glasso algorithm is a simple modification of Glasso, but now subject to its k-root inverse to be close to the k-root of Σ. Note that although the proposed methodology needs to select an additional parameter k, it improves the statistical performance without increasing the computational time significantly. Moreover, once the specific k-root and the penalty parameter (associated with the original Glasso framework) are selected, the proposed procedure requires less computational time than that of Glasso. Therefore, they propose a new k-root l_1 penalized maximum likelihood estimates as the objective function, which is as follows:

$$\mathcal{L}(\mathcal{D}|\mathcal{K})_{r\text{-}GLasso} = \log|\hat{\mathcal{K}}^{-\frac{1}{k}}| - tr\{\Sigma^{\frac{1}{k}}\hat{\mathcal{K}}^{-\frac{1}{k}}\} - \lambda||\hat{\mathcal{K}}^{-\frac{1}{k}}|| \tag{2.38}$$

Other than GLasso and its modified versions, several Lasso based methods have been developed to discover GGMs by proposing either new objective function or regularization parameters. Rothman, Bickel, Levina, and Zhu (2008) use the matrix decomposition to estimate the precision matrix, which can be simply written as

$$\mathcal{K} = W^T W \tag{2.39}$$

where, $W = [w_{ij}]$ is a lower triangle matrix. They call their method ***S**parse **P**ermutation **I**nvariant **C**ovariance **E**stimator* or *SPICE*. Rothman et al. (2008) define a new objective function, which is as follows:

$$\mathcal{L}(\mathcal{D}|\mathcal{K})_{SPICE} = tr\mathcal{K}\Sigma - \log|\mathcal{K}| + \lambda \sum_{i \neq j} |\mathcal{K}_{ij}|^q \tag{2.40}$$

SPICE also suffers from the slow computation.

Lam and Fan (2009) modified SPICE (Rothman et al., 2008) to improve the rate of convergence and sparsity of the precision matrix by introducing a new penalised MLE, and a new regularization parameter. They use Gaussian quasi-likelihood:

$$\mathcal{L}(\mathcal{D}|\mathcal{K})_{SCAD} = tr\Sigma\mathcal{K} - \log|\mathcal{K}| + \sum_{i \neq j} p_\lambda(|\mathcal{K}_{ij}|) \tag{2.41}$$

where, $p_\lambda = \lambda^2 - (|\mathcal{K}| - \lambda)^2$ is the new regularization parameter.

Shen, Pan, and Zhu (2012) focus on consistent and sharp parameter estimation to improve rate of covergence by computing L_0-constrained likelihood. According to Shen et al. (2012), L_0-constrained or regularized likelihood is as below:

$$\mathcal{L}(\mathcal{D}|\mathcal{K})_{Shen} = tr\mathcal{K}\Sigma - \log|\mathcal{K}| + \frac{\lambda_1}{\lambda_2} \sum_{i=1}^{d} min(|\Sigma_i|, \lambda_2) \tag{2.42}$$

where λ_1 is the regularization parameter (i.e.$\lambda = \frac{\log n}{2}$) to control the sparsity. λ_2 is another regularization parameter which controls the degree of approximation (i.e. decides which individual coefficient to be shrunk towards zero.) None of the regularization parameters are data-dependent.

T. Sun and Zhang (2013) identify that previous Lasso based methods suffer from the problem of poor rate of convergence. T. Sun and Zhang (2013) assume that Σ is a nonnegative-definite data matrix and $\mathcal{K}$ is a positive-definite target matrix with $\Sigma\mathcal{K} \approx I$. They describe the relationship between positive-definite matrix inversion and linear regression, and propose an estimator for $\mathcal{K}$ via scaled Lasso. They use the same objective function of GLasso, but new regularization parameter $\lambda = \sqrt{\frac{2\log d}{n}}$ is estimated. All of the above methods specify a specific sparsity pattern through a single regularization parameter. But in the real world, sparsity pattern is node specific.

From the compressed sensing and high dimensional linear regression literature, it is now well understood that constrainted l_1 minimization provides an effective way for reconstructing a sparse signal without specifying the sparsity pattern. Therefore, Cai et al. (2011) introduce constrained l_1 minimization in their linear programming method and penalized likelihood to estimate the precision matrix. They call their method ***C**onstrained l_1-minimization for **I**nverse **M**atrix **E**stimation* (*CLIME*). The penalized likelihood of CLIME is as follows:

$$\mathcal{L}(\mathcal{D}|\mathcal{K})_{CLIME} = -\log|\hat{\mathcal{K}}| + tr\{\Sigma\hat{\mathcal{K}}\} + \lambda||\hat{\mathcal{K}}||_\infty \tag{2.43}$$

Moreover, W. Liu and Luo (2015) focus on both rate of convergence and node specific sparsity pattern. Their method adds a separate regularization parameters to each column (assuming the sparsity of each node are different from others). Instead of using row fashion, they use column-by-column fashion for faster computation. They call their approach ***S**parse **C**olumn-wise **I**nverse **O**perator* (*SCIO*). The new objective function proposed in SCIO is as follows:

$$\mathcal{L}(\mathcal{D}|\mathcal{K})_{SCIO} = -\log|\hat{\mathcal{K}}| + tr\{\hat{\mathcal{K}}\Sigma\} + \sum_{i=1}^{d}\lambda_i||\mathcal{K}|| \tag{2.44}$$

where $\lambda_i = \frac{4i}{n}$. Though SCIO performs well in predicting the precision matrix on synthetic data, but due to the poor estimation of regularization parameter, it predicts many false associations.

All of the above mentioned methods suffer from the estimation of regularization parameter. H. Liu (2017) proposed a tuning insensitive method based on SQRT Lasso. It automatically adapts the unknown sparsity pattern by setting regularization parameter λ constant to $\pi\sqrt{\frac{\log d}{n}}$. They call this method *TIGER* (***T**uning-**I**nsensitive **G**raph **E**stimation and **R**egression*).

However, current Lasso based methods for graphical model structure discovery with continuous variables suffer from the following drawbacks:

- discover many false edges i.e. the false discovery rate is higher (T. Wang et al., 2016)(Avagyan et al., 2017)(Chiong & Moon, 2017).
- sacrificing the computational cost (Avagyan et al., 2017),(H. Liu, 2017).
- The regularization parameters are not data dependant (Chiong & Moon, 2017),(Hirose, Fujisawa, & Sese, 2017).

In this thesis, we propose a method with an minimum message length based objective function to discover the many true edges and reduce the false discovery rate.

2.5 Context-specific graphical models structure discovery

Graphical models represent multivariate distributions by explicitly expressing inter-dependencies, particularly suitable in the analysis of the high dimensional data (Lauritzen, 1996). In standard GGM, it is assumed that all observations are generated from the same graphical model (Meilă & Jordan, 2000). However, the real datasets exhibit heterogeneity, which can be accommodated through the use of mixtures of GGMs, to let each cluster exhibit different inter-dependencies among variables, a.k.a ”context-specific-dependencies” (Meilă & Jordan, 2000; Rodriguez et al., 2011). Moreover, typically there are far less number of samples (i.e. observations) compared to the number of variables, from an unknown number of components (i.e. cluster) which makes the conditional dependency discovery challenging, particularly for the high dimensional heterogenous data. Therefore, some methods have been developed to discover the context-specific GGMs from heterogeneous data.

Methods that discover the structure of context-specific graphical models first cluster the data into K components, and then estimate the precision matrix of each component. Meilă and Jordan (2000) are the pioneers to discover context-specific dependencies from the high dimensional continuous data. They use K-mean clustering to cluster the data into K components, and Chow and Liu (1968) proposed *ChowLiu* algorithm to predict context-specific trees. This method uses the MLE as the objective function. Moreover, this method is developed for both discrete and continuous data. Due to the use of MLE as objective function, this method is not efficient when the number of samples is far less than the number of variables. It also does not discover the context-specific graphical models.

Guo et al. (2011) first develop a method to discover context-specific Gaussian graphical models by cluster the data into K components, and then predict the GGM of each component separately using J. Friedman et al. (2008)'s proposed GLasso. They modified GLasso for heterogeneous data using two regularization parameters λ_1 and λ_2 to control the sparsity of the precision matrices. λ_1 and λ_2 control the sparsity of the presence of an edge between two nodes in any of the categories and the difference between categories, respectively. Guo et al. (2011) call their modified GLasso *Fused Graphical Lasso* (*FGL*). To estimate context-specific GGMs, Guo et al. (2011) propose a penalized likelihood as the objective function:

$$\mathcal{L}(\mathcal{D}|\mathcal{K})_{FGL} = \sum_{k=1}^{K} \Big[-\log|\mathcal{K}^{(k)}| + tr\{\Sigma^{(k)}\mathcal{K}^{(k)}\}\Big] + \lambda_1 \sum_{k}\sum_{i\neq j} |\mathcal{K}_{k,ij}| + \lambda_2 \sum_{k<k'}\sum_{i,j} |\mathcal{K}_{k,ij} - \mathcal{K}_{k',ij}| \tag{2.45}$$

Guo et al. (2011) assume that the number of components K in the data is user define.

Rodriguez et al. (2011) propose a Bayesian approach to predict context-specific

GGMs from data. They use Dirichlet prior process to compute the prior probabilities of the mixture model, and structures of context-specific graphical models. They use MCMC sampling to find the optimal graph structures. The advantages of Rodriguez et al. (2011)'s proposed method are two-fold: (a) the number of the components in the data is not user defined and (b) using sampling to estimate optimal graph structures and clusters. Mohan et al. (2012) observe that all context-specific GGMs shares almost similar graphical structure with small difference. Whereas, both FGL and Rodriguez et al. (2011)'s proposed method do not predict the common graph structure.

Danaher, Wang, and Witten (2014) improve FGL for predicting the common graph structure along with context-specific GGMs by proposing a new objective function. This objective function captures not only the information across the clusters but also similar pattern of sparsity across all of the precision matrix. Danaher et al. (2014) call their method *Joint Graphical Lasso* (*JGL*). The modified objective function is as follows:

$$\mathcal{L}(\mathcal{D}|\mathcal{K})_{JGL} = \frac{1}{2}\sum_{i=1}^{K}\Big\{n_i\Big(\log|\mathcal{K}_i| - tr\{\Sigma_i\mathcal{K}_i\}\Big)\Big\} - P(\theta) \tag{2.46}$$

where $P(\theta)$ is the penalized function, which encourages precision matrices $\mathcal{K}_1, \cdots, \mathcal{K}_K$ share certain characteristics, such as values of the non-zero elements. Moreover, they consider that estimated precision matrices tend to be sparse. The penalty function is

$$P(\theta) = \lambda_1 \sum_{k}\sum_{i\neq j}|\mathcal{K}_{k,ij}| + \underbrace{\lambda_2 \sum_{k<k'}\sum_{i,j}|\mathcal{K}_{k,ij} - \mathcal{K}_{k',ij}|}_{\text{penalty-1}} + \underbrace{\lambda_2 \sum_{i\neq j}\Big(\sum_{k=1}^{K}\mathcal{K}_{k,ij}^2\Big)^{\frac{1}{2}}}_{\text{penalty-2}} \tag{2.47}$$

where, λ_1 and λ_1 are non-negative tuning parameters. The penalty-1 term borrows information aggressively across classes, encouraging not only similar network structure but also similar edge values. On the other hand, the penalty-2 term encourages a similar pattern of sparsity across all the precision matrices. Later, Ma and Michailidis (2016) improve JGL using group Lasso (Breheny & Huang, 2009) instead of using GLasso (J. Friedman et al., 2008), named as *JSEM*. In JGL and JSEM, it is assumed that the number of components in the data is user-defined.

Peterson, Stingo, and Vannucci (2015) improved the Rodriguez et al. (2011) proposed Bayesian approach to discover the common graph structure. They use a Markov Random Field prior (F. Li & Zhang, 2010; Stingo & Vannucci, 2011) for graph structures, spike-and-slab prior (George & McCulloch, 1997) on the off-diagonal entries for network similarity. One common limitation of the above mentioned methods is that they only borrow strength in the graph space, leaving the strength of selected edges (e.g. partial correlation) to be modeled/estimated independently.

Tao, Huang, Wang, Xi, and Li (2016) address the above mentioned drawback by assuming that all Gaussian graphs share a similar structure, but the values of the same edges in each graph can be different. The corresponding non zero elements in each precision matrix are allowed to have different signs. They combine the fast coordinate descent algorithm of GLasso with the majorization-minimization algorithm[18] to derive a new re-weighted algorithm which compute values of the same edges in each graph. They modified the $P(\theta)$ term of the objective function of JGL to discover the better graphs structures, which is as below,

$$P(\theta) = \lambda_1 \sum_{i=1}^{K} \|\mathcal{K}_i\|_0 + \lambda_2 \sum_{i=1}^{K} \{ \sum_{j=1, i\neq j}^{K} \|D(\mathcal{K}_i, 0) \bigodot \mathcal{K}_j\|_0 \} \tag{2.48}$$

where, $D(\mathcal{K}_i, 0)$ is an indicator matrix where each element d_{ij} is defined as:

$$d_{ij} = \begin{cases} 1 & \text{if } |\mathcal{K}_{ij}| \geq \tau \\ 0 & \text{Otherwise.} \end{cases} \tag{2.49}$$

$D(\mathcal{K}_i, 0) \bigodot \mathcal{K}_j$ only keeps elements of $\mathcal{K}_i$, where the corresponding elements of $\mathcal{K}_j$ are 0, and $\bigodot$ stands for dot product. The advantage of using updated $P(\theta)$ is that it maintains differences of corresponding edge values among graphs when estimating similarities in structure.

Except methods proposed by Rodriguez et al. (2011) and Peterson et al. (2015), all

[18] The MM algorithm is an iterative optimization method which exploits the convexity of a function in order to find their maxima or minima.

methods predict the components of the data at once. Therefore, Gao et al. (2016) uses the EM algorithm to cluster the data and to estimate context-specific GGMs. In the E-step, they use multivariate Gaussian mixture model to cluster the data. In the M-step, they use JGL to predict context-specific GGMs with their common structure and estimates their parameters. Gao et al. (2016) modified the $P(\theta)$ term of the objective function of JGL, which is as follows

$$P(\theta) = \lambda_1 \sum_{k} \sum_{i \neq j} J_T(|\mathcal{K}_{k,ij}|) + \lambda_2 \sum_{k<k'} \sum_{i,j} J_T(|\mathcal{K}_{k,ij} - \mathcal{K}_{k',ij}|) \tag{2.50}$$

where $J_T(|z|) = min(|z|, \tau)$ is the truncated Lasso penalty (TLP) (Shen et al., 2012) to control the degree of approximation. Gao et al. (2016) call their method *New-Structural-Pursuit* (*New-SP*).

At the same time W. Sun, Hao, Liu, and Cheng (2016) also proposed another similar EM algorithm, where in the E-step the method clusters the data and in the M-step estimates γ_i and μ_i (where γ_i and μ_i are the mixing coefficient and mean of the cluster i) and context-specific GGMs via a penalized procedure. They named their framework ***S****imultaneous* ***C****lustering* ***A****nd estimatio****N*** of heterogeneous graphical models (SCAN). In the E-step, they also define the penalized log-likelihood function for complete data, which is as follows:

$$\mathcal{L}(\mathcal{D}|\mathcal{K})_{SCAN} = \frac{1}{n} \sum_{i=1}^{n} \Big\{ \sum_{j=1}^{K} \Big(\log \gamma_j + \mathcal{L}_j(\mathcal{D}_j, \theta_j) \Big) - P(\theta_i) \Big\} \tag{2.51}$$

where, $P(\theta_i) = \lambda_1 \underbrace{\sum_{j=1}^{K} \Big(\sum_{l=1}^{d} |\mu_{jl}| \Big)}_{\mathcal{P}_1(\theta_1)} + \lambda_2 \underbrace{\sum_{j=1}^{K} \Big(\sum_{i \neq l} |\mathcal{K}_{ijl}| \Big)}_{\mathcal{P}_3(\theta_1)} + \lambda_3 \underbrace{\sum_{i \neq l} \Big(\sum_{j=1}^{K} \mathcal{K}_{jil}^2 \Big)^{\frac{1}{2}}}_{\mathcal{P}_3(\theta_1)}$ is the penalty function which impose sparsity of the estimated cluster mean by $\mathcal{P}_1(\theta_i)$, precision matrix by $\mathcal{P}_2(\theta_i)$ and similarity between among all estimated precision matrices by $\mathcal{P}_3(\theta_i)$. In the M-step, the update of the coefficient of class member is as below:

$$\gamma_i = \frac{1}{n} \sum_{j=1}^{n} \frac{\gamma_i P(\mathcal{D}_{ij}, \theta_i)}{\sum_{k=1}^{K} \gamma_l P(\mathcal{D}_{kj}, \theta_k)} \tag{2.52}$$

W. Sun et al. (2016) uses JGL to estimate the precision matrices $\mathcal{K}$.

Later, Fop, Murphy, and Scrucca (2017) improved the EM based methods to discover context-specific GGMs by introducing N. Friedman (1998)'s proposed *structural EM* algorithm (SEM algorithm). The algorithm allows the estimation of model parameters and inferring graph configurations by combining the standard EM algorithm and the penalized EM algorithm with a graph structure search. The SEM algorithm is used to maximize the penalized likelihood with respect to model parameters and graph structures. The algorithm alternates between the two standard steps, Expectation and Maximization. In addition, the M step includes the structure learning step to search for the optimal graph configurations within the mixture components. The penalised likelihood is as shown as below:

$$\mathcal{L}(\mathcal{D}|\mathcal{K})_{Fop} = \sum_{i=1}^{K}\left(\sum_{j=1}^{n}\left\{\hat{z}_{ji}\log\gamma_i pdf^i(D_{ji}|\mu_i,\Sigma_i,G_i)\right\}\right) - \sum_{i=1}^{K}P(\theta) \tag{2.53}$$

where, $P(\theta)$ is a function that penalizes the graph complexity. In the context of Gaussian graphical model selection, a natural penalty function is such that the score corresponds to the Bayesian Information Criterion (BIC) (Schwarz, 1978) of a Gaussian graph covariance model. In this case the function is given by:

$$P_{BIC}(\theta) = \frac{|E_i|\log n}{2} \tag{2.54}$$

For large dataset, when n and d are of comparable size, this score may select graphs that are overly complex. In this case, Foygel and Drton (2010) suggest an extended Bayesian information criterion (eBIC). The corresponding $P(\theta)$ function is given by:

$$P_{eBIC}(\theta) = \frac{|E_i|\log n}{2} + 2\binom{mE}{E_i}E_i\log d \tag{2.55}$$

The Erdos-Rényi model is a popular model for random graphs. Under this model, the probability of a graph G_i with E_i arcs is given by $\alpha^{E_i}(1-\alpha)^{T-E_i}$, where α is the probability that two nodes are associated (Erdős & Rényi, 1959). From this quantity, the following penalty function can be derived:

$$P_{RL}(\theta) = -E_i \log \alpha - (T - E_i) \log (1 - \alpha) \tag{2.56}$$

The previous $P(\theta)$ functions penalize in the same way graphs with equal number of edges but dissimilar configurations. However, in some situations some form of association structures may be preferred to others a priori. To assign different penalization to different structures defined on the same number of arcs, Fop et al. (2017) consider the following penalty function:

$$P_{PL}(\theta) = \gamma_i \sum_{j=1}^{d} \log (deg_{ji} + 1) \tag{2.57}$$

where is deg_{ji} is the degree of node j of graph G_i. The penalty is derived from a power law on the nodes of a graph of the form $\prod_j (deg_{ji} + 1)^{\gamma_i}$.

All of the methods use a specific sparsity pattern on each context-specific GGMs. Whereas, Cai et al. (2011) observed that the sparsity pattern of each node is different from others. B. Wang, Singh, and Qi (2016) focus on this issue and extend the CLIME (Cai et al., 2011) for multi GGMs settings. They named extended CLIME, SIMULE (*detecting* ***S****hared and* ***I****ndividual parts of* ***MUL****tiple graphs* ***E****xplicitly*).

However, current methods to discover the context-specific GGMs suffer from the following drawbacks:

- discover many false edges i.e. the false discovery rate is higher.
- sacrificing too much the computational cost.
- The objective function is not well defined.

In this thesis, we propose a method with an minimum message length based objective function to discover the context-specific GGMs with their common structure which reduces the false discovery rate.

In conclusion of this chapter, we focus on resolving the major drawbacks of existing methods by setting four research objectives (discussed in Chapter 1). In next subsequent four chapters, we discuss our methods to resolve drawbacks of the all of the above mentioned methods.

Chapter 3

Tumour heterogeneity prediction - Single sample

The research in this chapter has been published in the following article: Mohammad S Rahman, Ann E. Nicholson and Gholamreza Haffari “HetFHMM: A novel approach to infer tumour heterogeneity using factorial hidden Markov model", *Journal of Computational Biology*, 25(2): 182-193, 2018, `https://doi.org/10.1089/cmb.2017.0101`

3.1 Introduction

In the chapter, we present a statistical model to identify cancer clones and their genetic make-up from mixed and noisy short reads of a tumour sample. Our model discovers cancer clones harbouring copy number variations (CNV) and/or single nucleotide variations (SNV) as mutations. It allows mutations to belong to multiple clones, a phenomenon exhibited in cancer biology (Stratton et al., 2009), hence leading to more accurate modelling and prediction.(c.f. experiments in Section 3.4). Furthermore, it infers the size of cancer clones through a notion called *clonal frequencies*, showing the relative number of cells belonging to a cancer clone in a tumour sample.

In Section 3.2, we present our model called HetFHMM (*Tumour **Het**erogeneity pre-*

*diction by **F**actorial **H**idden **M**arkov **M**odel*) for detecting heterogeneity in cancer, which is based on factorial hidden Markov models. Given NGS (next generation sequencing) short reads of a patient, we present inference algorithm for predicting clonal frequency and genetic architecture. Since the gold-labeled data for this problem is not readily available, we then outline our evaluation setup in Section 3.4 to assess model predictions in this unsupervised learning scenario. We provide empirical results in Section 3.5 comparing our model against strong existing models from the literature in predicting clonal frequency and genetic make up.

3.2 Model

Our goal is to identify the tumour clones and their genetic make up from NGS short reads of a tumour sample. The genetic make up of a clone is the set of mutations harboured by the clone, and the genotype of those mutations. We use similar genotypes as of Table 2.1 by considering that they not only represent a point mutation but also the copy number of the mutated part of the genome.

The input to our model includes a list of mutated genome locations, the number of NGS short reads with match and mismatch nucleotides to the reference genome at those genomic locations, and the number of maximum clones believed to exist in the sample. The output of our model is then the set of mutations belonging to each clone, the genotype of those mutations in each clone, and the frequency of each clone.

The basis of our model is that proximity on the genome induces inter-dependencies among mutations, in the sense that adjacent mutations tend to have similar genotypes and belong to the same clone (Ha et al., 2012). More specifically, we model each clone as a sequence of random variables where each of which corresponds to a mutated location on the genome. Each random variable takes a genotype as its value, and the genotype of consecutive random variables are inter-dependent. These random variables are “latent”, but we note that they give rise to the “observed” counts from the data. We suggest a probabilistic graphical model which postulates a generative model of how the data is generated; we then reason about the latent variables of the model using the statistical inference.

Let us assume that we are given a collection of T mutated genome locations to find the genetic make up of a maximum of K clones. We denote the genotype of the clone k at the genomic location t by $\mathcal{G}_{t,k}$, and the frequency of the clone k by ϕ_k where $\sum_{k=1}^{K} \phi_k = 1$. We assume that the clone 1 corresponds to the normal cells, where the genotype for all genomic locations is AB where A and B represent the two alleles inherited from parents (A. Li et al., 2011). To infer the genotypes $\mathcal{G}_{t,k}$ as well as the cellular frequencies ϕ_i, the observed data for our model include the total short reads N_t covering the mutation t (aka the read depth), the number of matched a_t and mismatched $\bar{a}_t$ short reads (note that $N_t = a_t + \bar{a}_t$), and log ratio of the read depths in the tumour and normal samples.

Our statistical model is based on factorial hidden Markov models (FHHMs) (Ghahramani & Jordan, 1997), where each chain corresponds to a clone and the observations correspond to the counts observed from the NGS data see Figure-3.1.

In our model, henceforth referred to by HetFHMM, the joint probability of the latent variables and the observations is written as

$$P(\boldsymbol{\mathcal{G}},\boldsymbol{O}|\boldsymbol{\phi}) = \Bigg(\Big(\prod_{k=1}^{K} P(\mathcal{G}_{1,k})\Big) P(\boldsymbol{O}_1|\boldsymbol{\mathcal{G}}_1,\boldsymbol{\phi})\Bigg)\Bigg(\prod_{t=2}^{T}\Big(\prod_{k=1}^{K} P(\mathcal{G}_{t,k}|\mathcal{G}_{t-1,k})\Big)P(\boldsymbol{O}_t|\boldsymbol{\mathcal{G}}_t,\boldsymbol{\phi})\Bigg) \tag{3.1}$$

where $\boldsymbol{\mathcal{G}}_t$ is the vector of genotypes for all clones at mutation t, $\boldsymbol{O_t} = \{a_t, N_t, l_t\}$ is the observed data at mutation t, and $\boldsymbol{\phi}$ is the vector of clonal frequencies for all clones. Let's have a closer look into the elements of the model: (i) the term $P(\mathcal{G}_{t,k}|\mathcal{G}_{t-1,k})$ is called the "transition probability" and determines the dependency of the genotype of the next mutation conditioned on that of the current mutation, and (ii) the term $P(\boldsymbol{O}_t|\boldsymbol{\mathcal{G}}_t,\boldsymbol{\phi})$ is called the "emission probability" and determines the relationship of the observed data $\boldsymbol{O}_t$ and the latent variables $\boldsymbol{\mathcal{G}}_t$ at mutation t (conditioned on the clonal frequencies $\boldsymbol{\phi}_t$). In what follows, we provide in-depth explanation of these terms.

Transition Probability: The transition probability $P(\mathcal{G}_{t,k} = q|\mathcal{G}_{t-1,k} = r)$, which

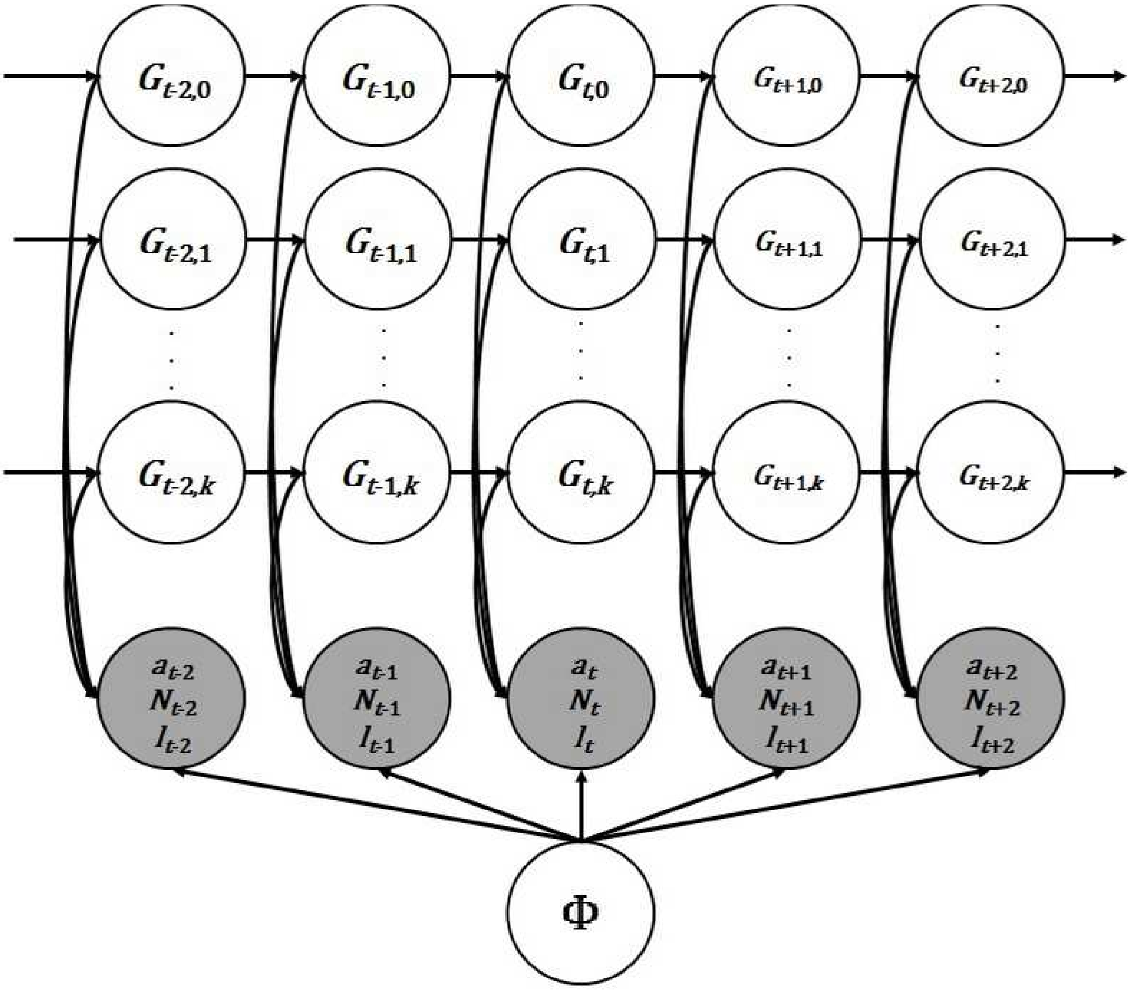


Figure 3.1: The probabilistic graphical model of our Factorial Hidden Markov Model for analysing heterogeneity (HetFHMM). Here, a_t, N_t and l_t are observation $\boldsymbol{O_t}$ of the model.

is denoted by $A_{t,k}(q,r)$ from the transition matrix $A_{t,k}$, captures the interdependencies between genotypes of adjacent mutations. Following Colella et al. (2007), we define the transition matrix for each chain as follows:

$$\underbrace{P(\mathcal{G}_{t,k} = q|\mathcal{G}_{t-1,k} = r)}_{A_{t,k}(q,r)} = \begin{cases} \rho_t & \text{if } q{=}r \\ \frac{1-\rho_t}{D_k-1} & \text{Otherwise.} \end{cases} \tag{3.2}$$

The stickiness of the genotype Markov process ρ_t is defines as

$$\rho_t \;=\; 1 - \frac{1}{2}(1 - e^{\frac{-d_t}{L}}) \tag{3.3}$$

where L is the average length of the sequence reads[1]. D_k is the dimension of the state space (i.e. the number of genotypes, which is 21 from Table 2.1), and d_t is the genomic distance between the mutant locations t and $t-1$.

Emission Probability: We decompose the emission probability for generating the observation $\boldsymbol{O}_t = \{a_t, N_t, l_t\}$ based on the hidden variables $\boldsymbol{\mathcal{G}}_t$ and $\boldsymbol{\phi}$, as follows:

$$P(\boldsymbol{O}_t|\boldsymbol{\mathcal{G}}_t, \boldsymbol{\phi}) = P(a_t|N_t, \boldsymbol{\mathcal{G}}_t, \boldsymbol{\phi})P(l_t|\boldsymbol{\mathcal{G}}_t, \boldsymbol{\phi}) \tag{3.4}$$

where a_t, N_t, and l_t are defined as before. We now elaborate the above two terms in the emission probability.

Following the previous work (Ha et al., 2012), we assume that a_t follows binomial distribution where the number of trials is N_t and the probability of success is as follows:

$$P_{b_t} = \frac{\sum_{k=1}^{K} \phi_k.r_{g_t^k}}{\sum_{k=1}^{K} \phi_k.c_{g_t^k}} \tag{3.5}$$

where $r_{g_t^k}$ and $c_{g_t^k}$ are number of reference allele and the copy number of the genotype in $\mathcal{G}_{t,k}$. For example, if the genotype is AAB, then $r_{g_t^k}$=2 and $c_{g_t^k}$=3.

The number of the short reads of tumour and normal cells are more than billion. As

[1] It was observed to be 2 Megabases (2×10^6 bases) in 104 breast tumours (rounded to the nearest Mb.) (Colella et al., 2007; Ha et al., 2012).

per central limit theorem, it is assumed that the log ratio of tumour-normal read depth is Gaussian distributed with mean μ_t and standard deviation σ. Mean of the log ratio of tumour-normal read depth is:

$$\tilde{\mu}_t = \frac{\sum_{k=0}^{K} \phi_k . c_{g_t^k}}{\phi_0 . c_{g_t^0} . \sum_{k=1}^{K} \phi_k . \psi} \tag{3.6}$$

where ψ be tumour ploidy[2] parameter which is set to 3[3]. $c_{g_t^0}$ is the copy number of genotype of normal clone at location t.

3.3 Inference

In order to infer the hidden variables $\boldsymbol{\mathcal{G}}$ and $\boldsymbol{\phi}$, we look for an imputation which maximises the likelihood of the data:

$$\begin{aligned} P(\boldsymbol{\mathcal{G}}, \boldsymbol{l}, \boldsymbol{a} | \boldsymbol{\phi}, \boldsymbol{N}, \sigma) = & \prod_{k=1}^{K} A_{1,k}(\mathcal{G}_{1,k}) \prod_{t=2}^{T} \prod_{k=1}^{K} A_{t,k}(\mathcal{G}_{t,k} | \mathcal{G}_{t-1,k}) \\ & \prod_{t=1}^{T} \mathcal{B}in(a_t | N_t, P_{b_t}) \times \mathcal{N}(l_t | \tilde{\mu}_t, \sigma^2) \end{aligned} \tag{3.7}$$

We repeatedly *alternate* between inferring the clone specific genotypes and the clonal frequencies until the convergence condition is met, i.e. we alternate between $\boldsymbol{\phi}$ and $\boldsymbol{G}$ to maximise the above likelihood function In what follows, we elaborate on these two phases of our optimisation algorithm.

Optimising $\boldsymbol{\phi}$ while $\boldsymbol{\mathcal{G}}$ is fixed. Maximising the likelihood function over $\boldsymbol{\phi}$ is a *constrained* optimisation problem since $\boldsymbol{\phi}$ is constrained to be a probability vector, i.e. $\sum_k \phi_k = 1$ and ϕ_k are non-negative. We make use of the *exponentiated gradient* (EG) algorithm to solve this constrained optimisation problem.

More formally, the constrained optimisation problem is as follows:

$$\min_{\boldsymbol{\phi} \in \triangle} -\mathcal{L}(\boldsymbol{\phi}) \tag{3.8}$$

[2] Ploidy is a measure of the number of chromosomes in a cell.

[3] According to Navin et al. (2014); Davoli, Uno, Wooten, and Elledge (2017), a typical tumour is triploid.

where $\mathcal{L}(\boldsymbol{\phi}) = -\log P(\boldsymbol{\mathcal{G}}, \boldsymbol{l}, \boldsymbol{a} | \boldsymbol{\phi}, \boldsymbol{N}, \sigma)$, and $\triangle$ is the simplex containing all probability vectors. To solve the above minimization problem, the EG updates are as follows:

$$\phi_k^{new} \propto \phi_k e^{-(\eta \nabla_{\phi_k} \mathcal{L}(\boldsymbol{\phi}))} \tag{3.9}$$

where η is the learning rate. After updating each component of the latent vector $\boldsymbol{\phi}$, the values are normalised so that they sum to one. For the EG updates, we need the derivatives, which are derived using the chain rule.

$$\begin{aligned}\mathcal{L}(\boldsymbol{\phi}) = & \sum_t \log \binom{N_t}{a_t} + a_t \log P_{b_t} + (N_t - a_t)\log(1 - P_{b_t}) \\ & + \log(\frac{1}{\sigma\sqrt{2\pi}}) - \frac{(l_t - \tilde{\mu}_t)^2}{2\sigma^2} + const \end{aligned} \tag{3.10}$$

$$\nabla \mathcal{L}(\boldsymbol{\phi}) = \sum_t [(\frac{a_t}{P_{b_t}} - \frac{N_t - a_t}{1 - P_{b_t}}) \cdot \nabla \tilde{\mu}_t + \frac{l_t - \tilde{\mu}_t}{\sigma^2} \cdot \tilde{\mu}_t] \tag{3.11}$$

$$\frac{dP_{b_t}}{d\phi_k} = \frac{c_{g_t^k} \cdot (r_{g_t^k} - P_{b_t})}{\sum_{k=0}^{K-1} \phi_i \cdot c_{g_t^k}} \tag{3.12}$$

$$\frac{d\tilde{\mu}_t}{d\phi_0} = \frac{c_{g_t^0} \cdot (1 - \tilde{\mu}_t)}{\phi_0 \cdot c_{g_t^0} + \sum_{k=1}^{K-1} \phi_k \cdot \psi} \tag{3.13}$$

$$\frac{d\tilde{\mu}_t}{d\phi_k} = \frac{c_{g_t^0} \cdot \psi \tilde{\mu}_t}{\phi_0 \cdot c_{g_t^0} + \sum_{k=1}^{K-1} \phi_k \cdot \psi} \tag{3.14}$$

Substitute $\frac{dP_{b_t}}{d\phi_k}$ and $\frac{d\tilde{\mu}_t}{d\phi_k}$ back to $\mathcal{L}(\boldsymbol{\phi})$, the gradient of the objective function can be found with respect to variable $\boldsymbol{\phi}$. We summarise the EG algorithm in algorithm 3.1.

Optimising $\boldsymbol{\mathcal{G}}$ while $\boldsymbol{\phi}$ is fixed. Since the exact inference in FHMM is intractable (Ghahramani & Jordan, 1997), we make use of Gibbs sampling as a Markov chain Monte Carlo (MCMC) method for approximate inference. Initially, we uniformly at random choose a genotype value for genotype variables in all chains except the normal chain where the genotypes are fixed to AB. Then, we sample each variable while the states of the rest of the variables are fixed. That is, the posterior probability of each genotype

Algorithm 3.1 EG algorithm for inferring clonal frequencies

```
while !$converged$ do
  for Clone $k = 0$ to $K$ do
    $\phi_k^{new} \leftarrow \phi_k^{old} \times \exp -\eta \nabla F(\phi_k^{old})$
  end for
  $\boldsymbol{\phi}^{new} \leftarrow normalize(\boldsymbol{\phi}^{new})$
  Compute $F(\boldsymbol{\phi}^{old})$
  Compute $F(\boldsymbol{\phi}^{new})$
  if $F(\boldsymbol{\phi}^{old}) - F(\boldsymbol{\phi}^{new}) > 0$ then
    $\boldsymbol{\phi} \leftarrow \boldsymbol{\phi}^{new}$
  end if
end while
```

for a hidden variable $\mathcal{G}_{t,k}$ is :

$$P(\mathcal{G}_{t,k}) \propto A_{t,k}(\mathcal{G}_{t,k}|\mathcal{G}_{t-1,k})A_{t+1,k}(\mathcal{G}_{t+1,k}|\mathcal{G}_{t,k}) \times \mathcal{B}in(a_t|N_t, P_{b_t})\mathcal{N}(l_t|\mu_t, \sigma) \tag{3.15}$$

The Gibbs sampling algorithm is shown in Algorithm 3.2.

Algorithm 3.2 Gibbs samplers for inferring clone specific genotypes

```
while !$converged$ do
  for mutation $t = 1$ to $T$ do
    for clone $k = 1$ to $K$ do
      for genotype $g = 1$ to $20$ do
        $P_{a_t} \leftarrow \mathcal{B}in(a_t|N_t, P_{b_t})$ ▷ $P_{b_t}$ is defined in eqn (3.5)
        $P_{l_t} \leftarrow \mathcal{N}(l_t|\tilde{\mu}_t, \sigma^2, \psi)$ ▷ $\tilde{\mu}_t$ is defined in eqn (3.6)
        $posterior[g] \leftarrow P(\mathcal{G}_{t,k} = g|\mathcal{G}_{t-1,k})P(\mathcal{G}_{t+1,k}|\mathcal{G}_{t,k} = g)P_{a_t}P_{l_t}$
      end for
      Sample $\mathcal{G}_{t,k}$ from normalised $posterior$
    end for
  end for
end while
```

3.4 Evaluation Framework

No existing model infers tumour clones and their genomic make-up along with the clonal frequencies from the tumour samples. Hence it is challenging to evaluate our model due to the lack of (a) appropriate data annotated with clones' details, (b) the most compatible models for comparison, and (c) suitable evaluation metrics. We focus on these issues in the rest of this section.

3.4.1 Real and Synthetic Data

Real Cancer Data. We make use of laboratory experimented Acute Myeloid Leukemia (AML) (Ding et al., 2012) as the real data. The data includes samples of 7 patients which were obtained after the chemotherapy treatment. The data is annotated with the clones and their mutations based on laboratory experiments which we use as gold standard; however, the annotation does not include the cellular prevalence of the clones.

Furthermore, Ding et al. (2012) performed the experiments in the laboratory to find the tumour heterogeneity evolution/ progression after the chemotherapy treatment. Due to unavailability of the real data, we use these available datasets. It is not a concern for our model whether the data is taken after or before the chemotherapy treatment.

Synthetic Cancer Data. To assess different aspects of our model, we generate synthetic cancer data containing the clone specific genotypes and their clonal frequencies (Table 3.1); Algorithm 3.3 summarises the process.

Table 3.1: Clone configurations for the synthetic data

Configuration	Clone data	Normal tissue	Tumour clones						
			1	2	3	4	5	6	7
1	3	0.20	0.28	0.52	-	-	-	-	-
	4	0.10	0.04	0.30	0.65	-	-	-	-
	5	0.01	0.05	0.12	0.22	0.60	-	-	-
	6	0.01	0.04	0.08	0.13	0.25	0.49	-	-
	8	0.01	0.02	0.04	0.07	0.13	0.15	0.22	0.36
1	3	0.30	0.20	0.50	-	-	-	-	-
	4	0.20	0.08	0.12	0.60	-	-	-	-
	5	0.18	0.06	0.09	0.15	0.52	-	-	-
	6	0.17	0.04	0.08	0.11	0.13	0.47	-	-
	8	0.15	0.02	0.04	0.06	0.08	0.10	0.13	0.42
1	3	0.17	0.38	0.45	-	-	-	-	-
	4	0.20	0.12	0.12	0.56	-	-	-	-
	5	0.11	0.11	0.11	0.15	0.54	-	-	-
	6	0.09	0.10	0.11	0.12	0.13	0.45	-	-
	8	0.03	0.05	0.07	0.09	0.10	0.10	0.16	0.40

To generate synthesis data, we first specify the number of clones and their clonal frequencies. We then generate the location of mutations on the genome. After randomly generating the location of the first mutation, we generate the gap between two consecutive mutations from a uniform distribution on [6K,7K] since the average distance between two consecutive mutations in the AML data is 6679 base-pair (Ding et al., 2012). After generating the locations, the genotypes and observations are sampled based on eqn (3.3) and eqn (3.5), respectively.

We would like to assess our model in data conditions where the interdependency assumptions made in the model about the adjacent genotypes do not hold. Therefore, we generate two more versions of the aforementioned datasets where the gap d_t in the computation of the stickiness of the genotypes Markov process ρ_t in eqn (3.3) is scaled by 3 and 1/3, which correspond to stronger and weaker interdependencies, respectively. This also provides a compelling test bed for the comparison against the competing models, PyClone and PhyloSub, which assume no dependency between the genotypes of adjacent mutations.

Algorithm 3.3 Synthetic data generation algorithm

```
Specify the clonal frequencies of clones $\Phi$
Generate mutation locations $\{1,..,T\}$
Set $\mathcal{G}_{1,0}$ .... $\mathcal{G}_{T,0}$ to AB $\triangleright$ the genotype of the normal clone
Uniformly at random generate the genotypes of $\mathcal{G}_{1,1}$... $\mathcal{G}_{K,1}$ $\triangleright$ first mutations
for each location $t \in \{1....T\}$ do
    for each clone $k \in \{1....\text{K}\}$ do
        Sample $\mathcal{G}_{t,k}$ from the transition matrix $A_{t,k}$ $\triangleright$ refer to eqn (4.3)
    end for
    Generate $a_t$ and $l_t$ based on the emission probability $\triangleright$ refer to eqn (3.4)
end for
```

3.4.2 Baseline Models

PyClone (Roth et al., 2014) and PhyloSub (Jiao et al., 2014) models discover the clonal architecture of a tumour, i.e. the frequency of clones and the set of mutations belonging to each clone. They cluster the mutations to identify the clones and to infer the clonal frequencies. PyClone takes the frequency of a cluster as the clonal frequency, whereas PhyloSub estimates the clonal frequencies differently.

HetFHMM produces two outputs: the clonal frequency and the clone specific genotypes (it is important to note that there is no existing method which infers the "clone specific genotypes" from the short reads). In HetFHMM, $\mathcal{G}_{t,i}$ specifies the genotype of the mutation t in the clone i, and whether the clone harbours this mutation or not.

Therefore, we can obtain the set of mutations belonging to a clone as well as the clonal frequencies from the output of HetFHMM, which can be then compared with the output of PyClone and PhyloSub.

3.4.3 Evaluation Metrics

We make use of *V-Measure* (Rosenberg & Hirschberg, 2007) and root mean-square distance ($RMSD$) to compare mutation clusterings and the clonal frequencies.

3.4.3.1 *V-Measure*

Rosenberg and Hirschberg (2007) has introduced an external based cluster evaluation metric *V-Measure* to quantify the quality of a predicted clustering with respect to the gold standard. It is an entropy-based measure which is a function of the completeness and homogeneity of predicted clusters with respect to the gold standard.

Homogeneity: Homogeneity is measure which computes the proportion of the members of a cluster contains only members of a single gold standard class (Rosenberg & Hirschberg, 2007). The notion of homogeneity is defined as follows:

$$h = 1 - \frac{H(Cls^g|Cls^p)}{H(Cls^g)} \tag{3.16}$$

where

$$H(Cls^g|Cls^p) = -\sum_{j=1}^{|Cls^p|}\sum_{i=1}^{Cls^g} \frac{b_{ij}}{|T|} \log \frac{b_{ij}}{\sum_{k=1}^{Cls^g} b_{ik}} \tag{3.17}$$

and

$$H(Cls^g) = -\sum_{j=1}^{Cls^g} \frac{\sum_{i=1}^{Cls^p} b_{ij}}{|Cls^g_j|} \log \frac{\sum_{i=1}^{Cls^p} b_{ij}}{|Cls^g_j|}$$

Cls^g, Cls^p, $|T|$, and b_{ij} are the set of gold standard clones, the set of the predicted clusters, the number of data points, and the number of data points that are the members of ith gold standard clone and jth predicted cluster.

Completeness: Completeness is measure which computes the proportion of the members of a gold standard clone contains only members of a predicted cluster

(Rosenberg & Hirschberg, 2007). The notion of completeness is defined as follows:

$$c = 1 - \frac{H(Cls^p|Cls^g)}{H(Cls^p)} \tag{3.18}$$

where

$$H(Cls^p|Cls^g) = -\sum_{j=1}^{|Cls^g|}\sum_{i=1}^{Cls^p}\frac{b_{ij}}{|T|}\log\frac{b_{ij}}{\sum_{k=1}^{Cls^p} b_{ik}} \tag{3.19}$$

and

$$H(Cls^g) = -\sum_{j=1}^{Cls^p}\frac{\sum_{i=1}^{Cls^g} b_{ij}}{|Cls_j^g|}\log\frac{\sum_{i=1}^{Cls^g} b_{ij}}{|Cls_j^g|}$$

$H(Cls^g|Cls^p)$ and $H((Cls^p|Cls^g)$ compute the non-homogeneity and incompleteness of the output, respectively. When $H(Cls^g|Cls^p)$ or $H(Cls^p|Cls^g)$ is zero, the output is either homogeneous or complete to the gold standard clone respectively. Moreover,the non-zero value of $H(Cls^g|Cls^p)$ and $H(Cls^p|Cls^g)$ express the degree of non-homogeneity and incompleteness, computed by equations 3.17 and 3.19 respectively.

The *V-measure* is then defined as the harmonic mean of the homogeneity and completeness scores:

$$V\text{-}measure = \frac{2 \times h \times c}{h + c} \tag{3.20}$$

For the prefect clustering[4], the *V-Measure* is one, and it is less than one for any imperfect clustering. *V-Measure* computes the degree of relative perfectness of clustered output.

3.4.3.2 *RMSD*:

We evaluate the predicted clonal frequencies with respect to the gold standard using *Root mean square distance* or *RMSD*. It is used to compute the distance or error between the clonal frequencies of predicted clusters and the gold standard:

$$RMSD \ = \sqrt{\frac{1}{|Cls^s|}\sum_{i=1}^{|Cls^s|}\|\phi_{Cls_i^s} - \phi_{Cls_i^g}\|^2 + \frac{1}{|\overline{Cls}^s|}\sum_{i=1}^{\overline{Cls}^s}\|\phi_{\overline{Cls}_i^s} - 0\|^2} \tag{3.21}$$

[4] Perfect clustering is the clustering output in which the predicted clusters are fully complete (i.e. c=1) and homogeneous ($h = 1$) to the gold standard clones.

where Cls^s and $\overline{Cls}^s$ are the significant and insignificant predicted clusters, respectively. Most of the methods produce many clusters than the gold standard. Among the predicted clusters, the some of the clusters contain the significant/ maximal number of mutations of gold standard clones, known as the significant clusters. The remaining clusters are known as the insignificant clusters. The better the quality of the predicted clones and their prevalences, the lower would be $RMSD$.

3.5 Results and discussions

In this section, we first compare the performance of HetFHMM vs PyClone and PhyloSub on synthetic data. Afterwards, we compare these models using real AML cancer data.

3.5.1 Synthetic Data

We investigate the mutation clusters and prediction accuracy of the cellular prevalence of each clone by computing $V\text{-}Measure$ and $RMSD$ scores, respectively. We let the number of clones in our model to be $\{3, 4, 5, 6, 7, 8\}$, and choose the one which produces the highest log-likelihood. For the synthetic data, we consider the clusters identified by the sampled clone-specific genotypes and predefined clonal frequencies as the gold standard.

Figure-3.2 shows the average $V\text{-}Measure$ results on the synthetic data. It contains four plots, where each of which corresponds to the results on data containing strong, normal, weak, and no dependency between the adjacent mutations. In each plot, there are five groups of results corresponding to synthetic data with different number of gold clones $\{3, 4, 5, 6, 8\}$. Based on Figure-3.2, some remarks are in order. Firstly, on all synthetic data conditions the performance of HetFHMM is superior compared to PyClone and PhyloSub. Further investigation of the results have revealed that HetFHMM has detected the same number of clones as the gold standard. However, PyClone and PhyloSub have detected much more clusters than the gold standard, which in turn has

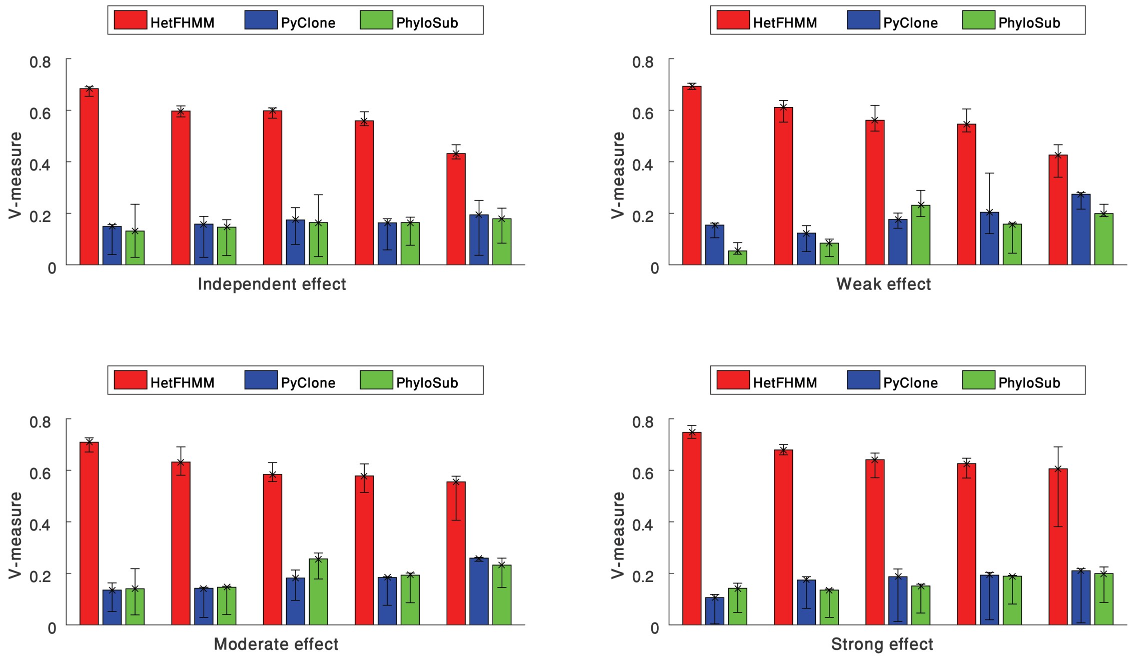


Figure 3.2: Average *V-measure* of outputs of HetFHMM, PyClone and PhyloSub on synthetic data. The four panels show the results on data generated with different degree of interdependency between adjacent mutations including strong, moderate, weak, and not interdependency. In each panel, there are multiple syntactic data, each of which corresponds to a different number of clones in $\{3, 4, 5, 6, 8\}$. For each data condition, our model (red bars) is compared with PhyloSub (green bars) and PyClone (blue bars).

affected their average V-$Measure$ scores. Secondly, as the interdependency between the adjacent mutations becomes stronger, the performance of HetFHMM improves. This is in contrast to the performances of PyClone and PhyloSub which do not change as adjacent mutations' influence changes. This is due to the fact that HetFHMM models the dependency between the adjacent mutations via the value of ρ in the transition probability (equation-4.3). Hence, in the data including strongly influenced adjacent mutations, ρ helps to predict the genotype of a mutation more accurately compared to the data containing weaker influence between adjacent mutations. On the other hand, PyClone and PhyloSub do not take into account the dependency between adjacent mutations, hence their performances do not change as the strength of the influence changes. Thirdly, the accuracy of cluster identification of HetFHMM decreases as the number of clones increase. This is in contrast to PyClone and PhyloSub which showed little improvement when the number of clones is increased (but still significantly outperformed by HetFHMM).

In the previous section, we have mentioned that a special type of synthetic data is generated where mutations do not have influences among themselves, which is the assumption made in PyClone and PhyloSub. Based on Figure-3.2, it is interesting to see that HetFHMM performs better than PyClone and PhyloSub even on this data condition. It indicates that HetFHMM can work better than the baselines no matter whether there is an interdependency between the adjacent mutations.

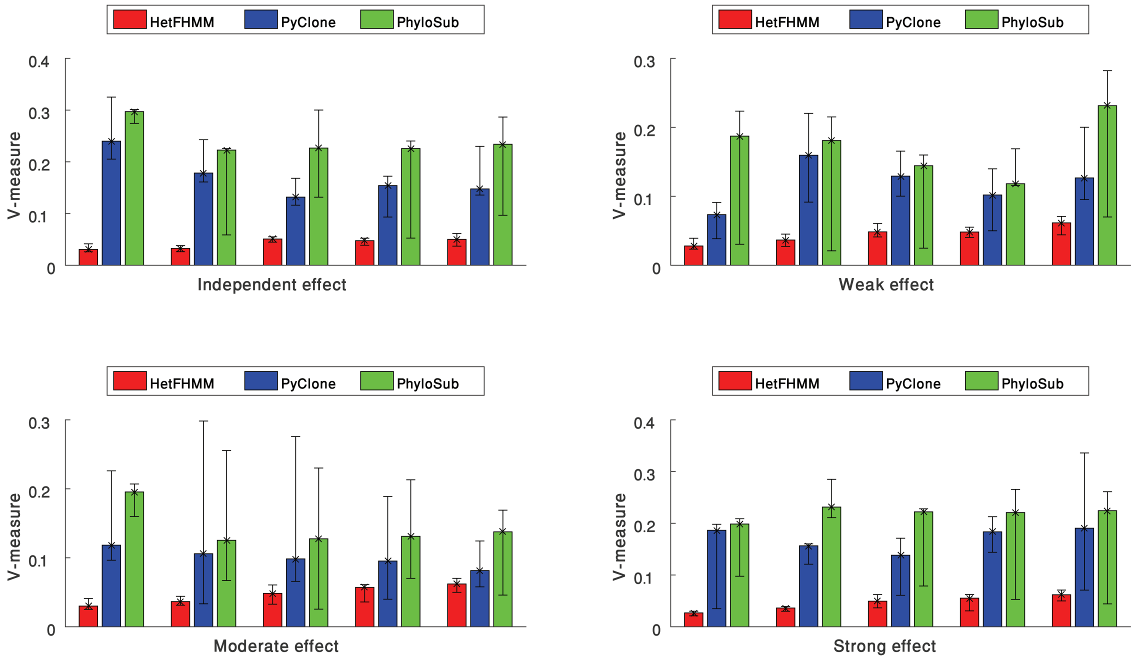


Figure 3.3: Average *RMSD* of outputs of HetFHMM, PyClone and PhyloSub on synthetic data. The four panels show the results on data generated with different degree of interdependency between adjacent mutations including strong, moderate, weak, and not interdependency. In each panel, there are multiple syntactic data, each of which corresponds to a different number of clones in $\{3, 4, 5, 6, 8\}$. For each data condition, our model (red bars) is compared with PhyloSub (green bars) and PyClone (blue bars).

In addition to *V-Measure* to evaluate the cluster output, we have evaluated the clonal frequencies predicted by HetFHMM, PyClone and PhyloSub using the *RMSD* score. Figure-3.3 presents the average *RMSD* errors for our model and the baselines on all synthetic data conditions. We see the similar trend that HetFHMM outperforms the baseline models on all data conditions. This is in part due to the more correctly predicted clones by HetFHMM.

3.5.2 Real Cancer Data

We apply our model to Ding et al. (2012)'s AML (described in subsection3.4.1) data, which is annotated with gold clones based on laboratory experiments. However, it is not annotated with the cellular prevalence of clones; therefore, we can only use *V-Measure* to test the quality of the predicted clones as *RMSD* is not applicable. For HetFHMM, we run the model with different number of clones from 3 to 10, and choose the one with the highest log likelihood.

Figure-3.4 shows the *V-Measure* scores as per patient sample. Based on the results, it is clear that HetFHMM has predicted the clusters more accurately than PyClone and PhyloSub for all patients. The accuracy of PyClone and PhyloSub has been almost the same for all cases, except the patients UPN869586 and UPN933124 where PhyloSub has detected more accurate clones compared to PyClone.

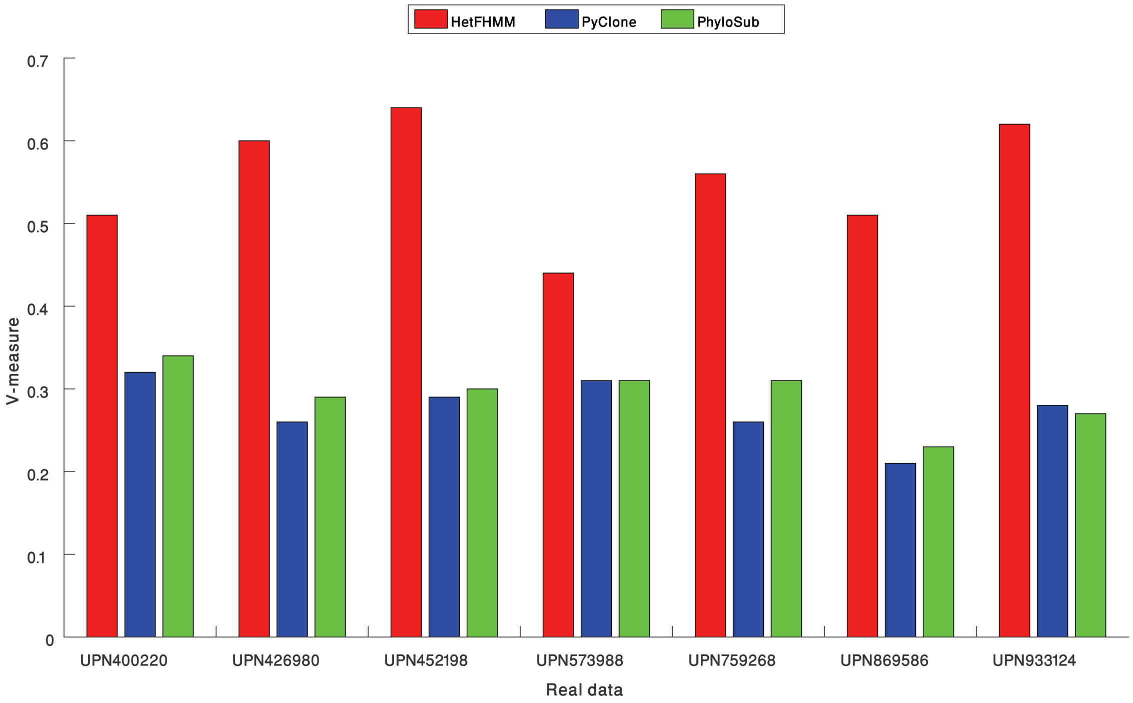


Figure 3.4: *V-Measure* of the clusters output of HetFHMM (red bars), PyClone (blue bars) and PhyloSub (green bars) on real cancer data from different AML patients.

3.6 Conclusion

We have develop a novel model, called HetFHMM, to identify the clonal architecture of a tumour sample based on next generation sequencing data. Our model discovers the mutations as well as the cellular prevalence of the clones in the sample. HetFHMM is based on Factorial Hidden Markov Models, whereby the genomic composition of each clone is represented by a hidden chain. The basis of the model is that the observed data is generated by a mixture of the underlying chains, where the mixing coefficients are the clonal prevalences. We make use of Gibbs sampling and exponentiated gradient algorithms to infer the clonal genomic compositions represented by the hidden chains as well as the clonal prevalences. The empirical results on synthetic and real cancer data confirms that our model outperforms strong baseline models PhyloSub and PyClone based on two evaluation metrics, i.e. V-$Measure$ and $RMSD$. The key to the stronger performance of HetFHMM compared to the baseline models is that it *jointly* infers the clone specific mutations and clonal frequencies to identify the tumour clones.

Following the literature, recent methods to infer tumour heterogeneity, are designed for multiple sample data and show better performance on multiple samples data. Whereas, HetFHMM can work on single sample data to infer tumour heterogeneity. Therefore, following research questions need to investigate to improve the performance of HetFHMM:

- How accurately will HetFHMM work on multiple sample data?
- How many samples will require to predict accurate or near accurate clone specific allelic composition of the mutations and clonal frequencies?
- Will HetFHMM perform better by incorporating long-range mutational interdependency?

We address the research questions as mentioned above to improve the performance of HetFHMM in next chpater.

Chapter 4

Tumour heterogeneity prediction - Multiple samples and long-range mutational influences

4.1 Introduction

In HetFHMM, as described in the Chapter 3, we identify tumour clones and their genetic make-up from short reads of a single tumour sample by assuming that tumour clones are not concentrated in a specific region. However, we know from the literature, this assumption is not realistic. More specifically, Ellenbroek and v. Rheenen (2014) found that each clone of a tumour forms and condenses inside a particular area of a tumour bulk. Since tumour sample specimens of a patient are taken from different parts of tumour bulk, clonal frequencies of clones within a tumour sample will be different depending on the area from which the sample was taken. Consequently, since observations (the count of short reads and log ratio of normal-tumour content) are interrelated with clone specific genotypes[1] and clonal frequencies, observations of each sample would be different from others. In contrast, the allelic composition of mutations will be the same irrespective of where the sample was taken.

[1] Clone specific allelic composition of mutations of clones

However, another limitation of the original version of HetFHMM is that it only represents inter dependencies between adjacent mutations. Ji et al. (2016) investigated that the relationship between mutations of cancer from three-dimensional mapping of the human genome, and found that due to the three-dimensional structure of DNA, cancer-causing genes with driver mutations mask the genomic functionalities of their nearest proteins and genes which cause the appearance of other mutations. However, if the DNA helix has unwrapped in one dimension, these driver mutations and the newly appeared mutations are not close to each other, and certainly will not be not adjacent. This type of influence among mutations is known as long-range mutational influences. Thus, the next challenge is to discover clones with genetic make-up from multiple sample data and long-range mutational influences.

In this chapter, we extend HetFHMM for multiple samples and long-range mutational influences. More specifically, we propose a factorial hidden Markov model as the extension of HetFHMM to identify tumour clones with their genetic make-up from multiple tumour samples by capturing long-range mutational influences. We call the extended HetFHMM method *__e__xtended __m__ultiple sample Tumour __Het__erogeneity prediction by __F__actorial __H__idden __M__arkov __M__odel (emHetFHMM)*. We test emHetFHMM against existing baselines: PyClone, PhyloSub and HetFHMM, and our method outperforms these baselines in the synthetic and the cancer data experiments.

We detail emHetFHMM with inference in Sections 4.2 and 4.3 respectively. We describe the experimental setup in Section 4.4 and present the experimental results comparing our method against strong baselines to infer clones in Section 4.5.

4.2 Model

HetFHMM identifies tumour clones and their genetic make-up from short reads of a tumour sample. We, now, extend HetFHMM to infer clones and their genetic make-up from short reads of multiple sample data, which also incorporates long-range mutational influences.

The input to emHetFHMM is similar to that of HetFHMM, but slightly different:

(a) a list of mutated genome locations,

(b) the number of NGS short reads with match and mismatch nucleotides to the reference genome at those genomic locations, which are sample specific as opposed to HetFHMM,

(c) the number of maximum clones believed to exist in the sample, and

(d) a list of mutations having long range influences among themselves.

The output of emHetFHMM is then (a) the set of mutations belonging to each clone and the genotype of those mutations in each clone, and (b) the frequency of each clone in each sample. The first output (genetic make-up of each clone) is identical to HetFHMM. Whereas, the sample specific clonal frequency is not the same as that of HetFHMM.

The observed data $\boldsymbol{O}_{t,x}$ at a location t of sample $\mathbf{x}$ includes the total short reads $N_{t,\mathbf{x}}$, the number of matched $a_{t,\mathbf{x}}$ and mismatched $\bar{a}_{t,\mathbf{x}}$ short reads (note that $N_{t,\mathbf{x}} = a_{t,\mathbf{x}} + \bar{a}_{t,\mathbf{x}}$), and log ratio of the read depths $l_{t,\mathbf{x}}$ in the tumour and normal samples. The observed data is used to infer the genotype of mutation t in the kth clone $\mathcal{G}_{t,k}$ as well as the clonal frequency of kth clone of the $\mathbf{x}$ sample $\phi_{k,\mathbf{x}}$. Similar to HetFHMM, emHetFHMM is based on Factorial Hidden Markov Model (FHHM) (Ghahramani & Jordan, 1997), where each chain corresponds to a clone and observations correspond to the counts collected from the data. In addition to the model, we consider the extra dependencies between genotypes to capture long-range mutational influences (see Figure-4.1). In emHetFHMM, the joint probability of latent variables and observations is written as

$$P(\boldsymbol{\mathcal{G}},\boldsymbol{O}|\boldsymbol{\Phi}) = \Bigg(\Big(\prod_{k=1}^{K} P(\mathcal{G}_{1,k})\Big)\Big(\prod_{\mathbf{x}\in X} P(\boldsymbol{O}_{1,\mathbf{x}}|\boldsymbol{\mathcal{G}}_1,\boldsymbol{\Phi}_{\boldsymbol{x}})\Big)\Bigg)\Bigg(\prod_{t=2}^{T}\Big(\prod_{k=1}^{K} P(\mathcal{G}_{t,k}|\mathcal{G}_{t-1,k}\vec{\mathcal{G}}_{-t,k})\Big)\Big(\prod_{\mathbf{x}\in X} P(\boldsymbol{O}_{t,\mathbf{x}}|\boldsymbol{\mathcal{G}}_t,\boldsymbol{\Phi}_{\boldsymbol{x}})\Big)\Bigg) \tag{4.1}$$

where $\boldsymbol{\mathcal{G}}_t$ is the vector of genotypes of all clones at location t. $\boldsymbol{O}_{t,\mathbf{x}} = \{N_{t,\mathbf{x}}, a_{t,\mathbf{x}}, l_{t,\mathbf{x}}\}$ is the observed data at mutation t of sample $\mathbf{x}$. $\boldsymbol{\Phi}$ is the matrix of sample specific clonal frequencies of all clones and $\boldsymbol{\Phi}_{\boldsymbol{x}}$ is the vector of frequencies of all clones of sample $\mathbf{x}$. The elements of the model are as follows:

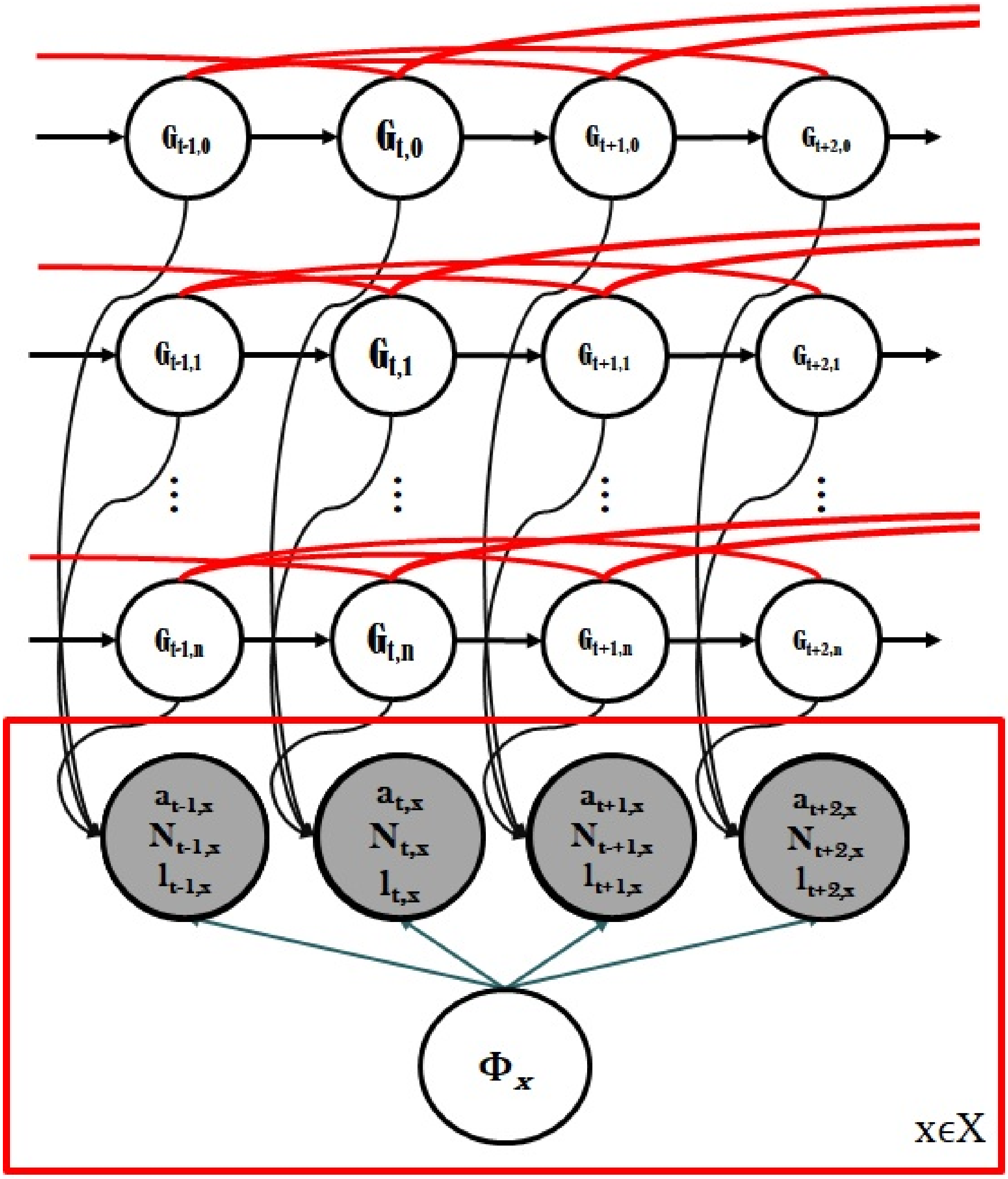


Figure 4.1: Probabilistic Graphical model of emHetFHMM for predicting tumour heterogeneity from multiple sample by capturing long range influences. Red arcs are representing the long-range influences among the mutations. The sample specific observations ($O_{t,\mathbf{x}} = \{N_{t,\mathbf{x}}, a_{t,\mathbf{x}}, l_{t,\mathbf{x}}\}$ and clonal frequencies are shown in red box.

(i) the term $P(\mathcal{G}_{t,k}|\mathcal{G}_{t-1,k},\vec{\mathcal{G}}_{-t,k})$ is called the "transition probability" which determines the dependency of the genotype of the current mutation of kth clone $\mathcal{G}_{t,k}$ conditioned on that of the previous mutation $\mathcal{G}_{t-1,i}$ and on mutations having long-range influences over current mutation $\vec{\mathcal{G}}_{-t,k}$. $\vec{\mathcal{G}}_{-t,k}$ is the vector of the genotype of mutations having long-range influences over the current mutation t.

In HetFHMM, we assume that only the adjacent mutations have effect on the current mutation. But in emHetFHMM, we include the long-range influences along with adjacent mutations as well.

(ii) the term $P(\boldsymbol{O}_{t,\mathbf{x}}|\boldsymbol{\mathcal{G}}_t,\boldsymbol{\Phi_x})$ is called the "emission probability" and determines the relationship of the observed data $\boldsymbol{O}_{t,\mathbf{x}} = \{N_{t,\mathbf{x}}, a_{t,\mathbf{x}}, l_{t,\mathbf{x}}\}$ and the latent variables $\boldsymbol{\mathcal{G}}_t$ at mutation t (conditioned on the clonal frequencies $\boldsymbol{\phi}_{k,\mathbf{x}}$).

In what follows, we provide in-depth explanation of these terms.

4.2.1 Transition Probability:

The transition probability $P(\mathcal{G}_{t,k}|\mathcal{G}_{t-1,k},\vec{\mathcal{G}}_{-t,k})$ captures long-range mutational dependencies along with adjacency influences, which is as followed:

$$P(\mathcal{G}_{t,k}|\mathcal{G}_{t-1,k},\vec{\mathcal{G}}_{-t,k}) = \underbrace{P(\mathcal{G}_{t,k}=q|\mathcal{G}_{t-1,k}=r)}_{A_{t,k}(q,r)} \times \prod_{j\in\vec{\mathcal{G}}_{-t,k}} \underbrace{P(\mathcal{G}_{t,k}=q|\vec{\mathcal{G}}^{j}_{-t,k})}_{E^{j}_{t,k}(q)} \tag{4.2}$$

where $A_{t,k}(q,r)$ and $E^{j}_{t,k}(q)$ are transition probabilities to capture inter-dependencies between the genotypes of adjacent and all long-range influenced mutations (located on the left and right side of the current mutation t), respectively. Following Colella et al. (2007), we define the transition probability matrix for adjacent mutations of each chain as follows:

$$\underbrace{P(\mathcal{G}_{t,k}=q|\mathcal{G}_{t-1,k}=r)}_{A_{t,k}(q,r)} = \begin{cases} \rho_t & \text{if } q{=}r \\ \frac{1-\rho_t}{D_k-1} & \text{Otherwise.} \end{cases} \tag{4.3}$$

ρ_t is an exponential function to define a prior probability that some genetic event (hidden state change) occurs between adjacent SNP loci a distance d_t apart:

$$\rho_t \;=\; 1 - \frac{1}{2}(1 - e^{\frac{-d_t}{L}}) \tag{4.4}$$

where L is the average length of the sequence reads[2]. D_k is the dimension of the state space (i.e. the number of genotypes, which is 21 from Table 3.1, and d_t is the genomic distance between the mutant locations t and $t-1$. In emHetFHMM, we use also same transition probabilities for adjacent mutations.

Additionally, we handle the presence of long-range mutational influences in emHetFHMM as follows. Long-range dependencies among mutations are special type of spatial effect. According to Ji et al. (2016), since the existence of three dimensional structure of DNA, mutations with long-range influences are located close to each other and the genes of these mutations form a pathway. Therefore, genes of the same pathway will skeleton the list of mutations having long-range influences. We use known gene-gene interaction networks to find genes and their pathways. From genes, their pathways, and their gene locations, we can identify mutations with long-range inter-dependencies.

In emHetFHMM, we introduce a new transition probabilities to capture long-range mutational influences in addition to exiting transition probabilities for adjacent mutations. The new transition probability that the genotype of mutations of the same pathways having long range dependencies is as followed:

$$\underbrace{P(\mathcal{G}_{t,k} = q|\vec{\mathcal{G}}^{j}_{-t,k})}_{E^{j}_{t,k}(q)} = \begin{cases} \tau & \text{if } q{=}\vec{\mathcal{G}}^{j}_{-t,k} \\ \frac{1-\tau}{D_k-1} & \text{Otherwise} \end{cases} \tag{4.5}$$

where $\vec{\mathcal{G}}^{j}_{-t,k}$ is the genotype of the mutation at location j of kth clone having long-range dependencies with current mutation t in the same pathway, and τ is the prior probability that genetic event occurs between two mutations having long-range dependencies.

[2] It was observed to be 2 Megabases (2×10^6 bases) in 104 breast tumours (rounded to the nearest Mb.) (Colella et al., 2007; Ha et al., 2012).

Since three dimensional location of mutation is not available, we propose three alternative ways to compute τ:

- Using one dimensional location, available in the data, to compute τ.

  $$\tau \; = \; 1 - \frac{1}{2}(1 - e^{\frac{-d_t^j}{L}}) \tag{4.6}$$

  where d_t^j is the one dimensional (1D) distance between jth and tth mutations.

- As mentioned earlier, Ji et al. (2016) investigated that due to the three-dimensional structure of DNA, cancer-causing genes with driver mutations mask the genomic functionalities of their nearest proteins and genes. This in turn causes the appearance of other mutations and the genotype of mutations of the same gene will be the same. Therefore, the gene location of a mutation is also an important factor to compute transition probabilities to capture long-range mutational influences.

  However, gene locations are not available in the observation $\boldsymbol{O}$. On the contrary, many tools are available to predict locations of pathways, genes and mutations accurately. From these tools, it is easy to predict the location of genes of the same pathway. We incorporate the gene location of mutations to compute τ which as follows:

  $$\tau \; = \; \Big(1 - \frac{1}{2}(1 - e^{\frac{-d_t^j}{L}})\Big)(e^{\frac{-d_{g_t^j}}{L_G}}) \tag{4.7}$$

  where $d_{g_t^j}$ is the scalar distance between genes having current and jth mutations. L_G is the average gene length[3].

- Two user-define values of τ which are 0.5 and 0.8

[3] The average gene length is 8446 base pairs (Jareborg, Birney, & Durbin, 1999).

4.2.2 Emission probabilities:

We decompose the emission probability for generating the observation $\boldsymbol{O}_{t,\mathbf{x}} = \{N_{t,\mathbf{x}}, a_{t,\mathbf{x}}, l_{t,\mathbf{x}}\}$ given that the hidden variables $\boldsymbol{\mathcal{G}}_t$ and $\boldsymbol{\Phi_x}$, as follows:

$$\begin{aligned} P(\boldsymbol{O}_{t,\mathbf{x}}|\boldsymbol{\mathcal{G}}_t, \boldsymbol{\Phi_x}) &= \prod_{\mathbf{x}\in X} P(a_{t,\mathbf{x}}|N_{t,\mathbf{x}}, \boldsymbol{\mathcal{G}}_t, \boldsymbol{\Phi_x}) P(l_{t,\mathbf{x}}|\boldsymbol{\mathcal{G}}_t, \boldsymbol{\Phi_x}) \\ &= \prod_{\mathbf{x}\in X} \mathcal{B}in(a_{t,\mathbf{x}}|N_{t,\mathbf{x}}, P_{b_{t,\mathbf{x}}}, \boldsymbol{\mathcal{G}}_t, \boldsymbol{\Phi_x}) \mathcal{N}(l_{t,\mathbf{x}}|\sigma^2, \tilde{\mu}_{t,\mathbf{x}}, \boldsymbol{\mathcal{G}}_t, \boldsymbol{\Phi_x}) \end{aligned} \quad (4.8)$$

where $a_{t,\mathbf{x}}$, $N_{t,\mathbf{x}}$, and $l_{t,\mathbf{x}}$ are sample specific observations as defined previously. As for basic HetFHMM (Chapter 3), we assume that $a_{t,\mathbf{x}}$ follows the binomial distribution where the number of trails is $N_{t,\mathbf{x}}$ and the probability of success $P_{b_{t,\mathbf{x}}}$ is as follows:

$$P_{b_{t,\mathbf{x}}} = \frac{\sum_{k=1}^{K} \phi_{k,\mathbf{x}}.r_{g_t^k}}{\sum_{k=1}^{K} \phi_{k,\mathbf{x}}.c_{g_t^k}} \quad (4.9)$$

where $r_{g_t^k}$ and $c_{g_t^k}$ are number of reference allele and the copy number of the genotype in $\mathcal{G}_{t,k}$ respectively.

Following HetFHMM, we assume that the log ratio of tumour-normal read depth is a Gaussian distributed with mean $\mu_{t,x}$ and standard deviation σ_x. The mean of the log ratio of tumour-normal read depth is:

$$\tilde{\mu}_{t,\mathbf{x}} = \frac{\sum_{k=0}^{K} \phi_{k,\mathbf{x}}.c_{g_t^k}}{\phi_{0,\mathbf{x}}.c_{g_t^0}. \sum_{k=1}^{K} \phi_{k,\mathbf{x}}.\psi} \quad (4.10)$$

where ψ is the tumour ploidy[4] parameter, which is set to 3.

4.3 Inference

emHetFHMM infers the clonal frequencies and genetic make-up of clones from multiple sample data, which also incorporate long-range mutational dependencies. Therefore, we must modify the inference algorithm of HetFHMM described in Section 3.3 for multiple

[4] Ploidy is a measure of the number of chromosomes in a cell.

sample data and for capturing the long range influences.

Since the exact inference in FHMM is intractable (Ghahramani & Jordan, 1997), we make use of Gibbs sampling as a Markov chain Monte Carlo (MCMC) method for approximate inference. To initialize, we randomly choose a genotype value for genotype variables in all chains except the normal chain where the genotypes are fixed to AB. Furthermore, we assign uniform clonal frequency to each sample. After initialization, first we infer the clone specific genotype from all of the samples, given that the sample specific clonal frequencies $\boldsymbol{\Phi_x}$ are fixed. The algorithm for sampling the clone specific genotypes from each sample x is shown in Algorithm 4.1.

Algorithm 4.1 Gibbs samplers for inferring clone specific genotypes

```
while !$converged$ do
    for mutation $t = 1$ to $T$ do
        for clone $k = 1$ to $K$ do
            for genotype $g = 1$ to $20$ do
                $P_{a_{t,\mathbf{x}}} \leftarrow \mathcal{B}in(a_{t,\mathbf{x}}|N_{t,\mathbf{x}}, P_{b_{t,\mathbf{x}}})$ ▷ $P_{b_{t,\mathbf{x}}}$ is defined in eqn (4.9)
                $P_{l_t} \leftarrow \mathcal{N}(l_{t,\mathbf{x}}|\tilde{\mu}_{t,\mathbf{x}}, \sigma_{\mathbf{x}}, \psi)$ ▷ $\tilde{\mu}_{t,\mathbf{x}}$ is defined in eqn (4.10)
                $E_{t,k} \leftarrow 1$
                for each mutation $i \in \vec{\mathcal{G}}_{-t,k}$ do
                    $E_{t,k} \leftarrow P(\mathcal{G}_{t,k}|\vec{\mathcal{G}}^{i}_{-t,k})$ ▷ $P(\mathcal{G}_{t,k}|\vec{\mathcal{G}}^{i}_{-t,k})$ is defined in eqn (4.5)
                end for
                $posterior[g] \leftarrow E_{t,k} \times P(\mathcal{G}_{t,k} = g|\mathcal{G}_{t-1,k}) \times P(\mathcal{G}_{t+1,k}|\mathcal{G}_{t,k} = g) \times P_{a_t} \times P_{l_t}$
            end for
            Sample $\mathcal{G}_{t,k}$ from normalised $posterior$
        end for
    end for
end while
```

After sampling genotypes, we predict sample specific clonal frequencies given that the clone specific genotypes are fixed. This step is discussed in detail in the Section 3.3. The algorithm to infer the sample specific cellular prevalences for each sample $\mathbf{x}$ is shown in Algorithm 4.2.

We repeatedly *alternate* between inferring the clone specific genotypes and clonal frequencies until a convergence condition is met, i.e. we alternate between $\boldsymbol{\Phi}$ and $\boldsymbol{\mathcal{G}}$ to maximise the above likelihood function. The complete likelihood function of the data

Algorithm 4.2 EG algorithm for inferring cellular prevalences of sample **x**

```
while !$converged$ do
    for Clone $k = 0$ to $K$ do
        $\phi^{new}_{k,\mathbf{x}} \leftarrow \phi^{old}_{k,\mathbf{x}} \times \exp -\eta\nabla F(\phi^{old}_{i,\mathbf{x}})$
    end for
    ${\boldsymbol{\Phi}_{\boldsymbol{x}}}^{new} \leftarrow normalize({\boldsymbol{\Phi}_{\boldsymbol{x}}}^{new})$
    Compute $F({\boldsymbol{\Phi}_{\boldsymbol{x}}}^{old})$
    Compute $F({\boldsymbol{\Phi}_{\boldsymbol{x}}}^{new})$
    if $F({\boldsymbol{\Phi}_{\boldsymbol{x}}}^{old}) - F({\boldsymbol{\Phi}_{\boldsymbol{x}}}^{new}) > 0$ then
        $\boldsymbol{\Phi}_{\boldsymbol{x}} \leftarrow {\boldsymbol{\Phi}_{\boldsymbol{x}}}^{new}$
    end if
end while
```

is as follows:

$$\begin{aligned} P(\boldsymbol{\mathcal{G}}, \boldsymbol{l}, \boldsymbol{a} | \boldsymbol{N}, \boldsymbol{\Phi}, \tilde{\mu}, \sigma, \psi) = & \prod\nolimits_{k=1}^{K}\left(\prod\nolimits_{t=1}^{T} A_{t,k}(\mathcal{G}_{t,k}|\mathcal{G}_{t-1,k}) \prod\nolimits_{j\in\vec{\mathcal{G}}_{-t,k}}\left(E^{j}_{t,k}(q)\right)\right) \\ & \prod\nolimits_{x\in X}\left(\prod\nolimits_{t=0}^{T}\mathcal{B}in(a_{t,\mathbf{x}}|N_{t,\mathbf{x}},P_b)\mathcal{N}(l_{t,\mathbf{x}}|\tilde{\mu}_{t,\mathbf{x}},\sigma^2)\right) \end{aligned}$$

The inference algorithm for emHetFHMM is shown as Algorithm 4.3.

Algorithm 4.3 Inference algorithm for emHetFHMM

```
Initialization of $S$
for Sample $\mathbf{x} = 1$ to $X$ do
    for Clone $k = 0$ to $K$ do
        $\phi_{k,\mathbf{x}} \leftarrow \frac{1}{K}$
    end for
end for
Initialization of Gibbs Sampling
for Clone $k = 1$ to $K$ do
    for Location $t = 0$ to $T$ do
        $\mathcal{G}_{t,k} \leftarrow$ randomly generated genotype
    end for
end for
while (!$converged$) do
    for sample $\mathbf{x} = 1$ to $X$ do
        Gibbs-Genotype($\mathbf{x}$); keeping $\Phi$ is fixed ▷ Algorithm-4.1
    end for
    for sample $\mathbf{x} = 1$ to $X$ do
        EG($x$); keeping $\mathcal{G}$ is fixed ▷ Algorithm-4.2
    end for
end while
```

4.4 Evaluation Framework

No existing model infers tumour clones and their genomic make-up along with the clonal frequencies from *multiple* tumour samples. Hence it is challenging to evaluate our model due to the lack of (a) appropriate data annotated with details of clones, (b) the most compatible models for comparison, and (c) suitable evaluation metrics. We focus on these issues as we describe the evaluation framework for emHetFHMM in the rest of this section.

4.4.1 Synthetic data

To assess different aspects of our model, we generate synthetic cancer data containing clone specific genotypes and sample specific clonal frequencies. To generate synthetic data, in step 1, we specify the number of clones $|K|$ and samples n. In this step, the number of samples for each patient would be 1, 2, 5, 10, 15, 20, 25 and 30. In step 2, we then specify the sample specific clonal frequencies. In the next step (step 3), we generate the mutant locations. First, we specify the first location randomly. Following the synthetic data generation of HetFHMM, we then generate the gap between two consecutive mutations from a uniform distribution on [6K,7K], since the average distance between two consecutive mutations in the AML data is 6679 base-pair (Ding et al., 2012).

After specifying the locations, we follow a couple of steps to generate genotypes of the clone specific mutations. In step 4, we set AB to genotypes of the normal clone. We then generate the genotype of the first location of each clone by random in step 5. In next the step (step 6), we generate genotype of the clone specific mutations. To generate the clone specific genotype of mutations, we use long-range mutational influences along with adjacent influences. We consider that τ is uniformly distributed in [0,1] to capture long-range mutational influences. As discussed earlier, mutations having long range dependencies can be found from the gene-gene interaction networks.

Algorithm 4.4 Synthetic data generation algorithm for multiple sample with long range influences

```
Step 1: Specify the number of clones $|K|$ and samples $N = \{1, 2, 5, 10, 15, 20, 25, \text{ and } 30\}$ of a patient.
Step 2: Specify sample specific clonal frequencies $\boldsymbol{\phi} = \{\boldsymbol{\phi_1} ... \boldsymbol{\phi_k}\}$
Step 3: Generate the location of mutations $1, 2, ...T$.
Step 4: Set $\mathcal{G}_{1,0} ..... \mathcal{G}_{T,0}$ to AB ▷ Genotype sequence of normal clone.
Step 5: Uniformly at random generate the genotypes of $\mathcal{G}_{1,1}$... $\mathcal{G}_{1,K}$ ▷ first mutations.
for each location $t \in \{1....T\}$ do
    for each clone $k \in \{1....\text{K}\}$ do
        Step 6: Sample $\mathcal{G}_{t,k}$ from transition matrix ▷ Equation 4.2
    end for
    for each number of samples $N = 1, 2, 5, 10, 15, 20, 25, \text{ and } 30$ do
        for sample $x$=1 to $N$ do
            Step 7: Generate $a_{t,x}$, $N_{t,x}$ and $l_{t,x}$ based on Emission probability matrix ▷ Equation 4.8
        end for
    end for
end for
```

We use the Reactome web database[5] to find the gene-gene interaction networks. In the Reactome database, all gene-gene interactions are experimentally confirmed in the laboratory. We used this open-source and peer-reviewed pathway database to find the possible gene-gene interactions when capturing the long-range mutational influences. We then do *gene mapping* to find out the location of the genes of the same pathway using Geneloc[6]. By comparing the locations of genes and mutations we tag the mutations, having long-range influences. Using the equation 4.2, we generate the clone specific genotypes. Finally in step 7, sample specific observations are sampled based on equation 4.8. We detail our synthetic generating process in Algorithm 4.4.

Following the synthetic data experiments of HetFHMM, we would like to assess *emHetFHMM* in data conditions where the inter-dependency assumptions made in the model about the adjacent genotypes do not hold. Therefore, we generate two more

[5] Reactome is a free, open-source, curated and peer-reviewed pathway database. The goal of this database is to provide intuitive bioinformatics tools for the visualization, interpretation and analysis of pathway knowledge to support basic research, genome analysis, modeling, systems biology and education. The weblink: `http://www.reactome.org/pages/download-data/`

[6] Geneloc (`http://genecards.weizmann.ac.il/geneloc/index.shtml`) is a web source to find the location of each gene in the human genome.

versions of the aforementioned datasets where the gap (d_t in the computation of the stickiness of genotypes Markov process ρ_t in eqn (4.4)) is scaled by 3 and 1/3, which correspond to stronger and weaker interdependencies, respectively. This also provides a compelling test bed for the comparison against the competing models, PyClone and PhyloSub, which assume no dependency between genotypes of adjacent mutations.

4.4.2 Real cancer data

We use two types of laboratory experimented cancer data, downloaded from *The Cancer Genome Atlas TCGA*[7]

- Acute Myeloid Leukemia (AML): This dataset has specimens from 2 patients. For the simplicity, we call the tumour specimen from the first patient LAML-1 and from the second LAML-2. LAML-1 and LAML-2 contained 3 and 5 samples respectively.

- Breast Invasive Carcinoma (BRCA): We download BRCA specimen data on three patients from TCGA. In the experiments we call data of the first patient BRCA-1, the second BRCA-2, and the third one BRCA-3. BRCA-1, BRCA-2 and BRCA-3 contained 2, 6 and 2 samples respectively.

All of these downloaded data were processed in the National Human Genome Research Institute (NHGRI)[8]. We use the laboratory experimented results (i.e. the clusters of the mutations of these data) from the same web source, which are considered as the gold standard.

4.4.3 Baseline Models

We compare the performance of emHetFHMM with PyClone, PhyloSub and HetFHMM. As mentioned in Chapter 3, PyClone (Roth et al., 2014) and PhyloSub (Jiao et al., 2014) models discover the clonal architecture of a tumour, i.e. the frequency of clones and

[7] TCGA is an useful web source which facilitates free cancer data for the researchers. Data portal address: `https://tcga-data.nci.nih.gov/docs/publications/tcga/?`

[8] `https://www.genome.gov/27569636/gds-data-access/`

the set of mutations belonging to each clone. They cluster the mutations to identify clones and to infer the clonal frequencies. PyClone takes the frequency of a cluster as the clonal prevalence, whereas PhyloSub estimates the clonal frequencies differently.

Similar to HetFHMM, emHetFHMM produces two outputs: the clonal frequencies and the clone specific genotypes (it is important to note that there is no existing method which infers the "clone specific genotypes" from the short reads except HetFHMM). In HetFHMM and emHetFHMM, $\mathcal{G}_{t,k}$ specifies the genotype of the mutation t in kth clone, and whether the clone harbours this mutation or not. Therefore, we can obtain the set of mutations belonging to a clone as well as the clonal prevalence from the output of emHetFHMM, which can be then compared with the output of PyClone, PhyloSub and HetFHMM.

4.4.4 Evaluation Metrics

We make use of $V\text{-}Measure$ (Rosenberg & Hirschberg, 2007) and root mean-square distance ($RMSD$) to compare mutation clusterings and the clonal frequencies, respectively which are already defined in the previous chapter.

4.5 Results

In this section, we first compare the performance of emHetFHMM[9] vs PyClone, PhyloSub and HetFHMM on the synthetic data. Afterwards, we compare these models using real cancer data (mentioned in section 4.4.2). In experiments on both synthetic and real cancer data, we also compare the performance of multiple sample extension of HetFHMM, which does not use long-range mutational influences and called *mHetFHMM*.

Moreover, we implement emHetFHMM with alternative calculated τ (the prior probability that genetic event occurs between two mutations having long-range dependencies discussed earlier) on synthetic data to select the best suitable way to compute τ based

[9] emHetFHMM is implemented in Matlab version 2014b

on the *V-Measure* and *RMSD* errors. We call the emHetFHMM that uses one dimensional location to calculate τ as emHetFHMM (1D distance). As mentioned earlier, we also use gene location as the alternative computing of τ to capture genotypes of mutations having long-ranges dependencies. Therefore, we call the gene location variant of τ used in emHetFHMM as emHetFHMM (Gene location). Moreover, we also use two user-defined alternative of τ (discussed earlier). These two user-defined τ create two variants of emHetFHMM which are called emHetFHMM($\tau = 0.50$) and emHetFHMM($\tau = 0.80$).

4.5.1 Synthetic Data

We investigate the mutation clusters and prediction accuracy of the clonal frequencies of each clone by computing *V-Measure* and *RMSD* scores, respectively. For the synthetic data, we consider the clusters identified by the sampled clone-specific genotypes and predefined sample specific clonal frequencies as the gold standard.

4.5.1.1 Experiments on the data having single sample

We first carry out experiments on the single sample data with 3 clones. Figure 4.2 presents the average *V-Measure* of outputs of HetFHMM, mHetFHMM and emHetFHMM with four alternative calculated τ (1D distance, gene location, 0.5 and 0.8). For single sample data, mHetFHMM detected statistically similar clustered output as that of HetFHMM. *V-Measure* calculated on the output of mHetFHMM is 0.516. Whereas, the accurate detection of clustered mutations of 3 clone data by HetFHMM is around 50.1% (On average *V-Measure* is 0.501 for all type of inter adjacent mutations dependencies: independent, weak, normal and strong). Both mHetFHMM and HetFHMM perform similarly on single sample data.

All variants of emHetFHMM outperform both mHetFHMM and HetFHMM. Since one of the important biological features of tumour (long-range mutational influences) is captured by emHetFHMM, it performs better than both mHetFHMM and HetFHMM.

In addition, Fig 4.2 shows that gene location alternative computation of τ of

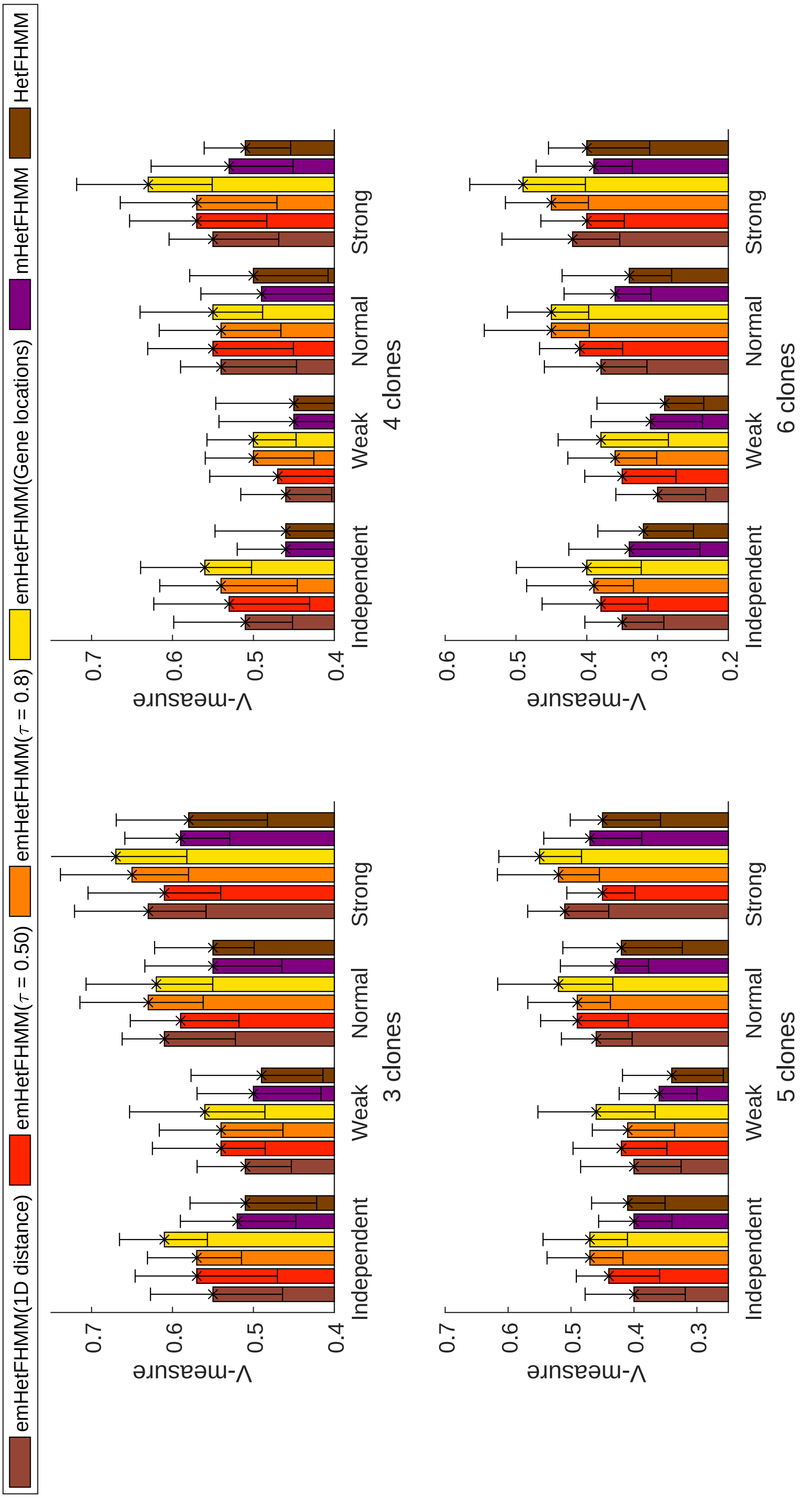


Figure 4.2: Average *V-Measure* of outputs of emHetFHMM and its variants; mHetFHMM and HetFHMM on single sampled {3,4,5 and 6} clones synthetic data with four adjacent effects.

emHetFHMM performs best over the other alternative computation of τ. As mentioned earlier that, the genotype of the mutations in the same gene are the same. Gene location variant of τ of emHetFHMM considers this feature to compute the transition probability to sample the genotype of mutations. Whereas, the 1D distance between the mutations and two user defined τ do not mention whatever these mutations are located in same gene or not. Therefore, these variants of τ of emHetFHMM do not perform well as gene location variant does. It helps to capture better long-range mutational influences and improves the clone specific genotypes prediction. Gene location variant of emHetFHMM performs very well over the other variants of emHetFHMM, mHetFHMM and HetFHMM. Therefore, it can be said that emHetFHMM is the generalized versions of HetFHMM.

4.5.1.2 Experiments on the data having more than 1 sample

We further carry out experiments on the 3 clone data with 2, 5, 10, 15, 20, 25 and 30 samples by mHetFHMM and emHetFHMM. Figure 4.4 presents the average *V-Measure* of outputs of mHetFHMM and emHetFHMM with its different variants. Similar to the previous experiment, gene location variants of emHetFHMM outperforms mHetFHMM and other variants of emHetFHMM.

Furthermore, Figure 4.4 shows that as the number of samples increases, both emHetFHMM and mHetFHMM predict clusters of mutations more accurately. Therefore, the availability of more samples from a cancer patient would help emHetFHMM to infer more accurate clones.

Observing *V-Measure* scores corresponding to the different number of samples, there is no statistically significant change in clusters predicted by emHetFHMM and mHetFHMM. The highest *V-Measure* is pretending when the number of samples is more than 25. Both emHetFHMM and mHetFHMM need approximately 25 to 30 samples to predict statistically better clusters of the mutations compare to PyClone and PhyloSub (Figure 4.3). Therefore, we can claim that, after capturing long-range mutational influences and multiple samples features into emHetFHMM, it can predict

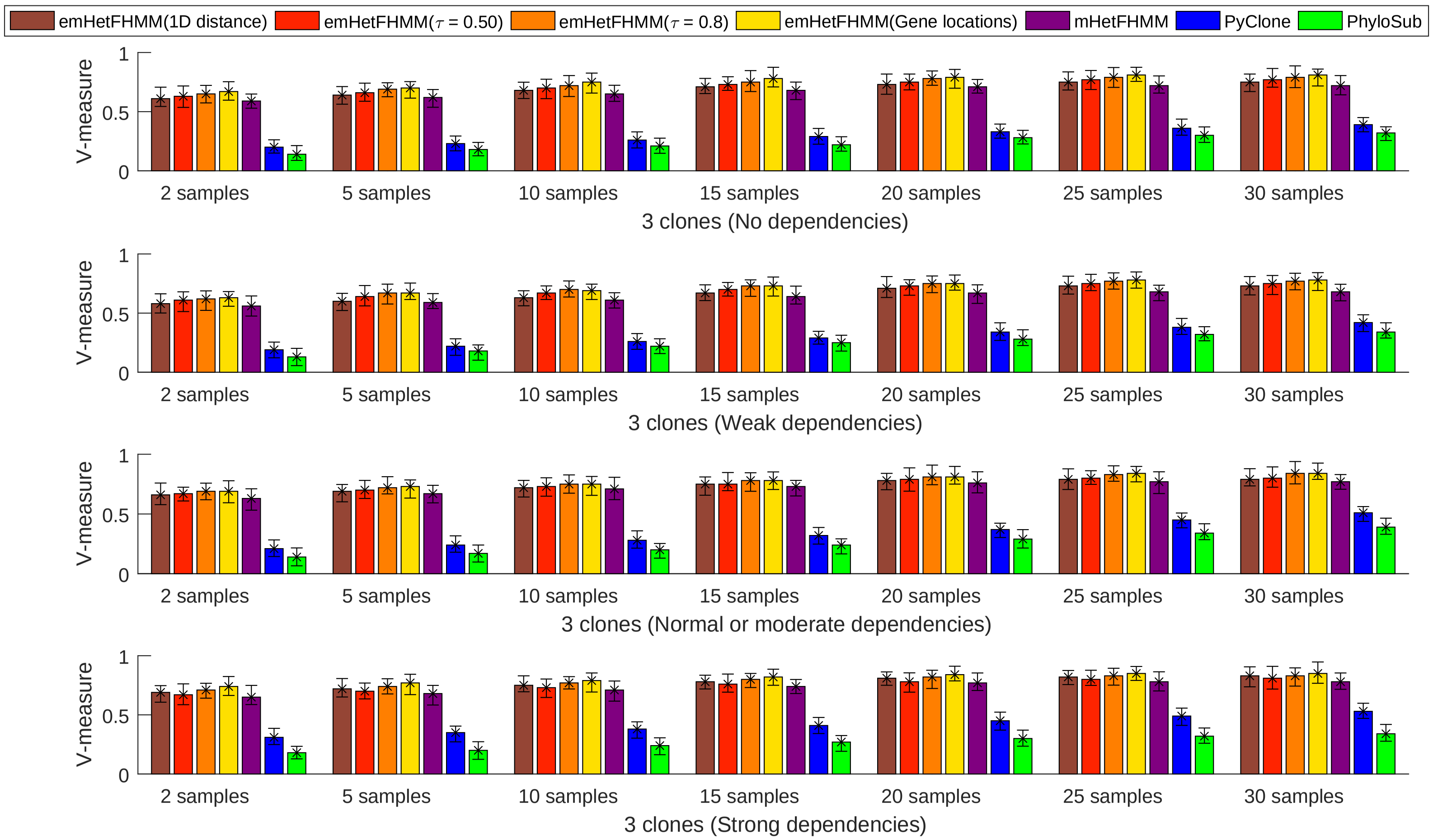


Figure 4.3: Average $V\text{-}Measure$ of outputs of emHetFHMM and its variants; mHetFHMM, PyClone and PhyloSub on 3 clones synthetic data with four adjacent effects.

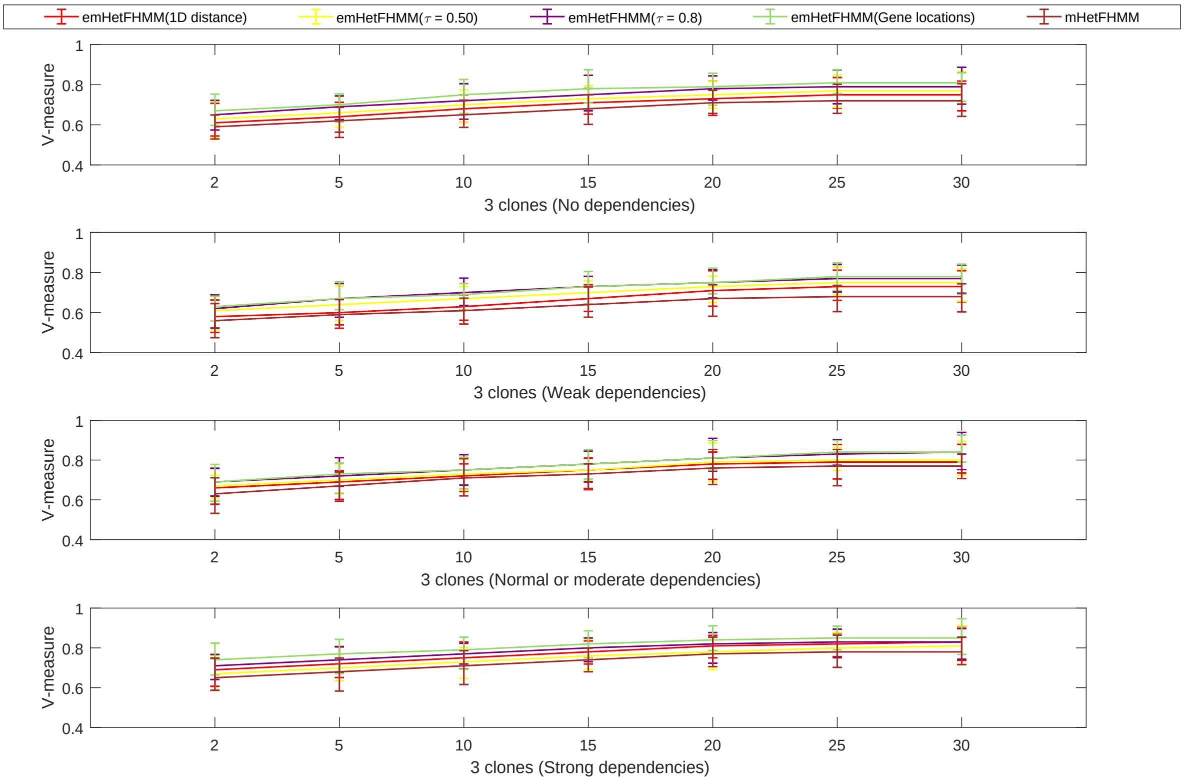


Figure 4.4: Improvement trend of outputs of emHetFHMM and its variants; and mHetFHMM on 3 clones synthetic data with four adjacent effects.

clone specific genotype of mutations from the data with 25 to 30 samples efficiently.

4.5.1.3 Investigating the performance on different adjacent effect

We also investigate the performance of emHetFHMM on four types of adjacent inter dependencies (discussed in previous section in details). Figure 4.4 shows that emHetFHMM with its different variants perform very well when mutations have strong effect on their adjacent mutations. Moreover, emHetFHMM performs well on moderate effect comparing to weak effect. Transition probabilities of emHetFHMM is also dependent on the distance between two adjacent mutations. Two mutations with strong adjacent effect are located very close to each other, and the probability that the genotype of these two mutations would be the same. Whereas, for two distant adjacent mutations, the probabilities of the genotypes may be different. Following the equation 4.2, emHetFHMM samples any of 19 different genotypes (other than same genotype) with equal probability, which might sample the genotype of a mutation differ from gold standard. Hence it affects the clustered outputs of emHetFHMM for 3 clone data with different adjacent effect.

Interestingly, Fig 4.4 narrates that emHetFHMM needs 20 to 25 samples of the data with strong adjacent effect, to infer accurate clone specific genotypes. Whereas data with other effects requires around 25 to 30 samples.

4.5.1.4 Performance of PyClone and PhyloSub on the same data

We compare outputs of PyClone and PhyloSub by computing $V\text{-}Measure$. Figure 4.3 shows the average $V\text{-}Measure$ of emHetFHMM, mHetFHMM, PyClone and PhyloSub. Similar to HetFHMM, both PyClone and PhyloSub are outperformed by all variants of emHetFHMM and mHetFHMM. The number of clusters predicted by PyClone and PhyloSub are quite larger than the gold standard. Even PhyloSub inferred around 3 to 6 more clusters than PyClone. Moreover, both methods depend on the existing genotype prediction methods to compute clonal frequencies of clones. Existing genotype prediction methods are not predicting clone specific genotypes and do not consider

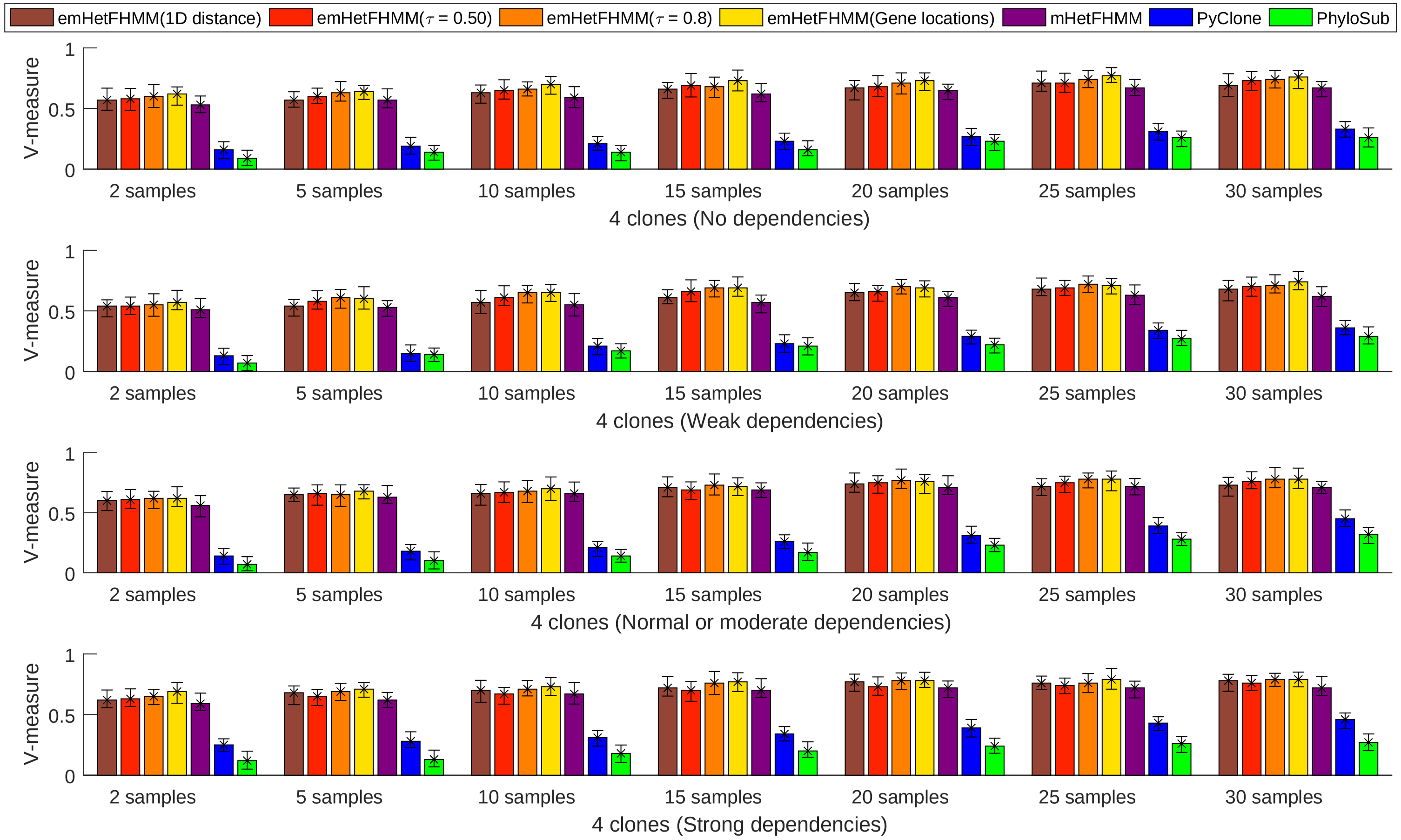


Figure 4.5: Average $V\text{-}Measure$ of outputs of emHetFHMM and its variants; mHetFHMM, PyClone and PhyloSub on 4 clones synthetic data with four adjacent effects.

the presence of long-range mutational influences. Hence, it affects the performance of PyClone and PhyloSub.

Moreover, both PyClone and PhyloSub perform well as the number of samples increases. According to the figure 4.3, PyClone and PhyloSub do not perform significantly statistically well when the number of sample is 30 as emHetFHMM does. Based on the results of this experiment, emHetFHMM outperforms the baselines PyClone and PhyloSub.

4.5.1.5 Experiments on the data with more than 3 clones

Figures from 4.5 to 4.7 present results of emHetFHMM, mHetFHMM, PyClone and PhyloSub. Similar to 3 clone experiments, gene location variant of emHetFHMM performs well for the data with 4, 5 and 6 clones. In addition, The number of samples affects outputs of emHetFHMM. Results of this experiment show that mHetFHMM, PyClone and PhyloSub are not as good as emHetFHMM.

4.5.1.6 Performance on predicting clonal frequencies

We next evaluate the inferred sample specific clonal frequencies based on $RMSD$ (discussed in details in previous section). Figures from 4.8 to 4.11 depict the average $RMSD$ of predicted clonal frequencies comparing with gold standard. Over the increase of the number of samples, $RMSD$ of predicted clonal frequencies are in the trend of decreasing. As we know that clonal frequencies are inter-related with clone specific genotypes. As the number of samples increases, emHetFHMM predicts more accurate clone specific genotypes and improves the estimation of clonal frequencies. Hence, the number of sample affects the accurate prediction of clonal frequencies.

Similar to the average $V\text{-}Measure$, gene location variants of emHetFHMM outperforms PyClone, PhyloSub and mHetFHMM on 3, 4, 5 and 6 clones' data. PyClone and PhyloSub do not infer the sample specific clonal frequencies. These models consider that clonal frequencies of all of the clones are same in all samples. Due to the assumption about clonal frequencies and the less accurate prediction of the clusters affected the

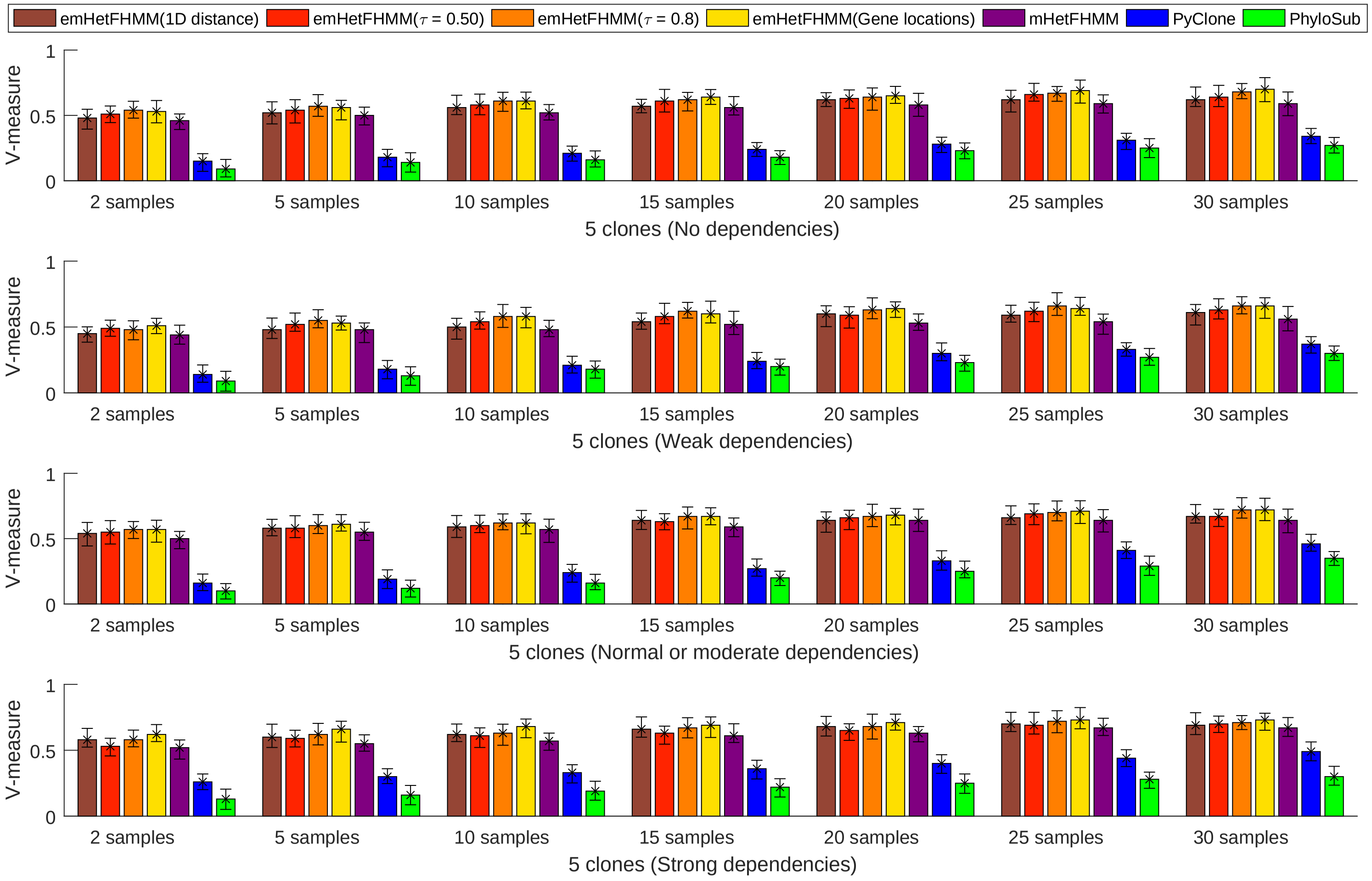


Figure 4.6: Average *V-Measure* of outputs of emHetFHMM and its variants; mHetFHMM, PyClone and PhyloSub on 5 clones synthetic data with four adjacent effects.

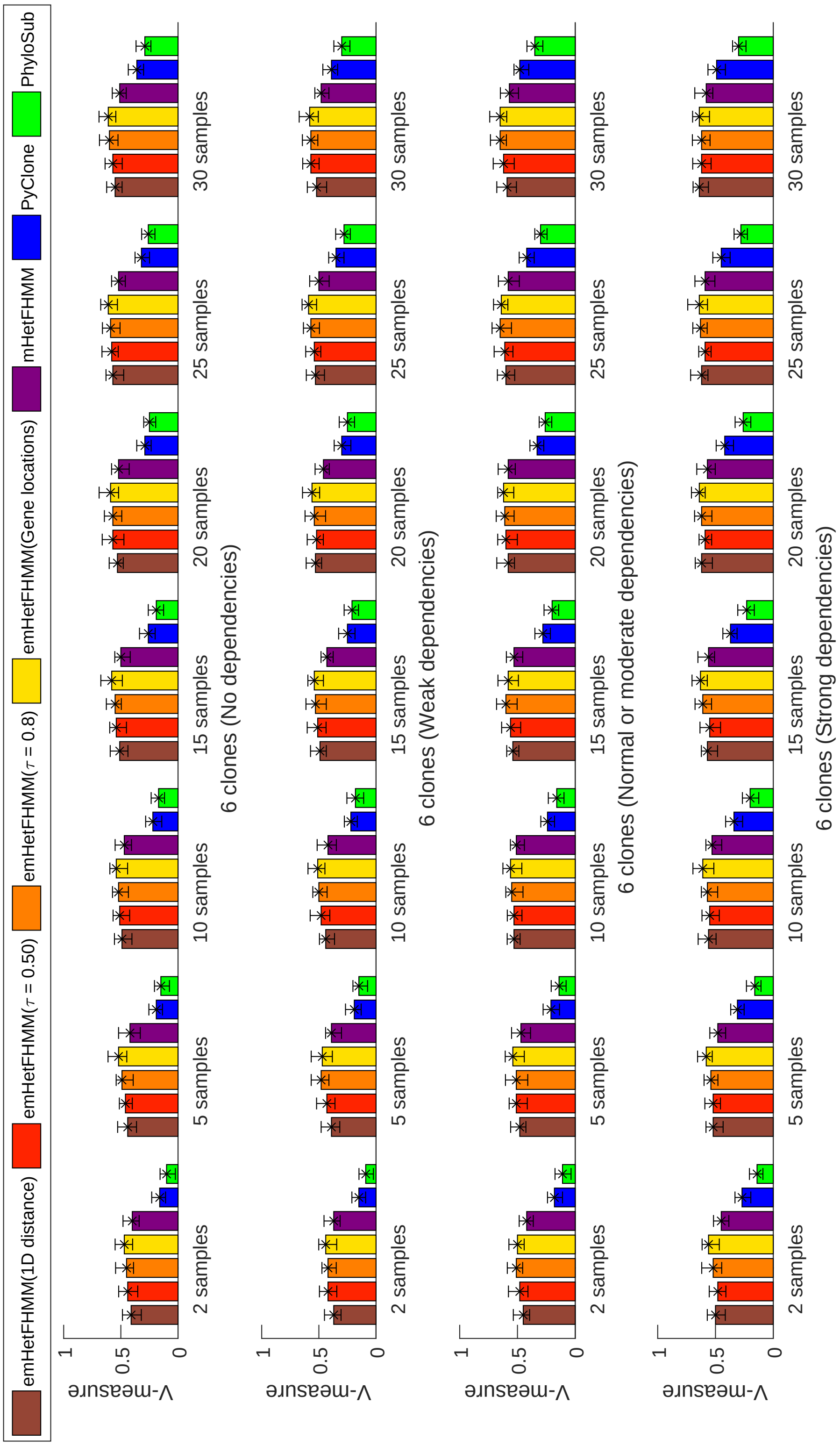


Figure 4.7: Average *V-Measure* of outputs of emHetFHMM and its variants; mHetFHMM, PyClone and PhyloSub on 6 clones synthetic data with four adjacent effects.

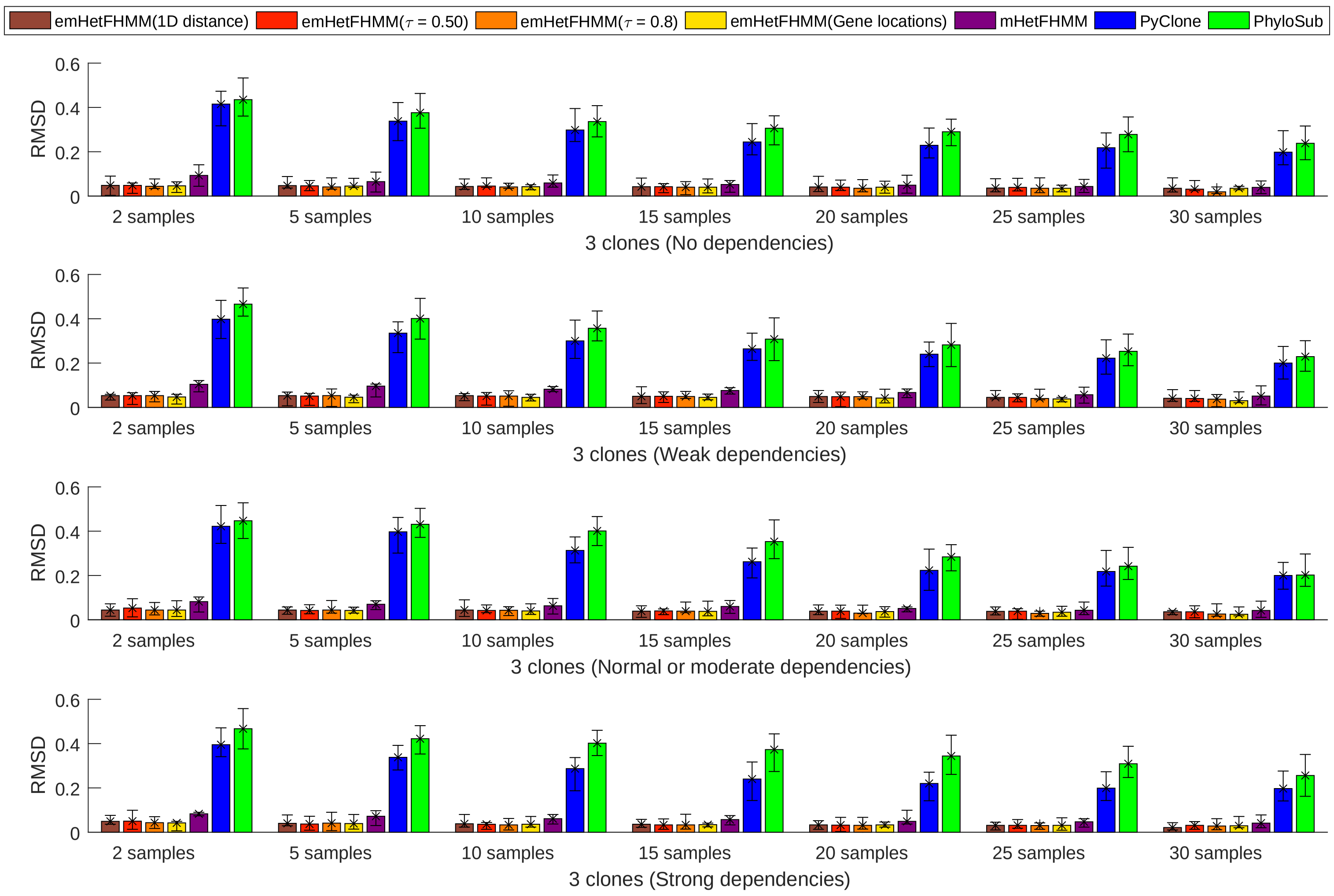


Figure 4.8: Average $RMSD$ error of the computed clonal frequencies by emHetFHMM and its variants; mHetFHMM, PyClone and PhyloSub on 3 clones synthetic data with four adjacent effects.

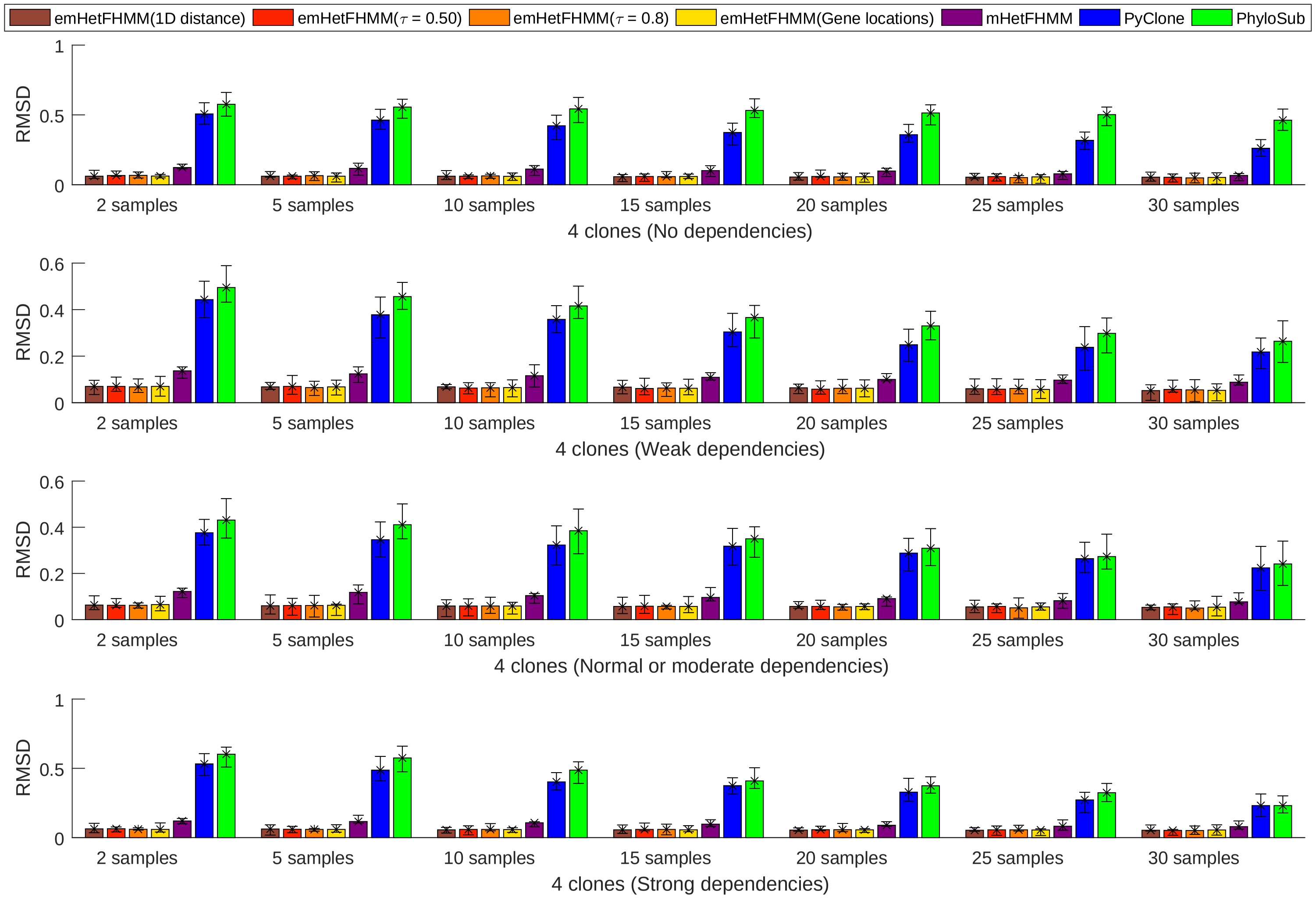


Figure 4.9: Average $RMSD$ error of the computed clonal frequencies by emHetFHMM and its variants; mHetFHMM, PyClone and PhyloSub on 4 clones synthetic data with four adjacent effects.

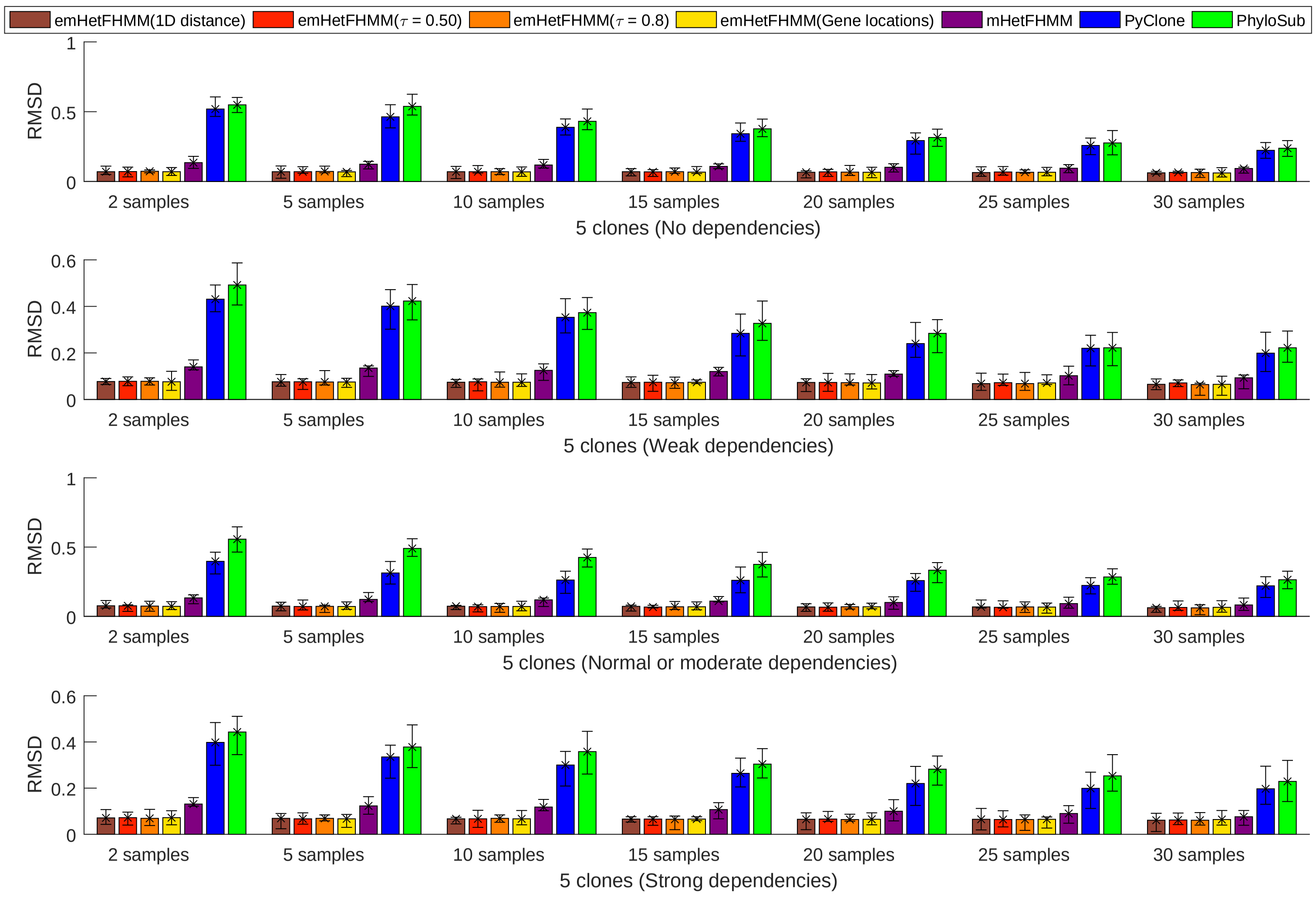


Figure 4.10: Average $RMSD$ error of the computed clonal frequencies by emHetFHMM and its variants; mHetFHMM, PyClone and PhyloSub on 5 clones synthetic data with four adjacent effects

prediction of clonal frequencies of PyClone and PhyloSub. Hence, from Figures from 4.8 to 4.11, we found that $RMSD$ error of PyClone and PhyloSub were larger than emHetFHMM. Based on results of these experiments on synthetic data, we can claim that emHetFHMM is one of the best methods for inferring tumour heterogeneity.

4.5.2 Real cancer data

We apply emHetFHMM and mHetFHMM to the cancer data downloaded from TCGA (described in Section 4.4.2), which is annotated with gold standard clones based on laboratory experiments. However, the data is not annotated with the clonal frequencies of clones; therefore, we only use $V\text{-}Measure$ to test the quality of the predicted clones as $RMSD$ is not applicable. For emHetFHMM and mHetFHMM, we run the model with different number of clones from 3 to 10, and choose the one with the highest log likelihood. In the synthetic data experiments, gene location variants of emHetFHMM performed well. It inspires us to use the gene location variants of emHetFHMM for TCGA real cancer data. Moreover, we also use the τ=0.8 variants of emHetFHMM.

Figure 4.12 presents the average $V\text{-}Measure$ of the output, inferred by emHetFHMM, mHetFHMM, PyClone and PhyloSub. Based on the average $V\text{-}Measure$, it is clear that gene location variant of emHetFHMM predicts clusters more efficiently than mHetFHMM, PyClone and PhyloSub for all AML and BRCA cancer data. Moreover, on BRCA-3 data, mHetFHMM predicts less than the number of gold clones and outperformed by PhyloSub and . We investigate the data of BRCA-3 and find that the distance between the most of the adjacent mutations are large. For having weak adjacent affect between mutations, mHetFHMM was not performed well as same as the data having the mutation with strong adjacent affect. Therefore, it affects the output of mHetFHMM on BRCA-3. Whereas, emHetFHMM captures the long-range mutational influences, which improves the performance and outperforms PhyloSub in BRCA-3 data.

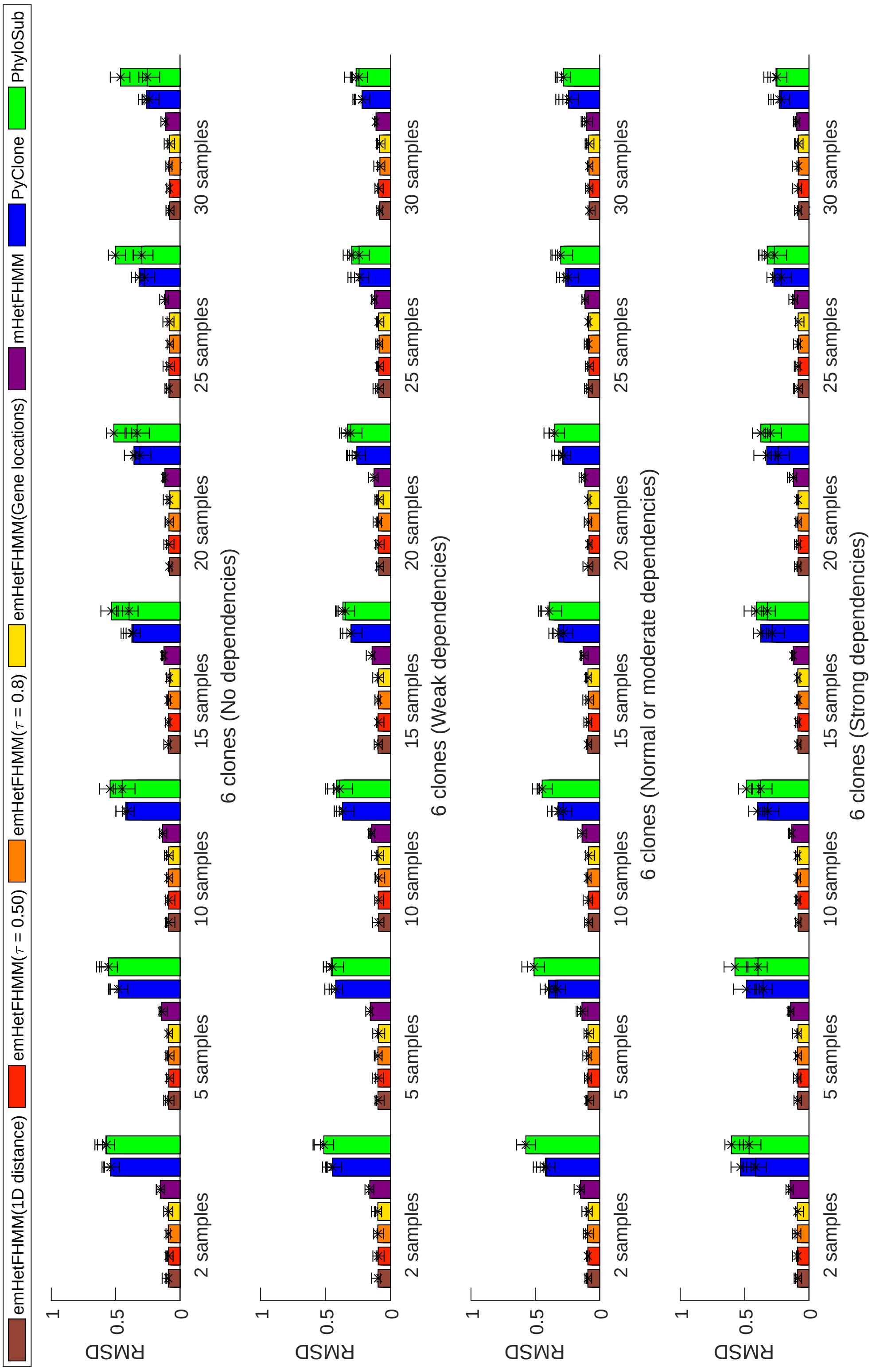


Figure 4.11: Average $RMSD$ error of the computed clonal frequencies by emHetFHMM and its variants; mHetFHMM, PyClone and PhyloSub on 6 clones synthetic data with four adjacent effects

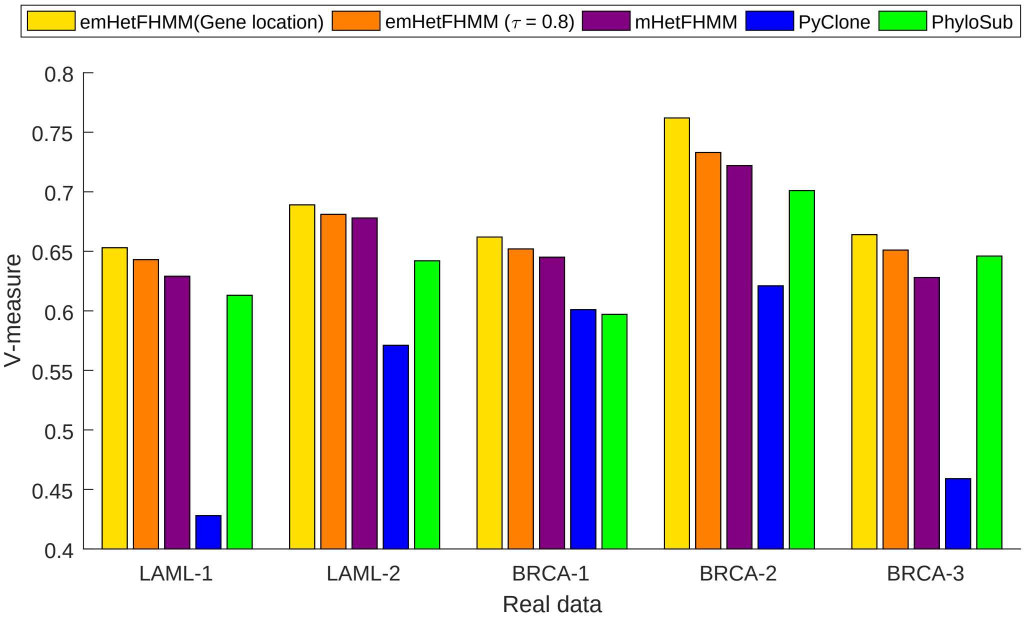


Figure 4.12: Average *V-Measure* of outputs of emHetFHMM and its gene location and $\tau = 0.8$ variants; mHetFHMM, PyClone and PhyloSub on real cancer data.

4.6 Conclusion

We extend HetFHMM to multiple samples data and long-range mutational dependencies, to identify the clonal architecture of a tumour more accurately from the next generation sequencing data. The resulting model emHetFHMM discovers mutations and their types in each clone along with sample specific clonal frequencies from the data. emHetFHMM is based on Factorial Hidden Markov Models, whereby the genomic composition of each clone is represented by a hidden chain. The basic idea of the model is that the observed data is generated by a mixture of the underlying chains, where the mixing coefficients are clonal frequencies. We make use of Gibbs sampling and exponentiated gradient algorithms to infer the clonal genomic compositions represented by hidden chains as well as clonal frequencies. The empirical results on the synthetic and the cancer data confirms that our model outperforms strong baseline models PhyloSub and PyClone based on two evaluation metrics, i.e. *V-Measure* and *RMSD*. Moreover, according to results on the synthetic data, emHetFHMM requires comparatively smaller number of samples with respect to PhyloSub and PyClone. Key to the stronger performance of emHetFHMM compared to the baseline models is that it uses the multiple samples and long-range mutational influences.

emHetFHMM infers tumour heterogeneity from multiple sample data and capture long-range mutational inter-dependency. We use the Reactome database[10] to find the gene-gene interaction networks and Geneloc[11] to find the location of each gene in the human genome. Using these two databases, we identify the long-range mutational dependencies. However, in the cancer data, many gene-gene interactions are hidden and not presented in Reactome database. Therefore, the cancer data can be used to reveal mutations with this new influences, which are missing in the gene-gene interaction networks available in Reactome or alike databases. Therefore, we extend our research to discover the gene-gene interaction networks from the cancer data from which find more long-range inter-dependency among the mutations will be captured. This unknown long-ranges dependencies certainly improve tumour heterogeneity prediction.

[10] `http://www.reactome.org/pages/download-data/`

[11] `http://genecards.weizmann.ac.il/geneloc/index.shtml`

Chapter 5

Network Structure Learning with Gaussian Graphical Models

The research in this chapter has been published/ submitted in the following article: Mohammad S Rahman and Gholamreza Haffari, “A statistically efficient and scalable method for exploratory analysis of high-dimensional data", Revised version submitted to *Journal of Data Mining and Knowledge Discovery.*

In emHetFHMM, we use *Reactome* database to find out the available gene-gene interaction network (i.e. gene-pathways) and *GeneLoc* to find out the location of the genes. Cordell (2009) has discovered that the genetic factor function primarily involves multiple other genes through a complex mechanism to play a significant role in the development of cancer cells. All of the interactions between the genes are not possible to predict through laboratory experiments which do not consider all potential genetic factor functions. Performing network analysis using large-scale gene expression datasets is an effective way to uncover new biological knowledge (Su, Meng, Ma, Bai, & Liu, 2016). Statistical interaction between two genes can describe the relationship between these two genes. However, large scale gene expression data involves continuous valued random variables, where it is critical to uncover the associations among the variables from the large sample data. Typically there are fewer samples compared to the number of variables, which makes the association discovery challenging, particularly for high-dimensional data.

Association discovery among data variables can be casted as discovering statistical dependencies among the random variables, expressed by the structure of an underlying probabilistic graphical model (Petitjean et al., 2013). However, current methods for graphical model structure discovery with continuous variables either do not scale well to datasets with large sample sizes (Yang & Lozano, 2015; H. Liu, 2017; Hirose et al., 2017); or poor objective function (J. Friedman et al., 2008; H. Liu, 2017); or suffer from high false discovery rates when the number of dimensions is much larger than the sample size (T. Wang et al., 2016; Avagyan et al., 2017; Chiong & Moon, 2017); or sacrificing too much the computational cost (Avagyan et al., 2017; H. Liu, 2017).

In this chapter, we propose a scalable and statistically efficient approach for undirected graphical model structure discovery for exploratory analysis of high-dimensional continuous data. Starting from the null graph, our approach incrementally adds the best edge maximising a test statistic using the graphical model. We start from the log-likelihood ratio test and note that it leads to small number of edges in the estimated graph, hence missing a large number of true associations. We then present a novel test statistic based on the minimum message length (MML) principle for statistical inference, where candidate models compete based on the length of their lossless compression of the data. An integral part of our test statistics is the maximum likelihood estimate for the parameters of the competing models. As we desire our method to be computationally efficient, we restrict the structure of the competing models to *chordal* graphs. They characterise *decomposable* probabilistic graphical models, which enjoy analytical solution for the maximum likelihood estimates of their parameters. As such, chordal graphs have been popular in probabilistic graphical models, eg see (Beeri, Fagin, Maier, & Yannakakis, 1983; Deshpande et al., 2001).

Our MML-based structure discovery is an information-theoretic method enjoying (a) low false discovery rate, (b) suitability for small number of samples when discovering statistical dependencies (associations) among large number of variables, and (c) scalability to large-scale problems involving thousands of variables. We call our method *ContChordalysis*, naming it after *Chordalysis* (Petitjean et al., 2013) which is a method for chordal graphical model structure discovery for discrete-valued random variables. We present extensive empirical results on synthetic and real-life datasets, and show that our method outperforms strong baselines in terms of both speed and the accuracy of the predicted associations between the random variables in the graphical model.

5.1 Structure Discovery in Decomposable Gaussian Graphical Models

Let $\mathcal{D} = \{X_1, \ldots, X_n\}$ be a training set consisting of n data points where $X_i \in \mathbb{R}^d$ and d is the number of dimensions (equivalently attributes, or random variables). Our aim is to discover the unobserved undirected graphical structure based on the observed/sampled vectors in $\mathcal{D}$.

We are interested in the undirected graphical structure $G = (V, E)$, where V is the set of vertices each of which corresponds to a random variable (or a dimension of the input vectors), and E is the set of edges capturing the statistical associations between random variables. A parameterisation of the model corresponds to multivariate functions assigned to subset of variables in maximal cliques of the graph. The probability density function corresponding to the graph is defined as

$$f(\mathcal{D}) \propto \prod_{C \in \mathcal{C}} P(\mathcal{D}^C)$$

where $\mathcal{C}$ is the set of maximal cliques, and $P(\mathcal{D}^C)$ is a clique-specific function defined on the subset of variables appearing in a clique C. In any distribution resulting from the graphical model, two random variables are statistically independent conditioned on the variables in a cut separating the two.

In this chapter, we assume that the observed input vectors have been generated from a multivariate Gaussian distribution $\mathcal{D} \sim \mathcal{N}_d(\mu, \Sigma)$, which means the cliques are also parameterised by Gaussian distributions. Therefore, our aim is to discover the structure of the so-called Gaussian graphical model. For computational convenience, we work with chordal graphical structures, leading to decomposable models covered in the next subsection.

5.1.1 Decomposable Models

Decomposable Models is a subclass of undirected graphical models which provides a usefully constrained representation in which model selection and parameter estimation can be done efficiently, which makes it suitable for large-scale problems.

Definition 5.1.1. *(Deshpande et al., 2001) A graphical model is decomposable if the associated graph G is chordal. A chordal graph is one in which all cycles of four or more vertices have a chord, which is an edge that is not part of the cycle but connects two vertices of the cycle.*

Let $\mathcal{M}$ be a decomposable model, and $f_{\mathcal{M}}$ be the probability density function of a Gaussian distribution corresponding to $\mathcal{M}$. It can be shown that (Lauritzen, 1996):

$$f_{\mathcal{M}}(X) = \frac{\prod_{C \in \mathcal{C}} P(\mathcal{D}^C)}{\prod_{S \in \mathcal{S}} P(\mathcal{D}^S)} \tag{5.1}$$

where $\mathcal{C}$ is the set of maximal cliques and $\mathcal{S}$ is the set of minimal separators corresponding to the chordal graph of the model $\mathcal{M}$. The importance of this result is that it relates the Gaussian distribution over all variables to those on the subsets of variables, i.e. Gaussian distributions over the variables involved in maximal cliques $P(\mathcal{D}^C)$ or minimal separators $P(\mathcal{D}^S)$. This amounts to a closed form solution for the maximum likelihood estimate (MLE) of the covariance matrix $\hat{\Sigma}$ of the Gaussian graphical model $P_{\mathcal{M}}$, through the MLE of the covariance matrices of the component models.

Theorem 1. *(Lauritzen, 1996) For a Gaussian graphical model corresponding to a* *chordal graph $G = (V, E)$, the maximum likelihood estimate of the covariance matrix is:*

$$\hat{\Sigma}^{-1} = n \left\{ \sum_{C \in \mathcal{C}} [(ssd_C)^{-1}]^V - \sum_{S \in \mathcal{S}} [(ssd_S)^{-1}]^V \right\} \tag{5.2}$$

where $[A]^V$ denotes extending a small matrix A defined on a subset of variables V to a larger matrix on all variables by setting extra entries to zero. ssd (Sum of Squared Distance) is defined as

$$ssd = \sum_{i=1}^{n} (X_i - \bar{X})(X_i - \bar{X})^T$$

where $\bar{X} = \frac{1}{n} \sum_{i=1}^{n} X_i$ is the empirical mean. The determinant of the estimate can be calculated as (Lauritzen, 1996):

$$|\hat{\Sigma}^{-1}| = n^{|V|} \frac{\prod_{S \in \mathcal{S}} |\ ssd_S|}{\prod_{C \in \mathcal{C}} |ssd_C|}. \tag{5.3}$$

The inverse of the covariance matrix is called the *precision* matrix $\mathcal{K} = \Sigma^{-1}$. Interestingly, the non-association between the variables expressed in the graph $G = (V, E)$ of the Gaussian graphical model translates to pattern of zeros in the precision matrix, i.e. $\mathcal{K}$ has zero for all entries where there is no edge between the corresponding pairs of vertices in E.

In order to discover the optimal decomposable graphical structure from a given training data, typically one of the following strategies is employed (Deshpande et al., 2001), (Petitjean et al., 2013):

- Forward Selection: Starting with the simplest model with no edge (i.e. $E = \emptyset$). Edges are added incrementally, as long as the new hypothesised models are not rejected according to an appropriate test statistics.

- Backward Elimination: Starting with the complete graph over the $|V|$ vertices, edges are deleted incrementally, as long as the new hypothesised models are not rejected according to an appropriate test statistics.

In this chapter, in order to make the graphical model discovery simple and easy, we adopt the forward selection strategy, and add the edges incrementally. As we want the resulting model to be decomposable, the addition of an edge has to be done with care.

5.1.2 Structure Discovery by Hypothesis Testing

Let $\mathcal{S}^+(G)$ denote all positive definite matrices whose zero patterns are consistent with the graph G, i.e. they have zero for all entries corresponding to non-existent edges of the graph G. Let G' be a candidate graph resulting from adding an edge (a,b) to the graph G, that is $G' = G \cup \{(a,b)\}$. In the forward selection strategy, we test the hypothesis that $\mathcal{K} \in \mathcal{S}^+(G)$ under the assumption that $\mathcal{K}' \in \mathcal{S}^+(G')$.

Theorem 2. *(Lauritzen, 1996) The exact deviance test for testing a decomposable model $\mathcal{K} \in \mathcal{S}^+(G)$ assuming a decomposable model $\mathcal{K}' \in \mathcal{S}^+(G')$ can be performed by rejecting the small values of*

$$r = \frac{|\hat{\mathcal{K}}|}{|\hat{\mathcal{K}}'|} \tag{5.4}$$

which is distributed as a beta distribution $\mathcal{B}(\frac{|V|-|C_{ab}|}{2}, \frac{1}{2})$ where C_{ab} is the maximal clique that contains the newly added edge (a,b). $\hat{\mathcal{K}}$ and $\hat{\mathcal{K}}'$ are the maximum likelihood estimates for the precision matrix of the Gaussians corresponding to G and G', respectively.

We will discuss how the test statistics can be computed efficiently in Section 5.1.3. As the graphical model is learnt incrementally by adding one edge at a time, we make intensive use of statistical testing. Multiple hypothesis testing is prone to many false discoveries. This is critical in our approach where we need to do a lot statistical testing due to the large size of the search space, which may lead to accepting modifications of the models more often than needed. This can be avoided by using *layered critical values* (Webb, 2008), a variant of the Bonferroni correction that increases the number of significant patterns discovered while still maintaining strict control over the risk of false discoveries. Given the p-value threshold β (usually $\beta = 0.1$), the layered p-value at iteration t of the algorithm is

$$\beta_t = \frac{\beta}{2^t |\mathcal{G}_t|} \tag{5.5}$$

Algorithm 5.1 ContChordalysis

```
Input: Dataset $D = \{X_i\}_{i=1}^{n}$, Significance level $\beta$
Output: The graph $G = (V, E)$
Initialise $G$ to be the graph without any edges
$t \leftarrow 1$
repeat
    $E^c \leftarrow$ CandidateEdge($G$) ▷ see Section 5.1.3
    for $e \in E^c$ do
        $r_e =$ testStatistic($D, G, e$) ▷ based on eqn (5.6)
    end for
    $e^* \leftarrow \arg\min_{e \in E^c} r_e$
    $pval = \frac{\beta}{2^t |E^c|}$
    if $r_{e^*} \leq pval$ then
        $G \leftarrow$ addEdge($G, e^*$) ▷ see Section 5.1.3
    end if
    $t \leftarrow t + 1$
until $(r_{e^*} > pval)$ or $(t > \frac{|V|(|V|-1)}{2})$
```

where t is the number of edges in the current best model, and $\boldsymbol{G}_{chordal}$ is the number of chordal graphs that can be formed by adding an edge to the current model.

The resulting structure discovery algorithm is presented in Algorithm 5.1. We call our algorithm *ContChordalysis*, to highlight that it is for structure discovery of chordal graphs for continuous valued variables.

5.1.3 Efficient Computation of the Test Statistics

We now turn to the question of how to efficiently compute the test statistics in eqn ((5.4)), which is particularly important for large-scale datasets.

(Deshpande et al., 2001) characterises the edges that can be added to a decomposable model while retaining its decomposability. Furthermore, it presents an efficient algorithm to enumerate all such edges in $O(|V|^2)$. This is achieved by a data structure called the *clique graph*, which keeps track of the maximal cliques $\mathcal{C}$ and minimal separators $\mathcal{S}$. Adding an edge to the graph and updating the underlying data structures also takes $O(|V|^2)$.

Theorem 3. *(Deshpande et al., 2001) If two decomposable models $\mathcal{M} \subset \mathcal{M}'$ differ only in one edge (a, b), (i.e., $(a, b) \in \mathcal{M}'$ and $(a, b) \notin \mathcal{M}$), then the maximal cliques and the*

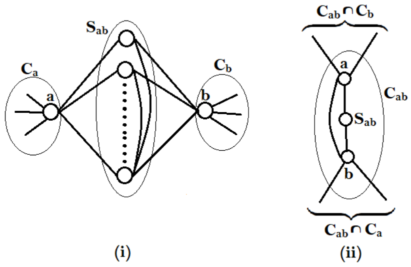


Figure 5.1: Structure of (i) the cliques C_a, C_b and separator S_{ab} in reference model; and (ii) newly formed clique C_{ab} and separators $C_{ab} \cap C_a$ and $C_{ab} \cap C_b$ in candidate model.

minimal separators (C, S) and (C', S') in these two models differ as follows:

- *If $C_a \not\subset C_{ab}$ and $C_b \not\subset C_{ab}$, then $C' = C + C_{ab}$ and $S' = S + C_{ab} \cap C_a + C_{ab} \cap C_b - S_{ab}$*
- *If $C_a \subset C_{ab}$ and $C_b \not\subset C_{ab}$, then $C' = C + C_{ab} - C_a$ and $S' = S + C_{ab} \cap C_b - S_{ab}$*
- *If $C_a \not\subset C_{ab}$ and $C_b \subset C_{ab}$, then $C' = C + C_{ab} - C_b$ and $S' = S + C_{ab} \cap C_a - S_{ab}$*
- *If $C_a \subset C_{ab}$ and $C_b \subset C_{ab}$, then $C' = C + C_{ab} - C_a - C_b$ and $S' = S - S_{ab}$*

where C_{ab} and S_{ab} are the maximal clique and minimal separator for the nodes a and b, and C_a and C_b are the maximal cliques including each of these nodes.

Thus, the change in the determinant of the MLE estimates of the precision matrix after adding an edge (a, b) is only dependent on the minimal separator of the two vertices S_{ab}, the newly formed clique C_{ab}, and the newly formed separators $C_{ab} \cap C_a$ and $C_{ab} \cap C_b$. This means we only have to compute the determinant terms relevant to the candidate edges that can be added to the current model. This immediately leads to the following theorem.

Theorem 4. *If two decomposable models $\mathcal{M} \subset \mathcal{M}'$ differ only in one edge (a, b), (i.e., $(a, b) \in \mathcal{M}'$ and $(a, b) \notin \mathcal{M}$), then*

$$\frac{|\hat{\mathcal{K}}|}{|\hat{\mathcal{K}}'|} = \frac{|ssd_{C_{ab}}| \cdot |ssd_{S_{ab}}|}{|ssd_{C_{ab} \cap C_a}| \cdot |ssd_{C_{ab} \cap C_b}|} \tag{5.6}$$

Proof. From Theorem 2, we need to compute the test statistics $r = \frac{|\hat{\mathcal{K}}|}{|\hat{\mathcal{K}}'|}$. From the equation (5.3) and Theorem 3, the test statistics is calculated as:

$$\frac{|\hat{\mathcal{K}}|}{|\hat{\mathcal{K}}'|} = \frac{n^{|V|} \frac{\prod_{S \in \mathcal{S}} |ssd_S|}{\prod_{C \in \mathcal{C}} |ssd_C|}}{n^{|V|} \frac{\prod_{S \in \mathcal{S}'} |ssd_S|}{\prod_{C \in \mathcal{C}'} |ssd_C}|} = \frac{\frac{\prod_{S \in \mathcal{S}} |ssd_S|}{\prod_{C \in \mathcal{C}} |ssd_C|}}{\frac{\prod_{S \in \mathcal{S} + C_{ab} \cap C_a + C_{ab} \cap C_b - S_{ab}} |ssd_S|}{\prod_{C \in \mathcal{C} + C_{ab}} |ssd_C|}}$$

which immediately gives the test statistics in the equation 5.6. We have considered the first case in Theorem 3 in above; moreover, considering the other three cases results in the same expression for the test statistics. □

5.1.4 Time Complexity Analysis

We now turn to the time complexity analysis of ContChordalysis in Algorithm 5.1. At the step t of the main loop of the algorithm, all candidate next graphs have t edges. The computation of the test statistics in line 8 is upper bounded by the evaluation of the maximal clique that can be formed by adding the tth edge to the graph. In the extreme case where all t edges form one clique, the maximum size of the clique would be[1] $k = \frac{1+\sqrt{1+8t}}{2}$. Hence the time complexity of computing the test statistics is $O(k^3) = O(t^{\frac{3}{2}})$, assuming computing the determinant of a k-by-k matrix is $O(k^3)$. The time complexity of enumerating the candidate edges in line 6 and adding a selected edge in line 13 is $O(|V|^2)$. Therefore, the time complexity of one pass over the main loop is $O(|V|^2 + |E^c|t^{\frac{3}{2}}) = O(|V|^2 t^{\frac{3}{2}})$ since the number of candidate edges E^c is upper bounded by the number of edges of the complete graph $\frac{|V|(|V|-1)}{2}$.

5.2 A Statistically Efficient Method for Structure Discovery

ContChordalysis is based on the maximum likelihood estimation, and uses multiple test correction to reduce the false discovery rate. However, the low-rate of false discoveries

[1] A clique of k nodes contains $\frac{k(k-1)}{2}$ edges. Setting the number of edges to t and solve the resulting quadratic equation yields $k = \frac{1+\sqrt{1+8t}}{2}$. (Petitjean et al., 2013)

comes at a price: it requires many samples to accept correct hypotheses. Moreover, ContChordalysis has a major functional drawback: it relies on the existence of the maximum likelihood estimates, which may not exist if the number of samples is less than the size of the largest clique in the graph. To overcome these drawbacks, we propose a new test statistic based on the minimum message length (MML).

The MML criterion provides an information-theoretic objective for statistical inference to find the best hypothesis for the observed data (Wallace & Boulton, 1968). It controls the false discovery rate, requiring far fewer samples to accept true hypotheses. MML relies on quantifying the amount of information required to convey losslessly the observed data in an explanation message. The best hypothesis is the one that can convey the entire data set in the shortest possible explanation message.

Let us consider a hypothesis (or model) $\mathcal{M}$ that offers an explanation of the observed data $\mathcal{D}$. Based on the fundamental rules of probability:

$$p(\mathcal{M}, \mathcal{D}) = p(\mathcal{M}) \times p(\mathcal{D}|\mathcal{M}) = p(\mathcal{D}) \times p(\mathcal{M}|\mathcal{D})$$

where $p(\mathcal{M})$ is the prior over hypotheses/models, $p(\mathcal{D}|\mathcal{M})$ is the likelihood, $p(\mathcal{D})$ is the prior probability of data, and $p(\mathcal{M}|\mathcal{D})$ is the posterior of $\mathcal{M}$ given $\mathcal{D}$. Using Shannon's communication theory, the amount of information for explaining $\mathcal{D}$ with $\mathcal{M}$ is:

$$I(\mathcal{M}, \mathcal{D}) = I(\mathcal{M}) + I(\mathcal{D}|\mathcal{M}) = I(\mathcal{D}) + I(\mathcal{M}|\mathcal{D}) \tag{5.7}$$

where $I(a) = -\log(p(a))$ gives the optimal code length to convey an event a whose probability is $p(a)$. This results in an objective criterion to compare two competing models $\mathcal{M}_1$ and $\mathcal{M}_2$ given the same data D:

$$I(\mathcal{M}_1|\mathcal{D}) - I(\mathcal{M}_2|\mathcal{D}) = I(\mathcal{M}_1) + I(\mathcal{D}|\mathcal{M}_1) - I(\mathcal{M}_2) - I(\mathcal{D}|\mathcal{M}_2). \tag{5.8}$$

A possible realisation of this framework is the transmission of data over a communication channel between the sender and the receiver. The sender sends $\mathcal{D}$ with an explanation message, so that the receiver can reconstruct back the original data losslessly from the message. The sender's message encodes both the model $\mathcal{M}$ and the data residual $p(\mathcal{D}|\mathcal{M})$. The receiver then reads in the model from the message, and decodes the original data from the residual. The goal of this communication game is to minimise the length of the explanation message, hence the use of the communication channel and resources (Wallace & Boulton, 1968). If the sender can find the best model on the data, the receiver will receive the most economic decodable explanation message; this is the basis of statistical inference based on the MML principle (Wallace & Boulton, 1968).

For our structure discovery setting, the encoding of the model in the message consists of the encoding of the chordal graph's topology G and the associated model parameters, which we elaborate in the rest of this section.

5.2.1 Encoding of the Graph

We now describe the encoding of the graphical structure G associated with the model $\mathcal{M}$ based on (Allisons, 2017) and (Petitjean et al., 2014). For this purpose, it is sufficient to send the edges of the graph: (a) the number of edges $|E|$, and (b) the particular combination of the edges that the graph exhibits if we have an enumeration of all possibilities. We do not need to encode the variables since it is common across all models, hence does not change the outcome of comparing messages.

We need $\log(|E_{Complete}| + 1)$ to encode the number of edges[2], where $|E_{Complete}| = \frac{|V| \times |V-1|}{2}$. For a given number of the edges, we ideally need to index and send only the chordal graphs. However, we are not aware of an analytical expression for the number of chordal graph with a fixed number of edges. Hence, we use the number of all graphs as an upper-bound, which results in sending more bits than necessary. The number of all possible graphs with a fixed number of edges $|E|$ is $\log \binom{|E_{complete}|}{|E|}$. Hence, the length of encoding the graph's topology is:

$$I(G) = \log(|E_{complete}| + 1) + \log \binom{|E_{complete}|}{|E|}. \tag{5.9}$$

[2] This includes zero for the null graph.

5.2.2 Encoding of the Parameters and the Data

Once the graph's topology has been encoded, we encode the parameters of the model as well as the data. To encode model parameters, we encode the parameters of all maximal cliques and minimal separators and then combine them. Let $k << |V|$ be the number of nodes in a maximal clique C (or alternatively, a minimal separator). Let $\mathcal{D}^C$ be the part of data set $\mathcal{D}$ corresponding to the variables in C. According to (Wallace & Boulton, 1968), the MML encoding of $\mathcal{D}^C$ and the parameters of the multivariate Gaussian distribution corresponding to a maximal clique (or minimal separator) C is:

$$I(C, \mathcal{D}^C) \overset{\text{def}}{=} \underbrace{\frac{m_C}{2}\log(q) + \frac{m_C}{2} - \overbrace{\log(h(\theta_C))}^{\text{Prior}} + \overbrace{\log\sqrt{|\mathcal{F}(\theta_C)|}}^{\text{Fisher information}}}_{I(C)} - \underbrace{\overbrace{\mathcal{L}(\mathcal{D}^C|\theta_C)}^{\text{log-likelihood}}}_{I(\mathcal{D}^C|C)} \tag{5.10}$$

where $m_C = \frac{d^2+3d}{2}$ is the number of free parameters, q is the lattice quantisation constant to reduce the quantisation error[3], and $\theta^C = (\mu^C, \Sigma^C)$. In what follows, we compute various components of $I(C, \mathcal{D}^C)$ in eqn (6.4), i.e. the prior probability, the Fisher information matrix, and the likelihood.

5.2.2.1 Prior probability of the parameters

Following (Dowe et al., 1996), we use a flat prior for μ^C and a conjugate inverted Wishart prior for Σ^C. Hence, the prior joint density over the parameters is $h(\theta^C) \propto |\Sigma^C|^{-\frac{k+1}{2}}$ where k is the size of the clique C.

5.2.2.2 Likelihood

The log-likelihood $\mathcal{L}(\mathcal{D}^C|\mu^C, \Sigma^C)$ of the relevant part of the data based on the multivariate Gaussian distribution corresponding the maximal clique C is

$$-\frac{nk}{2}\log 2\pi - \frac{n}{2}\log|\Sigma^C| - \frac{1}{2}\sum_{i=1}^{n}(X_i - \mu^C)\Sigma^{C^{-1}}(X_i - \mu^C)^T \tag{5.11}$$

[3] Quantisation error results from limited precision in machines when representing real numbers.

where k is the size of the clique $|C|$. The MLE estimates are given by:

$$\hat{\mu}^C = \frac{1}{n}\sum_{i=1}^{n} X_i^C \quad , \quad \hat{\Sigma}^C = \frac{1}{n-1}\sum_{i=1}^{n}(X_i^C - \hat{\mu}^C)(X_i^C - \hat{\mu}^C)^T \tag{5.12}$$

5.2.2.3 Fisher information of the parameters

We need to evaluate the second order partial derivatives of $-\mathcal{L}(D^C|\mu^C, \Sigma^C)$ for computing the Fisher information for the parameters (Wallace & Boulton, 1968). Let $|\mathcal{F}(\mu^C, \Sigma^C)|$ represent the determinant of the Fisher information matrix which is the product of $|\mathcal{F}(\mu^C)|$ and $|\mathcal{F}(\Sigma^C)|$, i.e. the determinant of the Fisher information matrices of μ^C and Σ^C, respectively.

Taking the second order partial derivatives of $-\mathcal{L}(\mathcal{D}^C|\mu^C, \Sigma^C)$ with respect to μ^C, we get $-\nabla_{\mu^{C^2}}\mathcal{L} = n\Sigma^{C^{-1}}$. So the determinant of the Fisher information matrix for μ^C is $|\mathcal{F}(\mu^C)| = n^k|\Sigma^C|^{-1}$.

To compute $|\mathcal{F}(\Sigma^C)|$, (Magnus & Neudecker, 1988) derived an analytical expression using the theory of matrix derivatives based on matrix vectorization:

$$|\mathcal{F}(\Sigma^C)| = n^{\frac{k(k+1)}{2}} 2^{-k} |\Sigma^C|^{-(k+2)}. \tag{5.13}$$

Hence, the determinant of the Fisher information matrix for μ^C and Σ^C is,

$$|\mathcal{F}(\mu^C, \Sigma^C)| = n^{\frac{k(k+3)}{2}} 2^{-k} |\Sigma^C|^{-(k+2)}. \tag{5.14}$$

5.2.2.4 Putting it all together

Substituting the prior probability, Fisher information and log-likelihood into eqn (6.4), the encoding of the parameters and data of a maximal-clique/minimal-separator is

$$I(C, \mathcal{D}^C) \;=\; \frac{n-1}{2}\log(|\Sigma_C|) + \frac{1}{2}\sum_{i=1}^{n}(x_i - \mu_C)\Sigma_C^{-1}(x_i - \mu_C)^T + c \tag{5.15}$$

where c is a constant.

After encoding the data and parameters corresponding to all maximal cliques and minimal separators, we need to combine them to get the length of the message needed to be sent from the sender to the receiver. Let us start by an example graphical model consisting of $C_{\{A;B\}}$ and $C_{\{B;C\}}$ as maximal cliques where S_B is their minimal separator. To send the parameters of this simple graphical model, one may think of sending the parameters of the multivariate Gaussian distributions corresponding to the maximal cliques and the separator $\{P(A,B), P(B,C), P(B)\}$, which include their means and covariance matrices. However, this encoding has redundancy as the parameters of the separator $P(B)$ can be reconstructed by the receiver from either $P(A,B)$ or $P(B,C)$ via marginalisation. Therefore, a more efficient encoding consists of sending the parameters of $\{P(A,B), P(B,C)\}$. This idea can be pushed further to send the parameters of $P(C|B)$ instead of $P(B,C)$, as the joint can be constructed from the conditional as well as the marginal $P(B)$ which is already computable from $P(A,B)$. Hence, a more efficient encoding may be that for sending $\{P(A,B), P(C|B)\}$ or $\{P(B,C), P(A|B)\}$. In general, there are exponentially many non-redundant sets of conditional and joint factors, from which the original joint distribution can be constructed. To find the minimum message length, we would need to have a search over this exponential space of sets. To avoid such search, we resort to a measure which sums up the encoding needed for the parameters of the maximal cliques, and deducts the encoding of the minimal separators to remove redundancy.

We resort to the following efficiently computable expression to approximate the message length consisting of the model and the data:

$$I(\mathcal{M}|\mathcal{D}) = I(G) + \sum_{C \in \mathcal{C}} I(C, \mathcal{D}^C) - \sum_{S \in \mathcal{S}} I(S, \mathcal{D}^S) \tag{5.16}$$

where $\mathcal{C}$ and $\mathcal{S}$ are the set of maximal cliques and minimal separators, respectively. A similar expression has been used in (Petitjean et al., 2014) to encode model parameters and data for discrete-valued graphical models.

Algorithm 5.2 ContChordalysis-MML

```
1:  Input: Dataset $D = \{X_i\}_{i=1}^n$
2:  Output: Graph $G = (V, E)$
3:  $t \leftarrow 0$
4:  Initialise $G$ to be the graph without any edges
5:  repeat
6:      $E^c \leftarrow$ CandidateEdge($G$) ▷ see Section 5.1.3
7:      for $e \in E^c$ do
8:          $s_e =$ MMLScore($D, G, e$) ▷ based on eqn (5.17)
9:      end for
10:     $e^* \leftarrow \arg\max_{e \in E^c} s_e$
11:     if $s_{e^*} > 0$ then
12:         $G \leftarrow$ addEdge($G, e*$) ▷ see Section 5.1.3
13:     end if
14:     $t \leftarrow t + 1$
15: until ($s_{e^*} < 0$) or ($t > \frac{|V|(|V|-1)}{2}$)
```

5.2.3 MML as Test Statistics

As mentioned earlier, we use forward selection to discover the graphical model. In forward selection, the reference model $\mathcal{M}$ and a candidate model $\mathcal{M}'$ are differed by and edge (a, b). According to MML theory, $\mathcal{M}'$ replaces $\mathcal{M}$ if encoding the message based on $\mathcal{M}'$ requires fewer bits than that of $\mathcal{M}$ i.e. $I(\mathcal{M}|\mathcal{D}, G) - I(\mathcal{M}'|\mathcal{D}, G') > 0$. Therefore, the MML score for comparing the reference and a candidate model is:

$$
\begin{aligned}
&I(\mathcal{M}|\mathcal{D}, G) - I(\mathcal{M}'|\mathcal{D}, G') = \\
&\log\left(\frac{|E_{complete}| - |E|}{|E_{complete}| - |E| - 1}\right) + I(C_{ab}, \mathcal{D}^{C_{ab}}) + I(S_{ab}, \mathcal{D}^{S_{ab}}) \\
&- I(C_{ab} \cap C_b, \mathcal{D}^{C_{ab} \cap C_b}) - I\big(C_{ab} \cap C_a, \mathcal{D}^{C_{ab} \cap C_a}\big).
\end{aligned} \tag{5.17}
$$

The resulting method, which we call ContChordalysis-MML, is summarised in Algorithm 5.2; it differs from Algorithm 5.1 only in lines from 8 to 11.

The time complexity of computing $I(C, \mathcal{D}^C)$ is $O(|C|^3)$, since it contains the inverse and the determinant of $|C|$-by-$|C|$ matrices. Therefore, the time complexity of Algorithm 5.2 is similar to that of the Algorithm 5.1, where one pass over the main loop is $O(|V|^2 + |E^c|t^{\frac{3}{2}}) = O(|V|^2 t^{\frac{3}{2}})$ in the round t of the main loop.

5.2.4 Relationship between Kolmogorove complexity and MML

Kolmogorov complexity has a similar objective function as MML. According to Wallace and Dowe (1999), there is no essential difference between Kolmogorov Complexity and Minimum Message Length approaches. They differ only in the choice of reference Turing machines. Any Universal machine is regarded as acceptable in Kolmogorov complexity, whereas MML usually restricts the reference machine to a non-universal form in the interest of computational feasibility. Furthermore, MML chooses the machine in form of the prior probabilities which are data dependent. This attention to the choice of machine allows MML, in domains where the possible theories are computable, to estimate complexities with errors of only a few digits. As a result, MML can be, and has routinely been, applied with some confidence to many problems of machine learning, inductive and statistical inference from finite bodies of real data.

5.3 Experiments and results

We compare the performance of our methods (ContChordalysis and ContChordalysis-MML) with five baselines on both synthetic and real-life datasets. We implement ContChordalysis and its variants in Matlab 2014b. All experiments are run on a desktop with Intel Core i5 3.2GHz CPU and 8GB of RAM.

5.3.1 Baselines

We compare our methods with five strong competing methods: TIGER (H. Liu, 2017), CLIME (Cai et al., 2011), Graphical Lasso (GLasso) (J. Friedman et al., 2008), rooted Graphical Lasso (r-GLasso) (Avagyan et al., 2017) and a recently proposed greedy approach called FoBa-gdt (J. Liu et al., 2014). TIGER uses SQRT-Lasso from (Belloni, Chernozhukov, & Wang, 2012) for estimating both the graph G and the precision matrix K. CLIME (Cai et al., 2011) uses linear programming in Lasso to estimate the precision matrix. GLasso (J. Friedman et al., 2008) uses coordinate descent algorithm in Lasso to estimate the precision matrix and the graph structrue G. r-GLasso (Avagyan et al.,

2017) is the most recent variant of the GLasso approach which estimates the kth rooted precision matrix to predict more accurate Gaussian graphical model structure. Finally, FoBa-gdt (J. Liu et al., 2014) is a forward-backward greedy approach to discover the graph G. All of the baselines use penalized log-likelihood as the objective function. Moreover, all of the baselines discover the Gaussian graphical model structures. For these two reasons, we compare our methods with the above mentioned baselines to evaluate the performance. All of the baselines are implemented in R packages and publicly available through CRAN.

5.3.2 Other scoring functions

In this chapter, we have proposed two test statistics based on MML and p-value to select an optimal solution. We compare the performance of the proposed test statistics to two scoring functions: (Altmueller & Haralick, 2004)'s proposed MDL (Minimum Descriptor Length) score and BIC (Bayesian Information Criterion) (Foygel & Drton, 2010).

$$\begin{aligned} Altmueller's\ score = & - \mathcal{L}(\mathcal{D}|\theta) + \log n \\ & + \sum_{i=1}^{|\mathcal{C}|} (|E_i| \log n) + \sum_{i=1}^{|\mathcal{C}|} |V_i| - \sum_{i=1}^{|\mathcal{S}|} |V_i| \end{aligned} \tag{5.18}$$

$$BIC = -2\mathcal{L}(\mathcal{D}|\theta) + |E| \log n \tag{5.19}$$

where $\mathcal{L}(\mathcal{D}|\theta) \;=\; \sum_{c=1}^{|\mathcal{C}|} \mathcal{L}(\mathcal{D}^c|\theta^c) - \sum_{s=1}^{|\mathcal{S}|} \mathcal{L}(\mathcal{D}^s|\theta^s)$.

We employ BIC and Altmueller's MDL scores to form additional variants of ContChordalysis, and call them *ConChordalysis-BIC* and *ContChordalysis-Altmueller*, respectively.

5.3.3 Performance Metrics

We evaluate results using standard performance metrics: precision, recall, and FMeasure. Precision is the fraction of correctly predicted edges (i.e. associations) with respect to all predicted edges. Recall is the fraction of correctly predicted edges with respect to the correct edges. Fmeasure is the harmonic mean of precision and recall, i.e. $\text{Fmeasure} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$.

In our synthetic data experiments, the graphs used for generating synthetic data are considered the gold standard. Both ContChordalysis and ContChordalysis-MML use decomposable model to discover the GGM.

5.3.4 Synthetic data

We generate synthetic data based on the various combinations of the number of data points n and the number of variables/dimensions $|V|$, where $n \in \{100, 1000, 20000\}$ and $|V| \in \{100, 1000\}$. For each combination, we generate graphs with different properties: random networks (RN) and small world network (SWN). We have 2 graph types and 6 combinations of the number of data points and variables. For each case, we generate a positive definite precision matrix K, where K_{ij} is nonzero if the corresponding edge exists in the graph. For each $2 \times 6 = 12$ configuration, we generate 3 datasets from the corresponding multivariate Gaussian distribution, and report the average results over these three randomly generated datasets.

In both synthetic data experiments, we make use of five-fold cross validation, thereby dividing the dataset into five partitions. We take any of these five partitions as the test set and use the other four partitions as the training sets to learn the regularization parameters of the competitive methods.

5.3.4.1 Random network (RN)

We carry out experiments on the datasets of random networks. Random network (RN) generation is a process where each edge is chosen with probability p as the result of a coin toss. To generate the random network, we consider a random ordering of all possible edges, and process them from left to right. For each edge, we toss a coin with probability p to see whether it should be chosen. We discard the edge if it does not form a chordal graph with all previously selected edges. We have generated graphs with the coin parameters $p = 0.5$. In this experiment, we evaluate the performance of our methods and baselines on chordal graphs.

Table 5.1 compares our methods with the baselines on synthetic data generated from random networks. ContChordalysis-MML outperforms the other methods in all configurations in terms of Precision, Recall, and FMeasure. As expected, for a particular dimension size, the FMeasure of all methods improves as the sample size n increases. Likewise, in almost all cases, the FMeasure decreases for a particular sample size as the number of dimensions increases. In other words, ContChordalysis-MML requires fewer samples compared to other methods to discover meaningful associations between the variables.

Interestingly, ContChordalysis-MML outperforms ContChordalysis based on the empirical results in Table 5.1. In ContChordalysis, the threshold α_t is geometrically decreased as new edges are added to the graph. Hence after a few steps, the threshold becomes very small, stopping the addition of new edges. Therefore, the number of edges in graphs discovered by ContChordalysis is small, which leads to missing a large number of true edges. This is confirmed by inspecting the number of edges in the graphs discovered by different methods, reported in TP and FP columns of Table 5.2.

MML extensively uses the covariance matrix to predict the association between the random variables. Whereas, MDL (Altmueller & Haralick, 2004) and BIC (Foygel & Drton, 2010) use the number of edges instead of the covariance matrix to resolve the over fitting problem of MLE and ignore the effect of covariance matrix. Therefore, MDL and BIC do not perform as well as MML.

(Giraud, 2014) point out the limitation of BIC on high dimensional data and performs well for data having small number of variables. We confirm this observation by noting that, it discovers smaller number of associations (TP and FP columns of

Table 5.1: Average recall (Re), precision (Pr), $Fmeasure$ (FM) and computational times (in seconds) of ContChordalysis and its variants, TIGER, CLIME, GLasso, r-GLasso and FoBa-gdt for random networks.

Methods	n = 100				n = 1000				n = 20000			
	Re	Pr	FM	T(s)	Re	Pr	FM	T(s)	Re	Pr	FM	T(s)
$\|V\|$ = 100												
ContChordalysis-MML	0.74	0.68	0.71	612.7	0.79	0.68	0.73	778.7	0.82	0.79	0.80	829.1
ContChordalysis	0.32	0.67	0.43	147.1	0.31	0.59	0.41	159.6	0.28	0.51	0.36	245.8
ContChordalysis-Altmueller	0.17	0.40	0.23	158.6	0.16	0.42	0.23	197.3	0.17	0.42	0.24	280.1
ContChordalysis-BIC	0.12	0.39	0.18	142.3	0.12	0.41	0.19	181.2	0.12	0.39	0.19	252.3
TIGER	0.61	0.64	0.62	685.9	0.60	0.66	0.63	804.7	0.51	0.62	0.56	922.8
CLIME	0.47	0.63	0.54	2071.3	0.41	0.61	0.49	4108.6	0.45	0.72	0.55	6501.1
GLasso	0.43	0.51	0.47	695.8	0.41	0.55	0.47	791.1	0.42	0.62	0.50	902.4
r-GLasso	0.47	0.54	0.50	689.3	0.46	0.59	0.52	801.3	0.46	0.65	0.54	926.5
FoBa-gdt	0.29	0.59	0.39	152.4	0.35	0.64	0.45	172.6	0.40	0.71	0.51	268.3
$\|V\|$ = 1000												
ContChordalysis-MML	0.69	0.66	0.67	1071.8	0.69	0.71	0.70	1193.8	0.72	0.80	0.76	1279.3
ContChordalysis	0.13	0.47	0.20	259.3	0.12	0.51	0.20	273.1	0.13	0.61	0.22	302.5
ContChordalysis-Altmueller	0.09	0.45	0.15	289.3	0.07	0.45	0.13	313.1	0.09	0.49	0.16	346.2
ContChordalysis-BIC	0.08	0.42	0.13	275.2	0.07	0.45	0.12	291.6	0.08	0.47	0.13	375.2
TIGER	0.49	0.65	0.56	1198.3	0.42	0.69	0.52	1263.4	0.39	0.72	0.51	1511.7
CLIME	0.41	0.61	0.49	11005.7	0.36	0.66	0.47	13759.7	0.32	0.70	0.44	16681.7
GLasso	0.34	0.49	0.40	1188.6	0.30	0.53	0.38	1245.2	0.28	0.59	0.38	1482.4
r-GLasso	0.36	0.51	0.42	1179.5	0.32	0.56	0.41	1237.8	0.30	0.61	0.40	1429.6
FoBa-gdt	0.16	0.58	0.25	272.8	0.16	0.63	0.25	289.7	0.15	0.69	0.25	327.6

Table 5.2: Average Confusion matrix of ContChordalysis-MML and its variants, TIGER, CLIME, GLasso, r-GLasso and FoBa-gdt for random networks.

Methods	$n = 100$				$n = 1000$				$n = 20000$			
	TP	TN	FP	FN	TP	TN	FP	FN	TP	TN	FP	FN
$\lvert V\rvert = 100$												
ContChordalysis-MML	800	3493	375	282	984	3242	462	262	1170	3213	310	257
ContChordalysis	344	3699	169	738	390	3433	271	856	397	3143	380	1030
ContChordalysis-Altmueller	179	3600	268	903	197	3433	271	1049	237	3197	326	1190
ContChordalysis-BIC	130	3666	202	952	153	3485	219	1093	175	3251	272	1252
TIGER	656	3500	368	426	742	3323	381	504	729	3077	446	698
CLIME	513	3568	300	569	508	3380	324	738	643	3274	249	784
GLasso	466	3421	447	616	514	3284	420	732	593	3161	362	834
r-GLasso	505	3439	429	577	578	3304	400	668	650	3173	350	777
FoBa-gdt	318	3648	220	764	434	3461	243	812	565	3293	230	862
$\lvert V\rvert = 1000$												
ContChordalysis-MML	151198	202483	77889	67930	191100	144489	78054	85857	241597	103549	60399	93955
ContChordalysis	27676	249163	31209	191452	34481	189415	33128	242476	45221	135037	28911	290331
ContChordalysis-Altmueller	19197	256911	23461	199931	20570	197404	25139	256387	30918	131770	32178	304634
ContChordalysis-BIC	17111	256744	23628	202017	18760	199616	22927	258197	26431	134144	29804	309121
TIGER	107874	222287	58085	111254	117180	169898	52645	159777	131020	112997	50951	204532
CLIME	90226	222687	57685	128902	99896	171082	51461	177061	108418	117484	46464	227134
GLasso	74165	203180	77192	144963	82088	149749	72794	194869	93510	98968	64980	242042
r-GLasso	78990	204480	75892	140138	88755	152808	69735	188202	101235	99224	64724	234317
FoBa-gdt	34949	255065	25307	184179	43586	196945	25598	233371	51744	140702	23246	283808

the table 5.2) than ContChordalysis. This consequently affects its recall, precision and FMeasure. (Altmueller & Haralick, 2004)'s score is a variant of BIC. Therefore, (Altmueller & Haralick, 2004)'s score performs similar to BIC and is outperformed by both ContChordalysis-MML and ContChordalysis.

TIGER, CLIME, GLasso and r-GLasso use Lasso to estimate the precision matrix. In Lasso, the regularization parameter λ in the penalized likelihood objective functions significantly affects the precision matrix estimation process. TIGER, CLIME, GLasso and r-GLasso are outperformed by ContChordalysis-MML. Similar to other baselines, FoBa-gdt is also outperformed by ContChordalysis-MML. FoBa-gdt uses a penalized likelihood as the objective function and it removes edges at backward elimination step until the objective function finds an optimal solution. Therefore, it removes many true edges which affects the recall, precision and FMeasure.

Particularly in the case when n=100 and $|V|$=1000, ContChordalysis-MML's precision is comparable with TIGER and CLIME. Whereas the recall and FMeasure of ContChordalysis-MML is much better.

Table 5.1 also shows that ContChordalysis runs faster than the baselines, although it suffers from the inaccurate prediction of associations. ContChordalysis-MML runs much faster than the baseline methods: TIGER, CLIME, GLasso, r-GLasso and FoBa-gdt. Therefore, ContChordalysis-MML is a statistically efficient and scalable method for predicting associations.

5.3.4.2 Synthetic data experiments: Small world network (SWN)

Small-world network (SWN) generation process where the resulted graphs have the power-law property. Many real-life networks are SWNs, e.g. social networks and gene networks (Watts & Strogatz, 1998). To generate a SWN, we make use of the Watts-Strogatz algorithm (Watts & Strogatz, 1998). In small world network experiment, we did not check chordality of the gold standard graph during generating the data. We observe similar trends to random network experiments in that ContChordalysis-MML outperforms other methods in terms of FMeasure. Speedwise, ContChordalysis is faster than the other methods followed by ContChordalysis-MML which is much faster than the baselines.

Interestingly, precision of ContChordalysis and its variants are not as good as TIGER. In the small world network experiments, we did not check the chordality of the graphs when generating the dataset. Therefore, to maintain the chordality, ContChordalysis and its MML variant predict many edges (showed in table 5.4) which affected the precision.

5.3.5 Acute Myeloid Leukemia gene expression data

We also apply our methods to TCGA cancer gene expression datasets: AML (Acute Myeloid Leukemia) to discover the gene network. We download both gene expression datasets from cBioPortal[4]. In the experiments, we focus on the cancer related transcription factors (TFs). The AML gene expression dataset contains 51 TFs[5] and 173 samples. For the gold standard, we use the *regulatory potential scores*[6] between a pair of genes, i.e. TFs for AML cancer based on TF ChIP-seq binding data from the Cistrome Cancer Database[7]. Following the previous work (C. Wang et al., 2013), an edge exist between two TFs if their regulatory potential score is at least 0.5.

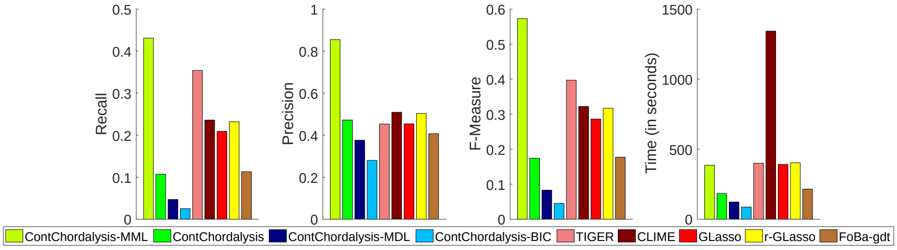


Figure 5.2: Average recall (Re), precision (Pr), Fmeasure (FM) and computational times (in seconds) of the discovered graphical structures by different methods for the AML cancer gene expression data.

Table 5.5 presents the number of edges predicted by the baselines and the variants of our method. ContChordalysis-MML recovers not only more edges than other methods

[4] http://www.cbioportal.org

[5] This dataset contains only TFs, no other genes

[6] Regulatory potential scores are a computational tool to aid in the identification of putative regulatory sites of the human genome

[7] http://cistrome.org/CistromeCancer

Table 5.3: Average recall (Re), precision (Pr), $Fmeasure$ (FM) and computational times (in seconds) of ContChordalysis-MML and its variants, TIGER, CLIME, GLasso, r-GLasso and FoBa-gdt for small world networks.

Methods	n = 100				n = 1000				n = 20000			
	Re	Pr	FM	T(s)	Re	Pr	FM	T(s)	Re	Pr	FM	T(s)
$\|V\|$ = 100												
ContChordalysis-MML	0.80	0.71	0.75	768.3	0.82	0.72	0.77	814.6	0.86	0.79	0.82	867.5
ContChordalysis	0.19	0.52	0.27	167.1	0.25	0.57	0.34	171.9	0.21	0.58	0.31	192.8
ContChordalysis-Altmueller	0.14	0.47	0.22	151.2	0.16	0.55	0.25	192.6	0.17	0.61	0.27	226.1
ContChordalysis-BIC	0.07	0.37	0.12	133.2	0.10	0.46	0.16	163.1	0.10	0.51	0.17	233.6
TIGER	0.56	0.81	0.66	881.4	0.57	0.83	0.68	887.3	0.50	0.85	0.63	916.7
CLIME	0.38	0.69	0.49	2763.8	0.40	0.72	0.52	4566.7	0.35	0.75	0.48	6897.5
GLasso	0.38	0.64	0.48	795.8	0.40	0.66	0.49	822.0	0.34	0.68	0.45	882.6
r-GLasso	0.40	0.66	0.50	791.2	0.42	0.69	0.52	817.3	0.38	0.72	0.49	879.3
FoBa-gdt	0.24	0.62	0.35	158.9	0.31	0.67	0.42	169.4	0.27	0.67	0.39	223.6
$\|V\|$ = 1000												
ContChordalysis-MML	0.66	0.69	0.67	3145.2	0.69	0.75	0.72	3480.9	0.77	0.83	0.80	3568.2
ContChordalysis	0.08	0.39	0.13	382.2	0.08	0.41	0.14	444.7	0.09	0.49	0.15	509.4
ContChordalysis-Altmueller	0.07	0.52	0.13	261.2	0.07	0.58	0.13	317.2	0.07	0.63	0.13	378.1
ContChordalysis-BIC	0.05	0.46	0.09	248.2	0.06	0.51	0.10	294.1	0.06	0.56	0.11	322.3
TIGER	0.38	0.70	0.49	3266.8	0.36	0.73	0.48	3728.4	0.36	0.78	0.50	4129.1
CLIME	0.38	0.74	0.50	12491.6	0.32	0.75	0.45	15501.3	0.30	0.78	0.44	18331.3
GLasso	0.31	0.59	0.41	3288.6	0.28	0.62	0.39	3545.2	0.28	0.67	0.40	3682.9
r-GLasso	0.35	0.63	0.45	3315.6	0.32	0.66	0.43	3552.4	0.31	0.70	0.43	3701.5
FoBa-gdt	0.09	0.44	0.15	322.6	0.11	0.49	0.17	382.4	0.12	0.59	0.20	451.9

Table 5.4: Average Confusion matrix of ContChordalysis-MML and its variants, TIGER, CLIME, GLasso, r-GLasso and FoBa-gdt for small world networks.

Methods	$n = 100$				$n = 1000$				$n = 20000$			
	TP	TN	FP	FN	TP	TN	FP	FN	TP	TN	FP	FN
$\|V\| = 100$												
ContChordalysis-MML	1100	3126	449	275	1168	3071	454	257	1506	2798	400	246
ContChordalysis	255	3340	235	1120	352	3260	265	1073	375	2927	271	1377
ContChordalysis-Altmueller	192	3359	216	1183	235	3333	192	1190	299	3007	191	1453
ContChordalysis-BIC	98	3409	166	1277	139	3362	163	1286	182	3025	173	1570
TIGER	774	3394	181	601	812	3359	166	613	871	3045	153	881
CLIME	524	3340	235	851	573	3303	222	852	610	2995	203	1142
GLasso	521	3283	292	854	563	3236	289	862	593	2920	278	1159
r-GLasso	550	3293	282	825	601	3255	270	824	658	2943	255	1094
FoBa-gdt	334	3371	204	1041	443	3307	218	982	475	2965	233	1277
$\|V\| = 1000$												
ContChordalysis-MML	165520	174348	74364	85268	211380	122693	70459	94968	274840	86272	56292	82096
ContChordalysis	19102	218836	29876	231686	25870	155926	37226	280478	32375	108869	33695	324561
ContChordalysis-Altmueller	18031	232069	16643	232757	22052	177185	15967	284296	25666	127491	15073	331270
ContChordalysis-BIC	11824	234832	13880	238964	17282	176549	16603	289066	21787	125446	17118	335149
TIGER	94428	208243	40469	156360	110491	152286	40866	195857	129449	106054	36510	227487
CLIME	95331	215218	33494	155457	98870	160196	32956	207478	107671	112196	30368	249265
GLasso	77778	194664	54048	173010	85585	140698	52454	220763	101410	92617	49947	255526
r-GLasso	86965	197638	51074	163823	97622	142863	50289	208726	110943	95017	47547	245993
FoBa-gdt	23092	219323	29389	227696	32375	159457	33695	273973	41770	113539	29025	315166

(column-2 of Table 5.5) but also more “unique edges" (column-4 of same table), i.e. those correct edges which are not detected by other methods. Figure 5.2 depicts the results, and shows that ContChordalysis-MML outperforms the other methods in terms of FMeasure.

We also compare the run time of the different methods to discover the graphical structures. As shown in Figure 5.2, the speed trend is similar to those observed in the synthesis experiments, where ContChordalysis-BIC is the fastest method, followed by ContChordalysis-MML, which is in turn faster than the baselines.

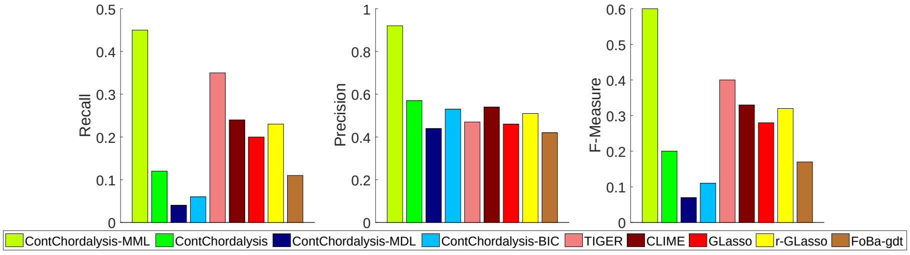


Figure 5.3: Average recall (Re), precision (Pr) and Fmeasure (FM) of the discovered graphical structures by different methods for the moralized AML cancer gene expression data.

The gold standard graph of the AML dataset is not chordal. Hence, we add some edges to the gold standard graph to make it chordal, which should give an upper bound on the performance of our methods along with the baselines. Therefore, we use the moralized AML dataset to find the upper bound on the performance of all of the comparing methods. Columns 5 to 7 of table 5.5 present the recovery of the number of moralized AML gold standard edges by the baselines and our method. Figure 5.3 depicts the upper bound on FMeasure/Precision/Recall of our methods and the baselines. Original gold standard graph is nearly chordal, and we add only 19 edges to make it chordal. Therefore, there is not a significant difference between the upper bound FMeasure and the original FMeasure. Most important findings of this experiment is that TIGER, CLIME, GLasso, r-GLasso and FoBa-gdt do not perform well on chordal graphs, which is reflected in their FMeasures.

Table 5.5: The number of matched, predicted and unique prediction of edges by different methods in the AML gene expression data.

Method	Number of predicted edges compared with gold standard data					
	Non-moralized			moralized		
	The number of true edges $(TP + FN)$: 550			The number of true edges $(TP + FN)$: 569		
	matched (TP)	Predicted $(TP + FP)$	Unique	matched (TP)	Predicted $(TP + FP)$	Unique
ContChordalysis-MML	**237**	**277**	**45**	**254**	**277**	**47**
ContChordalysis	59	125	5	71	125	4
ContChordalysis-Altmueller	26	69	0	37	69	0
ContChordalysis-BIC	14	50	0	22	50	0
TIGER	195	430	31	201	430	33
CLIME	130	255	4	137	255	3
GLasso	115	253	0	116	253	0
r-GLasso	128	254	8	130	254	7
FoBa-gdt	62	152	2	64	152	2

5.3.6 Breast cancer gene expression data

We also apply our methods to TCGA cancer gene expression datasets: BRCA (Breast invasive Carcinoma) to infer the gene network. We download BRCA gene expression datasets from cBioPortal. Similar to AML experiments, we focus on the cancer related transcription factors (TFs). The BRCA gene expression dataset contains 729 TFs[8] and 528 samples. For the gold standard, we use the *regulatory potential scores* between a pair of genes, i.e. TFs for BRCA cancer based on TF ChIP-seq binding data from the Cistrome Cancer Database. Following the previous work: (C. Wang et al., 2013) and AML experiment, an edge exist between two TFs if their regulatory potential score is at least 0.5.

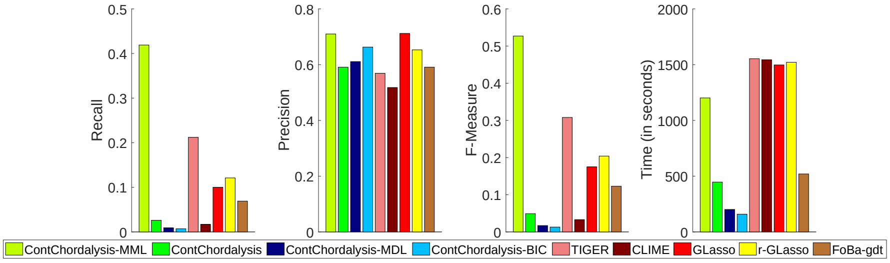


Figure 5.4: Average recall (Re), precision (Pr), Fmeasure (FM) and computational times (in seconds) of the discovered graphical structures by different methods for the breast invasive carcinoma (BRCA) gene expression data.

Figure 5.4 presents recall, precision, FMeasure and the running time of the variants of our method versus the baselines. We again see the same trend that ContChordalysis-MML outperforms other methods in terms of the performance measures. We list the number of edges predicted by ContChordalysis (including its variants) and the baselines in Table 5.6. We see that ContChordalysis-MML discovers more true edges than any other of compared methods. It also detects more than 3000 true gene-pairs which are not detected by the baselines.

`BRCA_1` is an important susceptible gene that may cause the appearance of breast and ovarian cancers in human body (Miki et al., 1994; Brose et al., 2002; Finch et al., 2006). (Pujana et al., 2007) has identified the list of genes which have interacted with `BRCA_1` to

[8] This dataset contains only TFs, no other genes

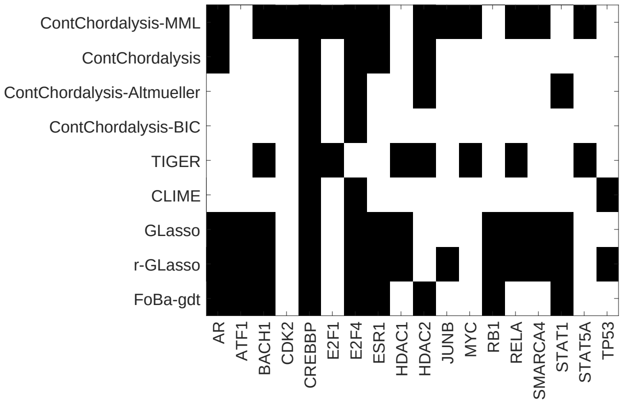


Figure 5.5: Discovery of interacting gene with `BRCA_1` which causes the appearance the breast cancer (BRCA) by different methods.

create cancer cells in breast and ovary. We are interested to assess the methods by comparing their discoveries on genes having interaction with `BRCA_1`. Figure 5.5 shows that ContChordalysis-MML spots 13 true gene-pairs, whereas TIGER/CLIME/GLasso/r-GLasso/FoBa-gdt detect 11/3/7/12/9 true gene-pairs respectively.

It is known that many gene-pairs can be responsible for the appearance of cancer cells in human body. We are also interested in knowing which important gene-pairs have been detected by the different methods. According to (Qin et al., 2016), gene-pairs with higher RTS (regulatory potential scores) are the important gene-pairs for the appearance of breast cancer. We select 50 important gene-pairs based on the RTS between the gene-pairs. Figure 5.6 shows that ContChordalysis-MML detected 37 gene-pairs, while TIGER the strongest baseline method and r-GLasso detect just 27 gene-pairs. CLIME, GLasso and FoBa-gdt discover fewer than 25 TF-pairs. Based on this evaluation, our method ContChordalysis-MML outperforms existing strong baselines by discovering more true important gene-pairs.

We also analyse the computational time of the different methods for discovering associations among the variables. Figure 5.4 shows that the variants of our method are faster than all the baselines.

Similar to the moralized experiment on AML gene expression data, we carry out an experiment to measure the upper bound on the performance of our methods and the baseline by making the gold standard graph chordal. Columns 5 to 7 of table 5.6 present

Table 5.6: The number of matched, predicted and unique prediction of edges by different methods in the BRCA gene expression data.

Method	Number of predicted edges compared with gold standard data					
	Non-moralized			moralized		
	The number of true edges $(TP + FN) : 25833$			The number of true edges $(TP + FN) : 26971$		
	matched (TP)	Predicted $(TP + FP)$	Unique	matched (TP)	Predicted $(TP + FP)$	Unique
ContChordalysis-MML	**10824**	**14627**	**3077**	**11374**	**14627**	**3241**
ContChordalysis	694	1126	88	807	1126	97
ContChordalysis-Altmueller	228	409	0	273	409	1
ContChordalysis-BIC	201	346	0	236	346	0
TIGER	5475	7873	1584	5603	7874	1891
CLIME	464	703	21	481	703	0
GLasso	2593	3642	61	2664	3642	57
r-GLasso	3129	4788	97	3257	4788	82
FoBa-gdt	1780	3011	11	1859	3011	7

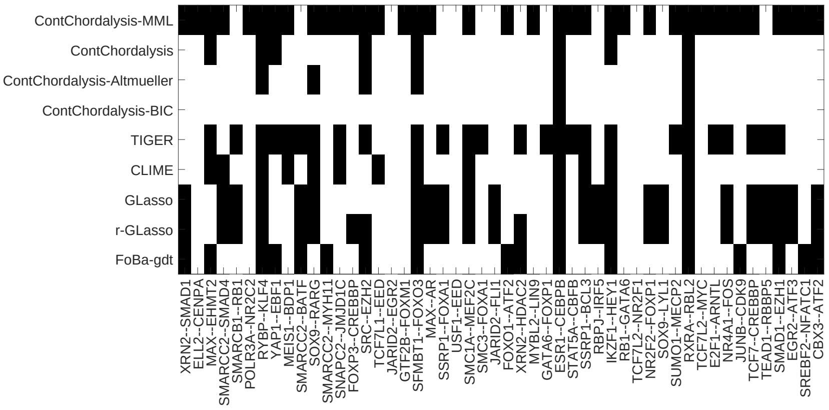


Figure 5.6: Discovery of 50 prominent gene-pairs, causing the breast cancer (BRCA), by different methods.

the recovery of the number of moralized BRCA gold standard edges by the baselines and our method. Figure 5.7 depicts the upper bound on FMeasure/Precision/Recall of our methods and baselines on BRCA dataset. Our MML approach predicted more accurate edges than other methods and outperformed the baselines.

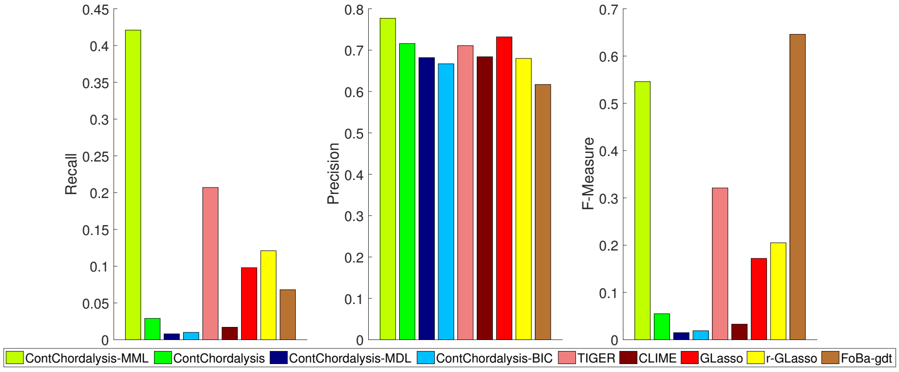


Figure 5.7: Average recall (Re), precision (Pr) and Fmeasure (FM) of the discovered graphical structures by different methods for the moralized BRCA cancer gene expression data.

5.3.7 Ancestry gene expression data

We also perform experiment on another real dataset that was used by TIGER, one of the most competitive baselines. This dataset contains unrelated individuals of Northern and Western European ancestry from Utah (CEU), whose genotypes are available from the Sanger Institute website[9] (Bhadra & Mallick, 2013). The number of samples n is 60 and the dimension size d is 100. (Bhadra & Mallick, 2013) have analyzed the data and found 55 significant interactions among the 100 chosen traits. (Mohammadi & Wit, 2015) used a Bayesian method to infer the gene network with 281 edges which include all of the significant interaction discovered by (Bhadra & Mallick, 2013). Moreover, among the 281 edges, (Mohammadi & Wit, 2015) identified 86 edges as significant interactions.

(H. Liu, 2017) used this dataset to evaluate the performance of TIGER. We only test ContChordalysis and ContChordalysis-MML on this data. Table 5.7 presents the number of edges predicted by the TIGER[10] and the variants of our method. From Table 5.7, we can say that our ContChordalysis-MML discovered more accurate graphical structure than TIGER and outperformed TIGER and ContChordalysis.

Table 5.7: The number of edges predicted by ContChordalysis, its MML variant and TIGER including the significant edges found by (Bhadra & Mallick, 2013) and (Mohammadi & Wit, 2015) from human gene expression data ancestry.

Methods	The number of gold standard edges predicted by		
	Total	(Bhadra & Mallick, 2013)	(Mohammadi & Wit, 2015)
ContChordalysis-MML	618	45	74
ContChordalysis	108	13	19
TIGER (H. Liu, 2017)	306	40	70

5.3.8 Patient classification data

We carry out another experiments on the problem of predicting the breast cancer patient with their types. In this experiment, r-GLasso (Avagyan et al., 2017), GLasso and CLIME predict the cancer patients with pCR (pathological complete response) and residual disease (RD) from the graphical model structure. These baselines predict the relationships between the patients, instead of the genes. Patients having the same disease type will be connected with each other, otherwise they are not connected. In this experiment, we use the same dataset[11] that (Avagyan et al., 2017) used which contains 22,283 gene expression levels of 133 patients. There are 34 patients with pCR

[9] ftp://ftp.sanger.ac.uk/pub/genevar

[10] This experiment is already carried out by (H. Liu, 2017) and reported in their chapter.

[11] available at http://bioinformatics.mdanderson.org/pubdata.html

Table 5.8: Average pCR and RD classification measurements

Method	Specificity	Sensitivity	MCC
ContChordalysis-MML	0.91	0.86	0.74
ContChordalysis	0.90	0.41	0.36
GLasso(J. Friedman et al., 2008)	0.75	0.61	0.33
r-GLasso(Avagyan et al., 2017)	0.69	0.84	0.48
CLIME(Cai et al., 2011)	0.71	0.84	0.49

and 99 patients with RD. Similar to (Avagyan et al., 2017), we consider the results of (Hess et al., 2006) as the gold standard.

To measure the prediction accuracy, (Avagyan et al., 2017) used specificity, sensitivity and Matthew correlation coefficient[12] (MCC). Moreover, they consider TP and TN as the number of correctly predicted pCR and RD patients, respectively, and FP and FN as the number of erroneously predicted pCR and RD patients, respectively. Therefore, we use specificity, sensitivity and MCC instead of recall, precision and FMeasure to evaluate the performance. (Avagyan et al., 2017) also compared their method with GLasso and CLIME on this dataset.

Table 5.8 presents the pCR and RD patients classification results. In the table, we have reported the results of GLasso, r-GLasso and CLIME from (Avagyan et al., 2017)'s paper. Based on the results of table 5.8, our method ContChordalysis-MML outperformed CLIME, GLasso and r-GLasso.

5.3.9 Real-life data experiment on Finance stock performance of the companies

We carry out further experiments on another real dataset: "Finance stock performance of the companies" used in (Petitjean & Webb, 2015), which contains 20 years financial performance of 490 companies. The number of samples in the dataset is 3450, where the financial footprints in individual days are considered as samples. Using this dataset, we identify the financial relationship between the companies. As we do not have any gold standard data for this dataset, we compute the log-likelihood of the held-out data

[12] The Matthews correlation coefficient (MCC) is used in machine learning as a measure of the quality of binary (two-class) classifications. $MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}}$

to evaluate the performance of the methods.

We make use of five-fold cross validation, thereby dividing the dataset into five partitions of size 690 samples. We take any of these five partitions as the test set and use the other four partitions as the training sets to learn the structure of the graphical model.

Table 5.9 shows the average log-likelihood of the models recovered by ContChordalysis-MML is higher than that for the other methods. Furthermore, the average log-likelihood per edge for ContChordalysis-MML is higher than the other methods. Overall, these results indicate that ContChordalysis-MML is more accurate in predicting the association between variables compared to the baselines.

Table 5.9: Average log-likelihood of different methods on the dataset of finance stock performance of the companies

Method	Log-Likelihood	Log-Likelihood per edge
ContChordalysis-MML	**-895.29**	**-0.19**
ContChordalysis	-989.72	-1.44
ContChordalysis-Altmueller	-922.53	-1.82
ContChordalysis-BIC	-995.61	-2.09
TIGER	-908.76	-0.24
CLIME	-969.63	-1.25
GLasso	-993.75	-0.39
rGLasso	-1011.17	-0.34
FoBa-gdt	-973.6	-1.17

5.4 Conclusion

We have proposed a scalable and statistically efficient approach for graphical model structure discovery involving continuous variables for exploratory data analysis. We introduce ContChordalysis and it variants, including a novel MML-based criterion, for structure discovery of Gaussian graphical models. Our methods are step-wise algorithms, where they add edges maximising a test statistics incrementally to the estimated graph. ContChordalysis makes use of log-likelihood ratio test, and ContChordalysis-

MML uses an information theoretic criterion based on minimum message length principle. Our methods work with chordal graphs and decomposable models to make the computation of the test statistics efficient. We have presented extensive empirical results on synthetic and real-life datasets, and shown that our ContChordalysis-MML method outperforms strong baselines in terms of both speed and the accuracy of the predicted associations from the data.

ContChordalysis-MML discovers the dependencies between random variables more accurately in faster ways and outperforms strong baselines, assuming that all observations are generated from the same underlying multivariate distribution. However, recent studies on cancer genome atlas network have found that gene expression data can be described as the mixtures of the small number of components harbouring different expression pathways (Mukherjee & Roriguez, 2016). Thus, real-life datasets exhibit heterogeneity, which can be accommodated through the use of mixtures of graphical models to let each component exhibit different conditional dependencies among variables, a.k.a "context-specific-dependencies" (Meilă & Jordan, 2000; Rodriguez et al., 2011). Moreover, Guo et al. (2011) and Rodriguez et al. (2011) emphasized that context-specific graphical structures share some edges with each other. Hence, as the next research goal, we investigate the discovery of context-specific dependencies among random variables in faster way with lower false discovery rate while there are far less number of samples (i.e. observations) compared to the number of variables generated from a mixture with unknown number of components. We address this target to improve the graphical model discovery method for the heterogeneous dataset in next chapter.

Chapter 6

Structure Learning with Mixture of Gaussian Graphical Models

6.1 Introduction

In ContChordalysis and its MML variant's setting, it is assumed that all observations are generated from a single underlying multivariate Gaussian distribution. However, recent studies on Cancer Genome Atlas Network have found that gene expression data can be better described by mixtures where different components harbour different expression pathways (Mukherjee & Roriguez, 2016). Typically there are far less number of samples (i.e. observations) compared to the number of variables, generated from the mixtures with *unknown* number of components. Therefore, real-life datasets exhibit heterogeneity, which can be better modeled through the use of mixtures of GGMs to let each component exhibit different conditional dependencies among the variables, a.k.a *context-specific-dependencies* (Meilă & Jordan, 2000; Rodriguez et al., 2011). However, high-dimensional heterogenous data make the context-specific conditional dependencies discovery challenging.

Initial methods: Chow and Liu (1968); Meilă and Jordan (2000) to discover context-specific graphical models predict the context-specific associations without any common conditional dependencies[1]. However, Guo et al. (2011); Rodriguez et al. (2011) emphasized that context-specific graphical structures share most of the edges, which are not well discovered by the above mentioned methods. Moreover, these methods discover context-specific graphical models with common dependencies by predicting many false edges.

In this chapter, we address and resolve the above issues (predicting many false conditional dependencies and discovering less common dependencies) by proposing a novel method to learn the mixtures of GGM based on an iterative algorithm, which iterates over the following two steps. First, it clusters the data into distinct clusters. Second, it employs the forward selection algorithm (Deshpande et al., 2001) for discovering the graphical model structure of each cluster. To discover context specific graphical models and their common structure, our method incrementally adds the best edge maximising a scoring function in the forward selection algorithm. In both steps of iterative algorithm, we use minimum message length (MML) as the objective function. Our MML-based approach is an information theoretic method enjoying (a) low false discovery rate, (b) suitability for the small number of samples when discovering statistical dependencies (associations) among large number of variables, and (c) scalability to large-scale problems involving thousands of variables. As mentioned in Chapter 4, an integral part of our MML based objective function is the maximum likelihood estimate for the parameters of the competing models, we restrict the structure of the competing models to *chordal* graphs which leads to decomposable models.

N. Friedman (1998) proposed a similar method called *Structural EM* algorithm to predict the cluster and their graphical models. But there are significant differences between the structural EM and our iterative algorithm. Structural EM algorithm is developed for directed acyclic graphs. Whereas our algorithm is proposed for decomposable undirected graph. Moreover, in each iteration, structural EM algorithm maximizes the maximum likelihood based scoring function. Whereas, our proposed iterative algorithm uses MML which is subject to minimize. Furthermore, instead of a single objective

[1] Common conditional dependencies are the conditional dependencies present in all components of the mixture.

function, we use two MML based objective functions: one for clustering and another for predicting context-specific GGMs.

We present extensive empirical results on synthetic and real-life datasets and show that our method leads to more accurate prediction of context-specific dependencies among variables, compared to the previous works.

6.2 Discovering a mixture of the decomposable GGMs

Let $\mathcal{D} = \{X_1, \ldots, X_n\}$ be a training set consisting of n data points where $X_i \in \mathbb{R}^d$ and d is the number of dimensions (equivalently, the number of random variables). We assume the data have been generated from a mixture of multivariate Gaussian distributions, where each component corresponds to a graphical model. Our aim is to discover the unobserved structure of undirected Gaussian graphical models, corresponding to the mixture components, based on the observed data $\mathcal{D}$:

$$P(\mathcal{D}) = \sum_{i=1}^{K} \gamma_i P(\mathcal{D}_i) \tag{6.1}$$

where γ_i is the mixing coefficient and $\mathcal{D}_i$ is the datapoints of cluster i. K is the number of the clusters/ components

Specifically, we are interested in the undirected graphical structures $\boldsymbol{G} = \{G_1, G_2, ..., G_k\}$ where $G_i = \{V, E_i\}$ is the context specific graphical structure of the component i, V is the set of vertices corresponding to random variables (or dimensions of the input vectors), E_i is the set of edges capturing context-specific statistical associations between random variables, and K is the number of components in the mixture model.

The input to the algorithm is the number of components K believed to exist in the data. The output is then the partitioned data with context-specific graphical model structures. The algorithm consists of two steps: (a) the clustering step, similar to the E step in the hard-EM algorithm, to partition the data and (b) the structure and parameter estimation step, similar to the M-step of the EM algorithm. In the estimation step, we employ ContChordalysis-MML, described in section previous chapter). Our algorithm keeps repeating the clustering, and structure and parameter estimation steps until it converges with respect to the objective function.

Our algorithm optimizes a minimum message length (MML) based objective function for estimating the structure of the context-specific graphical models. We call our iterative algorithm: ***P**artition **a**nd **G**raphical model discovery **I**terative **A**lgorithm based on **MML*** or PaGIAM, as summarized in Algorithm 6.1.

Algorithm 6.1 PaGIAM

```
INPUT: Data $\mathcal{D}$ and the number of components $k$
OUTPUT: Context specific graphical structures of the mixture model $\mathcal{G}$ and partitioned data.
Initialization:
Randomly partition $\mathcal{D}$ into $K$ cluster(s)
Compute $\gamma$ from partitioned $\mathcal{D}$
$MML_c \leftarrow \infty$
repeat
    $MML_p \leftarrow MML_c$
    structure and parameter estimation step:
    for $i \leftarrow 1$ to $K$ do
        $G_i \leftarrow$ ContChordalysis-MML($\mathcal{D}_i$)
    end for
    Clustering step:
    for $i \leftarrow 1$ to $n$ do
        $a \leftarrow argmax_{t \in k}\big(\gamma_t P(D^i|G_t, \theta_t)\big)$
        $\mathcal{D}_a \leftarrow \mathcal{D}_a \bigcup D^i$
    end for
    Compute $\gamma$ from new partition of $\mathcal{D}$
    Compute $MML_c$ using equation 6.10
until $MML_c \geq MML_p$
```

The computational time complexity of ContChordalysis-MML is $O(|V|^2t^{1.5})$[2], where t is the number of iteration. It is a time inefficient algorithm for the large dataset which

[2] In the Chapter 5, we describe that the computation of the MML based objective function of ContChordalysis-MML is upper bounded by the evaluation of the maximal clique that can be formed by adding the tth edge to the graph. Let consider a clique of x nodes contains $\frac{x(x-1)}{2}$ edges. Setting the number of edges to t and solve the resulting quadratic equation yields $x = \frac{1+\sqrt{1+8t}}{2}$. (Petitjean et al., 2013). In the extreme case where all t edges form one clique, the maximum size of the clique would be $x = \frac{1+\sqrt{1+8t}}{2}$. Hence the time complexity of computing the test statistics is $O(x^3) = O(t^{\frac{3}{2}})$, assuming computing the determinant of a x-by-x matrix is $O(x^3)$.

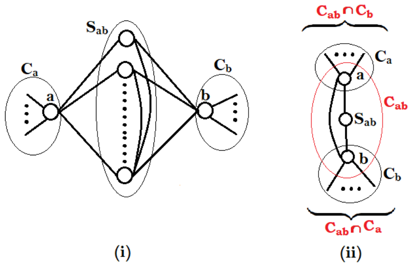


Figure 6.1: Structure of (i) the cliques C_a, C_b and separator S_{ab} in reference model; and (ii) newly formed clique C_{ab} and separators $C_{ab} \cap C_a$ and $C_{ab} \cap C_b$ in candidate model.

further slow down our mixture model structures discovery. In the next section, we propose a more scalable version of ContChordalysis-MML for faster computation. After discussing the scalable ContChordalysis-MML algorithm, we detail our MML based objective function for mixture model structure discovery.

6.2.1 Scalable ContChordalysis-MML

ContChordalysis and its MML variant are forward selection algorithms which adds the best edge to the candidate graphical structure, check the candidature of remaining edges, and re-compute the scoring function (i.e. objective function) for all candidate edges. However, the edge candidature checking and score computation make the forward selection strategy slow for a very large number of random variables.

6.2.1.1 Faster computation of MML scores of the edges

In ContChordalysis and its variants, we re-examine and re-compute the score of each candidate edge. According to the theorem 3 of the Chapter 5, addition of an edge (a, b) in a candidate model affects the separator between the two nodes a and b: S_{ab} and creates one new clique C_{ab} and two new separators $C_{ab} \cap C_a$ and $C_{ab} \cap C_b$. All the other separators and cliques remain unchanged. Figure 6.1(i) shows that the reference graph structure before adding the edge (a, b). Figure 6.1 (ii) shows that an edge (a, b) is added to candidate graph structure which forms new clique C_{ab} (red coloured) by merging the minimal separator S_{ab} and, nodes a and b and two new separators $C_{ab} \cap C_a$ and $C_{ab} \cap C_b$ (red coloured). All other maximal cliques (black colours) and minimal separators are

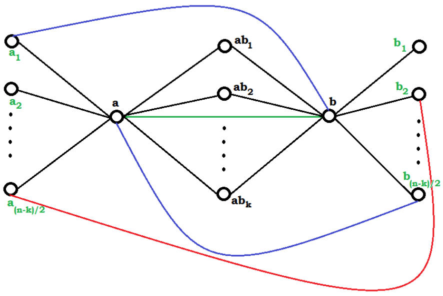


Figure 6.2: After adding the edge (a, b) (green coloured edge) changes the status of candidate edges

unaffected. Therefore, it leads to following theorem

Theorem 5. *For any edge (a, b), MML score at any step t would remain unchanged if the minimal separator S_{ab} between nodes a and b is unchanged.*

Proof. According to ContChordalysis, the efficient computation of the change in the MML score for encoding the data and parameters due to add an edge (a, b) follows following equation:

$$\Delta MML^*_{ab} = I(C_{ab}, \mathcal{D}^{C_{ab}}) + I(S_{ab}, \mathcal{D}^{S_{ab}}) - I(C_{ab} \cap C_b, \mathcal{D}^{C_{ab} \cap C_b}) - I\big(C_{ab} \cap C_a, \mathcal{D}^{C_{ab} \cap C_a}\big) \tag{6.2}$$

In this equation, S_{ab} is the minimal separator between the nodes a and b. Between the steps t and $t-1$, when S_{ab} is unchanged, C_{ab}, $C_{ab} \cap C_a$ and $C_{ab} \cap C_b$ are also unchanged where $C_{ab} = S_{ab} \cup \{a, b\}$. Hence, the change in the MML score for encoding the parameters and data due to the edge (a, b) would remain unchanged. □

Based on the above theorem, it is not required to re-compute the MML score for all candidate edges at every step. Re-computation of the score is only required for the candidate edges whose minimal separator have been changed after adding the edge (a, b). According to Petitjean and Webb (2015), the addition of an edge (a, b) to the candidate model affects the minimal separators between following pairs of nodes: (i) a and the neighbours of b, (ii) b and the neighbours of a, and (iii) neighbours of a and b. Figure 6.2 shows that after adding the edge (a, b) (green coloured edge) into the candidate model, the edges between the node b and the neighbours of node a (blue coloured edges) (similarly the node a and the neighbours of node b) form the cycle with length 3 and maintain the chordality in the graph structure. Therefore, first two types of pairs of nodes ((i) and (ii) types of pairs of nodes) form cycles with length of 3. Therefore, the edges whose scores need to be re-computed after the addition of the edge (a, b) to the model are: $(a, N(b))$ and $(N(a), b)$, where $N(a)$ and $N(b)$ be the neighbours of node a and b respectively. The number of edges whose scores are re-computed, would be at most $2n$. Hence, the computational complexity of each step, compared to before is improved to $O(|V|)$.

6.2.1.2 Faster computation of candidature checking of the edges

According to Petitjean and Webb (2015), addition of an edge (a, b) changes the minimal separators between the nodes pairs between the neighbours of nodes a and b. Node pairs between neighbours of nodes a and b may form a cycle in the model whose length would be more than 3. Figure 6.2 shows that after adding the edge (a, b) (green coloured edge) into the candidate model, the edges between the neighbours of node a and node b (red coloured edges) form the cycle with length 4. The length of a cycle must not be 4 or more in a chordal graph those edges which form a cycle with length 4 or more, would loose their status to become candidate edges. Therefore, instead of re-examine the candidature of all edges, we only check the candidature of node pairs between the neighbours of nodes a and b.

The computational time complexity to re-examine candidature of the node pairs between the neighbours of nodes a and b would be $O(|V|)$. Based on the above observation, the computational time complexity improves to $O(|V|)$. The Improved ContChordalysis-MML algorithm is summarised in Algorithm 6.2.

6.2.2 The MML objective function for mixture of GGMs

The MML based objective function encodes the hypothesis in a message including the encoding of the clusters of the data and the associated cluster parameters, and the graphical model structures including its maximal cliques and minimal separators parameters.

Algorithm 6.2 Scalable ContChordalysis-MML

```
Require: $\mathcal{D}$
Ensure: Graphical structure $\mathcal{G} = \{V, E\}$.
Initial step:
all edges are candidate edges.
$E \leftarrow \emptyset$
for each edge $(a, b) \in E^c$ do
    $MML^*_{(a,b)} \leftarrow I(C_{ab}, \mathcal{D}^{C_{ab}}) + I(S_{ab}, \mathcal{D}^{S_{ab}}) - I(C_{ab} \cap C_a, \mathcal{D}^{C_{ab}} \cap C_a) - I(C_{ab} \cap C_b, \mathcal{D}^{C_{ab}} \cap C_b)$
end for
repeat
    $MML^*_{(a,b)} \leftarrow argmin_{(x,y) \in E^c} MML^*_{(x,y)}$
    $MML(a, b) \leftarrow I(G') - I(G) + MML^*_{(a,b)}$
    if $MML_{(a,b)} < 0$ then
        $E \leftarrow E + (a, b)$
        $E^c \leftarrow E^c - (a, b)$
        Readjust $E^c$ ▷ discussed in Section 6.2.1.2
        Recompute $ML_{(x,y)}$ ▷ discussed in Section 6.2.1.1
        $t \leftarrow t + 1$
    end if
until $MML_{(a,b)} > 0$ or $t = \frac{n(n-1)}{2}$
```

As we use the ContChordalysis-MML to discover the graphical model structure of each cluster, therefore, we use same encoding of the maximal cliques and minimal separators parameters and the graphical model structures (discussed in details in Chapter 5). In this section, we elaborate the encoding of the clusters of the data and the associated cluster parameters and the data.

6.2.2.1 Encoding the clusters

We now describe the encoding of clusters which includes their contents and coefficient; and the number of clusters in the mixture. Firstly, we encode the number of clusters, for which we need $\log(K)$ bits. We then encode the coefficient of clusters (i.e. mixing coefficient). According to (Boulton & C.S.Wallace, 1969), to encode the coefficient of each cluster, we need $\log n_i - \log n$ bits, where n and n_i are the total number of datapoints in $\mathcal{D}$ and $\mathcal{D}_i$ respectively. Therefore, to encode coefficients of all of the clusters , we need $\sum_{i=1}^{k} \big(\log n_i - \log n\big)$ bits. Finally, we encode the content of clusters by encoding the cluster indicator vector $\vec{z}_i$ for reach data point $\mathcal{D}_i$. The cluster indicator vector $\vec{z}_i$ contains a numerical value between 1 and K to indicate cluster membership of datapoints. To encode $\vec{z}_i$ vectors, we need $\sum_{i=1}^{n} \log K$ bits in total. Therefore, the minimum message length to encode all clusters is:

$$I(\mathcal{H}) = \log K + \sum_{i=1}^{K} \big(\log n_i - \log n \big) + \sum_{i=1}^{n} \log K \tag{6.3}$$

6.2.2.2 Encoding of the Parameters and the Data

Once clusters have been encoded, we encode parameters of clusters as well as the data. We encode parameters and data of all clusters separately and then combine them. According to (Wallace & Boulton, 1968), the MML encoding of $\mathcal{D}_i$ and parameters of the multivariate Gaussian distribution corresponding to a cluster i, denoted by $I(\mathcal{D}_i, \theta_i)$, is:

$$I(\mathcal{D}_i, \theta_i) = \underbrace{\Bigg(- \overbrace{\log(p(\theta_i))}^{\text{Prior}} + \overbrace{\log \sqrt{|\mathcal{F}(\theta_i)|}}^{\text{Fisher information}} \Bigg)}_{I(\theta_i)} + \underbrace{\Bigg(- \overbrace{\mathcal{L}(\mathcal{D}_i|\theta_i)}^{\text{log-likelihood}} \Bigg)}_{I(\mathcal{D}_i|\theta_i)} \tag{6.4}$$

where $\theta_i = (\mu_i, \Sigma_i)$. In what follows, we compute various terms of $I(\theta_i, \mathcal{D}_i)$ in eqn (6.4), i.e. the prior probability, the Fisher information matrix, and the likelihood.

Prior probability of the parameters: Following the previous work (Dowe et al., 1996), we use a flat prior for μ_i (Oliver et al., 1996) and a conjugate inverted Wishart prior for Σ_i (Guavain & Lee, 1998). Hence, the prior joint density over the parameters is:

$$p(\theta_i) \propto |\Sigma_i|^{-\frac{d+1}{2}} \tag{6.5}$$

Likelihood: The log-likelihood $\mathcal{L}(\mathcal{D}_i|\mu_i, \Sigma_i)$ of the relevant part of the data is

$$-\frac{n_i}{2} \log 2\pi - \frac{n_i}{2} \log |\Sigma_i| - \frac{1}{2} \sum_{j=1}^{n_i} (D_{ij} - \mu_i) \Sigma_i^{-1} (D_{ij} - \mu_i)^T \tag{6.6}$$

where, $\mu_i = \frac{1}{n\,i} \sum_{j=1}^{n_i} D_{ij}$ and $\Sigma_i = \frac{1}{n_i - 1} \sum_{j=1}^{n_i} (D_{ij} - \mu_i)(D_{ij} - \mu_i)^T$.

Fisher information of the parameters: We need to evaluate the second order partial derivatives of $-\mathcal{L}(\mathcal{D}_i|\mu_i, \Sigma_i)$ to compute the Fisher information for the parameters

(Wallace & Boulton, 1968). Let $|\mathcal{F}(\mu_i, \Sigma_i)|$ represent the determinant of the Fisher information matrix which is the product of $|\mathcal{F}(\mu_i)|$ and $|\mathcal{F}(\Sigma_i)|$, i.e. the determinant of Fisher information matrices of μ_i and Σ_i, respectively (Oliver et al., 1996). Taking the second order partial derivatives of $-\mathcal{L}(\mathcal{D}_i|\mu_i, \Sigma_i)$ with respect to μ_i, we get $-\nabla^2_{\mu_i}\mathcal{L} = n_i\Sigma_i^{-1}$. So the determinant of the Fisher information matrix for μ_i is $|\mathcal{F}(\mu_i)| = n_i^d|\Sigma_i|^{-1}$. To compute $|\mathcal{F}(\Sigma_i)|$, (Dwyer, 1967) derived an analytical expression using the theory of matrix derivatives based on matrix vectorization:

$$|\mathcal{F}(\Sigma_i)| = n_i^{\frac{d(d+1)}{2}} 2^{-n_i} |\Sigma_i|^{-(d+2)}$$

Hence, the determinant of the Fisher information matrix for μ_i and Σ_i is

$$|\mathcal{F}(\mu_i, \Sigma_i)| = n_i^{\frac{d(d+3)}{2}} 2^{-d} |\Sigma_i|^{-(d+2)} \tag{6.7}$$

Putting it altogether, we have

$$I(\theta_i) = -\frac{1}{2}\log|\Sigma_i| \tag{6.8}$$

Substituting prior probabilities, Fisher information and log-likelihood in equation 6.4, encoding of parameters and data of a particular cluster is

$$I(\mathcal{D}_i, \theta_i) = \frac{n_i - 1}{2}\log(|\Sigma_i|) + \frac{1}{2}\sum_{j=1}^{n_i}(D_{ij} - \mu_i)\Sigma_i^{-1}(D_{ij} - \mu_i)^T \tag{6.9}$$

Therefore, MML to encode the both hypothesis (i.e. clusters and graph structures, and their parameters) and data is:

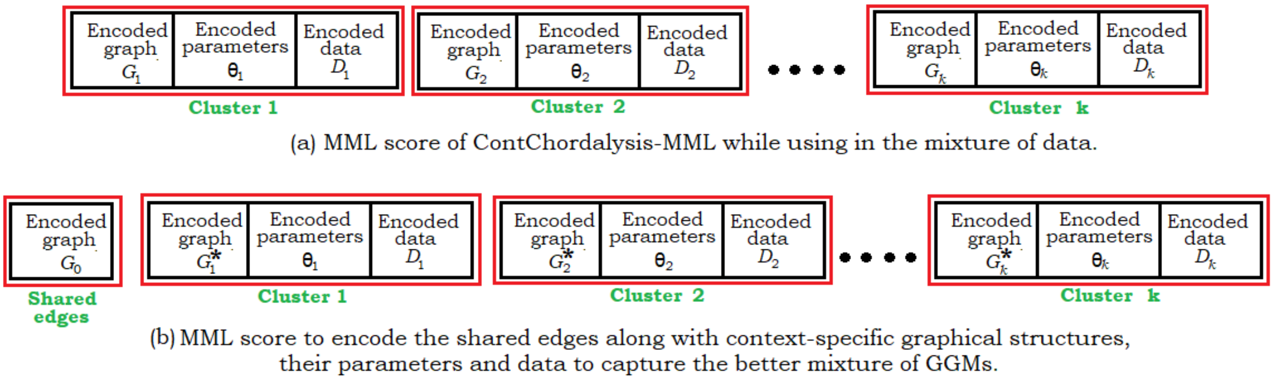
$$\begin{aligned} MML \quad = \quad & \underbrace{\log K + \sum_{i=1}^{n}\log K + \sum_{i=1}^{K}\log n_i - \log n}_{\text{Encoding the cluster}} + \underbrace{\sum_{i=1}^{K}\frac{n_i - 1}{2}\log(|\Sigma_i|) + \frac{1}{2}\sum_{j=1}^{n_i}(D_{ij} - \mu_i)\Sigma_i^{-1}(D_{ij} - \mu_i)^T}_{I(\mathcal{D}_i, \theta_i)} + \\ & \sum_{i=1}^{K}\Bigg(\underbrace{(n_i - 1)\{\sum_{c\in\mathcal{C}}\log|\Sigma_i^c| - \sum_{s\in\mathcal{S}}\log|\Sigma_i^s|\}}_{\text{encoding clique and separator parameters}} + \underbrace{\log(|E_{complete}| + 1) + \log\binom{|E_{complete}|}{|E_i|}}_{\text{Graph structure}}\Bigg) \end{aligned} \tag{6.10}$$

where D_{ij} is the jth data point of cluster i. $|E_{complete}|$ is the number of edges of a complete graph. Σ_i^C is the covariance matrix of a clique (separator) C of cluster i. E_i is a set of context-specific associations of cluster i.

Guo et al. (2011) investigated that all context specific graphical models share many edges among themselves. Whereas, ContChordalysis-MML does not discover the shared dependencies from the mixture of data. Moreover, it discovers each graphical model independently which makes the PaGIAM method slower. In next section, we detail a new GGM discovery method to discover shared and context specific graphical model structures using decomposable models and MML.

6.3 Capturing shared edges to improve the mixture of GGMs discovery

The method mentioned in section 6.2 discovers the structure of context-specific graphical models without leveraging the shared edges among them. However, in heterogeneous data, context-specific graphical structures share a significant number of edges. Modelling the shared structure can help the GGM structure discovery by pooling the statistics together. Following the previous chapter, we find that MML has performed very well as the test statistics for discovering graphical models. Therefore, we extend our approach to discover the shared edges and use them for a more effective encoding of the model in the MML sence. We learn the model based on an MML-based score. Our approach works with chordal graphs, leading to "decomposable" probabilistic graphical models, enjoying efficient computation of the MML scoring function. According to Wallace and Boulton (1968), the minimum message length finds the best model for the observed data by comparing the two competing models given the same data $\mathcal{D}$. To find the best structures, we encode the graph structures of shared edges (we call it super graph G_0) along the context-specific GGM structures $\{G_1, G_2, ..., G_K\}$, their parameters and the data in messages and compare their message lengths. While using ContChordalysis-MML in PaGIAM, we encode the graph topology G_i, the parameters θ_i and the data $\mathcal{D}_i$ of each cluster i and then combine them. It does not encode the super graph topology (See the Figure 6.3 (a)) and predicted less number of true edges or dependencies. To improve the better structures discovery from the heterogeneous data, we encode the super graph G_0 topology, the topology of context specific GGMs $\{G_1, G_2, ..., G_K\}$, their parameters $\{\theta_1, \theta_2, ..., \theta_K\}$ and data $\{\mathcal{D}_1, \mathcal{D}_2, ..., \mathcal{D}_K\}$. To minimize the number of required bits of MML, we only encode the edges of the context-specific GGMs which are not present in super graph, which is $G_i^* = G_i - G_0$. Figure 6.3 (b) shows the encoding of super graph, context-specific GGMs, their parameters and data to compute the MML score for better discovery of the context-specific GGMs.



(a) MML score of ContChordalysis-MML while using in the mixture of data.

(b) MML score to encode the shared edges along with context-specific graphical structures, their parameters and data to capture the better mixture of GGMs.

Figure 6.3: Difference between the computed MML scores of ContChordalysis-MML and proposed method

Therefore, the MML to encode the super graph, context-specific GGMs, their parameters and the data is as follows:

$$\overbrace{\underbrace{I(G_0)}_{\text{Super graph}} + \sum_{i=1}^{K} \underbrace{I(G_i^*)}_{\text{Context-specific GGMs}}}^{\text{Encoding all graph structures } I(\mathcal{G})} + \overbrace{\sum_{i=1}^{K} \left(\underbrace{I(\theta_i)}_{\text{Parameters of models}} + \underbrace{I(\mathcal{D}_i|\theta_i, G_i)}_{\text{Data fit to the models}} \right)}^{\text{Parameters of context specific graphs with data, } I(\mathcal{D}, \theta).} \tag{6.11}$$

where G_0 is the shared graphical structures which we call the super graph structure and $G_i = G_i^* + G_0$ is the context specific graph structure of ith component. G_i^* is the context specific graphical structure of component i without shared edges. Moreover, θ_i is the parameters of context specific model of component i. $\theta_i = \{\mu_i, \Sigma_i\}$ where μ_i and Σ_i mean vector and covariance matrix of graphical structure of component i. According to (Wallace & Boulton, 1968), equation 6.11 deduces to

$$I(G_0) + \sum_{i=1}^{K} I(G_i^*) + \sum_{i=1}^{K} \Bigg[\underbrace{-\log \frac{p(\theta_i)}{\sqrt{|\mathcal{F}(\theta_i)|}}}_{I(\theta_i)} \underbrace{- \sum_{j=1}^{n_i} \big(\mathcal{L}(D_{ij}|\theta_i) + \log K\big)}_{I(\mathcal{D}_i|\theta_i, G_i)} \Bigg] \tag{6.12}$$

where, extra $\log K$ bits are added with each datapoint to select its component id. For our two-level GGM structure discovery setting, the encoding of the model in the message consists of the encoding of topologies of the super and context specific chordal graphs and the associated model parameters, which we elaborate in the rest of this section.

6.3.1 Encoding the graph structures

We now describe the encoding of super and context specific graphical structures. For this purpose, it is sufficient to send the number of nodes and the connected pair of edges of each graphical structures. According to (Allisons, 2017), to encode the number of nodes, we need $\log n$ bits. Let us consider a super graph having $|E_0|$ number of edges. Therefore, to encode edges of super graph, we need $\log \binom{|E_{complete}|}{|E_0|}$, where $|E_{complete}| = \frac{d(d-1)}{2}$. We encode the component specific edges of context specific graphical structure G_i to prevent the multiple appearances of same edges in different graph structures. Therefore required bits to encode any context specific graph structure G_i^* is $\log \binom{|E_{complete}|}{|E_i - E_0|}$.

$$I(G_0) + \sum_{i=1}^{K} G_i^* = \log n + \log \binom{|E_{complete}|}{|E_0|} + \sum_{i=1}^{k} \log \binom{|E_{complete}|}{|E_i - E_0|} \tag{6.13}$$

6.3.2 Encoding the parameters and data

Once the graphs' topologies have been encoded, we encode the parameters of all context specific graphical model structures of the mixture of GGMs as well as the data independently and then merge them. To encode parameters and data of each context specific graphical structures, we encode parameters and data of all maximal cliques and minimal separators separately and then combine them. Moreover, ContChordalysis-MML encodes parameters and data of all maximal cliques and minimal separators of a graphical structure separately and then combines them. According to the previous chapter, to encode the parameters of a maximal clique (or minimal separator) C of a graphical model, we require

$$\log \frac{p(\theta_i^C)}{\sqrt{|\mathcal{F}(\theta_i^C)|}} = \frac{1}{2}\log|\Sigma_i^C| + Constant \tag{6.14}$$

Furthermore, we require following bits to encode the data of a maximal clique (or minimal separator) C of each context specific graphical model i:

$$\mathcal{L}(\mathcal{D}_i^C|\theta_i^C) = -\frac{1}{2}\sum_{i=1}^{n_i}(D_{ij}^C - \mu_i^C)\Sigma_i^C(D_{ij}^C - \mu_i^C)^T - \frac{n}{2}\log|\Sigma_i^C| \tag{6.15}$$

6.3.3 MML based Model Selection

In forward selection, a reference model $\mathcal{M}$ and a candidate model $\mathcal{M}'$ are differed by and edge (a,b). According to MML, $\mathcal{M}'$ replaces $\mathcal{M}$ if encoding the message based on $\mathcal{M}'$ requires less number of bits than that of $\mathcal{M}$ i.e. $I(\mathcal{M}'|\mathcal{D},G') - I(\mathcal{M}|\mathcal{D},G) < 0$, where G and G' are the graphical structures of reference and candidate models, respectively.

According to Deshpande et al. (2001), the edge (a,b) removes the separator S_{ab} from the reference model and creates a new clique C_{ab} and separators $C_{ab} \cap C_a$ and $C_{ab} \cap C_b$ in the candidate model. The rest of all maximal cliques and minimal separators remain unchanged in both models. Therefore, we only need to compute the MML score to encode the affected and newly appeared maximal cliques and minimal separators and their data (discussed in previous section in details).

In the mixture of GGMs, we follow the following steps to discover the structures of super and context-specific graphical models:

1. At the beginning, we consider that the super and all context-specific graphs are null graphs

2. We incrementally add the best edges e to either both the super graph G_0 and all context specific graphs or one of the context specific graphs by comparing reference and candidate models of super and context specific graphical models simultaneously.

3. We update the candidate edge list E^c (discussed in Section 6.2.1.2).

4. We re-compute the MML scores of the edges whose minimal separators and maximal cliques are changed after adding the e (discussed in Section 6.2.1.1).

5. We remove the best edges e from the candidate edge lists. Moreover, since the best edges e are added to any context-specific GGMs, we exclude e from the candidate edges of other context-specific graphs and super graphs to make the process simple.

Here, we present the MML based scoring function to compare the reference and the candidate models of super and context specific graphical models. We compute two types MML scoring functions:

1. MML score when an edge will be added to super and all context-specific graphs.

2. MML score when an edge will be added to any of the context-specific graphs.

6.3.3.1 MML score when an edge (a, b) is to be added to super graph

When a candidate edge is added to the super graph structure, it affects graph structures of both the super and context specific and their parameters. Therefore the MML difference between the candidate and reference graphical structures is as followed:

$$\begin{aligned} I(G') - I(G) &= \underbrace{\log n + \log \binom{|E_{complete}|}{|E_0|+1} + \sum_{i=1}^{K} \binom{|E_{complete}|}{|E_i - E_0|+1}}_{\text{Candidate graphical model}} \\ &\quad \underbrace{- \log n - \log \binom{|E_{complete}|}{|E_0|} - \sum_{i=1}^{K} \binom{|E_{complete}|}{|E_i - E_0|}}_{\text{Reference graphical model}} \\ &= \log \frac{|E_{complete}| - |E_0|}{|E_0|+1} + \sum_{i=1}^{k} \log \frac{|E_i - E_0|}{|E_{complete}| - E_i + E_0 + 1} \end{aligned} \tag{6.16}$$

The addition of an edge to the super graph affects the covariance matrices of affected and newly appeared cliques and separators of all context-specific graphical models.

Therefore, we encode covariance matrices of affected and newly formed separators and cliques of all context-specific graphical models and we need following bits

$$\begin{aligned}
I(\mathcal{D},\theta') - I(\mathcal{D},\theta) &= \sum_{i=1}^{K} \big(I(\theta_i') - I(\theta_i) + I(\mathcal{D}_i|\theta_i', G_i') - I(\mathcal{D}_i|\theta_i, G_i)\big) \\
&= -\frac{1}{2}\sum_{i=1}^{K} \left[\log \frac{|\Sigma_i^{C_{ab}}| \cdot |\Sigma_i^{S_{ab}}|}{|\Sigma_i^{C_{ab}\cap C_a}| \cdot |\Sigma_i^{C_{ab}\cap C_b}|}\right] \\
&\quad -\sum_{i=1}^{K} \left\{ \sum_{j=1}^{n_i} \Big(\mathcal{L}(D_{ij}^{C_{ab}}|\theta_i^{C_{ab}}) + \mathcal{L}(D_{ij}^{S_{ab}}|\theta_i^{S_{ab}}) - \mathcal{L}(D_{ij}^{C_{ab}\cap C_a}|\theta_i^{C_{ab}\cap C_a}) \right. \\
&\quad \left. -\mathcal{L}(D_{ij}^{C_{ab}\cap C_b}|\theta_i^{C_{ab}\cap C_b}) + \log K \Big) \right\} \qquad (6.17)
\end{aligned}$$

Therefore, the MML score difference between reference and candidate models are

$$I(\mathcal{M}'|\mathcal{D}, G') - I(\mathcal{M}|\mathcal{D}, G) = \underbrace{I(G') - I(G)}_{\text{Equation 6.16}} + \underbrace{I(\mathcal{D},\theta') - I(\mathcal{D},\theta)}_{\text{Equation 6.17}} \qquad (6.18)$$

6.3.3.2 MML score when an edge (a, b) is to be added to a context specific graph

The addition of candidate edge (a, b) to the context-specific graph G_i of cluster i affects only the corresponding graph structure and its parameters and rest of all are remain unchanged. Therefore the MML score difference between the candidate and reference graphical structures is as followed:

$$\begin{aligned}
I(G') - I(G) &= \underbrace{\log(n!) + \log\binom{|E_{complete}|}{|E_0|} + \sum_{j=1}^{K \text{ and } i\neq j} \binom{|E_{complete}|}{|E_j - E_0|} + \log\binom{|E_{complete}|}{|E_i - E_0| + 1}}_{\text{Candidate model}} \\
&\quad \underbrace{-\log(n!) - \log\binom{|E_{complete}|}{|E_0|} - \sum_{i=1}^{K \text{ and } j\neq i} \binom{|E_{complete}|}{|E_j - E_0|} - \log\binom{|E_{complete}|}{|E_i - E_0|}}_{\text{Reference model}} \\
&= \log \frac{1}{|E_i - E_0| + 1} \qquad (6.19)
\end{aligned}$$

As the edge (a, b) is added to the G_i and no change in reference and candidate model of the rest of all context specific graphical structures, we encode the data and parameters of G_i is as followed:

$$\begin{aligned}
I(\mathcal{D}, \theta') - I(\mathcal{D}, \theta) \;=\;& I(\theta'_i) - I(\theta_i) + I(\mathcal{D}_i|\theta_i, G'_i) - I(\mathcal{D}_i|\theta_i, G_i) \\
& -\frac{1}{2}\left[\log \frac{|\Sigma_i^{C_{ab}}| \cdot |\Sigma_i^{S_{ab}}|}{|\Sigma_i^{C_{ab}\cap C_a}| \cdot |\Sigma_i^{C_{ab}\cap C_b}|}\right] - \sum_{j=1}^{n_i} \Bigg(\mathcal{L}(D_{ij}^{C_{ab}}|\theta_i^{C_{ab}}) \\
& +\mathcal{L}(D_{ij}^{S_{ab}}|\theta_i^{S_{ab}}) - \mathcal{L}(D_{ij}^{C_{ab}\cap C_a}|\theta_i^{C_{ab}\cap C_a}) - \mathcal{L}(D_{ij}^{C_{ab}\cap C_b}|\theta_i^{C_{ab}\cap C_b}) \\
+\log k \Bigg) \qquad & \qquad (6.20)
\end{aligned}$$

Therefore, the MML score difference between the reference and candidate models are

$$I(\mathcal{M}'|\mathcal{D}, G') - I(\mathcal{M}|\mathcal{D}, G) = \underbrace{I(G') - I(G)}_{\text{Equation 6.19}} + \underbrace{I(\mathcal{D}, \theta') - I(\mathcal{D}, \theta)}_{\text{Equation 6.20}} \qquad (6.21)$$

6.3.4 The forward-selection Algorithm

Our algorithm to discover the context-specific GGM structures along with shared edges is presented in the Algorithm 6.3. In the algorithm, at step 1, we initialize all graphs as null graphs. We then compute the MML score to encode the parameters and data of all edges at step 2. In next step, we then add the best edges incrementally either in both super and all context-specific graphical structures or one of the context-specific graphical model structure based on the MML. After adding the best edges, we update the candidate edge lists E^c by maintaining the chordality of all context-specific graph structures and MML scores of candidate edges (discussed in Section 6.2.1 in details). We call our MML based context-specific GGMs discovery algorithm as ***t**he context-specific **G**aussian graphical models **D**iscovery using **MML*** or *tGDM.* In Gaussian graphical models step of the PaGIAM algorithm (algorithm 6.1), we use tGDM algorithm to discover context specific graphical models instead of ContChordalysis-MML. We call the updated

Algorithm 6.3 tGDM

```
Initialize all $G$ to be empty graphs and $E^c \leftarrow E$.
$t \leftarrow 0$
for each edge $(a, b) \in E^c$ do
    $MML^{*(0)}_{(a,b)} \leftarrow I(\mathcal{D}, \theta') - I(\mathcal{D}, \theta)$ $\triangleright$ Equation 6.17
    for $i = 1$ to $K$ do
        $MML^{*(i)}_{(a,b)} \leftarrow I(\mathcal{D}, \theta') - I(\mathcal{D}, \theta)$ $\triangleright$ Equation 6.20
    end for
end for
repeat
    $MML^{(0)}_{(a,b)} \leftarrow \big(argmin_{(x,y) \in E_c} MML^{*(0)}_{(x,y)}\big) + I(G') - I(G)$ $\triangleright$ Equation 6.16
    for $i = 1$ to $K$ do
        $MML^{(i)}_{(a,b)} \leftarrow \big(argmin_{(x,y) \in E_c} MML^{*(i)}_{(x,y)}\big) + I(G') - I(G)$ $\triangleright$ Equation 6.19
    end for
    $x \leftarrow \arg\min_k MML^{(i)}_{(a,b)}$
    if $MML^{x}_{(a,b)} < 0$ then
        Add $(a, b)$ to graph $G_x$
        if x=0 then
            Add $(a, b)$ to all of the context-specific graphs
        end if
        Readjust $E^c$ $\triangleright$ See section 6.2.1
        Recompute $ML_{(x,y)}$ $\triangleright$ See section 6.2.1
        $t \leftarrow t + 1$
    end if
until $MML^{x}_{(a,b)} \geq 0$ or $t = \frac{n(n-1)}{2}$
```

PaGIAM algorithm as the PaGIAM-tGDM algorithm.

6.4 Evaluation Framework

We compare the performance of our method: PaGIAM-HGDM with strong baselines on synthetic data and real cancer data.

6.4.1 Synthetic data

6.4.1.1 Parameters for synthetic data

We generate synthetic multi-dimensional dataset based on the mixture of Gaussian. We cover a wide range of datasets with different properties by changing different aspects, as follows:

- V: the number of variables, ranges in $\{10, 100, 1000, 5000 \text{ and } 10000\}$.
- n: the number of samples, ranges in$\{100, 1000, 10000, 50000\}$.
- K: the number of clusters, ranges $\{1, 2, 3, 4, 5\}$.
- $|C|$: maximal clique size for graphical structures in the mixture model. According to (Barabási & Albert, 2002), in real-world networks, every new node is born with some edge connections with existing nodes. It produces the connected graph. Therefore, the minimal clique size would be at least 2. We consider that $|C|$ varies from 2 to 6.
- α: Controlling the spread of sampled mixing coefficients in the mixture models. There are two possibilities of mixing coefficients: (a) all clusters having equal frequencies or (b) some clusters having non-equal frequencies. We assume that the mixing coefficients are Dirichlet distributed, with concentration parameter α which ranges in {100, 10, 1, 0.1}. When $\alpha = 100$, approximately all coefficient are equal. Whereas, the frequencies (i.e., mixing coefficient) tend to be different when $\alpha = 0.1$.

- δ: Controlling the statistical associations between random variables. The statistical association between two nodes ranges in between 1 to -1 to express the degree of associations. As it approaches zero, two variables are not statistically associated. Being closer to either 1 or -1 shows a the stronger association between variables. In the experiment, we consider a parameter δ which inversely controls the statistical association between the random variable which ranges in $\{1, 5, 10, 25, 50, 100, 250, 500\}$. We refer to this parameter as Inverse correlation parameter.

To assess the performance of our methods with the baselines, we vary each of parameters as mentioned earlier, in turn, having set base configuration to V = 1000, n = 10000, k = 3, $|C|$ = 3, α = 100 and δ = 50. Moreover, we also assess the performance by varying the number of samples as mentioned while setting the base configuration to V=10000, k = 3, $|C|$ = 3, α = 100 and δ = 50.

6.4.1.2 Graph structure generation

For each experimental setup, we first generate the graph structures and then the dataset. To generate the graph structures, we maintain the real-world networks properties (Clauset et al., 2007): (a) many small nodes are connected with few hubs, known as the power-law property, (b) The average path between two nodes is short and (c) new nodes prefer to attach to well-connected nodes over less-well connected nodes, known as the preferential attachment property.

Barabási and Albert (2002) proposed a model to generate scale-free graphs having the above mentioned properties. We use Barabási-Albert (BA) method to generate the graph structures with properties of real-world networks. This model facilitates us by controlling the number of nodes $|V|$, and the maximal clique size $|C|$ which controls the edge density of the graph.

To generate the graph structures for the synthetic data, we follow the following steps:

(a) First we generate K number of context-specific graphs using Barabási-Albert (BA) method.

(b) We then identify the super graph structure $G_0 = \{V, E_0\}$ by $E_0 = \bigcap_{i=1}^{k} E_i$.

Moreover, as we use decomposable models to discover graphical structures, we add an additional condition that both generated super and context-specific graphs are chordal. Moreover, if the identified super graph G_0 is not chordal, we add edges to make it chordal. We then added these new edges of super graph to all context-specific graphs.. We use candidate edge selection process of ContChordalysis algorithm to maintain the chordality of generated graphs.

6.4.1.3 Synthetic Gaussian data generation

Having the graph structure, we generate the context-specific precision matrix of G_i using following equation

$$\Sigma_i^{-1} \sim \begin{cases} (1/\delta) \cdot 1/d \cdot adj_i(x,y) & \text{if node } x \neq \text{node } y \text{ and } adj_i(x,y) = 1 \\ 1 \cdot adj_i(x,y) & \text{if node } x = \text{node } y \text{ and } adj_i(x,y) = 1 \\ 0 & adj_i(x,y) \neq 1 \end{cases}$$

where adj_i is the adjacency matrix of a component i. Finally, we generate data using $\mathcal{D} = \bigcup_{i=1}^{k} \left\{\mathcal{D}_i \sim \mathcal{N}_d(0, \Sigma_i)\right\}$, where the number of samples of cluster i would be $n_i = n{\cdot}\gamma_i$ and $\gamma_i \sim \mathcal{D}ir(\alpha)$.

6.4.2 Real data

We use two gene expression datasets to evaluate our methods: Breast cancer and Glioblastoma tumour data.

6.4.2.1 Breast cancer

Breast cancer is a hormone related cancer (Ma & Michailidis, 2016) and it has two major subtypes:

- Estrogen-receptor-positive (ER+). It is estimated that around 80% of all breast cancer are ER+. Survival rate of this cancer is better than ER-. It responds to

Table 6.1: Presence (cross marked) of genes and their pathways in each subtype of glioblastoma tumour cells

Gene-pair	Subtype-1	Subtype-2	Subtype-3	Subtype-4	References
PIK3IP-CDKN1A	X		X		(Verhaak et al., 2010)
PIK3IP-CDKN2C		X	X	X	(Brennan et al., 2013)
PIK3IP-CDKN3	X				(Brennan et al., 2013)
AKTIP-CDKN1A			X		(McLendon et al., 2008)
AKTIP-CDKN2C	X			X	(Brennan et al., 2013)
AKTIP-CDKN3		X		X	(Verhaak et al., 2010)
AKTIP-CCND2			X		(Mirzaa et al., 2014)
IDH1-FGFR3	X	X	X	X	(Verhaak et al., 2010)
IDH1-CCND2		X			(McLendon et al., 2008)
IDH1-CDKN2C	X				(Narita et al., 2002)

hormone therapy.

- Estrogen-receptor-negative (ER-). Its survival rate is poorer. Due to the absence of estrogen receptor hormone, it does not respond to hormone therapy.

Presence of estrogen receptor hormone in breast cancer plays an important role in therapeutic strategies and survival rates. We use a breast cancer dataset containing gene expression of 4512 genes from an Affymatrix HU95aV2 microarray for 148 samples which have been chemically synthesized by Pittman et al. (2004). This breast cancer data is the mixture of estrogen receptivity (ER+/ER-) subtypes, where each tumour sample in the dataset has additional classification tags based its estrogen receptivity (ER+/ER-). Moreover, we consider Pittman et al. (2004)'s chemically discovered gene pairs as the gold standard.

6.4.2.2 Glioblastoma tumour

Verhaak et al. (2010) studied the glioblastoma tumour samples gene expression data with 173 samples and 8271 genes. Verhaak et al. (2010)'s chemically synthesized a dataset containing tumour samples of four disease subtypes. They did not identify whether a gene-pair is present in a subtype or not. Whereas, Narita et al. (2002); McLendon et al. (2008); Brennan et al. (2013); Mirzaa et al. (2014) identified 10 important gene-pairs that causes the appearance of glioblastoma tumour cells. In table 6.1, we report them together with their presence in each disease subtype.

In the Glioblastoma tumour experiment, we investigate the performance of our methods and the baselines to predict the above mentioned 10 prominent gene-pairs from this large data.

6.4.3 Evaluation metrics

We evaluate results using context-specific recall, precision and FMeasure . Recall is the fraction of correctly predicted edges with respect to true edges. Precision is the fraction of correctly predicted edges (i.e. associations) with respect to all predicted edges. FMeasure is the harmonic mean of precision and recall, i.e. FMeasure = $\frac{2\times\text{Precision}\times\text{Recall}}{\text{Precision}+\text{Recall}}$. The average FMeasure is assumed as the accuracy of a method.

The algorithm tGDM generates $k+1$ graphical structures. The corresponding gold standard graph of a predicted graph is unknown in our synthetic data experiments, since potentially each discovered graph can be matched with each of the ground truth clusters. Therefore, we compute False Positive Rate (FPR)[3], False Negative Rate (FNR)[4] and error[5] for the best matched predicted graph of a gold standard graph. The predicted network G having a minimal error with respect to a gold standard G_{gold} is the best matched discovered graph G of the corresponding gold standard G_{gold}.

6.4.4 Baselines for synthetic data experiments

We compare the performance of our MML based scoring approach with two scoring function: AIC (Akaike Information Criterion) (Akaike, 1973) and BIC (Bayesian Information Criterion) (Schwarz, 1978) as variants of both PaGIAM and tGDM.

The AIC and BIC variants of PaGIAM algorithm refer to PaGIAA and PaGIAB respectively and their AIC and BIC scoring functions are as followed:

$$AIC_{PaGIAA} = -2\mathcal{L}(\mathcal{D}|\theta) + 2k \quad \text{and} \quad BIC_{PaGIAB} = -2\mathcal{L}(\mathcal{D}|\theta) + k\log n$$

[3] $FPR = \frac{FP}{TP+FP}$ where TP is the number of the predicted edges present in gold standard and FP is the number of the predicted edges not present in gold standard.

[4] $FNR = \frac{FN}{TN+FN}$ where TN is the number of the predicted conditional independence present in gold standard and FN is the number of the predicted conditional independence not present in gold standard.

[5] $error = FNR + FPR$.

Similarly, the AIC and BIC variants of tGDM algorithms are referred to by tGDA and tGDB respectively, and their AIC and BIC scoring functions are as follows:

$$AIC_{tGDA} = -2\mathcal{L}(\mathcal{D}|\theta) + 2|E| \quad \text{and} \quad BIC_{tGDB} = -2\mathcal{L}(\mathcal{D}|\theta) + |E|\log n$$

In the synthetic data experiments, we consider PaGIAM-tGDA, PaGIAB-tGDM, PaGIAA-tGDM, and PaGIAB-tGDM are the baselines of our MML based methods. We also compare the performance of the PaGIAM-ContChordalysis-MML approach discussed in Section 6.2.

6.4.5 Baselines for real data experiments

In real data experiments, we evaluate our methods: *PaGIAM-tGDM* and *PaGIAM-Contchordalysis-MML*[6] with recent strong baselines: New-SP (*New-Structural-Pursuit*) (Gao et al., 2016) and JSEM (*Joint Structural Estimation Method*)(Ma & Michailidis, 2016). New-SP and JSEM estimate context-specific GGMs with shared edges in the framework of a Gaussian mixture model.

New-SP uses the hard EM algorithm (Dempster et al., 1977) to cluster the data and Danaher et al. (2014) proposed Joint Fused Graphical Lasso method to estimate the context-specific GGMs. Beside, JSEM uses Graphical Lasso (J. Friedman et al., 2008) and Group Lasso (Breheny & Huang, 2009) for inferring the context-specific GGM structures. Both methods use penalized likelihood as objective function to discover the context-specific GGMs.

6.5 Results

We have implemented PaGIAM-tGDM and its variants in Matlab 2016b. New-SP and JSEM are developed by r-packages and available in CRAN. All experiments are run on a desktop with Intel Core i5 3.2GHz CPU and 8GB of RAM.

[6] Except PaGIAM-tGDM and PaGIAM-Contchordalysis-MML, all other baselines of synthetic data experiments do not perform well. For this reason, we do not use these baselines for the real world data.

Moreover, in the experiment, we start with our PaGIAM (Algorithm 6.1) assuming the number of components k is one in the mixture model and then keep on increasing the number of components till MML outlines the best-partitioned data, and super and context-specific graphical models.

6.5.1 Synthetic data

We first compare PaGIAM-tGDM with its variants to discover context-specific GGMs with shared edges on the synthetic data with different experimental setups (discussed in Section 6.4.1).

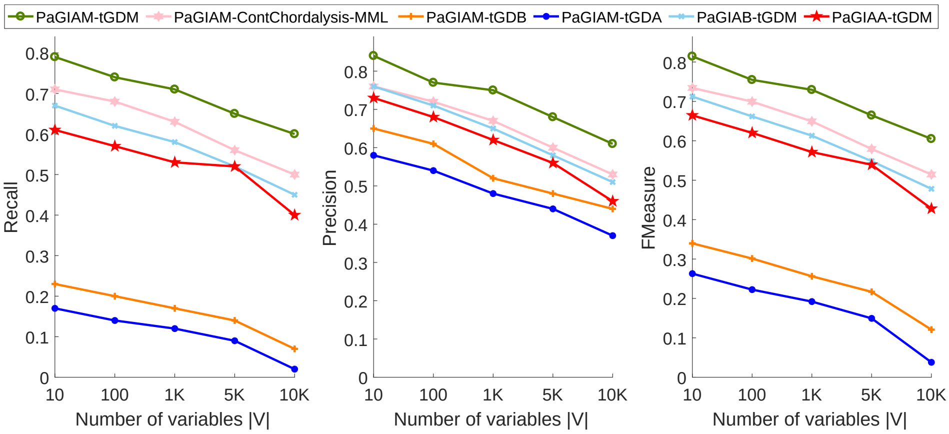


Figure 6.4: Performance of PaGIAM-tGDM and its variants on the synthetic datasets with the different number of variables $|V|$

Varying the number of dimensions $|V|$: In this experimental setup, we change the number of variables (i.e. the number of graph nodes) in 10, 100, 1000, 5000 and 10000. Figure 6.4 shows the recall, precision and FMeasure of the outputs of PaGIAM-tGDM and other baselines. PaGIAM-tGDM outperforms all of the competitive baselines. From Fig 6.4, recall, precision and FMeasure of all methods decreases with the increase in the number of variables $|V|$. However, FMeasure of PaGIAM-tGDM is still higher compared to other baselines in the high dimensional data. Therefore PaGIAM-tGDM discovers the hierarchical Gaussian graphical models from the data with a very large number of variables.

Interestingly, PaGIAM-ContChordalysis-MML performs well with respect to FMeasure and edge detection and outperforms all of the methods except PaGIAM-tGDM. From the evaluation results of PaGIAM-tGDM and PaGIAM-ContChordalysis-MML, we can say that PaGIAM algorithm (Algorithm 6.1) partitions the data more accurately. The accurate clustered data helps tGDM and ContChordalysis-MML to discover the graphical models more precisely. ContChordalysis-MML discovers each context-specific graphical model as an individual entity from clustered data. Whereas, tGDM discovers context-specific graphical models from clustered data along with super graph. Therefore, the results of tGDM is better than ContChordalysis-MML.

Recall, precision and FMeasures of PaGIAB-tGDM and PaGIAA-tGDM are not good as PaGIAM-tGDM. N. Friedman (1998) reported that BIC and AIC do not work well to partition the data and produce many wrong clustered data. Due to the presence of wrong data in each cluster, tGDM used inside PaGIAB and PaGIAA does not able to detect many true context specific edges which affect their recall, precision and FMeasures.

Giraud (2014) point out limitations of BIC and AIC that they do not perform well in large high dimension data. PaGIAM-tGDB and PaGIAM-tGDA use BIC and AIC as the scoring functions to add edges to the candidate graphical models. tGDB and tGDA detect less number of edges compared to other methods. Therefore, they do not perform well as PaGIAM-tGDM does.

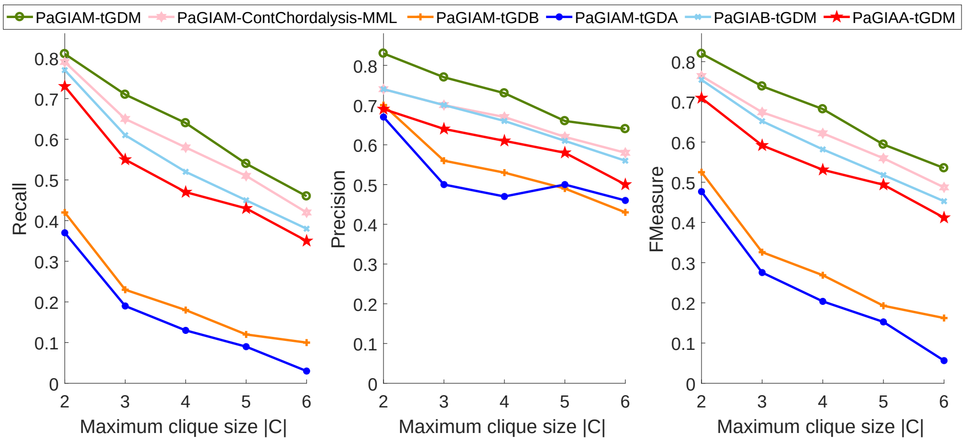


Figure 6.5: Performance of PaGIAM-tGDM and its variants on the synthetic datasets by varying the maximal clique size $|C|$

Varying the maximal clique size $|C|$: In this experiment, we vary maximal clique sizes from 2 to 6. Fig 6.5 shows the recall, precision and FMeasure of the outputs of our method and other baselines. PaGIAM-tGDM outperforms all of the competitive baselines. While the maximal clique size is two, the degree of all vertices is one. All methods except PaGIAM-tGDB and PaGIAM-tGDA, detect most of the true edges and their FMeasure are higher. Over the increment of the maximum size of cliques in the graph, FMeasure of all methods decreases. In this experiment, maximal size of cliques in the graphs inversely affect the FMeasure. However, among all methods, our PaGIAM-tGDM detects many true edges than other methods whatever the size of maximal cliques in the graph and therefore, FMeasure of PaGIAM-tGDM is higher than others.

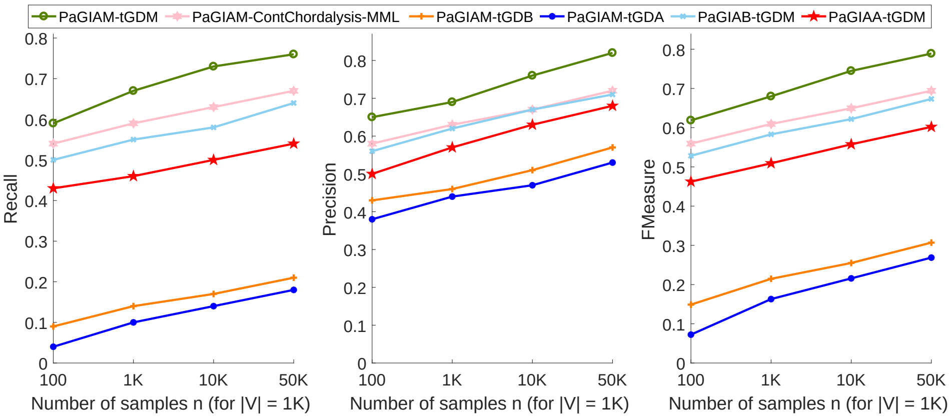


Figure 6.6: Performance of PaGIAM-tGDM and its variants on the synthetic datasets by changing the sample size n, considering the number of variables is 1K

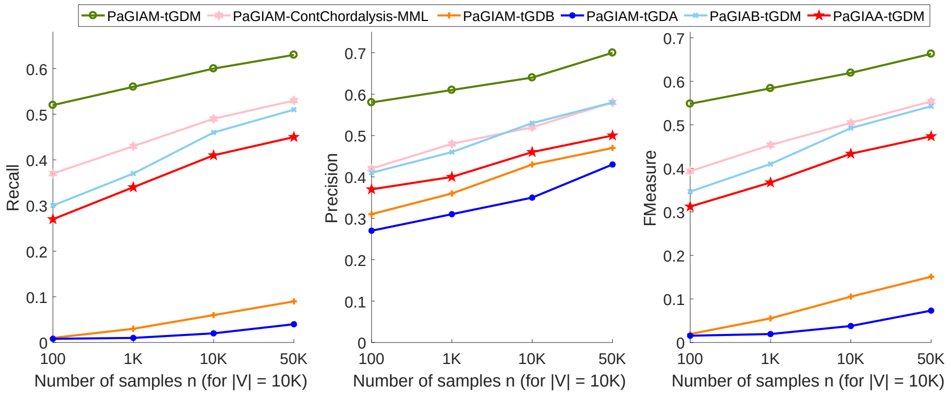


Figure 6.7: Performance of PaGIAM-tGDM and its variants on the synthetic datasets by changing the sample size n, considering the number of variables is 10K

Varying the number of data points n: We carried out two experiments by varying the size of samples where the number of variables are 1000 and 10000. In both experiments, recall, precision and FMeasure results are shown in Figures 6.6 and 6.7 and PaGIAM-tGDM outperforms all other methods. Over the increase of the number of samples, PaGIAM-tGDM detects many true edges accurately and increases the FMeasure. Similar trends also found in other methods, but not as good as PaGIAM-tGDM. Hence, PaGIAM-tGDM can work on any size of multivariate Gaussian distribution data efficiently.

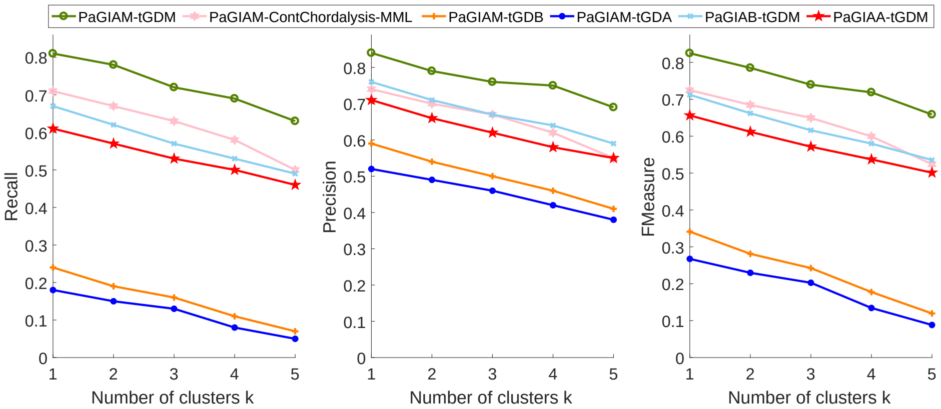


Figure 6.8: Performance of PaGIAM-tGDM and its variants on the synthetic datasets by varying the number of the components k in the mixture model

Varying the number of components k: Figure 6.8 reports the performance of methods with respect to the different number of components (i.e. clusters) in the mixture. As the number of clusters increases, recall, precision and FMeasure of all methods decreases. While the number of clusters increases, the amount of wrongly clustered data also increases which affect the results of all methods. Similarly, PaGIAM-tGDM outperforms all of the competitive baselines.

Varying the concentration parameter α for controlling the mixing coefficient: Figure 6.9 presents the performance of methods by varying the frequency parameter. As the frequency parameter α increases, the randomness of the cluster proportion decreases and tends to uniform. It affects the results of all methods. Our PaGIAM-tGDM outperforms all method which indicates PaGIAM-tGDM can work on any kind of heterogeneous data with different cluster proportion.

Varying the correlation control parameter δ: Correlation expresses the statistical association between random variables which strongly influences covariance matrices.

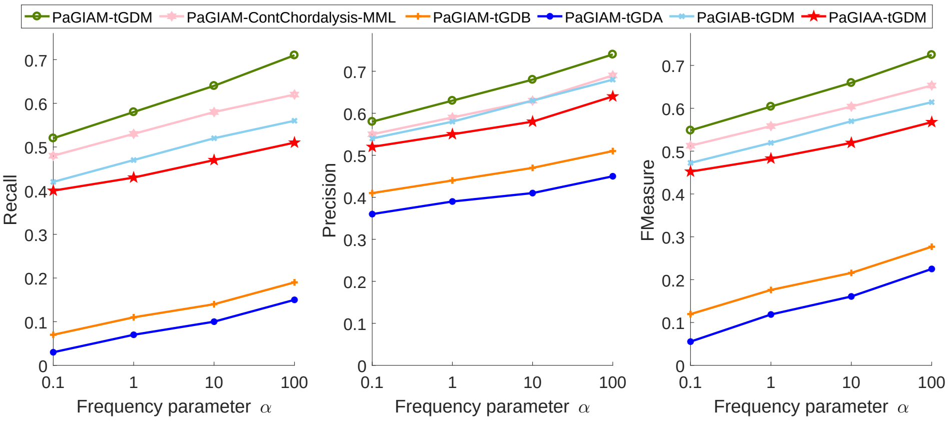


Figure 6.9: Performance of PaGIAM-tGDM and its variants on the synthetic datasets by varying the frequency parameters α

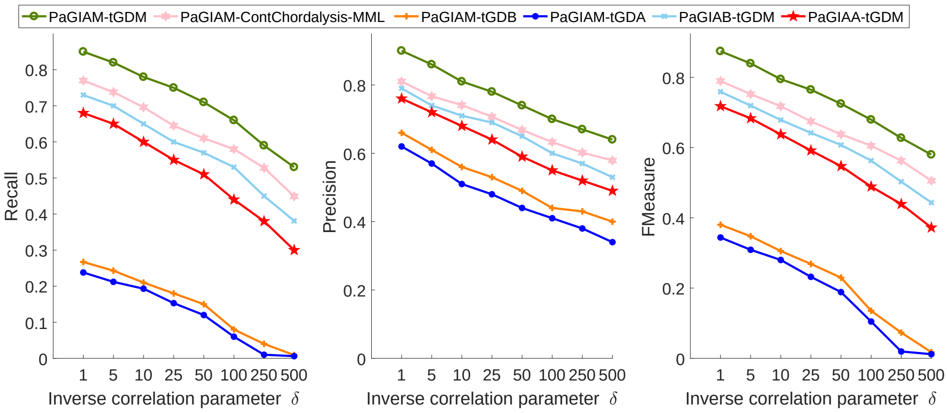


Figure 6.10: Performance of PaGIAM-tGDM and its variants on the synthetic datasets with different value of the correlation control parameter δ

According to Figure 6.10, increase in the value of the inverse correlation parameter δ inversely impacts the covariance matrices and causes the decrease of $FMeasure$. Our PaGIAM-HGDM can detect more than 55% true edges even when very small correlation exists between variables. Whereas, other methods cannot detect even 50% of true edges.

On account of clustering data and discovering context specific graphical model with shared edges accurately, PaGIAM-HGDM outperforms all methods in different experimental setups. Therefore, PaGIAM-tGDM is a statistically efficient method to predict the context-specific independencies from heterogeneous data.

6.5.2 Real world data

We compare our PaGIAM-tGDM and its ContChordalysis-MML variant with strong baselines: New-SP and JSEM to discover the context-specific graphical models from the breast cancer and the Glioblastoma tumour data.

6.5.2.1 Breast cancer data

Figure 6.11 presents recall, precision and Fmeasure of our method versus baselines. We again see the same trend that PaGIAM-tGDM outperforms other methods in terms of the performance measures.

New-SP and JSEM use the Graphical Lasso (GLasso) and penalized likelihood as their objective function to find the optimal context-specific graphical model structures. In GLasso, the regularized parameter is not estimated properly from the data which affected the penalized likelihood and the estimation of context-specific graphical models with their super graph. Hence, New-SP and JSEM statistically do not perform significantly well as PaGIAM-tGDM does.

It is known that many gene-pairs can be responsible for the appearance of the cancer cells in human body. We are interested to know important gene-pairs that have been detected by the methods. According to Pujana et al. (2007), we select 50 important gene-pairs that cause the appearance of cancer cell in human breast tissues. However, among the top 50 important gene pairs, most of them appear in both subtypes and the

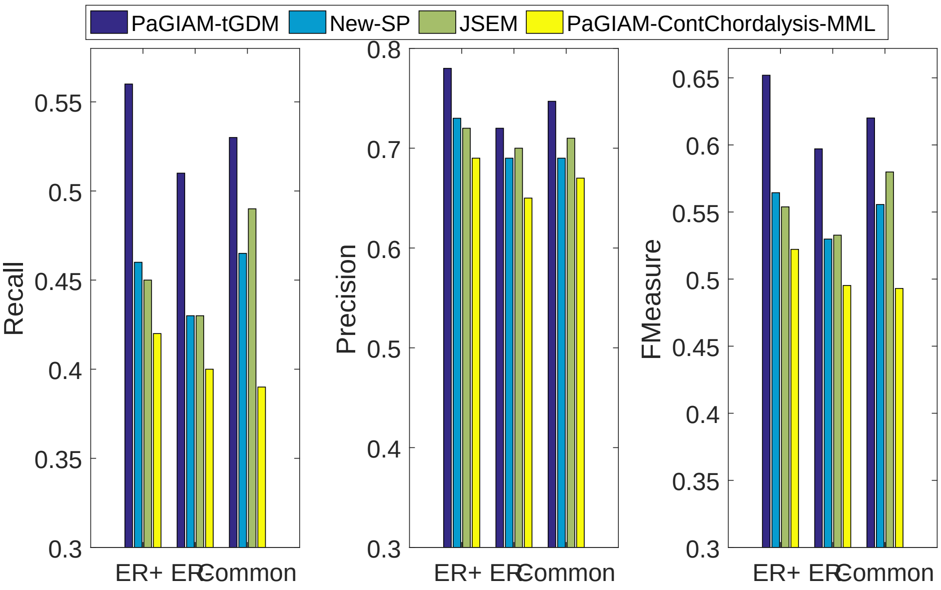


Figure 6.11: Performance of our method with existing baselines on BRCA data

rest appear in different subtypes.

Figure 6.12 shows that PaGIAM-tGDM detected 22 gene-pairs present in both ER+ and ER- subtypes. Whereas, the strongest baseline JSEM detects just 15 gene-pairs in both ER+ and ER- subtypes. New-SP and PaGIAM-ContChordalysis-MML discovers less than 15 gene pairs. Based on this evaluation, our method PaGIAM-tGDM outperforms exiting strong baselines by discovering more true important gene-pairs.

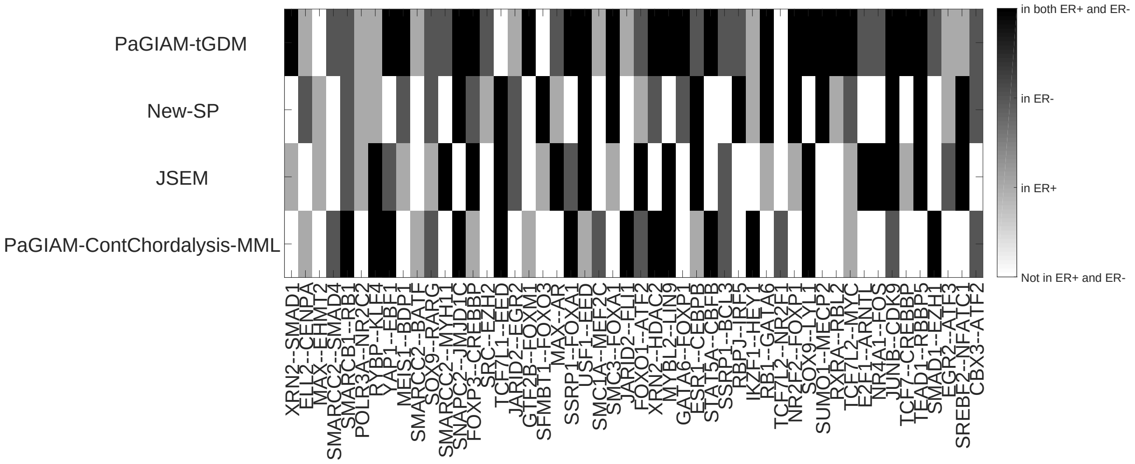


Figure 6.12: Discovery of 50 important breast cancer causing gene-pairs present in ER+ and ER- subtypes by competing methods.

6.5.2.2 Glioblastoma tumour data

We also test PaGIAM-tGDM on (Verhaak et al., 2010)'s Glioblastoma tumour data with New-SP, JSEM, and PaGIAM-ContChordalysis-MML. Due to unavailability of gold standard data, we compare the appearance of 10 gene-pairs in Glioblastoma tumour and its subtypes discovered by different methods (mentioned in Section 6.4.2.2 in details). Figure 6.13 shows the discovery of 10 gene-pairs in different subtypes of Glioblastoma by different methods along with gold standard. PaGIAM-tGDM detects eight gene-pairs including their presence in subtypes accurately. Whereas, New-SP and JSEM detects seven and six gene-pairs accurately. Based on the results of this experiment, PaGIAM-tGDM detects important gene-pairs accurately including their presence in different subtypes of Glioblastoma tumour data.

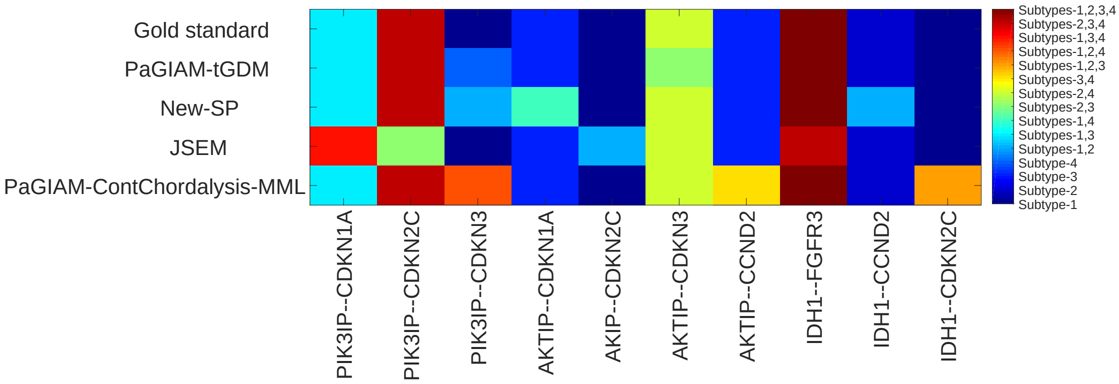


Figure 6.13: Discovery of 10 important gene-pairs of Glioblastoma tumour by competing methods.

Overall, the results of synthetic and real cancer data indicate that PaGIAM-tGDM is more accurate in predicting the context-specific dependencies compare to baselines.

6.6 Conclusion

We have proposed a statistically efficient method to discover the context-specific Gaussian graphical models (GGMs) structure from high dimensional heterogeneous Gaussian data. We introduce PaGIAM-tGDM based on a novel MML-based criterion for clustering and structure discovery of the context-specific GGMs and their shared edges. Our

PaGIAM-tGDM partitions and discovers the context-specific GGM structures from heterogeneous data accurately. We have presented extensive empirical results on synthetic and real-life cancer datasets, and shown that our PaGIAM-tGDM method outperforms strong baselines in terms of the accurate prediction of context-specific associations from the data. We can claim that our PaGIAM-tGDM is one of the strong baselines to predict the context-specific GGMs from heterogeneous data accurately.

Chapter 7

Conclusion

Tumour heterogeneity has been a progressive research area in recent years. While researchers have been addressed many challenges in tumour heterogeneity, many other challenges are yet to be resolved. In this thesis, we addressed a set of crucial and paramount issues that arose from the existing research works to infer the tumour heterogeneity. In the previous chapters of this thesis, we proposed four methods to infer tumour heterogeneity and gene-gene interaction networks from the real cancer data. Here, we conclude this thesis with a summary of the research and its contributions. We then discuss the possible future directions based on the thesis contributions.

7.1 Research summary and contribution

Cancer is a collection of cells (i.e. tumour clones) which are structurally disordered and have a set of mutations in their DNAs. As stated at the introduction chapter, DNA sequencing provides an opportunity to infer tumour clones, and their genomic disorder and mutations from the collection of short reads. Therefore, tumour heterogeneity researches have been focused on inferring the tumour clones from short reads by either clustering or predicting genotypes of mutations. All previous tumour heterogeneity inference methods assumed that (1) type of the mutation of a location is not clone-specific, (2) clonal frequencies of all clones are different and (3) the appearance of a mutation is strongly dependent on position of the adjacent mutation. Whereas, according to Kandoth et al. (2013) and Vogelstein et al. (2013), the type of the mutation at a genomic location varies from clone to clone (i.e. type of the mutation is clone-specific.). Moreover, Ellenbroek and v. Rheenen (2014) observed that in a tumour sample, multiple clones have same clonal frequencies. Ji et al. (2016) stated about the presence of the long-range mutational effect along with position specific effects of adjacent mutations in the tumour sample.

In Chapter 3, we addressed the first two issues (mutations are clone-specific and multiple clones have same frequencies) and presented a method using *Factorial Hidden Markov Model* (FHMM) (Ghahramani & Jordan, 1997). As it is known that, FHMM is a Hidden Markov Model which has multiple chains instead of a single chain. It is also assumed that the observations of FHMM are generated from hidden states of these chains. We were interested to predict genotype of mutations by assuming that the type of a mutation varies from clone to clone. FHMM gave us the facilities of multiple hidden states of a genomic location which might be different from each other. Using FHMM, we selected a separate chain for each of the tumour clone including the normal clone, and each hidden state of a chain represented the genotype of a mutation which was clone-specific. Moreover, we used *exponentiated gradient* (EG) descent algorithm to estimate the clonal frequencies. Our FHMM and EG algorithm based method simultaneously inferred the clone-specific genotype of mutations and clonal frequencies, respectively. We called our method *HetFHMM*.

Evaluation of HetFHMM was a significant challenge for us. None of the existing methods inferred the clone-specific genotype of mutations. Moreover, no method estimated the clonal frequencies and genotypes together from the data. We proposed a new evaluation framework for HetFHMM to evaluate its clone and their frequencies prediction, which was another research contribution to tumour heterogeneity prediction. We tested HetFHMM on synthetic and real cancer data and compare with existing strong baselines: PyClone and PhyloSub. Our method HetFHMM predicted more precise tumour clones, their genomic make-up and frequencies, and outperformed both PyClone and PhyloSub.

In HetFHMM, we made assumption that tumour clones were not concentrated in a specific region and no existence of long-range mutational influences. Moreover, Ellenbroek and v. Rheenen (2014) observed that each clone of a tumour formed and condensed inside a particular region of a tumour. Therefore, the clonal frequencies from different samples are different. Hence, in the Chapter 4, we presented the extension of HetFHMM for multiple samples data and the long-range mutational influences.

The long-range mutational influences were not available in observations of HetFHMM. Whereas, Ji et al. (2016) observed that since the existence of three dimensional structure of DNA, mutations with long-range influences are located close to each other. Moreover, it is also observed that the genes of these mutations interacted with each other and formed a pathway. Therefore, the genes of same pathway skeletoned the list of mutations having long-range influences. Hence, we used known gene-gene interaction networks to find genes and their pathways. From genes, their pathways, and their gene locations, we identified the mutations with long-range dependencies. We used gene-gene interaction network explicitly to find long-range mutational influences. Therefore, we proposed a new transition probabilities for the long-range mutational influences. We proposed three alternative ways to compute the transition probabilities for the long-range mutational influences by using either (1) 1 dimensional (1D) distance between these mutations, or (2) distance between the genes of these mutations, or (3) user defined values. From the experimental results, it was found that using the distance between the genes of these mutations having long-range influences, along with their 1D distance, emHetFHMM performed very well and predicted more accurate tumour clones and their frequencies. Moreover, results of the synthetic and the cancer data experiments showed that emHetFHMM performed very well for the multiple samples and long-ranges mutational influences, and outperformed HetFHMM, PyClone and PhyloSub. The research contributions of HetFHMM and emHetFHMM are four-fold, which are as follows:

- They inferred the clone-specific genotypes of mutations,

- They predicted clonal frequencies along with genotypes,
- They predicted tumour clones with same frequencies, and
- emHetFHMM extensively handled and used the long-range mutational influences.

In emHetFHMM, we used the *Reactome* database to find out the available gene-gene interaction network (i.e. gene-pathways) and *GeneLoc* to find out the location of genes. Cordell (2009) has discovered that the genetic factor function primarily involves multiple other genes through a complex mechanism to play a significant role in the development of cancer cells. Therefore, all interactions between genes were not possible to predict through laboratory experiments. Performing network analysis using large-scale gene expression datasets is an effective way to uncover new and unknown gene-gene interactions (Su et al., 2016). Statistical associations between two genes describes the relationship between these two genes in the form of graphical models. However, large scale gene expression data involves continuous (i.e. Gaussian distributed) valued random variables, where it is critical to uncover the associations among the variables from the large sample data. Typically, there are fewer samples compared to the number of variables, which makes the association discovery challenging, particularly for high-dimensional data. In the Chapter 5, we presented a method call *ContChordalysis* to discover the association between the variables (i.e. genes) in form of Gaussian Graphical Model (GGM), while the number of samples was far less than that of variables. The method ContChordalysis was designed based on the forward selection of greedy approach. Forward selection algorithm started with null graph and added best edges increasing by maximizing the objective function. For developing an efficient objective function, we employed decomposable model in forward selection, which facilitated us to estimate the Maximum Likelihood Estimation from marginal probabilities. First we developed an objective function based on likelihood. We then developed a minimum message length (MML) based objective function. It is well known that MML based objective function enjoys low false discovery rate, suitability for small number of samples, and scalability to large-scale problems involving thousands of variables. Using a MML based objective function, ContChordalysis selected the best edges which minimized the message length. We called our MML based ContChordalysis, *ContChordalysis-MML*. We generated the synthetic data for the random network and the small world network. In the synthetic data experiments, ContChordalysis-MML outperformed strong baselines: TIGER, CLIME, GLasso, r-GLasso and FoBa-gdt. Similar results were also observed in AML and BRCA cancer gene expression data. From the literature, it was revealed that most of the existing methods were suffered from the computation of objective function to discover the graphical models. MML based objective function found the better network structure from the high-dimensional data efficiently which was our one of the major research contributions of this thesis.

In ContChordalysis-MML, we assumed that all observations were generated from a single underlying multivariate distribution. However, the recent studies on Cancer Genome Atlas Network have found that gene expression data can be better described by mixtures where different components harbour different expression pathways (Mukherjee & Roriguez, 2016). Typically there were far less number of samples compared to the number of variables, generated from the mixtures with unknown number of components. The real-life datasets exhibit heterogeneity, which can be better modeled through the use of mixtures of GGMs to let each component exhibit different conditional dependencies among the variables, called as context-specific-dependencies (Meilă & Jordan, 2000). Many researches had been made to discover the context-specific GGMs, but suffered from predicting the many false context-specific associations and discovering less common associations. Whereas, Guo et al. (2011); Rodriguez et al. (2011) observed that all of the context-specific GGMs share most of the edges between themselves. In the Chapter 6, we addressed and resolved the drawbacks of existing methods by proposing a novel method to learn the mixtures of GGMs based on an iterative algorithm, which iterated over the following two steps: (1) clustered the data into distinct clusters and (2) employed the forward selection algorithm for discovering the context-specific GGM structure of each cluster along with the common structure. In the both steps of the iterative algorithm, we used minimum message length (MML) as the objective function. We called our method *PaGIAM-tGDM*. In the synthetic and the cancer data experiments, we observed that PaGIAM-tGDM found statistically better context-specific GGMs. The research contribution of ContChordalysis-MML and PaGIAM-tGDM are as follow:

- Use of MML based objective function, which has theoretical guarantee to work on the high-dimensional data efficiently. Even the experimental results were also showed the better performance,
- Reduction of false edge (i.e. association) discovery rate, and
- Working capability on the cancer gene expression data to reveal the hidden and unknown gene-gene interactions.

Currently, emHetFHMM inferred the more accurate tumour clones and their clonal composition than existing baselines. Then again ContChordalysis-MML and PaGIAM-tGDM discovered the GGMs more efficiently and accurately. Hence we can say that our proposed methods should help the cancer researchers to discover the more accurate tumour clones and to develop for effective patient specific cancer therapy.

7.2 Limitation and future works

In this thesis, we developed four computational methods to discover tumour heterogeneity and Gaussian graphical models. There are several clear avenues for further work and improvement to predict tumour heterogeneity and to discover Gaussian graphical models, that reduce some of the assumption made on our methods.

- In HetFHMM and its multiple sample and long-range mutational influence extension, our method required the maximum number of clones that may exist in the tumour sample to be user defined. In real cancer data, the number of the tumour clones varies from data to data. Therefore, an extension would be to predict the number of clones using non-parametric Bayesian models, such as infinite Factorial Hidden Markov Models (iFHMMs) (Gael, Teh, & Ghahramani, 2009) .

- In HetFHMM and its extension, we used short read counts of total and reference allele, and log ratio of normal-tumour content as observations. Moreover, next generation sequencing offers us the variety of tumour heterogeneity data: gene expression, DNA methylation etc. DNA methylation data helps us to understand the activities of methyl group in DNA structures and how mutations appear in tumour samples in cause of methyl group. Gene expression data helps us to recognize which genes are responsible for mutation appearance. We did not use these type of data, in HetFHMM and emHetFHMM. HetFHMM and emHetFHMM would be extended further by integrating multiple genomic data types as a joint latent variable model.

- We developed minimum message length and decomposable models based method to learn the structure of graphical models to express the structural relationships among random variables, assuming their joint distribution was normal. Although multivariate Gaussian distributions are good approximations for many real world phenomena, we believe that there are real life data which may be better captured by other forms of distributions. Therefore, we are interested to extend our research work to capture a broader class of distributions governing the data.

- As stated earlier, next-generation sequencing provides us multiple genomic datatypes: DNA sequencing, copy number, gene expression and DNA methylation data. In ContChordalysis and PaGIAM-tGDM, we assumed that the dataset was single type, i.e. gene-expression data. Therefore, it is possible to extend the Gaussian graphical models discovery from the multiple data-types high dimensional data as multivariate joint distribution.

- In emHetFHMM experiments, we did not carry out further experiments using gene-gene interactions predicted by ContChordalysis-MML and PaGIAM-tGDM. Therefore, one possible future work is to carry out the experiments on emHetFHMM using gene-gene interaction predicted by our graphical model discovery methods.

- Moreover, our graph discovery methods are designed to predict chordal graph. If these methods are not limited to chordal graphs, we cannot compute the MLE efficiently in closed-form. Hence, we are unable to perform the experiments if the gold standard graph is not chordal. This challenging problem can be regarded as the future extension.

References

Adams, R., Ghahramani, Z., & M.I.Jordan. (2010). Tree-structured stick breaking for hierarchical data. *Proceedings of the 24th Annual Conference on Neural Information Processing Systems 2010*, 19-27.

Akaike, H. (1973). Information theory and an extension of the maximum likelihood principle. *Proceedings of Second International Symposium on Information Theory*, 267-281.

Allisons, L. (2017, January). *Encoding General Graphs.* Retrieved from `http://www.allisons.org/ll/MML/Structured/Graph/`

Altmueller, S., & Haralick, R. M. (2004). Approximating High Dimensional Probability Distributions. *Proceedings of 17th International Conference on Pattern Recognition , 2004 (ICPR'04)*, *2*, 299-302.

Anand, P., Kunnumakara, A. B., Sundaram, C., Harikumar, K. B., Tharakan, S. T., Lai, O. S., ... Aggarwal, B. B. (2008). Cancer is a preventable disease that requires major lifestyle changes. *Springer Pharmaceutical Research*(9), 2097-2116. Retrieved from `https://doi:10.1007/s11095-008-9661-9`

Avagyan, V., Alonso, A. M., & Nogales, F. J. (2017). Improving the graphical lasso estimation for the precision matrix through roots of the sample covariance matrix. *Journal of Computational and Graphical Statistics*, *Online Publication*. Retrieved from `https://doi.org/10.1080/10618600.2017.1340890`

Banerjee, O., Ghaoui, L., & d'Aspremont, A. (2007). Model selection through sparse maximum likelihood estimation for multivariate gaussian or binary data. *Journal of Machine Learning Research*, *9*, 485-516.

Barabási, A.-L., & Albert, R. (2002). Statistical mechanics of complex networks. *Reviews of Modern Physics*, *74*(1), 47-97.

Beeri, C., Fagin, R., Maier, D., & Yannakakis, M. (1983). On the desirability of acyclic database schemes. *Journal of the ACM*, 379-513.

Belloni, A., Chernozhukov, V., & Wang, L. (2012). Square-Root Lasso: Pivotal Recovery of Sparse Signals via Conic Programming. *Biometrika*, *98*, 791-806.

Bhadra, A., & Mallick, B. (2013). Joint high-dimensional bayesian variable and covariance selection with an application to eqtl analysis. *Biometrics*, *69*(2), 447-457.

Boulton, D., & C.S.Wallace. (1969). The information content of a multistate distribution. *Journal of theoretical biology*, *23*, 269-278.

Breheny, P., & Huang, J. (2009). Penalized methods for bi-level variable selection. *Statistics and its Inference*, *2*(3), 369-380.

Brennan, C., et al. (2013). The somatic genomic landscape of gliobalstoma. *Cell*, *155*(2), 462-477.

Brose, M., et al. (2002). Cancer risk estimates for BRCA1 mutation carriers identified in a risk evaluation program. *Journal of the National Cancer Institute*, *94*(18), 1365–1372.

Cai, T., Liu, W., & Luo, X. (2011). A constrained l_1 minimization approach to sparse precision matrix estimation. *Journal of the American Statistical Association*, *106*(494), 594–607. Retrieved from `https://doi.org/10.1198/jasa.2011.tm10155`

Caraco, Y. (1998). Genetic determinants of drug responsiveness and drug interactions. *Journal of Therapeutic Drug Monitoring*, *20*(5), 517-524.

Chiong, K. X., & Moon, H. R. (2017). Estimation of graphical lasso using the $l_{1,2}$ norm. *The Econometrics Journal*. Retrieved from `https://doi.org/10.1111/ectj.12104`

Chow, C., & Liu, C. (1968). Approximating discrete probability distributions with dependence trees. *IEEE Transactions on Information Theory*, *14*(3), 462-467.

Clauset, A., et al. (2007). Power-law distributions in empirical data. *SIAM review*, *51*, 661-703.

Colella, S., Yau, C., Taylor, J. M., Mirza, G., Butler, H., Clouston, P., ... Ragoussis, J. (2007). Quantisnp: an objective bayes hidden-markov model to detect and accurately map copy number variation using snp genotyping data. *Nucleic Acid Research*, *35*(6), 2013-2025. Retrieved from `http://doi: 10.1093/nar/gkm076`

Cordell, H. (2009). Detecting gene–gene interactions that underlie human diseases. *Nature Reviews Genetics*, *10*, 392-404. Retrieved from `http://doi:10.1038/nrg2579`

Danaher, P., Wang, P., & Witten, D. M. (2014). The Joint Graphical Lasso for inverse covariance estimation across multiple classes. *Journal of Royal Statistical Society*, *76*(2), 373-397. Retrieved from `https://doi:10.1111/rssb.12033`

Davoli, T., Uno, H., Wooten, E., & Elledge, S. (2017). Tumor aneuploidy correlates with markers of immune evasion and with reduced response to immunotherapy. *Science*, *355*(6322), 1-33. Retrieved from `http://doi:10.1126/science.aaf8399`

Dempster, A., et al. (1977). Maximum Likelihood from incomplete data via the EM algorithm. *Journal of Royal Statistical Society*, *39*(1), 1-39.

Deshpande, A., et al. (2001). Efficient stepwise selection in decomposable models. *Proceedings of the Seventeenth Conference on Uncertainty in Artificial Intelligence*, 128-135.

Deshwar, A., Vembu, S., & Morris, Q. (2015). Comparing nonparametric bayesian tree priors for clonal reconstruction of tumours. *Proceedings of the Pacific Symposium on Biocomputing 2015*.

Deshwar, A., Vembu, S., Yung, C., Jang, G., Stein, L., & Morris, Q. (2015). *Genome Biology*, *16*(35).

Ding, L., Ley, T. J., Larson, D. E., Miller, C. A., Koboldt, D. C., Welch, J. S., ... Vickery, T. L. (2012). Clonal evolution in relapsed acute myeloid leukaemia revealed by whole-genome sequencing. *Nature*, *481*, 506-512. Retrieved from `http://doi:10.1038/nature10738`

Dowe, D., et al. (1996). MML estimation of the parameters of the spherical Fisher distribution. *Algorithmic Learning Theory*, *1160*, 213-227.

Dwyer, P. (1967). Some applications of matrix derivatives in multivariate analysis. *Journal of the American Statistical Association*, *62*, 607-625.

El-Kebir, M., Oesper, L., Acheson-Field, H., & Raphael, B. (2015). Reconstruction of clonal trees and tumor composition from multi-sample sequencing data. *Oxford Journal of Bioinformatics*, *31*, 62-70. Retrieved from `http://doi:10.1093/bioinformatics/btv261`

Ellenbroek, S. I. J., & v. Rheenen, J. (2014). Imaging hallmarks of cancer in living mice. *Nature Reviews Cancer*, *14*, 406-418. Retrieved from `http://doi:10.1038/nrc3742`

Erdős, P., & Rényi, A. (1959). On random graph i. *Publicationes Mathematicae*, *6*(1), 290-297.

Finch, A., et al. (2006). Salpingo-oophorectomy and the risk of ovarian, fallopian tube, and peritoneal cancers in women with a BRCA1 or BRCA2 mutation. *Journal of the American Medical Association*, *296*(2), 185–192.

Fop, M., Murphy, T. B., & Scrucca, L. (2017). Model-based clustering with sparse covariance matrices. *arXiv*. Retrieved from `arXiv:1711.07748`

Foygel, R., & Drton, M. (2010). Extended Bayesian Information Criteria for Gaussian Graphical Models. *Proceedings of 24th Annual Conference on Neural Information Processing Systems*, *23*, 604-612.

Freese, E. (1959). The difference between spontaneous and base-analogue induced mutations of phage t4. *Proceedings of the National Academy of Sciences of the United States of America*, *45*(4), 622–633.

Friedman, J., et al. (2008). Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, *9*, 432-441.

Friedman, N. (1998). The Bayesian structural EM algorithm. *Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence (UAI)*, 129-138.

Gael, J., Teh, Y., & Ghahramani, Z. (2009). The infinite factorial hidden markov model. *Advances in Neural Informa-*

tion Processing Systems, *21*, 1697-1704. Retrieved from http://papers.nips.cc/paper/3518-the-infinite-factorial-hidden-markov-model.pdf

Gao, C., Zhu, Y., Shen, X., & Pan, W. (2016). Estimation of multiple networks in Gaussian mixture models. *Electronic Journal of Statistics*, *10*, 1133-1154. Retrieved from https://doi:10.1214/16-EJS1135

George, E., & McCulloch, R. (1997). Approaches for bayesian variable selection. *Statistica Sinica*, *7*(2), 339-374. Retrieved from http://www.jstor.org/stable/24306083

Ghahramani, Z., & Jordan, M. I. (1997). Factorial hidden markov models. *Journal of Machine Learning*, *29*(2), 245-273. Retrieved from http://doi:10.1023/A:1007425814087

Giraud, C. (2014). *Introduction to high-dimensional statistics.* Chapman and Hall/CRCs.

Guavain, J.-L., & Lee, C.-H. (1998). Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains. *IEEE Transactions on Speech and Audio Processing*, *2*(2).

Guo, J., et al. (2011). Joint estimation of multiple graphical models. *Biometrika*, *98*(1), 1-15.

Ha, G., Roth, A., Khattra, J., Ho, J., Yap, D., Prentice, L. M., ... Shah, S. P. (2014). Titan: inference of copy number architectures in clonal cell populations from tumour whole-genome sequence data. *Genome Research*, *24*(11), 1881-1893. Retrieved from http://doi:10.1101/gr.180281.114.

Ha, G., Roth, A., Lai, D., Bashashati, A., Ding, J., Goya, R., ... Shah, S. P. (2012). Integrative analysis of genome-wide loss of heterozygosity and monoallelic expression at nucleotide resolution reveals disrupted pathways in triple-negative breast cancer. *Genome Research*, *20*(10), 1995-2007. Retrieved from http://doi:10.1101/gr.137570.112.

Hajirasouliha, I., Mahmoody, A., & Raphael, B. J. (2014). A combinatorial approach for analyzing intra-tumour heterogeneity from high-throughput sequencing data. *Oxford Journal of Bioinformatics*, *30*(12), i78-i86. Retrieved from http://doi:10.1093/bioinformatics/btu284

Hanahan, D., & Weinberg, R. (2011). Hallmarks of cancer: The next generation. *Cell*, *144*(5), 646-674. Retrieved from https://dx.doi.org/10.1016/j.cell.2011.02.013

Harris, R. (2017). Cancer is partly caused by bad luck, study finds. *SHOTS: Health news from National Public Radio.* Retrieved from https://www.npr.org/sections/health-shots/2017/03/23/521219318/cancer-is-partly-

Hess, L., et al. (2006). Pharmacogenomic Predictor of Sensitivity to Preoperative ChemotherapyWith Paclitaxel and Fluorouracil, Doxorubicin, and Cyclophosphamide in Breast Cancer. *Journal of Clinical Oncology*, *24*, 4236-4244.

Hirose, K., Fujisawa, H., & Sese, J. (2017). Robust sparse gaussian graphical modeling. *Journal of Multivariate Analysis*, *161*, 172-190. Retrieved from https://doi.org/10.1016/j.jmva.2017.07.012

Jalali, A., Johnson, C., & Ravikumar, P. (2011). On learning discrete graphical models using greedy methods. *Proceedings of the 24th dvances in Neural Information Processing Systems (NIPS)*, 1935-1943.

Jareborg, N., Birney, E., & Durbin, R. (1999). Comparative analysis of noncoding regions of 77 orthologous mouse and human gene pairs. *Genome Research*, *9*, 815-824. Retrieved from https://doi:10.1101/gr.9.9.815

Ji, X., Dadon, D. B., Powell, B. E., Fan, Z. P., Borges-Rivera, D., Shachar, S., … Young, R. A. (2016). 3d chromosome regulatory landscape of human pluripotent cells. *Cell Stem Cell*, *18*(2), 262-275. Retrieved from http://doi:10.1016/j.stem.2015.11.007

Jianga, Y., Qiu, Y., Minn, A., , & Zhang, N. (2016). Assessing intratumor heterogeneity and tracking longitudinal and spatial clonal evolutionary history by next-generation sequencing. *Proceedings of National*

Academy of Science of USA, *113*(37), E5528-E5537. Retrieved from `https://doi.org/10.1073/pnas.1522203113`

Jiao, W., Vembu, S., Deshwar, A. G., Stein, L., & Morris, Q. (2014). Inferring clonal evolution of tumours from single nucleotide somatic mutations. *BMC Bioinformatics*, *15*(35), 1-16. Retrieved from `http://doi:10.1186/1471-2105-15-35`

Johnson, C., Jalali, A., & Ravikumar, P. (2012). High-dimensional sparse inverse covariance estimation using greedy methods. *Proceedings of the 15th International COnference on Artificial Intelligence and Statistics.*

Joseph, C., Wigley, E. D. A. S. A. S. L. C.-R. F., Boin, F., Fava, A., Thoburn, C., Kinde, I., . . . Rosen, A. (2014). Association of the autoimmune diseases scleroderma with an immunologic response to cancer. *Science*, *343*(6167), 152-157. Retrieved from `http://doi:10.1126/science.1246886.`

Kandoth, C., McLellan, M. D., Vandin, F., Ye, K., Niu, B., Lu, C., . . . Ding, L. (2013). Mutational landscape and significance across 12 major cancer types. *Nature*, *502*, 333-339. Retrieved from `http://doi:10.1038/nature12634`

Koller, D., & Friedman, N. (2009). *Probabilistic graphical models: Principles and techniques - adaptive computation and machine learning.* The MIT Press.

Lam, C., & Fan, J. (2009). Sparsistency and rates of convergence in large covariance matrix estimation. *The Annals of Statistics*, *37*(6B), 4254-4278. Retrieved from `https://doi:10.1214/09-AOS720`

Lauritzen, S. (1996). *Graphical models.* Oxford statistical science series.

Ledoit, O., & Wolf, M. (2004). A well-conditioned estimator for large-dimensional covariance matrices. *Journal of Multivariate Analysis*, *88*(2), 365-411. Retrieved from `https://doi.org/10.1016/S0047-259X(03)00096-4`

Li, A., Liu, Z., Lezon-Geyda, K., Sarkar, S., Lannin, D., Schulz, V., . . . Tuck, D. (2011). Gphmm: an integrated hidden markov model for identification of copy number alteration and loss of heterozygosity in complex tumour samples using whole genome snp arrays. *Nucleic Acids Research*, *39*(12), 4928-4941. Retrieved from `http://doi:10.1093/nar/gkr014`

Li, F., & Zhang, N. (2010). Bayesian variable selection in structured high-dimensional covariate spaceswith applications in genomics. *Journal of the American Statistical Association*, *105*(491), 1202-1214. Retrieved from `http://www.jstor.org/stable/27920144`

Liu, H. (2017). TIGER: A tuning-insensitive approach for optimally estimating Gaussian graphical models. *Electronic Journal of Statistics*, *11*(1), 241-294.

Liu, J., Fujimaki, R., & Ye, J. (2014). Forward-backward greedy algorithm for general convex smooth functions over a cardinality constraint. *Proceedings of the 31st International Conference on International Conference on Machine Learning*, *32*, 503-511.

Liu, W., & Luo, X. (2015). Fast and adaptive sparse precision matrix estimation in high dimensions. *Journal of Multivariate Analysis*, *135*, 153-162. Retrieved from `https://doi.org/10.1016/j.jmva.2014.11.005`

Ma, J., & Michailidis, G. (2016). Joint Structural Estimation of Multiple Graphical Models. *Journal of Machine Learning Research*, *17*, 1-48.

Magnus, J., & Neudecker, H. (1988). *Matrix differential calculus with applications in statistics and econometrics*. Willey, New York.

Malikic, S., McPherson, A., Donmez, N., & Sahinalp, C. (2015). Clonality inference in multiple tumor samples using phylogeny. *Oxford Journal of Bioinformatics*, *31*(9), 1349-1356. Retrieved from `http://doi: 10.1093/bioinformatics/btv003`

McLendon, R., et al. (2008). Comprehensive genomic characterization defines human glioblastoma genes and core pathways. *Nature*, *455*(7216), 1061-1068.

Meilă, M., & Jordan, M. I. (2000). Learning with Mixtures of Trees. *Journal of Machine Learning Research*, *1*, 1-48.

Meinshausen, N., & Buhlmann, P. (2006). High-dimensional graphs and variable selection with the lasso. *The Annals of Statistics*, *34*(3), 1436-1462. Retrieved from `https://doi:10.1214/009053606000000281`

Merlo, L., Pepper, J., Reid, B., & Maley, C. (2006). Cancer as an evolutionary and ecological process. *Nature review cancer*, *6*, 924-935. Retrieved from `https://doi:10.1038/nrc2013`

Miki, Y., et al. (1994). A Strong Candidate for the Breast and Overian Cancer Susceptinility Gene `BRCA_1`. *Science*, *266*, 66-71.

Miller, C., White, B., Dees, N., Griffith, M., Welch, J., Griffith, O., … Ding, L. (2014). Sciclone: Inferring clonal architecture and tracking the spatial and temporal patterns of tumour evolution. *PLOS Computational Biology*, *10*(8). Retrieved from `https://doi.org/10.1371/journal.pcbi.1003665`

Mirzaa, G., et al. (2014). De novo CCND2 mutations leading to stabilization of cyclin D2 cause megalecephaly-polymicrogyria-polydactyly-hydrocephalus syndrome. *Nature Genetics*, *46*(5), 510-514.

Mohammadi, A., & Wit, E. C. (2015). Bayesian structure learning in sparse Gaussian graphical models. *Bayesian Analysis*, *10*, 109-138.

Mohan, K., Chung, M. J.-Y., Han, S., Witten, D., Lee, S.-I., & Fazel, M. (2012). Structured learning of gaussian graphical models. *Advanced Neural Inf Process Systems*, 629-637.

Mukherjee, C., & Roriguez, A. (2016). GPU-powered Shotgun Stochastic Search for Dirichlet process mixtures of Gaussian Graphical Models. *Journal of Computational Graph Statistics*, *25*(3), 762-788.

Narita, Y., et al. (2002). Mutant epidermal growth factor receptor signalling downregulates p27 through activation of the phosphatidylinositol 3-kinase/AKT pathway in glioblastomas. *Cancer research*, *62*(22), 6764-6769.

Navin, N., Kendall, J., Troge, J., Andrews, P., Rodgers, L., McIndoo, J., … Wigler, M. (2014). Tumor evolution inferred by single cell sequencing. *Nature*, *472*(7341), 90-94. Retrieved from `http://doi:10.1038/nature09807`

Nowell, P. (1976). The clonal evolution of tumor cell populations. *Science*, *194*(4260), 23-28. Retrieved from `https:// doi:10.1126/science.959840`

Oliver, J., et al. (1996). Unsupervised learning using MML. *Proceedings of the 13th International Conference Machine Learning*, 364-372.

Olsvik, O., Wahlberg, J., Petterson, B., Uhlén, M., Popovic, T., Wachsmuth, I. K., &

Fields, P. I. (1993). Use of automated sequencing of polymerase chain reaction generated amplicons to identify three types of cholera toxin subunit b in vibrio cholerae 01 strains. *Journal of Clinical Microbiology*, *31*(1), 22-25.

Peterson, C., Stingo, F. C., & Vannucci, M. (2015). Bayesian inference of multiple gaussian graphical models. *Journal of the American Statistical Association*, *110*(509), 159-174. Retrieved from `https://doi.org/10.1080/01621459.2014.896806`

Petitjean, F., et al. (2013). Scaling log-linear analysis to high-dimensional data. *Proceedings of IEEE International Conference on Data Mining*, 597-606.

Petitjean, F., et al. (2014). A Statistically Efficient and Scalable Method for Log-Linear Analysis of High-Dimensional Data. *Proceedings of IEEE International Conference on Data Mining (ICDM)*, 110-119.

Petitjean, F., & Webb, G. (2015). Scaling log-linear analysis to datasets with thousands of variables. , 469-477.

Pittman, J., et al. (2004). Integrated modeling of clinical and gene expression information for personalized prediction ofdisease outcomes. *Proceedings of the National Academy of Sciences of United State of America*, *101*, 8431-8436.

Popic, V., Salari, R., Hajirasouliha, I., Haghighi, D. K., West, R., & Batzoglou, S. (2015). Fast and scalable inference of multi-sample cancer lineages. *Genome Biology*, *16*(91). Retrieved from `http://doi:10.1186/s13059-015-0647-8`

Pujana, M. A., et al. (2007). Network modeling links breast cancer susceptibility and centrosome dysfunction. *Nature genetics*, *39*, 1338-1349.

Qin, Q., et al. (2016). ChiLin: a comprehensive ChIP-seq and DNase-seq quality control and analysis pipeline. *BMC Bioinformatics*, *17*, 1274-1286.

Rodriguez, A., , Lenkoski, A., & Dobra, A. (2011). Sparse covariance estimation in heterogeneous samples. *Electronic Journal of Statistics*, *5*, 981-1014.

Rosenberg, A., & Hirschberg, J. B. (2007). V-measure: A conditional entropy-based external cluster evaluation. *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, 410-420. Retrieved from http://hdl.handle.net/10022/AC:P:21139.

Roth, A., Khattra, J., Yap, D., Wan, A., Laks, E., Biele, J., ... Shah, S. (2014). Pyclone: statistical inference of clonal population structure in cancer. *Nature Methods*, *11*, 396-398. Retrieved from http://doi:10.1038/nmeth.2883

Rothman, A., Bickel, P., Levina, E., & Zhu, J. (2008). Sparse permutation of invariant covariance estimation. *Electronic Journal of Statistics*, *2*, 494-515. Retrieved from https://doi:10.1214/08-EJS176

Sanger, F., & Coulson, A. (1975). A rapid method for determining sequences in dna by primed synthesis with dna polymerase. *Journal of Molecular Biology*, *94*(3), 441-448. Retrieved from http://doi:10.1016/0022-2836(75)90213-2

Satas, G., & Raphael, B. (2017). Tumor phylogeny inference using tree-constrainted importance sampling. *Bioinformatics*, *33*, 1152-1160.

Schwarz, G. (1978). Estimating the dimension of a model. *The Annals of Statistics*, *6*, 461-464.

Sharma, S., Javadekar, S. M., Pandey, M., Srivastava, M., Kumari, R., & Raghavan, S. (2015). Homology and enzymatic requirements of microhomology-dependent alternative end joining. *Cell death and disease*, *19*(6). Retrieved from http://doi:10.1038/cddis.2015.5

Shen, X., Pan, W., & Zhu, Y. (2012). Likelihood-based selection and sharp parameter estimation. *Journal of American Statistical Association*, *107*(497), 223-232. Retrieved from https://doi:10.1080/01621459.2011.645783

Shokirov, B. (2013). Test for normality of the gene expression data. *Methods in Molecular Biology*, *972*, 193-208.

Stingo, F., & Vannucci, M. (2011). Variable selection for discriminant analysis with markov random field priors for the analysis of microarray data. *Oxford Journal of Bioinformatics*, *27*(4), 495-501. Retrieved from http://doi:10.1093/bioinformatics/btq690

Stratton, M. R., Campbell, P. J., & Futreal, P. A. (2009). The cancer genome. *Nature*, *458*, 719-724. Retrieved from http://doi:10.1038/nature07943

Strino, F., Parisi, F., Micsinai, M., & Kluger, Y. (2013). Trap: a tree approach for fingerprinting subclonal tumour composition. *Nuclic Acids research*, *41*(17), 1-15. Retrieved from `http://doi:10.1093/nar/gkt641`

Su, L., Meng, X., Ma, Q., Bai, T., & Liu, G. (2016). Lprp: A gene–gene interaction network construction algorithm and its application in breast cancer data analysis. *Interdisciplinary Sciences: Computational Life Sciences*. Retrieved from `https://doi.org/10.1007/s12539-016-0185-4`

Sun, T., & Zhang, C.-H. (2013). Sparse matrix inversion with scaled lasso. *Journal of Machine Learning Research*, *14*, 3385-3418.

Sun, W., Hao, B., Liu, Y., & Cheng, G. (2016). Simultaneous clustering and estimation of heterogeneous graphical models. Retrieved from `arXiv:1611.09391`

Tao, Q., Huang, X., Wang, S., Xi, X., & Li, L. (2016). Multiple gaussian graphical estimation with jointly sparse penalty. *Signal Processing*, *128*(11), 88-97. Retrieved from `https://doi.org/10.1016/j.sigpro.2016.03.009`

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (methodological). Wiley*, *58*(1), 267-288.

Verhaak, R., et al. (2010). Integrated genomic analysis identifies clinically relevant subtypes of glioblastoma characterized by abnormalities in PDGFRA, IDH1, EGFR and NF1. *Cancer cell*, *17*(1), 98-110.

Vogelstein, B., Papadopoulos, N., Velculescu, V., Zhou, S., Jr, L. D., & Kinzler, K. (2013). Cancer genome landscapes. *Science*, *339*(6127), 1546-1558. Retrieved from `http://doi:10.1126/science.1235122.`

Wallace, C., & Boulton, D. (1968). An information measure for classification. *The Computer journal*, *11*, 185-194.

Wallace, C., & Dowe, D. (1999). Minimum message length and kolmogorov complexity. *The Computer Journal*, *42*(4), 270-283. Retrieved from `https://doi.org/10.1093/comjnl/42.4.270`

Wang, B., Singh, R., & Qi, Y. (2016). A constrained l_1 minimization approach for estimating multiple sparse gaussian or nonparanormal graphical models. *Springer Machine Learning*, *106*(9-10), 1381-1417. Retrieved from `https://doi:10.1007/s10994-017-5635-7`

Wang, C., et al. (2013). Solving log-determinant optimization problems by a newton-cg primal proximal point algorithm. *SIAM Journal on Optimization*, *20*, 2994-3013.

Wang, T., Ren, Z., Ding, Y., Fang, Z., Sun, Z., MacDonald, M. L., ... Chen, W. (2016). Fastggm: An efficient algorithm for the inference of gaussian graphical model in biological networks. *PLOS Computational Biology*, *12*(2), 1-16. Retrieved from `https://doi.org/10.1371/journal.pcbi.1004755`

Watts, D., & Strogatz, S. (1998). Collective dynamics of 'small-world' networks. *Nature*, *393*, 420-442.

Webb, G. (2008). Layered critical values: a powerful direct-adjustment approach to discovering significant patterns. *Journal of Machine Learning*, *71*, 307-323.

West, D. B. (2001). *Introduction to graph theory.* Pearson.

Xia, H., Liu, Y., Wang, M., Feng, H., & Li, A. (2013). A novel hmm for analyzing chromosomal aberrations in heterogeneous tumour samples. *Proceedings of The 7th International Conference on Systems Biology (ISB)*, 92-96. Retrieved from `http://doi:10.1109/ISB.2013.6623800`

Yang, E., & Lozano, A. (2015). Robust gaussian graphical modeling with the trimmed graphical lasso. *Proceeding of the 28th International Conference on Neural Information Processing Systems*, *2*, 2602-2610.

Yau, C. (2013). Oncosnp-seq: a statistical approach for the identification of somatic copy number alternations from next generation sequencing of cancer genomes. *Oxford Journal of Bioinformatics*, *29*(19), 2482-2484. Retrieved from `http://doi:10.1093/bioinformatics/btt416`

Yu, Z., Li, A., & Wang, M. (2017). Climat-het: detecting subclonal copy number alterations and loss of heterozygosity in heterogeneous tumor samples from whole-genome sequencing data. *BMC Medical Genomics*, *10*(15), 1-10. Retrieved from `http://doi:10.1186/s12920-017-0255-4`

Yu, Z., Liu, Y., Shen, Y., Wang, M., , & Li, A. (2014). Climat: accurate detection of copy number alteration and loss of heterozygosity in impure and aneuploid tumor samples using whole-genome sequencing data. *Oxford Journal of Bioinformatics*, *30*(18), 2576-2583. Retrieved from `http://doi:10.1093/bioinformatics/btu346`

Yuan, K., Sakoparnig, T., Markowetz, F., & Beerenwinkel, N. (2015). Bitphylogeny: a probabilistic framework for reconstructing intra-tumor phylogenies. *Genome Biology*, *16*(36), 1-16. Retrieved from `http://doi:10.1186/s13059-015-0592-6`

Yuan, M., & Lin, Y. (2007). Model selection and estimation in the gaussian graphical model. *Biometrika*, *94*(1), 19-35. Retrieved from `https://doi:10.1093/biomet/asm018`

Zare, H., Wang, J., Hu, A., Weber, K., Smith, J., Nickerson, D., ... Noble, W. (2014). Inferring clonal composition from multiple sections of a breast cancer. *PLoS Computation Biology*, *10*(7). Retrieved from `http://doi:10.1371/journal.pcbi.1003703`