Design of Bayesian Hierarchical Models for Accurate Detection of Somatic Mutations
（高精度な体細胞変異検出のための階層ベイズモデルの設計）

by

Takuya Moriyama
森山 卓也

A Doctor Thesis
博士論文

Submitted to
the Graduate School of the University of Tokyo
on December 6, 2019
in Partial Fulfillment of the Requirements
for the Degree of Doctor of Information Science and Technology
in Computer Science

Thesis Supervisor: Satoru Miyano　宮野 悟
Professor of Computer Science

ABSTRACT

Cancer is driven by genomic alterations. Cells suffer from stimulations, e.g., tobacco, alcohol, ultraviolet, oxidative stress and infections, and genomes in cells iterate the process of genomic alteration and repairment every day. Accumulated genomic alterations that avoided DNA repair cause abnormal functions in cells and eventually leads to cancer. Information on genomic alterations is essential for cancer research and cancer therapy because of the causality from genomic alterations to cancer. For cancer research, researchers infer the evolutionary process of cancer from genomic alterations and search for a novel therapy based on the inferred evolutionary process. For cancer therapy, cancer genome medicine, i.e., cancer therapy for each patient based on individual genomic alteration profile, is now becoming reality due to the lowering cost of next-generation sequence (NGS) technology. Therefore, the development of accurate detection methods of genomic alteration from next-generation sequence data sets is one of the most important problems in the field of cancer genomics.

Postnatal genomic alterations are called somatic mutations. In general, at least one tumor and matched normal sequence data sets are utilized to detect somatic mutations from NGS data sets. Mainly two types of approaches exist for the detection method of somatic mutations: single-tumor-based approach and multiple-tumor-based approach. The single-tumor-based approach detects somatic mutations by using a single-regional tumor and a matched normal sequence data sets, and the multiple-tumor-based approach uses multi-regional tumors and a matched normal sequence data sets. Although NGS data specific properties or biological prior knowledge are reported to be important for performance improvement, these properties or prior knowledge are not sufficiently leveraged in both types of approaches.

For the single-tumor-based approach, Bayesian-hierarchical-model-based methods have been developed to leverage NGS data specific properties. However, these existing Bayesian-hierarchical-model-based methods have focused on the modeling of single NGS data specific property, and they are not designed to incorporate multiple properties simultaneously.

For the multiple-tumor-based approach, existing methods have focused on the statistical modeling of biological prior knowledge, e.g., the mutation sharing assumption or the property of the tumor phylogenetic tree. The design of the existing statistical models for mutation sharing assumption are based on the prior knowledge that accuracy can be improved by applying a lower threshold for mutation call if at least one tumor sample has the somatic mutation. For applying the mutation sharing assumption, it is important to infer whether at least one tumor sample has the somatic mutation with high confidence and it is expected that the number of detected candidates and the specificity of the candidate detection are beneficial for that purpose. However, the existing statistical models leverage the number of detected candidates but not leverage the specificity of candidate detection. Furthermore, existing methods cannot use NGS data specific properties because of the restriction of the statistical modelings. As for existing methods that use the tumor phylogeny, the results of the performance evaluation experiment are contradictory; they are evaluated as poorly performed methods in some reports, but evaluated as excellent methods in other reports. Therefore, it is not well examined whether or not the property of tumor phylogeny is effective for the detection of somatic mutation.

Hence, existing methods cannot sufficiently use NGS data specific properties or biological prior knowledge and we can expect a further improvement of detection accuracy for both types of approaches. In this thesis, we consider manners of leveraging these properties or prior knowledge and propose methods for accurate detection of somatic mutations.

First, for the single-tumor-based approach, we propose a novel somatic mutation calling method named as OHVarfinDer. There have been no enough researches about the construction of the Bayesian hierarchical model to incorporate multiple NGS data specific properties. In this point, our method explicitly integrates multiple Bayesian hierarchical models into one model by partitioning-based model integration. In this model integration approach, we introduce an observed indicator variable for each observed data point, which indicates the corresponding model to generate the data point, and these indicator variables enable the integration of multiple Bayesian hierarchical models. This

approach of model integration is different from the Bayesian model averaging because this approach does not require any weight parameter settings in Bayes factor computation if the weight parameters are equal between numerator and denominator. We evaluated our proposed method based on both simulation data sets and real data sets. For the simulation data sets, our method performs comparably with other existing methods when a single property is available and outperforms existing methods when multiple properties are available. For the real data sets, we utilized TCGA benchmark data sets and our method outperforms existing methods in most cases.

Second, we propose a novel multiple-tumor-based mutation calling method named as MultiMuC. For leveraging the mutation sharing assumption, the existing methods have focused on the number of detected candidates but not incorporated the specificity of detection or NGS data specific properties. For leveraging the specificity of detection, our method introduces two types of latent variables. The first type of variable represents the existence of at least one detected mutation candidate and the second type of variable represents the sufficient number of detected candidates with high confidence. Through introducing these latent variables, our method uses the number of candidates and detection specificity. For leveraging NGS data specific properties, we focus on leveraging the data generation probabilities of the stochastic models in existing mutation calling methods that incorporate such NGS data specific properties. In general, existing mutation calling methods only output Bayes factors or posterior mutation event probabilities and we cannot directly obtain data generation probabilities. We guaranteed that we can obtain the consistent posterior distribution or maximum a posteriori state even when only Bayes factors are available. Based on this idea, we constructed a Bayesian hierarchical model by using the Bayes factors obtained from mutation calling results. Therefore, our proposed method can use NGS data specific properties through leveraging data generation probabilities within existing mutation calling methods. We evaluated the proposed method by a simulation based on real data sets. In this simulation, we set multiple tumor phylogenetic trees and multiple clonal composition rates and generated multiple tumor sequence data sets based on them. The performance evaluation demonstrates that our proposed method can improve the accuracy of existing single-tumor-based mutation calling methods by incorporating the mutation sharing assumption.

Finally, we examine whether or not tumor phylogeny is effective for the detection of somatic mutations. For this purpose, we assume a stochastic model for generating the results of mutation calling. Under this assumption, we evaluate the expected specificity and sensitivity of the tumor-phylogeny-based detection method and the non-tumor-phylogeny-based detection method. We also derived a sufficient condition from which the tumor-phylogeny-based detection method has superior specificity of detection. From these evaluations, we revealed when the tumor phylogeny is effective for the detection of mutations and showed that we may improve the detection accuracy in a particular situation.

論文要旨

癌はゲノムの変異により起きる病気である。細胞は、タバコ、アルコール、紫外線、酸化ストレス、感染症などの刺激を受け、日々ゲノムに変異を蓄積させては、DNA 修復系による修復を繰り返している。DNA 修復を免れ、後天的に蓄積したゲノムの変異はやがては細胞の機能に異常をもたらし、癌を引き起こす。癌はゲノムの変異を原因とする病気であることから、ゲノム変異の情報は、癌の研究や治療において、不可欠な情報である。癌研究においては、体細胞変異の情報を用いて癌の進化の過程を推定し、これをもとに新たな治療方針の模索が進められている。また、癌治療においては、次世代シークエンサー（NGS）技術の発展に伴い、低コストでゲノム情報を取得できるようになったため、NGS データから検出したゲノム変異の情報から患者ごとに治療方針を提案する癌ゲノム医療が現実に推し進められている。そのため、NGS データから高精度にゲノム変異を検出する手法の開発は癌ゲノム分野における重要課題の一つである。

後天的に起きたゲノムの変異は体細胞変異と呼ばれる。通常、体細胞変異を NGS データから検出する際は、腫瘍由来のシークエンスデータと正常組織由来のシークエンスデータがそれぞれ少なくとも一つ以上利用される。体細胞変異を検出する方法としては大きく二つの方法があり、一つ目は腫瘍一検体のシークエンスデータに基づく方法で、二つ目は多検体の腫瘍に基づく方法である。一検体のシークエンスデータに基づく方法では、腫瘍一検体と一つの対応する正常細胞のシークエンスデータが用いられ、多検体の腫瘍に基づく方法では、多検体の腫瘍と一つの対応する正常組織のシークエンスデータが用いられる。体細胞変異検出手法の性能改善には、NGS データ特異的な性質や生物学的な事前知識の適用が重要と報告されているが、単一検体に基づく方法、多検体に基づく方法の両方で十分に活用されていない。

単一検体に基づく方法に関しては、NGS データ特異的な性質を利用するために、階層ベイズモデルを基に検出手法が開発されてきた。しかし、これらの既存手法における階層ベイズモデルにおいては、単一の性質のモデル化に焦点を当てており、複数の性質を同時に考慮する設計は為されていない。

多検体に基づく方法に関しては、体細胞変異が共有される性質や癌の進化系統樹のもつ性質などの、生物学的な事前知識の利用に注目が置かれている。まず、体細胞変異が共有される性質を利用する手法に関しては、少なくとも一つの検体が変異をもつ場合、検出の閾値を下げると精度が改善できるという知見をもとに統計モデルが設計されている。ここで、変異が共有される性質を利用するには、少なくとも一つの検体が変異をもつことを高い確度で判定することが重要であり、検出される候補変異数と、検出の特異度が重要であると考えられる。しかしながら、既存の統計モデルでは変異数のみを利用し、検出の特異度までは考慮されていない。さらには、既存の統計モデルの設計の問題により、NGS データ特異的な性質は利用できない。次に、系統樹の性質を利用する手法に関しては、相反する性能評価の報告が上がっており、一部の論文では性能が悪いとして報告する一方で、他方では性能が高いと報告されている。そのため、系統樹の性質が体細胞変異検出にとって有用な性質かどうかはそもそも十分に考察されていない。

以上のことから、既存手法においては、NGS データ特異的な性質や生物学的な事前知識の適用は十分になされておらず、体細胞変異の検出精度には依然として改善余地があると期待される。本学位論文においては、NGS データ特異的な性質や生物学的な事前知識の適

用方法を考案し、高精度な体細胞変異検出手法を提案する。

まず、一検体腫瘍に基づく方法に関して、体細胞変異の検出を行う手法 OHVarfinDer を提案する。既存研究において、NGS データ特異的な性質を複数同時に統計モデルに加味する階層ベイズモデルの方法に関しては十分な研究がなされていなかった。この点に関し、我々の提案手法では分割に基づくモデル統合方法により、明示的に複数の階層ベイズモデルを一つの階層ベイズモデルとして統合する。この方法では、各観測変数に対し、観測を生成した統計モデルを示す新たな観測変数を導入することで、複数の階層ベイズモデルの統合が可能になる。この統合方法はベイズモデル平均化と異なり、ベイズファクターの計算において、分子と分母で重みパラメータが等しい場合では、事前に重みパラメータなどの設定が不要である。我々は、シミュレーションデータと実データに基づき、提案手法の評価を行った。シミュレーションデータによる評価では、単一の性質が利用できる場合では他の既存手法と同程度の性能を示し、複数の性質が利用可能な場合においては既存手法を上回る性能を示した。実データに関しては、TCGA のベンチマークデータに基づく評価を行い、ほとんどの場合で既存手法を上回る性能を示した。

次に、多検体腫瘍に基づく体細胞変異検出手法 MultiMuC を提案する。変異共有の性質の利用においては、既存手法では検出される変異候補の数に着目しているが、変異検出の特異度や NGS データ特異的な性質は考慮していない。変異検出の特異度を利用するために、提案手法では二種類の潜在変数を導入する。一つ目の潜在変数は少なくとも一つの変異候補が検出されているかを表し、二つ目の潜在変数は変異候補が高い特異度で検出されていて、変異候補数も十分多くある状態を表す。提案手法では、これらの潜在変数の導入によって、検出された変異候補の数と検出特異度の両方を利用する。また、NGS データ特異的な性質を利用するために、そのような性質を加味した変異検出手法の確率モデル内のデータ生成確率の利用に着目した。通常、それらの変異検出手法からは、ベイズファクター、ないしベイズファクターに変換可能な事後確率のみが得られ、データの生成確率は直接利用することはできない。我々は、このようなベイズファクターしか得られない状況においても、事後分布の推定や最大事後確率推定には影響が無いことを示した。このアイデアから、変異検出手法の出力として得られるベイズファクターをもとに階層ベイズモデルを構築した。そのため、提案手法では、既存の変異検出手法内のデータ生成確率を通じて、NGS データ特異的な性質を利用することができる。我々は、実データに基づくシミュレーションにより、提案手法の性能評価を行った。このシミュレーションでは、複数の癌の系統樹構造とクローンの混合比率を用意することで、癌のシークエンスデータを複数生成した。この性能評価によって、我々の手法は、多数の検体において変異が共有されていることを利用し、既存手法の精度をさらに改善可能であることを示した。

最後に、がんの進化系統樹の性質が体細胞変異検出に対して有用かどうかを考察する。この考察では、変異検出の結果を生成する確率モデルに仮定をおいた元で、系統樹を用いて変異検出を行う手法と、系統樹を用いずに変異検出を行う手法の感度と特異度の期待値を評価した。また、系統樹を用いた検出手法の方が高い特異度を示すための十分条件を導出した。これらの評価から、どのような状況下でがんの進化系統樹が変異検出に有用かを明らかにし、特定条件下において変異検出の精度向上に有用たり得ることを示した。

Acknowledgements

First of all, I would like to thank my supervisor, Professor Satoru Miyano, for giving me the best environment for bioinformatics research. I could get pre-training of bioinformatics research through working as academic support in the Miyano laboratory before entering the master course. With these supports, I could accomplish this work. I would like to thank the members of the thesis committee, Associate Professor Tetsuo Shibuya, Professor Reiji Suda, Professor Masami Hagiya, Lecturer Issei Sato, and Professor Koji Tsuda for their constructive comments. I am also grateful to Professor Seiya Imoto and Dr. Rui Yamaguchi. They supported my research activities and provide meaningful statistical insights and discussions. They also cultivated my academic writing and presentation skills. Dr. Yuichi Shiraishi, Kenichi Chiba, and Assistant Professor Shuto Hayashi collaborate with me and provide me helpful advice for my research work. Moreover, I would like to thank all members of the Laboratory of DNA Information Analysis, Laboratory of Sequence Analysis in Human Genome Center, and Division of Health Medical Data Science and Division of Health Medical Computational Science in Health Intelligence Center. Especially, I am thankful to Assistant Professor Takanori Hasegawa, Assistant Professor Zhang Yao-zhong, Lecturer Atsushi Niida, Assistant Professor Kotoe Katayama, Dr. Masanori Kakuta, Dr. Satoshi Ito, Dr. Rika Kasazima, Eigo Shimizu, Hiroko Tanaka, Mitsuhiro Komura, Ayako Tomiyasu, Asako Suzuki. Lastly, I would like to thank my family for supporting and encouraging me.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Overview

More than 10% of people suffer from cancer and dies [6]. This malignant disease is driven by genomic alterations [80]. Genomic alterations are propelled by stimulations, e.g., tobacco, alcohol, ultraviolet, oxidative stress, and infections, and suppressed by the DNA repair process and human cells eventually accumulate genetic alterations with age. In general, the causal genomic alterations for cancer is different between cancer patients [11]. Therefore, genome sequencing and obtaining profiles of genomic alterations are important to recommend the optimal therapy for each patient. The lowering cost of the Next-generation sequence (NGS) thechnology [52] and the development of decision support systems enable clinical sequencing for individual optimal cancer therapy.

Postnatal genomic alterations are termed somatic mutations. Before 2004, we can detect somatic mutations in only limited regions through southern blotting or sanger sequencing [38, 69, 73]. After 2004, NGS technology appeared and enables comprehensive detection of somatic mutations [49]. For example, Illumina sequencer enabled the massive amount of DNA sequencing through the sequence by synthesis (Fig. 1.1). Based on the NGS technology, each somatic mutation is detected by comparing sequence data sets from tumor and matched normal tissues.

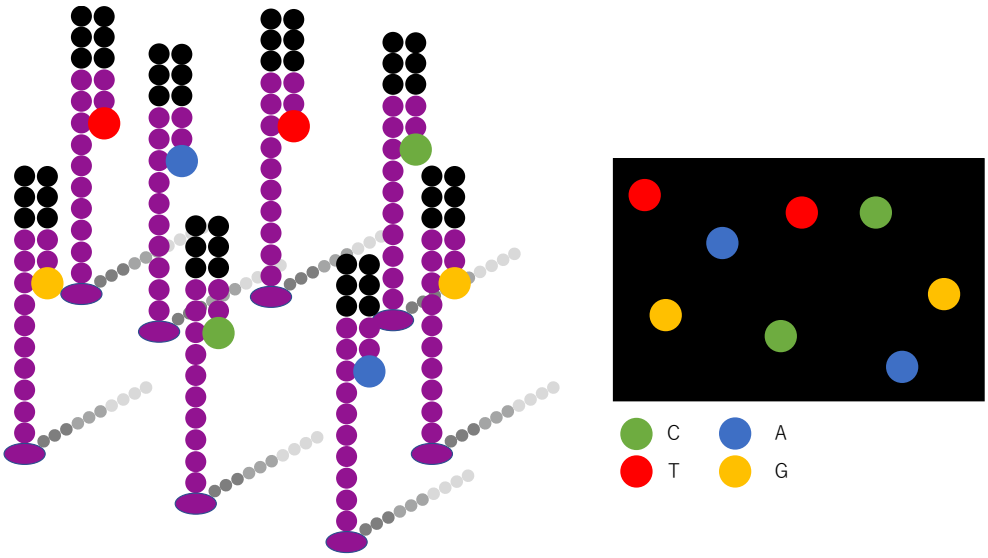


Figure 1.1: Sequence by synthesis conducted in Illumina sequencer.

Due to the causality from somatic mutations to cancer, profiles of somatic

mutations provide essential and beneficial information for cancer research and therapy. The following examples demonstrate the importance of somatic mutations. The first example is intratumor heterogeneity [74, 47, 22]. Tumor cells evolve and diverge by accumulating somatic mutations, and this tumor evolutionary process leads to the intratumor heterogeneity: tumor tissue within a patient is composed of heterogeneous tumor cells with a different set of somatic mutations. Intratumor heterogeneity affects the drug resistance; the existence of small tumor cell population (sub-clonal population) with drug resistance leads to a cancer recurrence after the treatment of molecular target drugs. Therefore, researchers in cancer genomics examine the intratumor heterogeneity and the relations between drug resistance and intratumor heterogeneity. The second example is mutation signatures [2, 60, 63]. Mutation signatures represent patterns of somatic mutations and sequence around them and they are introduced to understand the relationship between stimulation and DNA damage. In order to infer mutation signatures for a better understanding of the relationship between stimulation and DNA damage, profiles of somatic mutations are required. The third example is clinical sequencing. Based on the obtained profile of genomic alterations from sequence data sets, doctor and decision support systems propose medical care for individual patients. A lot of medical organizations examine the optimal decision support system [30] and the protocol of genome sequencing [50].

As supported by these examples, the detection of somatic mutations is a basis for the field of cancer genomics and continuous improvement of detection accuracy is important. Especially, to detect somatic mutations in the sub-clonal tumor cell population for a better proposal of treatment, we require accurate detection methods because fractions of sub-clonal tumor cells can be less than 5%. Therefore, a great deal of effort has been made to improve the performance of mutation call.

For single-tumor-based mutation call, i.e., mutation call from a tumor and a matched normal sequence data set, incorporating sequence-data-specific properties is reported to be important for performance improvement. Nakamura et al. reported sequence-specific error profiles of Illumina sequencers and examined sequencing error-prone sites that are susceptible to sequence errors [57]. In order to exclude sequencing error-prone sites, Shiraishi et al. evaluated the susceptibility to sequence errors from multiple normal sequence data sets and improved performance [72]. Albers et al. focused on the fact that the homopolymer sequence is susceptible to sequence errors. They designed a read generation probability that generates homopolymer sequences in smaller probability and they proposed a Bayesian method to call indels [1]. Usuyama et al. found that each somatic mutation tends to occur on one side of the haplotypes unlike sequence errors and designed a Bayesian statistical model for mutation call to incorporate the haplotypic bias [77]. In our previous work, we focused on the fact that the overlapping part of paired-end reads gives effective information to reduce sequence errors and constructed a Bayesian method to incorporate these overlapping regions [55]. Cibulskis et al. used strand bias of candidate mutation in the pre-filtering step to remove false positives [10].

For multiple-tumor-based mutation call, i.e., mutation call from multiple tumors and a matched normal sequence data sets within a patient, incorporating properties for multi-regional tumor sequence data sets are reported to improve detection performance. Josephidou et al. focused on the assumption of mutation sharing: if we can predict at least one tumor region has the mutation, then we can be more confident to detect a mutation in more tumor regions by lowering the original threshold of detection. Based on this assumption, they developed a

Bayesian method to improve sensitivity when at least one tumor sequence data set has a mutation [32]. Salari et al. focused on the property of tumor phylogenetic tree that patterns of mutated samples are limited, and they retrieve a maximum set of compatible mutations to improve detection performance [68].

In spite of these previous efforts, existing methods have several problems and there is room for further performance improvement. For single-tumor-based approaches, the majority of the existing methods focused on one sequence-data-specific property and constructed a Bayesian hierarchical model based on the property. However, existing Bayesian methods have not proposed the design of Bayesian hierarchical models to incorporate multiple sequence-data-specific properties explicitly. For multiple-tumor-based approaches, existing methods have focused on the multi-regional-specific assumptions, i.e., mutation sharing and the existence of tumor phylogenetic tree, which are applicable only to multi-regional tumor sequence data sets. However, existing methods have not proposed the design of Bayesian hierarchical models to incorporate sequence-data-specific properties that are available even in single-tumor-based approaches. Furthermore, the existing studies do not show clear answers to the following basic questions: whether the property of tumor phylogeny is valuable for the detection of somatic mutations.

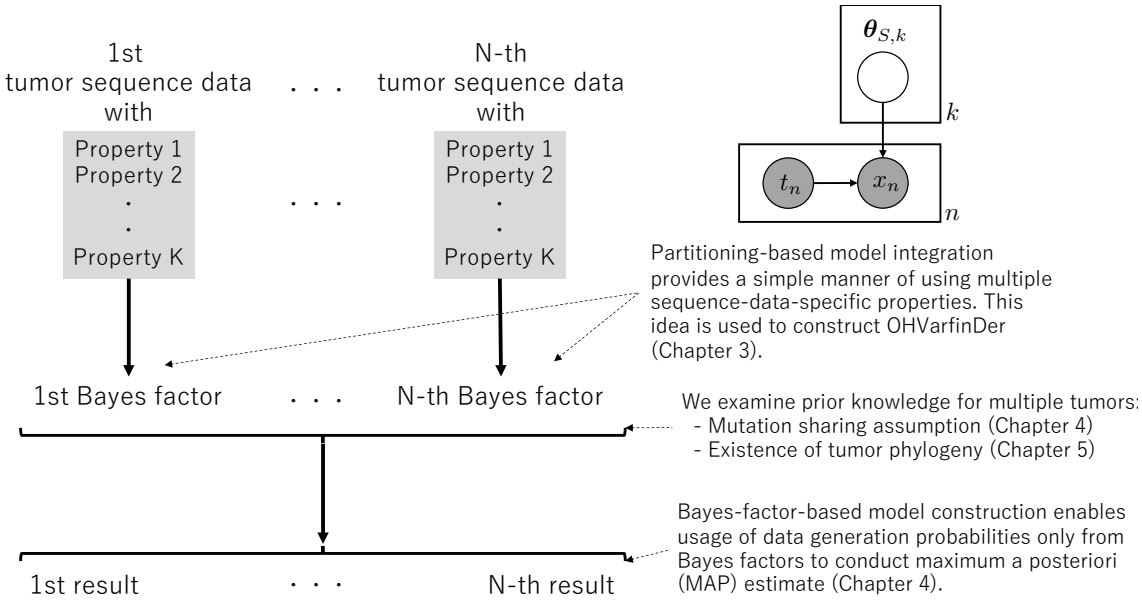


Figure 1.2: Summary of the contributions to the design of Bayesian hierarchical models for detection of somatic mutations.

1.2 Contribution of This Thesis

This thesis contributes to the design of Bayesian hierarchical models to incorporate multiple properties that are available in both settings of single-regional and multi-regional tumor sequence data sets for accurate detection of somatic mutations. The summary of this thesis is represented at Fig. 1.2.

1.2.1 Incorporation of Multiple Sequence-Data-Specific Properties in Single-Regional Tumor Sequence Data Set

In Chapter 3, we propose a design of a Bayesian model to incorporate multiple sequence-data-specific properties in single-regional tumor data set. In this design of a Bayesian model, we integrate multiple generative models into one model by introducing observed parameters for partitioning. Unlike the Bayesian model averaging, this manner of integration has an advantage: our design does not require additional hyperparameter settings for integration when the probabilities of observed parameters for partitioning are the same among numerator and denominator in the Bayes factor. Based on this design, we constructed a mutation calling method termed OHVarfinDer. We evaluated the performance of the proposed method based on pure simulation data sets and TCGA 4 mutation calling benchmark data sets.

1.2.2 Incorporation of the Mutation Sharing Assumption in Multi-Regional Tumor Sequence Data Sets

In Chapter 4, we proposed a Bayesian mutation calling method of MultiMuC for multi-regional tumor sequence data sets. We constructed a Bayesian model of MultiMuC based on the idea of the mutation sharing assumption and Bayes-factor-based model construction.

For using the mutation sharing assumption, we especially focused on the "No-TP(True Positive)" case: even if we could detect mutation candidates in multiple regions, no true mutations exist, unfortunately. The reason for focusing on the No-TP case is that the application of the mutation sharing assumption under the No-TP case can lead to performance degradation. We found that we can decrease the probability of the No-TP case by increasing the specificity of detection or the number of detected candidates. Based on this investigation, we used the specificity of detection and the number of detected candidates to avoid the No-TP case. For Bayes-factor-based model construction, this manner of model construction is helpful for incorporating data generation probabilities from the results of single-tumor-based mutation calling methods. In the practical setting, data generation probabilities are not directly available from the results of other mutation calling methods and we can only use Bayes factors at best. We showed that Bayes factor is sufficient for maximum a posteriori (MAP) estimate; we can obtain consistent MAP state even when all the data generation probabilities are not available but Bayes factors are available. We evaluated that the proposed method can improve the detection performance of the existing single-tumor-based mutation calling methods and outperforms existing multiple-tumor-based mutation calling methods based on a real-data-based simulation study.

1.2.3 Evaluating the Effectiveness of Tumor Phylogenetic Tree

In Chapter 5, we considered the effectiveness of tumor phylogeny for multiple-tumor-based mutation calling. Under setting several assumptions, we evaluate the performance of a tumor-phylogeny-based mutation calling method and a non-tumor-phylogeny-based mutation calling method.

For the problem setting, we assume two mutation profiles for the same multiple tumor samples within a patient are given from distinct genomic regions: reliable profile and unreliable profile. The purpose is to predict each mutation in a patient (and not to predict each mutation in each tumor region) from the unreliable profile. From the evaluation of the performance of a tumor-phylogeny-based

mutation calling method, we can suggest that the tumor-phylogeny-based mutation calling method can predict each mutation in a patient with high specificity and moderate sensitivity even when the original unreliable profile has lower specificity of prediction. This evaluation suggests that tumor phylogeny is effective for predicting each mutation in a patient in particular situations.

Chapter 2

Preliminaries

2.1 Next-Generation Sequence Data Sets and Mutation Call

For the detection of somatic mutations, a next-generation sequencer (NGS) is used in general. In this section, we will briefly explain a workflow of retrieving a sequence data set and mutation calling, and we use the Illumina's sequencer [52] as an example of a next-generation sequencer.

2.1.1 Workflow of Obtaining a Sequence Data Set by NGS Technology

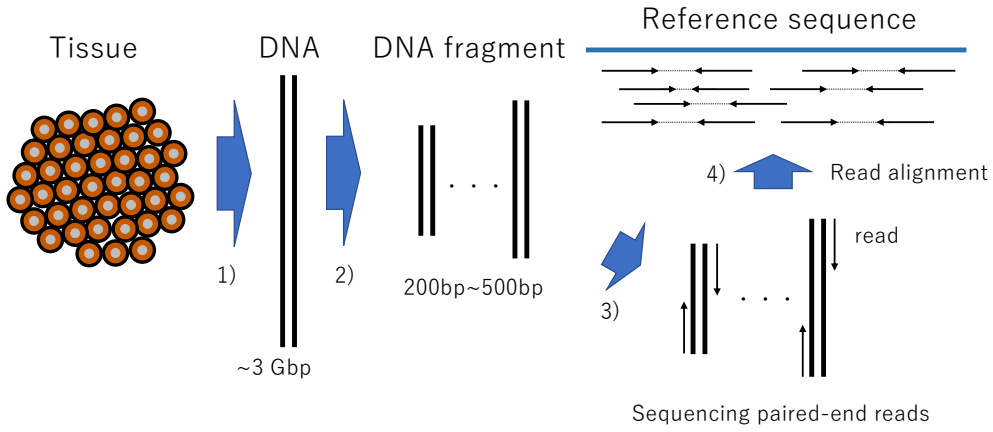


Figure 2.1: General workflow of retrieving sequence data set through NGS.

The workflow of obtaining a sequence data set from human tissue by NGS is summarized in Fig. 2.1.

1) Extracting DNA molecules from a tissue. The total size of the human genome is about 3Gbp.

2) Split DNA molecules randomly into fragments. In general, the length of each fragment is from 200bp to 500bp.

3) Retrieving paired-end reads from each edge of the original fragment. The length of each read is from 100bp to 150bp.

4) Alignment of paired-end reads to the reference sequence.

The above workflow is conducted to a tumor and a matched normal tissue for for mutation calling. For the read alignments of DNA and RNA, a lot of alignment tools are available [46, 34, 14, 35, 79, 42, 25, 82, 43, 45, 44, 56]. Examples of obtained pair of sequence data sets are shown in Figs. 2.2 and 2.3.

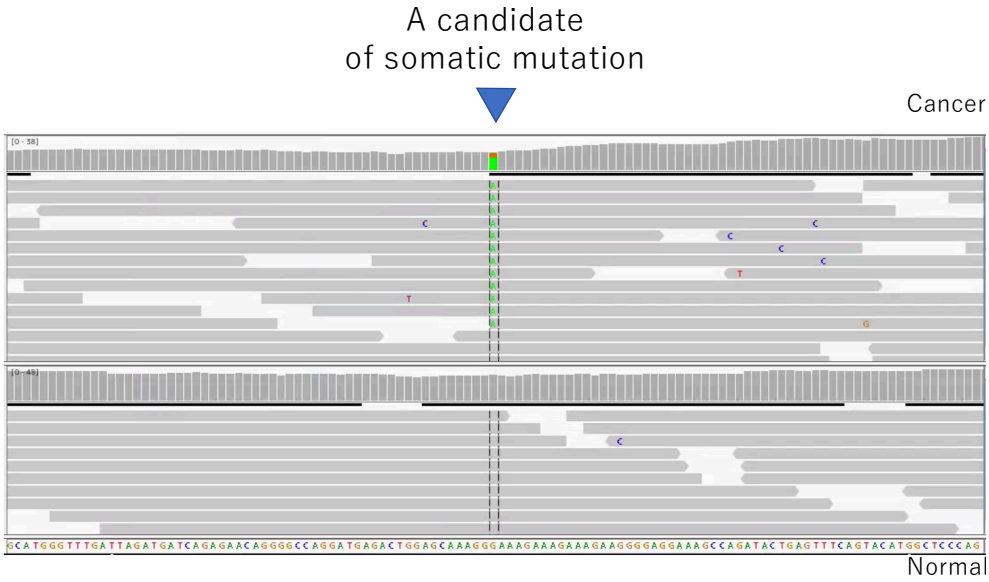


Figure 2.2: Examples of a tumor and a matched normal sequence data sets in a non-erroneous position.

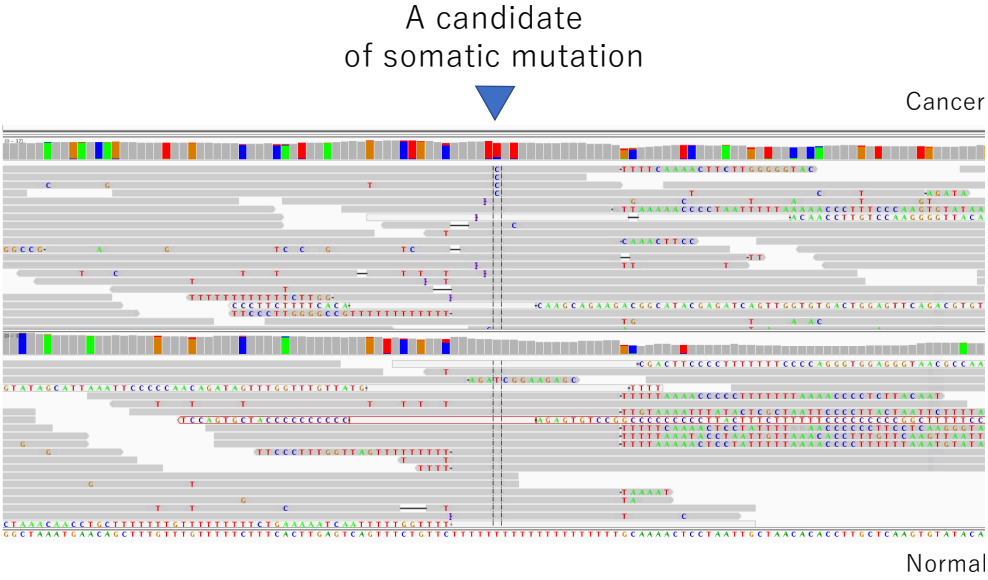


Figure 2.3: Examples of a tumor and a matched normal sequence data sets in an erroneous position.

2.1.2 Mutation Calling from Sequence Data Sets

In single-tumor-based mutation calling approaches, mutation calling is conducted from a pair of sequence data set, i.e., a tumor and matched normal sequence data set. Mutation calling is a two-step process of pre-filtering and classification steps. In the pre-filtering step, mutation candidates are collected by setting thresholds on values, e.g., variant allele frequency, read coverage, the number of variant supporting reads. An example of pre-filtering is shown in Fig. 2.4. After the pre-filtering step, mutation calling methods evaluate and classifies whether each candidate is a true somatic mutation or not. Most of the mutation calling methods output the evaluated scores, e.g., P-values, Bayes factors, posterior event probabilities of mutation.

In multiple-tumor-based approaches, mutation calling is conducted from mul-

tiple tumors and a matched normal sequence data sets. In the general case, mutation calling is also a two-step process of pre-filtering and classification steps. Unlike single-tumor-based approaches, multiple tumor sequence data sets are used in both pre-filtering and classification steps.

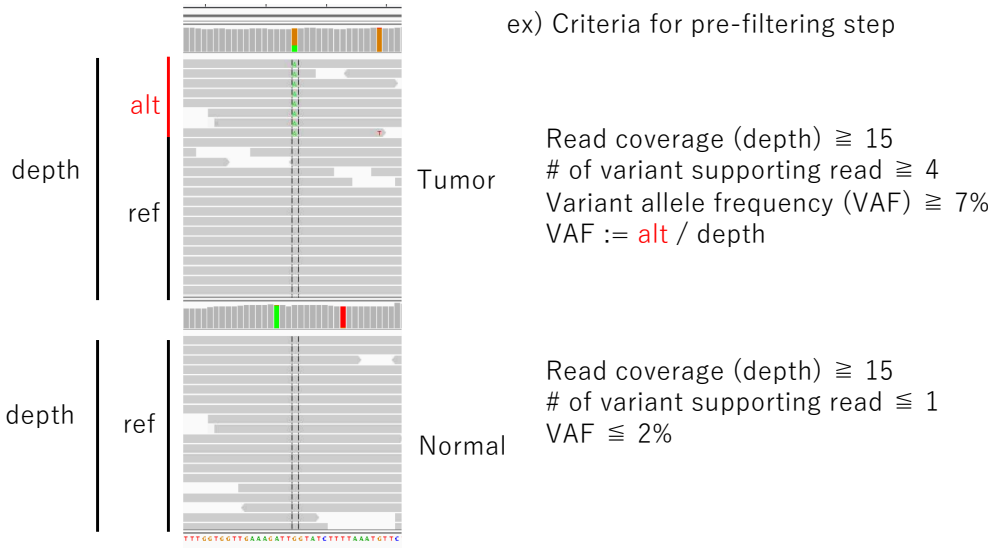


Figure 2.4: An example of criteria for collecting mutation candidates.

2.2 Computational Techniques for Stochastic Models

Through this thesis, we solve the problem of mutation calling based on the constructed stochastic models. By ignoring a strict type definition of each set of variables for simplicity, we express a joint distribution in a constructed stochastic model as $\Pr(D, \Theta|M)$, where D is a set of observed variables representing data set, Θ is a set of unobserved random variables representing parameters, and M represents the model and defines the form of the joint distribution for (D, Θ).

After this section, we introduce a set of computational techniques: variational Bayes and Markov chain Monte Carlo method (MCMC). We apply variational Bayes for evaluating marginal likelihoods in Chapter 3 and apply MCMC for inferring the MAP (maximum a posteriori) state in Chapter 4.

$$\Pr(D|M) = \int \Pr(D, \Theta|M) d\Theta, \tag{2.1}$$

$$\Theta_{\mathrm{MAP}} = \arg\max_{\Theta} \Pr(\Theta|D, M). \tag{2.2}$$

2.3 Variational Bayes

We review variational Bayes which is a method for lower bounding the marginal likelihood and optimizing the lower bound in an iterative manner by assuming a stochastic model [4, 3].

2.3.1 Assumed Stochastic Model

For the stochastic model, we assume that $\boldsymbol{y} := \{y_i\}_{i=1,\cdots,n}$ represents a set of observed variables and $\boldsymbol{x} := \{x_i\}_{i=1,\cdots,n}$ represents a set of hidden variables, where n is the total number of observations. $\boldsymbol{\theta}$ represents a set of parameters on which each data set (x_i, y_i) is dependent. We represent the graphical model for the assumed stochastic model shown in Fig. 2.5.

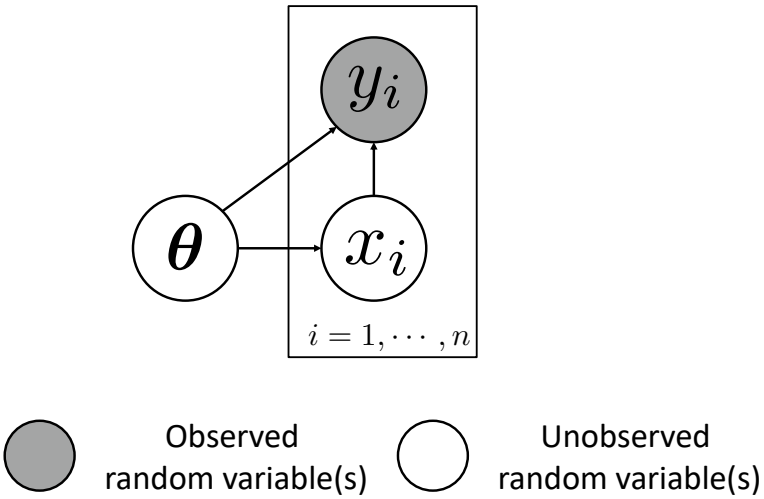


Figure 2.5: The graphical model for the assumed stochastic model.

2.3.2 Lower Bound for the Marginal Likelihood

The marginal likelihood of the assumed model, $\Pr(\boldsymbol{y})$ can be lower bounded by introducing a free distribution of $q(\boldsymbol{x}, \boldsymbol{\theta})$ and applying Jensen's inequality.

$$\begin{aligned}
\ln \Pr(\boldsymbol{y}) &= \ln \int d\boldsymbol{\theta} d\boldsymbol{x} \Pr(\boldsymbol{y}, \boldsymbol{x}, \boldsymbol{\theta}) \\
&= \ln \int d\boldsymbol{\theta} d\boldsymbol{x} \, q(\boldsymbol{x}, \boldsymbol{\theta}) \frac{\Pr(\boldsymbol{y}, \boldsymbol{x}, \boldsymbol{\theta})}{q(\boldsymbol{x}, \boldsymbol{\theta})} \\
&\geq \int d\boldsymbol{\theta} d\boldsymbol{x} \, q(\boldsymbol{x}, \boldsymbol{\theta}) \ln \frac{\Pr(\boldsymbol{y}, \boldsymbol{x}, \boldsymbol{\theta})}{q(\boldsymbol{x}, \boldsymbol{\theta})} := \mathcal{L}(q(\boldsymbol{x}), q(\boldsymbol{\theta})). \qquad (2.3)
\end{aligned}$$

From the equality condition of the Jensen's inequality, we can maximize the lower bound of Eq. (2.3) when $q(\boldsymbol{x}, \boldsymbol{\theta}) = \Pr(\boldsymbol{x}, \boldsymbol{\theta} | \boldsymbol{y})$ and we can obtain the exact value of marginal likelihood. However, obtaining the posterior distributions of $\Pr(\boldsymbol{x}, \boldsymbol{\theta} | \boldsymbol{y})$ requires the value of the marginal likelihood of $\Pr(\boldsymbol{y})$ and hence we do not simplify the original problem. Instead, we simplify the problem by setting a constraint of $q(\boldsymbol{x}, \boldsymbol{\theta}) = q_{\boldsymbol{x}}(\boldsymbol{x}) q_{\boldsymbol{\theta}}(\boldsymbol{\theta})$ and maximize the lower bound of Eq. (2.3) for each distribution of $q_{\boldsymbol{x}}(\boldsymbol{x})$ and $q_{\boldsymbol{\theta}}(\boldsymbol{\theta})$.

We can also see that maximization of the lower bound is equal to the minimization of the Kullback-Leibler (KL) divergence between $q_{\boldsymbol{x}}(\boldsymbol{x})q_{\boldsymbol{\theta}}(\boldsymbol{\theta})$ and $\Pr(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y})$.

$$\begin{aligned}\ln \Pr(\boldsymbol{y}) - \mathcal{L}(q_{\boldsymbol{x}}(\boldsymbol{x}), q_{\boldsymbol{\theta}}(\boldsymbol{\theta})) &= \int d\boldsymbol{\theta} d\boldsymbol{x}\; q_{\boldsymbol{x}}(\boldsymbol{x}) q_{\boldsymbol{\theta}}(\boldsymbol{\theta}) \ln \frac{q_{\boldsymbol{x}}(\boldsymbol{x}) q_{\boldsymbol{\theta}}(\boldsymbol{\theta})}{\Pr(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y})} \\ &= \mathrm{KL}[q_{\boldsymbol{x}}(\boldsymbol{x}) q_{\boldsymbol{\theta}}(\boldsymbol{\theta}) || \Pr(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y})] \geq 0. \end{aligned} \tag{2.4}$$

Variational Bayes is summarized by the iterative procedures that consist of two steps: VBE step and VBM step. We describe $q_{\boldsymbol{x}}^{(t)}(\boldsymbol{x})$ as the distribution for $\boldsymbol{x}$ obtained by the t-th VBE step and $q_{\boldsymbol{\theta}}^{(t)}(\boldsymbol{\theta})$ as the distribution for $\boldsymbol{\theta}$ obtained by the t-th VBM step.

2.3.3 VBE Step: Minimize the KL Divergence w.r.t. $q_{\boldsymbol{x}}(\boldsymbol{x})$

In the VBE step, we minimize the KL divergence w.r.t. $q_{\boldsymbol{x}}(\boldsymbol{x})$. By ignoring the constant values with respect to $q_{\boldsymbol{x}}(\boldsymbol{x})$, we can see the optimal $q_{\boldsymbol{x}}(\boldsymbol{x})$ is $F^{(t)}(\boldsymbol{x})/Z_{\boldsymbol{x}}$.

$$\begin{aligned}
&\mathrm{KL}[q_{\boldsymbol{x}}(\boldsymbol{x}) q_{\boldsymbol{\theta}}^{(t)}(\boldsymbol{\theta}) || \Pr(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y})] \\
&= \int d\boldsymbol{\theta} d\boldsymbol{x}\; q_{\boldsymbol{x}}(\boldsymbol{x}) q_{\boldsymbol{\theta}}^{(t)}(\boldsymbol{\theta}) \ln \frac{q_{\boldsymbol{x}}(\boldsymbol{x}) q_{\boldsymbol{\theta}}^{(t)}(\boldsymbol{\theta})}{\Pr(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y})} \\
&= \int d\boldsymbol{x} q_{\boldsymbol{x}}(\boldsymbol{x}) \left\{ \int d\boldsymbol{\theta}\; q_{\boldsymbol{\theta}}^{(t)}(\boldsymbol{\theta}) (\ln q_{\boldsymbol{x}}(\boldsymbol{x}) + \ln q_{\boldsymbol{\theta}}^{(t)}(\boldsymbol{\theta}) - \ln \Pr(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y})) \right\} \\
&= \int d\boldsymbol{x} q_{\boldsymbol{x}}(\boldsymbol{x}) \left\{ \ln q_{\boldsymbol{x}}(\boldsymbol{x}) - \int d\boldsymbol{\theta}\; q_{\boldsymbol{\theta}}^{(t)}(\boldsymbol{\theta}) \ln \Pr(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y}) \right\} + \text{const} \\
&= \int d\boldsymbol{x} q_{\boldsymbol{x}}(\boldsymbol{x}) \left\{ \ln q_{\boldsymbol{x}}(\boldsymbol{x}) - \ln \left(\exp \left(\int d\boldsymbol{\theta}\; q_{\boldsymbol{\theta}}^{(t)}(\boldsymbol{\theta}) \ln \Pr(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y}) \right) \right) \right\} + \text{const} \\
&= \int d\boldsymbol{x} q_{\boldsymbol{x}}(\boldsymbol{x}) \left\{ \ln q_{\boldsymbol{x}}(\boldsymbol{x}) - \ln \left(F^{(t)}(\boldsymbol{x}) \right) \right\} + \text{const} \\
&= \int d\boldsymbol{x} q_{\boldsymbol{x}}(\boldsymbol{x}) \left\{ \ln q_{\boldsymbol{x}}(\boldsymbol{x}) - \ln \left(\frac{F^{(t)}(\boldsymbol{x})}{Z_{\boldsymbol{x}}} \right) - \ln Z_{\boldsymbol{x}} \right\} + \text{const} \\
&= \mathrm{KL} \left[q(\boldsymbol{x}) \middle|\middle| \frac{F^{(t)}(\boldsymbol{x})}{Z_{\boldsymbol{x}}} \right] + \text{const},
\end{aligned}$$

where

$$\begin{aligned}
F^{(t)}(\boldsymbol{x}) &:= \exp \left(\int d\boldsymbol{\theta}\; q_{\boldsymbol{\theta}}^{(t)}(\boldsymbol{\theta}) \ln \Pr(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y}) \right), \\
Z_{\boldsymbol{x}} &:= \int d\boldsymbol{x} F^{(t)}(\boldsymbol{x}),
\end{aligned}$$

For the form of the distribution $F^{(t)}(\boldsymbol{x})/Z_{\boldsymbol{x}}$, each x_i is independent with the other latent variables.

$$\begin{aligned}
F^{(t)}(\boldsymbol{x}) &= \exp \left(\int d\boldsymbol{\theta}\; q_{\boldsymbol{\theta}}^{(t)}(\boldsymbol{\theta}) \ln \Pr(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y}) \right) \\
&= \exp \left(\int d\boldsymbol{\theta}\; q_{\boldsymbol{\theta}}^{(t)}(\boldsymbol{\theta}) \ln \frac{\Pr(\boldsymbol{x}, \boldsymbol{\theta}, \boldsymbol{y})}{\Pr(\boldsymbol{y})} \right) \\
&\propto \exp \left(\int d\boldsymbol{\theta}\; q_{\boldsymbol{\theta}}^{(t)}(\boldsymbol{\theta}) \ln \Pr(\boldsymbol{x}, \boldsymbol{\theta}, \boldsymbol{y}) \right) \\
&= \exp \left(\int d\boldsymbol{\theta}\; q_{\boldsymbol{\theta}}^{(t)}(\boldsymbol{\theta}) \left(\ln \Pr(\boldsymbol{\theta}) + \sum_i \ln \Pr(x_i, y_i|\boldsymbol{\theta}) \right) \right)
\end{aligned}$$

$$\propto \exp\left(\int d\boldsymbol{\theta}\, q_{\boldsymbol{\theta}}^{(t)}(\boldsymbol{\theta}) \sum_i \ln \Pr(x_i, y_i|\boldsymbol{\theta})\right)$$
$$= \prod_i \exp\left(\int d\boldsymbol{\theta}\, q_{\boldsymbol{\theta}}^{(t)}(\boldsymbol{\theta}) \ln \Pr(x_i, y_i|\boldsymbol{\theta})\right).$$

Therefore, the optimal $q(\boldsymbol{x})$ for the $(t+1)$-th procedure can be obtained as follows.

$$q_{\boldsymbol{x}}^{(t+1)}(\boldsymbol{x}) = \prod_i q_{x_i}^{(t+1)}(x_i),$$
$$q_{x_i}^{(t+1)}(x_i) := \frac{\exp\left(\int d\boldsymbol{\theta}\, q_{\boldsymbol{\theta}}^{(t)}(\boldsymbol{\theta}) \ln \Pr(x_i, y_i|\boldsymbol{\theta})\right)}{Z_{x_i}},$$
$$Z_{x_i} := \int dx_i \exp\left(\int d\boldsymbol{\theta}\, q_{\boldsymbol{\theta}}^{(t)}(\boldsymbol{\theta}) \ln \Pr(x_i, y_i|\boldsymbol{\theta})\right).$$

2.3.4 VBM Step: Minimize the KL Divergence w.r.t. $q_{\boldsymbol{\theta}}(\boldsymbol{\theta})$

In the VBM step, we minimize the KL divergence w.r.t. $q_{\boldsymbol{\theta}}(\boldsymbol{\theta})$. By ignoring the constant values with respect to $q_{\boldsymbol{\theta}}(\boldsymbol{\theta})$, we can see the optimal $q_{\boldsymbol{\theta}}(\boldsymbol{\theta})$ is $G^{(t+1)}(\boldsymbol{\theta})/Z_{\boldsymbol{\theta}}$.

$$\begin{aligned}
&\mathrm{KL}[q_{\boldsymbol{x}}^{(t+1)}(\boldsymbol{x}) q_{\boldsymbol{\theta}}(\boldsymbol{\theta}) || \Pr(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y})] \\
&= \int d\boldsymbol{\theta} d\boldsymbol{x}\, q_{\boldsymbol{x}}^{(t+1)}(\boldsymbol{x}) q_{\boldsymbol{\theta}}(\boldsymbol{\theta}) \ln \frac{q_{\boldsymbol{x}}^{(t+1)}(\boldsymbol{x}) q_{\boldsymbol{\theta}}(\boldsymbol{\theta})}{\Pr(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y})} \\
&= \int q_{\boldsymbol{\theta}}(\boldsymbol{\theta}) d\boldsymbol{\theta} \left(\int d\boldsymbol{x} q_{\boldsymbol{x}}^{(t+1)}(\boldsymbol{x}) \ln \frac{q_{\boldsymbol{x}}^{(t+1)}(\boldsymbol{x}) q_{\boldsymbol{\theta}}(\boldsymbol{\theta})}{\Pr(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y})}\right) \\
&= \int q_{\boldsymbol{\theta}}(\boldsymbol{\theta}) d\boldsymbol{\theta} \left(\ln q_{\boldsymbol{\theta}}(\boldsymbol{\theta}) - \int d\boldsymbol{x} q_{\boldsymbol{x}}^{(t+1)}(\boldsymbol{x}) \ln \Pr(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y})\right) + \text{const} \\
&= \int q_{\boldsymbol{\theta}}(\boldsymbol{\theta}) d\boldsymbol{\theta} \left(\ln q_{\boldsymbol{\theta}}(\boldsymbol{\theta}) - \int d\boldsymbol{x} q_{\boldsymbol{x}}^{(t+1)}(\boldsymbol{x}) (\ln \Pr(\boldsymbol{x}, \boldsymbol{\theta}, \boldsymbol{y}) - \ln \Pr(\boldsymbol{y}))\right) + \text{const} \\
&= \int q_{\boldsymbol{\theta}}(\boldsymbol{\theta}) d\boldsymbol{\theta} \left(\ln q_{\boldsymbol{\theta}}(\boldsymbol{\theta}) - \int d\boldsymbol{x} q_{\boldsymbol{x}}^{(t+1)}(\boldsymbol{x}) (\ln \Pr(\boldsymbol{x}, \boldsymbol{\theta}, \boldsymbol{y}))\right) + \text{const} \\
&= \int q_{\boldsymbol{\theta}}(\boldsymbol{\theta}) d\boldsymbol{\theta} \left(\ln q_{\boldsymbol{\theta}}(\boldsymbol{\theta}) - \ln \Pr(\boldsymbol{\theta}) - \int d\boldsymbol{x} q_{\boldsymbol{x}}^{(t+1)}(\boldsymbol{x}) \ln \Pr(\boldsymbol{x}, \boldsymbol{y}|\boldsymbol{\theta})\right) + \text{const} \\
&= \int q_{\boldsymbol{\theta}}(\boldsymbol{\theta}) d\boldsymbol{\theta} \left(\ln q_{\boldsymbol{\theta}}(\boldsymbol{\theta}) - \ln \Pr(\boldsymbol{\theta}) \exp\left(\int d\boldsymbol{x} q_{\boldsymbol{x}}^{(t+1)}(\boldsymbol{x}) \ln \Pr(\boldsymbol{x}, \boldsymbol{y}|\boldsymbol{\theta})\right)\right) + \text{const} \\
&= \mathrm{KL}\left[q_{\boldsymbol{\theta}}(\boldsymbol{\theta}) \middle|\middle| \frac{G^{(t+1)}(\boldsymbol{\theta})}{Z_{\boldsymbol{\theta}}}\right] + \text{const},
\end{aligned}$$

where

$$G^{(t+1)}(\boldsymbol{\theta}) := \Pr(\boldsymbol{\theta}) \exp\left(\int d\boldsymbol{x} q_{\boldsymbol{x}}^{(t+1)}(\boldsymbol{x}) \ln \Pr(\boldsymbol{x}, \boldsymbol{y}|\boldsymbol{\theta})\right),$$
$$Z_{\boldsymbol{\theta}} := \int d\boldsymbol{\theta} G^{(t+1)}(\boldsymbol{\theta}).$$

From this, the optimal $q_{\boldsymbol{\theta}}(\boldsymbol{\theta})$ for the $(t+1)$-th procedure can be obtained as follows.

$$q_{\boldsymbol{\theta}}^{(t+1)}(\boldsymbol{\theta}) = \frac{G^{(t+1)}(\boldsymbol{\theta})}{Z_{\boldsymbol{\theta}}}.$$

2.3.5 A Conjugate Exponential Model

Here, we introduce the case of conjugate exponential (CE) model in which we can obtain simple solutions in the VBE and VBM steps. We call the assumed stochastic model is CE model if the model satisfies the following conditions. For the case that we assume the different CE models, see [24, 23].

Condition (1).
The complete-data likelihood is in the exponential family:

$$\Pr(x_i, y_i|\boldsymbol{\theta}) = g(\boldsymbol{\theta}) f(x_i, y_i) \exp(\boldsymbol{\phi}(\boldsymbol{\theta})^T \boldsymbol{u}(x_i, y_i)), \tag{2.5}$$

where $\boldsymbol{\phi}(\boldsymbol{\theta})$ is the vector of natural parameters, $\boldsymbol{u}$ and f are the functions that define the exponential family, and $g(\boldsymbol{\theta})$ is a normalizing constant:

$$g(\boldsymbol{\theta})^{-1} = \int dx_i dy_i f(x_i, y_i) \exp(\boldsymbol{\phi}(\boldsymbol{\theta})^T \boldsymbol{u}(x_i, y_i)). \tag{2.6}$$

Condition (2).
The prior distribution of parameters is conjugate to the complete-data likelihood:

$$\Pr(\boldsymbol{\theta}|\eta, \boldsymbol{\nu}) = h(\eta, \boldsymbol{\nu}) g(\boldsymbol{\theta})^{\eta} \exp(\boldsymbol{\phi}(\boldsymbol{\theta})^T \boldsymbol{\nu}), \tag{2.7}$$

where η and $\boldsymbol{\nu}$ are hyperparameters of the prior, and h is a normalizing constant:

$$h(\eta, \boldsymbol{\nu})^{-1} = \int d\boldsymbol{\theta} g(\boldsymbol{\theta})^{\eta} \exp(\boldsymbol{\phi}(\boldsymbol{\theta})^T \boldsymbol{\nu}). \tag{2.8}$$

If the assumed stochastic model is a CE model, the solution at each VBE and VBM step can be obtained as follows.

VBE step:

$$q_{\boldsymbol{x}}^{(t+1)}(\boldsymbol{x}) = \prod_{i=1}^{n} q_{x_i}^{(t+1)}(x_i), \tag{2.9}$$

$$q_{x_i}^{(t+1)}(x_i) \propto f(x_i, y_i) \exp(\overline{\boldsymbol{\phi}}^T \boldsymbol{u}(x_i, y_i)) = \Pr(x_i, y_i|\overline{\boldsymbol{\phi}}), \tag{2.10}$$

VBM step:

$$q_{\boldsymbol{\theta}}^{(t+1)}(\boldsymbol{\theta}) = h(\overline{\eta}, \overline{\boldsymbol{\nu}}) g(\boldsymbol{\theta})^{\overline{\eta}} \exp(\boldsymbol{\phi}(\boldsymbol{\theta})^T \overline{\boldsymbol{\nu}}), \tag{2.11}$$

where

$$\overline{\boldsymbol{\phi}} = \int d\boldsymbol{\theta} q_{\boldsymbol{\theta}}(\boldsymbol{\theta}) \boldsymbol{\phi}(\boldsymbol{\theta}), \tag{2.12}$$

$$\overline{\eta} = \eta + n, \tag{2.13}$$

$$\overline{\boldsymbol{\nu}} = \nu + \sum_{i=1}^{n} \overline{\boldsymbol{u}}(y_i), \tag{2.14}$$

$$\overline{\boldsymbol{u}}(y_i) = \int dx_i q_{x_i}(x_i) \boldsymbol{u}(x_i, y_i). \tag{2.15}$$

2.4 Markov Chain Monte Carlo Methods

We introduce Markov chain Monte Carlo methods (MCMC). The methods enable sampling of random variables from complicated distributions, e.g., posterior distributions in Bayesian statistics.

Let $\pi(\cdot)$ be a probability distribution on a state space $\mathcal{X}$ and $\pi_u(\cdot)$ is a corresponding unnormalized probability density function on $\mathcal{X}$ and $0 < \int_{x\in\mathcal{X}} \pi_u(x)dx < \infty$. (We assume $\mathcal{X}$ is a continuous state space, e.g., $\mathbb{R}^d$, but the other settings that $\mathcal{X}$ is a discrete state space is also possible [59].) In MCMC, we generate a Markov chain, a series of random variables on $\mathcal{X}$, $X^{(0)}, X^{(1)}, \cdots$ following the transition probability $T(x, \cdot)$.

$$X^{(n+1)}|X^{(0)}, \cdots, X^{(n)} \sim T(X^{(n)}, \cdot). \tag{2.16}$$

A theory of MCMC [76, 66] guarantees if a Markov chain has $\pi(\cdot)$ as a stationary distribution and the chain is Harris recurrent, aperiodic, then the distribution of $X^{(n)}$ converges to $\pi(\cdot)$ for any $X^{(0)} \in \mathcal{X}$. A Markov chain has $\pi(\cdot)$ as a stationary distribution if

$$\int_{s\in\mathcal{X}} \pi(ds)T(s, dt) = \pi(dt). \tag{2.17}$$

For Eq. (2.17), reversibility of a Markov chain is a sufficient condition.

Definition 2.4.1. *A Markov chain on a state space $\mathcal{X}$ is reversible with respect to a probability distribution $\pi(\cdot)$ on $\mathcal{X}$, if*

$$\pi(ds)T(s, dt) = \pi(dt)T(t, ds), \quad s, t \in \mathcal{X}. \tag{2.18}$$

Property 2.4.1. *If a Markov chain is reversible with respect to $\pi(\cdot)$, then the chain has $\pi(\cdot)$ as a stationary distribution.*

Proof.

$$\int_{s\in\mathcal{X}} \pi(ds)T(s, dt) = \int_{s\in\mathcal{X}} \pi(dt)T(t, ds) = \pi(dt) \int_{s\in\mathcal{X}} T(t, ds) = \pi(dt).$$

□

2.4.1 The Metropolis-Hastings Algorithm

The Metropolis-Hastings algorithm [51, 28] generates a markov chain in two steps. In the first step, we generate $X^{(n+1)} \in \mathcal{X}$ from the proposal distribution of $Q(X^{(n)}, \cdot)$ of which probability density function is $q(X^{(n)}, \cdot)$. In the second step, we accept the proposal with probability of $\alpha(X^{(n)}, X^{(n+1)})$ and otherwise $X^{(n+1)} = X^{(n)}$, where

$$\begin{aligned} \alpha(s, t) &= \min\left[1, r^*\right], \\ r &= \frac{\pi_u(t)q(t, s)}{\pi_u(s)q(s, t)}, \quad s, t \in \mathcal{X}. \end{aligned}$$

A Markov chain generated by the above Metropolis-Hastings algorithm has $\pi(\cdot)$ as a stationary distribution by reversibility.

Property 2.4.2. *The Metropolis-Hastings algorithm produces a Markov chain $\{X^{(n)}\}$ which is reversible with respect to $\pi(\cdot)$.*

Proof. We need to show

$$\pi(ds)T(s,dt) = \pi(dt)T(t,ds). \tag{2.19}$$

Letting $c := \int_{x\in\mathcal{X}} \pi_u(x)$, the left side of Eq. (2.19) can be written as

$$\begin{aligned}
\pi(ds)T(s,dt) &= \left[c^{-1}\pi_u(s)ds\right]\left[q(s,t)\alpha(s,t)dt\right] \\
&= c^{-1}\pi_u(s)q(s,t)\min\left[1, \frac{\pi_u(t)q(t,s)}{\pi_u(s)q(s,t)}\right] dsdt \\
&= c^{-1}\min\left[\pi_u(s)q(s,t), \pi_u(t)q(t,s)\right] dsdt,
\end{aligned}$$

which is symmetric in s and t. Therefore, the left side is equal to the right side. □

2.4.2 The Gibbs Sampler

We assume that the state space $\mathcal{X}$ is d-dimensional, i.e., $\mathcal{X} = \mathbb{R}^d$, and we write $\boldsymbol{X} = (X_1, \cdots, X_d)$.

In the Gibbs sampler [21] for i-th component, we sample the i-th component from the conditional distribution in which all the other components are observed.

$$X_i^{(n+1)} \sim \Pr(\cdot | X_1^{(n)}, \cdots, X_{i-1}^{(n)}, X_{i+1}^{(n)}, \cdots, X_d^{(n)}),$$

where

$$\Pr(A | X_1^{(n)}, \cdots, X_{i-1}^{(n)}, X_{i+1}^{(n)}, \cdots, X_d^{(n)}) = \frac{\int_{x\in A} \pi_u(X_1^{(n)}, \cdots, X_{i-1}^{(n)}, x, X_{i+1}^{(n)}, \cdots, X_d^{(n)})dx}{\int_{x\in\mathbb{R}} \pi_u(X_1^{(n)}, \cdots, X_{i-1}^{(n)}, x, X_{i+1}^{(n)}, \cdots, X_d^{(n)})dx}, \quad A \subseteq \mathbb{R}. \tag{2.20}$$

Then, we conduct the above sampling for $i = 1, \cdots, d$ repeatedly.

Each Gibbs sampler for i-th component can be seen as a special case of the Metropolis-Hastings algorithm; we set Eq. (2.20) as the proposal distribution and the acceptance probability is always 1.

$$\begin{aligned}
r^* &= \frac{\pi_u(X_1^{(n)}, \cdots, X_{i-1}^{(n)}, X_i^{(n+1)}, X_{i+1}^{(n)}, \cdots, X_d^{(n)})}{\pi_u(X_1^{(n)}, \cdots, X_{i-1}^{(n)}, X_i^{(n)}, X_{i+1}^{(n)}, \cdots, X_d^{(n)})} \cdot \frac{\pi_u(X_1^{(n)}, \cdots, X_{i-1}^{(n)}, X_i^{(n)}, X_{i+1}^{(n)}, \cdots, X_d^{(n)})}{\pi_u(X_1^{(n)}, \cdots, X_{i-1}^{(n)}, X_i^{(n+1)}, X_{i+1}^{(n)}, \cdots, X_d^{(n)})} \\
&= 1.
\end{aligned}$$

2.5 Bayes Factor

Jeffreys developed a methodology for quantifying the evidence in favor of a scientific theory in his paper [31] of which centerpiece was a number, now called the Bayes factor.

The Bayes factor is given by the following equations.

$$\text{Bayes factor} \quad = \quad \frac{\Pr(D|H_1)}{\Pr(D|H_0)},$$

where D denote data set that is generated under one of two hypotheses H_0 and H_1. By considering the posterior odds ratio as follows, the Bayes factor is equal to the posterior odds ratio when both prior hypothesis probabilities are equal.

$$\begin{aligned} \text{posterior odds} \quad &= \quad \frac{\Pr(D|H_1)}{\Pr(D|H_0)} \cdot \frac{\Pr(H_1)}{\Pr(H_0)} \\ &= \quad \text{Bayes factor} \cdot \text{prior odds}. \end{aligned}$$

In the simplest case that there is no free parameter in the probability density $\Pr(D|H_0)$ and $\Pr(D|H_1)$, the Bayes factor is simply given by its likelihoods. In the other case in which there are unknown free parameters in both hypotheses, the Bayes factor is given by evaluating the marginal likelihood of each hypothesis.

$$\text{Bayes factor} \quad = \quad \frac{\int \Pr(D|\boldsymbol{\theta}_1, H_1)\Pr(\boldsymbol{\theta}_1|H_1)d\boldsymbol{\theta}_1}{\int \Pr(D|\boldsymbol{\theta}_0, H_0)\Pr(\boldsymbol{\theta}_0|H_0)d\boldsymbol{\theta}_0},$$

where $\boldsymbol{\theta}_i$ represents the parameters in the hypothesis of H_i.

Kass [33] gave an interpretation of the Bayes factor as listed in the following table.

Table 2.1: Interpretation of the Bayes factor

Bayes factor	Strength of evidence
1 to 3	Not worth more than a bare mention
3 to 20	Positive
20 to 150	Strong
≥ 150	Very strong

2.6 Bayesian Model Averaging

Conducting an inference based on a single best model without considering model uncertainty can result in underestimating the uncertainty about quantities of interest. To this problem, averaging over all of the candidate models, called Bayesian model averaging [64, 29], provides a coherent approach for accounting for model uncertainty. Let $M_1, \cdots, M_K$ be all the models considered, $\boldsymbol{\theta}_k$ be the the model parameters for k-th model, and Δ be a quantity of interest, e.g., a future observation or a model parameter. Then the posterior distribution of Δ given data D is

$$\Pr(\Delta|D) = \sum_{k=1}^{K} \Pr(\Delta|M_k, D)\Pr(M_k|D),$$

where

$$\Pr(M_k|D) = \frac{\Pr(D|M_k)\Pr(M_k)}{\sum_{i=1}^{K} \Pr(D|M_i)\Pr(M_i)},$$
$$\Pr(D|M_k) = \int \Pr(D, \boldsymbol{\theta}_k|M_i)\Pr(\boldsymbol{\theta}_k|M_i)d\boldsymbol{\theta}_k.$$

2.7 Phylogenetic Tree

We review the definition of phylogenetic trees and related topics [26] under infinite sites assumption [37]. For different assumptions, e.g., finite sites assumption and Dollo parsimony, see [86, 85, 16]. Let $T \in \{0,1\}^{c\times k}$ be a binary matrix for a mutation profile, where $c \in \mathbb{Z}_{>0}$ represents the number of cell types and $k \in \mathbb{Z}_{>0}$ represents the number of mutations. A phylogenetic tree is defined for the mutation profile T as follows.

2.7.1 Definition of a Phylogenetic Tree

Definition 2.7.1. *A phylogenetic tree $\mathcal{T} = (V, E)$ for $T \in \{0,1\}^{c\times k}$ is a rooted tree that satisfies the following condition.*

$$\exists f : \mathcal{S} \to F_{\mathcal{T}}, \exists g : \mathcal{M} \to E, \forall v \in F_{\mathcal{T}}, \forall s \in f^{-1}(\{v\}),$$
$$g^{-1}(P_{\mathcal{T}}(v)) = \{m \in \mathcal{M} | s \text{ has mutation } m\},$$

where

$$V : A \text{ set of vertices of } \mathcal{T},$$
$$E : A \text{ set of edges of } \mathcal{T},$$
$$F_{\mathcal{T}} : A \text{ set of leaves of } \mathcal{T},$$
$$\mathcal{S} : A \text{ set of indices for samples in } T,$$
$$\mathcal{M} : A \text{ set of indices for mutations in } T,$$
$$P_{\mathcal{T}}(v) := \{e \in E | e \text{ is included in the path from the root of } \mathcal{T} \text{ to } v\}.$$

The above definition of the phylogenetic tree requires the consistency between the mutation profile T and the interpreted rooted tree $\mathcal{T}$. Fig. 2.6 shows a mutation profile and the corresponding phylogenetic tree as an example. By tracking from a leaf to the root of the phylogenetic tree, we can check the mutations that samples on the leaf have.

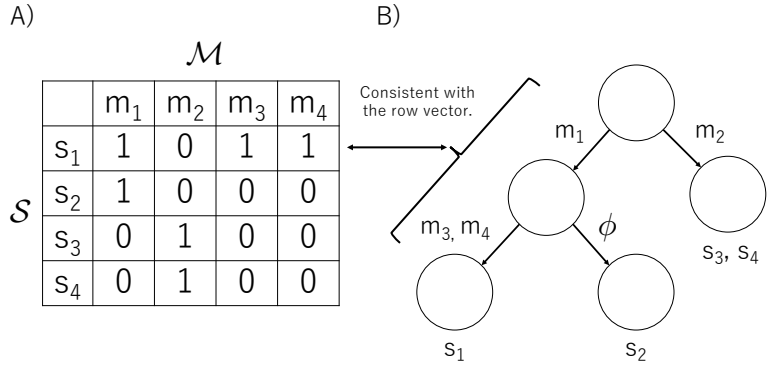


Figure 2.6: (A) shows a mutation profile as an example. (B) shows the phylogenetic tree for the mutation profile of (A).

Partially, we will review the phylogeny problem: given a mutation profile $T \in \{0,1\}^{c\times k}$, determine whether there is a phylogeny tree for T, and if so, build one. In this thesis, we will only review the decision problem about the existence of a corresponding phylogenetic tree. For the existence of the corresponding phylogenetic tree, there exists an equivalent condition. We would like to introduce the following notation, for simplification.

Definition 2.7.2.

$$O_k := \{s \in \mathcal{S} | s \text{ has mutation } k\}$$

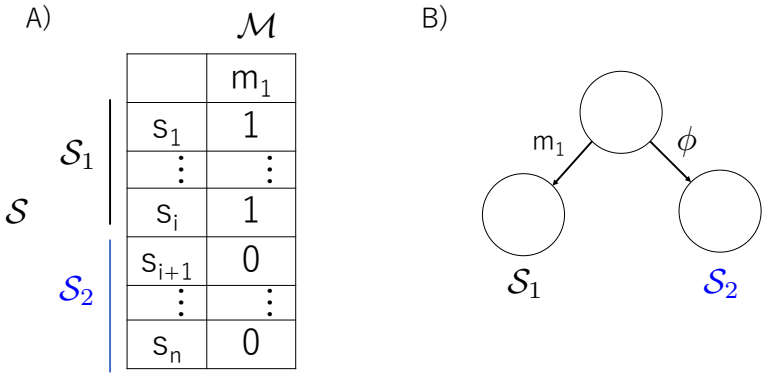


Figure 2.7: A mutation profile and a corresponding phylogenetic tree when $k = 1$. (A) shows a mutation profile T and (B) shows a corresponding phylogenetic tree $\mathcal{T}$.

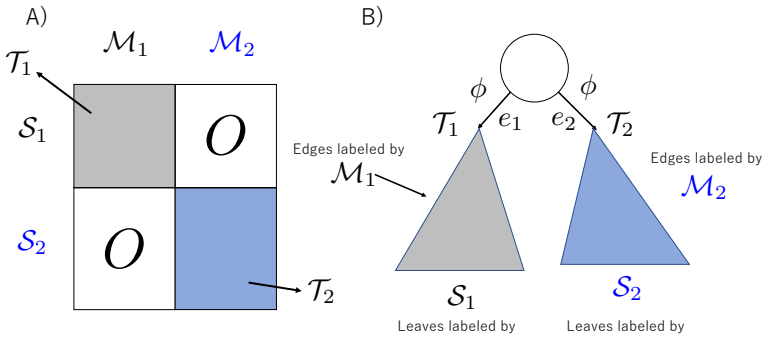


Figure 2.8: A mutation profile and a corresponding phylogenetic tree when $k > 1$ and no common mutations exist. (A) shows a mutation profile T and (B) shows a corresponding phylogenetic tree $\mathcal{T}$.

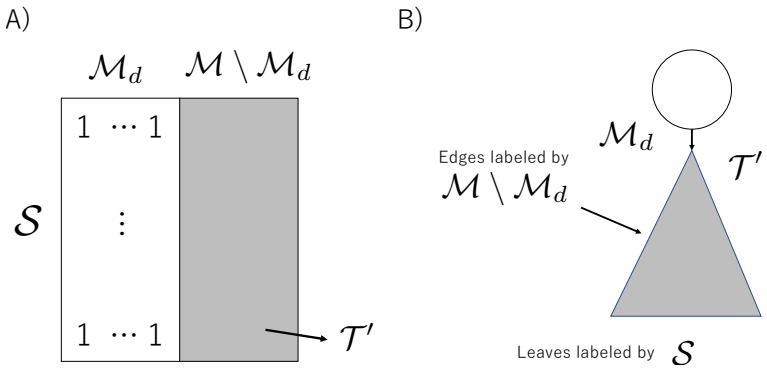


Figure 2.9: A mutation profile and a corresponding phylogenetic tree when $k > 1$ and a set of common mutations exists. (A) shows a mutation profile T and (B) shows a corresponding phylogenetic tree $\mathcal{T}$.

2.7.2 Equivalent Conditions of Having a Phylogenetic Tree

The equivalent conditions for having a phylogenetic tree can be given as follows.

Lemma 2.7.1.

$T \in \{0,1\}^{c\times k}$ *has a phylogenetic tree*
$\Leftrightarrow \forall i,j \in \{1,\cdots,k\}$ *s.t. the condition of (i) or (ii) or (iii) is satisfied,*
(i)$O_i \cap O_j = \phi$, (ii)$O_i \subseteq O_j$, (iii)$O_i \supseteq O_j$.

Proof.

$\Rightarrow \because)$
Because $k \in \mathbb{Z}_{>0}$, $\mathcal{M} \neq \phi$. Let $m_1, m_2 \in \mathcal{M}$. (m_1, m_2 can be the same index.)
From the hypothesis, there exists a phylogenetic tree $\mathcal{T}$ and function $g : \mathcal{M} \to E$.
Let $e_1 := g(m_1), e_2 := g(m_2)$. There can be three cases as follows.
$\cdot$ (a) $e_1 = e_2$.
$\cdot$ (b) e_1 is descended from e_2, or e_2 is descended from e_1.
$\cdot$ (c) e_1 is not descended from e_2, and e_2 is not descended from e_1.
In case of (a), the condition (ii) and (iii) are satisfied.
In case of (b), the condition (ii) or (iii) is satisfied.
In case of (c), there exists no common corresponding samples and (i) is satisfied.

$\Leftarrow \because)$
Let $T \in \{0,1\}^{c\times k}$ be the mutation profile.
We prove the following statement by induction with recpect to k.
T satisfies at least one condition in (i), (ii), and (iii) $\Rightarrow T$ has a phylogenetic tree.
In case of $k = 1, c \in \mathbb{N}$.

In this case, there is only one column vector.

Therefore, we can split all the samples into two set, like Fig. 2.7.

Then, the tree shown in Fig. 2.7 gives a phylogenetic tree for T.

In case of $k > 1, c \in \mathbb{N}$.

Let, $\mathcal{M}_d := \{m \in \mathcal{M} | O_m = \mathcal{S}\}$.

First, we prove the statement in case of $\mathcal{M}_d = \phi$.

Let, $\mu := \underset{i\in\mathcal{M}}{\arg\max} |O_i|$.

Because $\mathcal{M}_d = \phi$, $1 \leq |O_\mu| < k$.

From the maximality of $\mu, \forall k \in \mathcal{M}$ *s.t.* $O_k \cap O_\mu = \phi$ or $O_k \subseteq O_\mu$.

We split mutations and samples into two sets as follows.

$\cdot \mathcal{M}_1 := \{k \in \mathcal{M} | O_k \cap O_\mu = \phi\}$,

$\cdot \mathcal{M}_2 := \{k \in \mathcal{M} | O_k \subseteq O_\mu\}$,

$\cdot \mathcal{S}_1 := \bigcup_{i\in\mathcal{M}_1} O_i$,

$\cdot \mathcal{S}_2 := \bigcup_{i\in\mathcal{M}_2} O_i$.

The above sets give partitions as follows.

$\cdot \mathcal{M}_1 \cup \mathcal{M}_2 = \mathcal{M}$,

$\cdot \mathcal{M}_1 \cap \mathcal{M}_2 = \phi$,

$\cdot \mathcal{S}_1 \cup \mathcal{S}_2 = \mathcal{S}$,

$\cdot \mathcal{S}_1 \cap \mathcal{S}_2 = \phi$.

$\because$) We check $\mathcal{S}_1 \cap \mathcal{S}_2 = \phi$.

$$\mathcal{S}_1 \cap \mathcal{S}_2 = \bigcup_{i \in \mathcal{M}_1} O_i \cap \bigcup_{j \in \mathcal{M}_2} O_i = \bigcup_{i \in \mathcal{M}_1, j \in \mathcal{M}_2} (O_i \cap O_j).$$

From the definitions, $O_i \cap O_\mu = \phi,\ O_j \subseteq O_\mu$.

Therefore, $\forall i \in \mathcal{M}_1, j \in \mathcal{M}_2\ s.t.\ O_i \cap O_j = \phi$, and $\mathcal{S}_1 \cap \mathcal{S}_2 = \phi$.

We can see the mutation profile can be represented like Fig. 2.8.

From $1 \leq |\mathcal{M}_1| < k$ and $1 \leq |\mathcal{M}_2| < k$,

we can construct phylogenetic trees of $\mathcal{T}_1, \mathcal{T}_2$ for $\mathcal{M}_1, \mathcal{M}_2$.

By jointing $\mathcal{T}_1, \mathcal{T}_2$ by edges of e_1, e_2, we obtain a novel rooted tree $\mathcal{T}$.

We show this rooted tree satisfies the following condition.

$\exists f : \mathcal{S} \to F_{\mathcal{T}}, \exists g : \mathcal{M} \to E, \forall v \in F_{\mathcal{T}}, \forall s \in f^{-1}(\{v\})$,

$g^{-1}(P_{\mathcal{T}}(v)) = \{m \in \mathcal{M} | s \text{ has mutation } m\}$.

$\because$)

Because $\mathcal{T}_1, \mathcal{T}_2$ are phylogenetic trees,

$\cdot \exists f_1 : \mathcal{S}_1 \to F_{\mathcal{T}_1}, \exists g_1 : \mathcal{M}_1 \to E_1, \forall v_1 \in F_{\mathcal{T}_1}, \forall s_1 \in f_1^{-1}(\{v_1\})$,
$g_1^{-1}(P_{\mathcal{T}_1}(v_1)) = \{m \in \mathcal{M}_1 | s_1 \text{ has mutation } m\}$,

$\cdot \exists f_2 : \mathcal{S}_2 \to F_{\mathcal{T}_2}, \exists g_2 : \mathcal{M}_2 \to E_2, \forall v_2 \in F_{\mathcal{T}_2}, \forall s_2 \in f_2^{-1}(\{v_2\})$,
$g_2^{-1}(P_{\mathcal{T}_2}(v_2)) = \{m \in \mathcal{M}_2 | s_2 \text{ has mutation } m\}$.

We show $\mathcal{T}$ is a phylogenetic tree for T.

We set f, g as follows.

$$f(s) := \begin{cases} f_1(s) & (s \in \mathcal{S}_1) \\ f_2(s) & (s \in \mathcal{S}_2) \end{cases}, g(m) := \begin{cases} g_1(m) & (m \in \mathcal{M}_1) \\ g_2(m) & (m \in \mathcal{M}_2) \end{cases}.$$

If $v \in F_{\mathcal{T}_1}$,

by the definition of $\mathcal{T}_1, \forall s \in f^{-1}(\{v\}) = f_1^{-1}(\{v\})$,

$g^{-1}(P_{\mathcal{T}}(v))$
$= g^{-1}(P_{\mathcal{T}_1}(v) \cup e_1)$
$= g_1^{-1}(P_{\mathcal{T}_1}(v)) \cup \phi$
$= \{m \in \mathcal{M}_1 | s \text{ has mutation } m\}$
$= \{m \in \mathcal{M} | s \text{ has mutation } m\}$.
($\because\ \mathcal{S}_1 \cap \mathcal{S}_2 = \phi, s$ does not have mutations in $\mathcal{M}_2$.)

If $v \in F_{\mathcal{T}_2}$,

we can prove in the same manner.

Next, we prove the statement in case of $\mathcal{M}_d \neq \phi$.

If $\mathcal{M}_d = \mathcal{M}$, the proof is trivial.

Otherwise, $\mathcal{M}_d \subsetneq \mathcal{M}$, we make a phylogenetic tree $\mathcal{T}'$ for $\mathcal{M} \setminus \mathcal{M}_d$.

Then, we build a rooted tree $\mathcal{T}$ (Fig. 2.9), and check the definition.

□

2.7.3 Modeling Variant Allele Frequencies in Bulk Tumor Sequence Data Sets

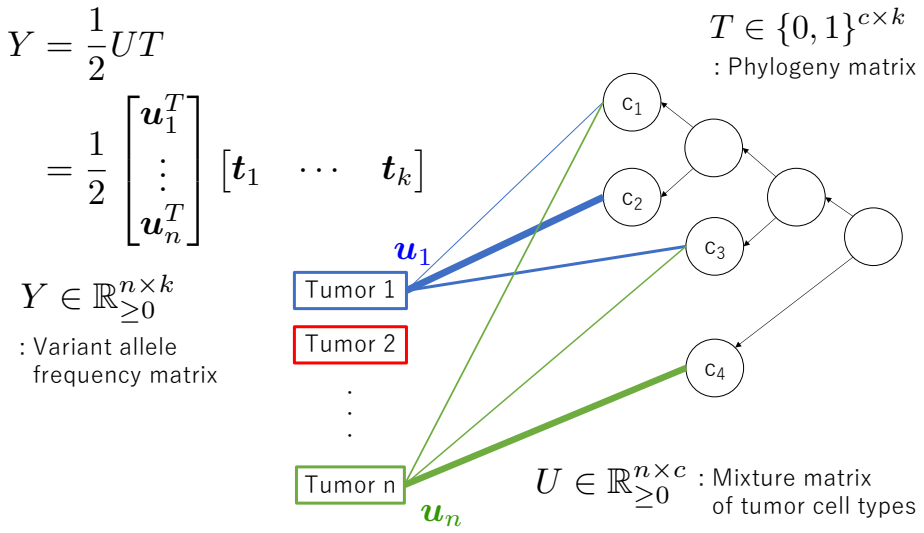


Figure 2.10: A modeling of the variant allele frequencies in multiple bulk sequence data sets.

To infer the hidden tumor phylogeny from the multiple tumor bulk sequence data sets, several modeling of the variant allele frequencies are assumed. Here, we introduce one of the simplest (and most often used) modeling. Let $T \in \{0,1\}^{c \times k}$ be a matrix which has a phylogeny, and $U \in \mathbb{R}_{\geq 0}^{n \times c}$ be a mixture matrix of which each row vector is a non-negative simplex. Each binary row vector in T represents the set of mutations held by each tumor cell type, and the corresponding phylogenetic tree satisfies the infinite sites assumption. Each non-negative valued row vector in U represents the mixture rate of the tumor cell types for each bulk sequence data.

We describe the matrix of the variant allele frequencies as $Y \in \mathbb{R}_{\geq 0}^{n \times k}$, where $Y_{i,j}$ represents the variant allele frequency of the j-th mutation at the i-th sequence data set. By assuming that every mutation occurs only on one haplotype, the variant allele frequencies can be expressed as follows (Fig. 2.10).

$$Y = \frac{1}{2}UT.$$

For the actual inference of the tumor phylogenetic tree from bulk sequencing data sets, see [75, 17, 18, 27, 48, 12, 84, 70].

Chapter 3

A Bayesian Model Integration for Mutation Calling through Data Partitioning

3.1 Overview

Detection of somatic mutations is a basis for the field of cancer genomics and continuous performance improvement of the detection accuracy is desired and conducted. A lot of researches have focused on the sequence-data-specific properties; they modeled single property in Bayesian statistical models explicitly and successfully improved the detection accuracy. However, no design of Bayesian statistical models has been proposed for using the multiple sequence-data-specific properties in such an explicit manner, and hence there is room for further performance improvement.

In this chapter, we introduce a design of a Bayesian statistical model termed partitioning-based model integration. Under this design, we set a partitioning rule to each sequenced read, label the type of data, and allocate a generation probability for each type. One advantageous point of this design is that there exists a case in which we do not require additional hyperparameters in combining multiple models unlike the Bayesian model averaging. Based on this design, we propose a novel mutation calling method of OHVarfinDer that leverages the multiple sequence-data-specific properties for better detection performance.

The organization of this chapter is as follows. First, we explain the existing methods. Second, we introduce the design of partitioning-based model integration. Third, we show the details of the Bayesian model in OHVarfinDer. Finally, we show some experimental results.

Contents of this chapter are mainly related to the published work of [55, 53].

3.2 Related Work

Here we would like to introduce existing mutation callers. Previous mutation callers can be mainly categorized into two types. The first type of mutation caller [83, 40] does not assume any specific probability distribution, and the score is based on Fisher's exact test. In this type of mutation caller, the number of reference-supporting reads and variant-supporting reads is counted on tumor and normal sample, and the P-value is computed based on these 2 by 2 contingency table. The second type of mutation caller assumes specific probability distributions and constructs stochastic models that can explicitly incorporate the sequence-data specific property, e.g., base quality, read mapping states [1], haplotype-specific somatic mutations [77], and overlapping part of paired-end reads [55].

3.2.1 VarScan2

VarScan2 [40] is based on the Fisher's exact test [19] method. In this approach, using tumor data and normal data, 2 by 2 contingency table is prepared. The contingency table is as follows.

	# reference-supporting reads	# variant-supporting reads
Tumor sample	a	b
Normal sample	c	d
Sum	a+c	b+d

Table 3.1: 2 by 2 contingency table used for Fisher's exact test.

Under the null hypothesis where there is no difference between the tumor and normal samples, the P-value is calculated using the hypergeometric distribution.

$$\text{P-value} = \frac{{}_{a+b}C_a \cdot {}_{c+d}C_c}{{}_nC_{a+c}},$$

where $n = a + b + c + d$. If the P-value is small enough to reject the null hypothesis, the candidate position is judged to have a somatic mutation. This method performs reasonably under any experimental settings by ignoring any other background information of the observed sequence data.

3.2.2 MuTect

The variant detection of MuTect [10] is done by comparing the likelihoods of an error data generation model and a mutated data generation model. For each site, we denote reference allele by $g \in \{A, C, G, T\}$. Error probability at candidate position of i-th read e_i is defined by the phred-like quality score q_i ($e_i := 10^{(\frac{-q_i}{10})}$). Variant detection is done by computing likelihoods under two models. M_0 is the model in which there is no variant and all the non-reference bases are generated by sequence errors, and M_f^m represents the model in which a variant allele $m \in \{A, C, G, T\}$ exists with variant allele frequency of f, and sequence error also occur on each read. The likelihood of M_f^m is as follows:

$$L\left[M_f^m\right] = \Pr(\{b_i\}|\{e_i\}, g, m, f) = \prod_{i=1}^{d} \Pr(b_i|e_i, g, m, f).$$

By assuming the independence among sequence errors, the base b_i is generated as follows:

$$\Pr(b_i|e_i, g, m, f) = \begin{cases} f\frac{e_i}{3} + (1-f)(1-e_i) & (b_i = g) \\ f(1-e_i) + (1-f)\frac{e_i}{3} & (b_i = m) \\ \frac{e_i}{3} & (\text{otherwise}) \end{cases}.$$

By incorporating the prior odds $\left(\frac{\Pr(m,f)}{(1-\Pr(m,f))}\right)$ and a decision threshold $\log_{10}\delta_T$, MuTect detect somatic mutations.

$$\text{LOD}_T(m, f) = \log_{10}\left(\frac{L[M_f^m]\Pr(m, f)}{L[M_0](1 - \Pr(m, f))}\right) \geq \log_{10}\delta_T$$

$$\Leftrightarrow \log_{10}\left(\frac{L[M_f^m]}{L[M_0]}\right) \geq \log_{10}\delta - \log_{10}\left(\frac{\Pr(m, f)}{1 - \Pr(m, f)}\right) =: \theta_T,$$

where $\theta_T = 6.3$ is used for somatic mutation detection in practice. This method is also used for the detection of germline variants, by setting a different decision threshold.

3.2.3 Strelka

The variant detection in Strelka [71] is based on the evaluation of a probability in which somatic mutation occurred in the tumor sample and diploid genotype is observed in the normal sample. The joint probability of somatic event occurrence and diploid genotype observation is denoted by $\Pr(E_S, G_n|D)$, where E_S denotes the somatic event, G_n denotes the genotype in the normal sample, and D denotes the data set. By applying Bayes' theorem, the joint probability is factorized as $\Pr(E_S, G_n|D) = \Pr(E_S|D)\Pr(G_n|D)$. For $\Pr(E_S|D)$, let f_t, f_n be the variant allele frequencies in tumor and normal sample, the somatic event can be seen as the event that different variant allele frequencies are observed. Then the somatic event probability is computed as follows:

$$\Pr(E_S|D) = \int_{f_t, f_n} \mathbb{I}_{\{f_t \neq f_s\}} \Pr(f_t, f_n|D) df_t df_n.$$

The author approximately computes the integral by splitting the $[0, 1]^2$ region into 10 by 10 cells. For $\Pr(G_n|D)$, the author computes this probability by a conventional single-sample Bayesian approach. Strelka uses $\Pr(E_S, G_n = \text{ref/ref}|D)$ as a score for variant detections.

3.2.4 HapMuC

Each true somatic mutation is reported to occur only on one haplotype, but sequence errors can occur on reads generated from both haplotypes. This bias of haplotype is reported to be a beneficial property for improving detection performance and HapMuC [77] uses this property by constructing different distributions of the original haplotype for each observed paired-end read and improved the detection performance in whole-genome sequence (WGS) data set.

3.2.5 OVarCall

If the original DNA template is shorter than twice the length of the read, there exist overlapping regions from which genome sequences are obtained twice. From the previous studies [61, 9], consistent non-reference bases are more probably a true mutation and inconsistent bases are more likely a sequence error and this property helps to decrease error rates. OVarCall [55] uses this property by setting distributions in which each pair of reads are generated from a common latent state, and improved the detection performance at whole exome sequence (WES) data set.

3.3 Proposed Design of Bayesian Model

3.3.1 Bayes Factor for Finding Mutations

We denote a data set as $\mathcal{R} := \{r_n\}_{n=1}^d$, where r_n is the n-th string consisting of $\{A, T, G, C\}$ and d is the depth on the mutation candidate position. We denote mutated and error data generation models as $\mathcal{M}_M, \mathcal{M}_E$, and corresponding set

of parameters as Θ_M, Θ_E. Next, the Bayes factor [33] is written as follows.

$$\mathrm{BF} = \frac{\Pr(\mathcal{R}|\mathcal{M}_M)}{\Pr(\mathcal{R}|\mathcal{M}_E)},$$

where

$$\Pr(\mathcal{R}|\mathcal{M}_S) = \int \Pr(\mathcal{R}, \Theta_S|\mathcal{M}_S)\Pr(\Theta_S)d\Theta_S, \;\; S \in \{M, E\}.$$

3.3.2 Model Integration by Bayesian Model Averaging

Before showing the proposed design, we would like to show how Bayesian model averaging can integrate models to compute Bayes factor. We assume $K \in \mathbb{N}$ models for each mutated and error data generation model, and denote these models as $\mathcal{M}_{M,k}, \mathcal{M}_{E,k}$, where $k \in \{1, ..., K\}$. We denote a corresponding set of parameters as $\Theta_{M,k}, \Theta_{E,k}$, where $k \in \{1, ..., K\}$, and $\Theta_{M,\mathrm{all}} := \bigcup_{k=1}^{K} \Theta_{M,k}$, and $\Theta_{E,\mathrm{all}} := \bigcup_{k=1}^{K} \Theta_{E,k}$. We assume disjointness between $\Theta_{M,\mathrm{all}}$ and $\Theta_{E,\mathrm{all}}$, i.e., $\Theta_{M,\mathrm{all}} \cap \Theta_{E,\mathrm{all}} = \phi$, and we do not assume disjointness within $\Theta_{S,1}, \cdots, \Theta_{S,K}$.

By following the idea of the Bayesian model averaging, we can construct an integrated generative model Fig. 3.1 and set the probability of the observed data as follows:

$$\Pr(r_n|H = \mathcal{M}_{S,k}, \Theta_{M,\mathrm{all}}, \Theta_{E,\mathrm{all}}) = \Pr(r_n|\Theta_{S,k}, \mathcal{M}_{S,k}), \quad S \in \{M, E\}, \tag{3.1}$$

where H is an unobserved random variable representing the model that generates the observed data set.

Under this stochastic model, we can calculate the Bayes factor as follows:

$$\begin{aligned}
\mathrm{BF} &= \frac{\Pr(\mathcal{R}|H \in \{\mathcal{M}_{M,1}, \cdots, \mathcal{M}_{M,K}\})}{\Pr(\mathcal{R}|H \in \{\mathcal{M}_{E,1}, \cdots, \mathcal{M}_{E,K}\})} \\
&= \frac{\Pr(H \in \{\mathcal{M}_{M,1}, \cdots, \mathcal{M}_{M,K}\}|\mathcal{R})}{\Pr(H \in \{\mathcal{M}_{E,1}, \cdots, \mathcal{M}_{E,K}\}|\mathcal{R})} \cdot \frac{\Pr(H \in \{\mathcal{M}_{E,1}, \cdots, \mathcal{M}_{E,K}\})}{\Pr(H \in \{\mathcal{M}_{M,1}, \cdots, \mathcal{M}_{M,K}\})} \\
&= \frac{\sum_{k=1}^{K} \Pr(\mathcal{R}|\mathcal{M}_{M,k})\Pr(\mathcal{M}_{M,k})}{\sum_{k=1}^{K} \Pr(\mathcal{R}|\mathcal{M}_{E,k})\Pr(\mathcal{M}_{E,k})} \cdot \frac{\Pr(H \in \{\mathcal{M}_{E,1}, \cdots, \mathcal{M}_{E,K}\})}{\Pr(H \in \{\mathcal{M}_{M,1}, \cdots, \mathcal{M}_{M,K}\})} \\
&= \frac{\sum_{k=1}^{K} \Pr(\mathcal{R}|\mathcal{M}_{M,k})h_{T,k}}{\sum_{k=1}^{K} \Pr(\mathcal{R}|\mathcal{M}_{E,k})h_{E,k}},
\end{aligned}$$

where

$$\begin{aligned}
h_{S,k} &:= \frac{\Pr(H = \mathcal{M}_{S,k})}{\Pr(H \in \{\mathcal{M}_{S,1}, \cdots, \mathcal{M}_{S,K}\})}, \\
\Pr(\mathcal{R}|\mathcal{M}_{S,k}) &= \int \Pr(\mathcal{R}|\Theta_{S,k}, \mathcal{M}_{S,k})\Pr(\Theta_{S,k}|\mathcal{M}_S)d\Theta_{S,k}.
\end{aligned}$$

Therefore, this manner of model integration requires additional $2K - 2$ hyperparameters of $h_{S,k}$.

3.3.3 Partitioning-Based Model Integration

We assume that we can observe indicator variable $t_n \in \{1, 2, ..., K\}$ with each data r_n and assume that the original data set is partitioned into K subsets and t_n indicates the subset of data to which r_n belongs. We also assume that the k-th subset of data is generated through the k-th model of $\mathcal{M}_{M,k}$ or $\mathcal{M}_{E,k}$. We denote this augmented data set as $\mathcal{R}_{\mathrm{aug}} := \{(r_n, t_n)\}_{n=1}^{d}$.

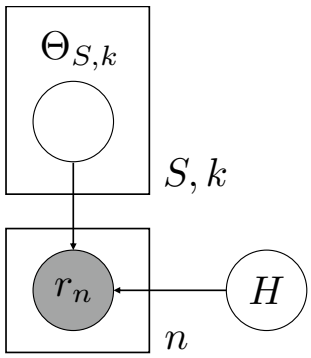


Figure 3.1: Graphical model for Bayesian model averaging. $S \in \{M, E\}$ represents the hypothesis.

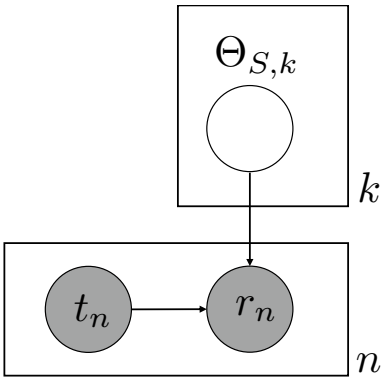


Figure 3.2: Graphical model for partitioning-based model integration. $S \in \{M, E\}$ represents the hypothesis.

We assume the graphical model of Fig. 3.2, and we set the probability of the observed data as follows:

$$\Pr(r_n|t_n, \Theta_{S,\text{all}}, \mathcal{M}_S) = \Pr(r_n|\Theta_{S,t_n}, \mathcal{M}_{S,t_n}). \tag{3.2}$$

Our purpose here is to compute the following Bayes factor.

$$\text{BF} = \frac{\Pr(\mathcal{R}_{\text{aug}}|\mathcal{M}_M)}{\Pr(\mathcal{R}_{\text{aug}}|\mathcal{M}_E)}.$$

From the graphical model in Fig. 3.2 and above assumptions of Eq. (3.2), the joint probability can be computed as follows.

$$\begin{aligned}
&\Pr(\mathcal{R}_{\text{aug}}, \Theta_{S,\text{all}}|\mathcal{M}_S) \\
&= \Pr(\Theta_{S,\text{all}}|\mathcal{M}_S)\Pr(\mathcal{R}_{\text{aug}}|\Theta_{S,\text{all}}, \mathcal{M}_S) \\
&= \Pr(\Theta_{S,\text{all}}|\mathcal{M}_S)\prod_n \Pr(r_n, t_n|\Theta_{S,\text{all}}, \mathcal{M}_S) \\
&= \Pr(\Theta_{S,\text{all}}|\mathcal{M}_S)\prod_n \Pr(r_n|t_n, \Theta_{S,\text{all}}, \mathcal{M}_S)\Pr(t_n|\mathcal{M}_S) \\
&= \Pr(\Theta_{S,\text{all}}|\mathcal{M}_S)\prod_n \Pr(r_n|\Theta_{S,t_n}, \mathcal{M}_{S,t_n})\Pr(t_n|\mathcal{M}_S),
\end{aligned}$$

where

$$S \in \{M, E\}, \Pr(t_n|\mathcal{M}_S) > 0.$$

From this joint probability, the marginal likelihood can be computed as follows.

$$\Pr(\mathcal{R}_{\text{aug}}|\mathcal{M}_S) = A_S \cdot \left\{ \prod_n \Pr(t_n|\mathcal{M}_S) \right\},$$

where

$$A_S := \int \Pr(\Theta_{S,\text{all}}|\mathcal{M}_{S,k}) \left\{ \prod_{k=1}^{K} \prod_{\{n|t_n=k\}} \Pr(r_n|\Theta_{S,k}, \mathcal{M}_{S,k}) \right\} d\Theta_{S,\text{all}}.$$

If we can assume $\Pr(t|\mathcal{M}_M) = \Pr(t|\mathcal{M}_E)$ for any $t \in \{1, ..., K\}$, we do not need to set $\Pr(t|\mathcal{M}_M), \Pr(t|\mathcal{M}_E)$ for computation of Bayes factor. This is because

$$\text{BF} = \frac{\Pr(\mathcal{R}_{\text{aug}}|\mathcal{M}_M)}{\Pr(\mathcal{R}_{\text{aug}}|\mathcal{M}_E)} = \frac{A_M \cdot \prod_n \Pr(t_n|\mathcal{M}_M)}{A_E \cdot \prod_n \Pr(t_n|\mathcal{M}_E)} = \frac{A_M}{A_E}.$$

This manner of model integration requires two conditions. The first condition is a partitioning rule on the data set and we can construct a corresponding generative model for each partitioned data set. The second condition is that partition probabilities should be the same among the tumor and error model ($\Pr(t|\mathcal{M}_M) = \Pr(t|\mathcal{M}_E)$). The merit of this manner is that partition probabilities $\Pr(t|\mathcal{M}_M), \Pr(t|\mathcal{M}_E)$ do not affect the Bayes factor and thus careful and explicit settings of these probabilities are not necessary.

3.4 Bayesian Hierarchical Modeling for Mutation Calling

3.4.1 Characteristic Information Sources for Mutation Calling

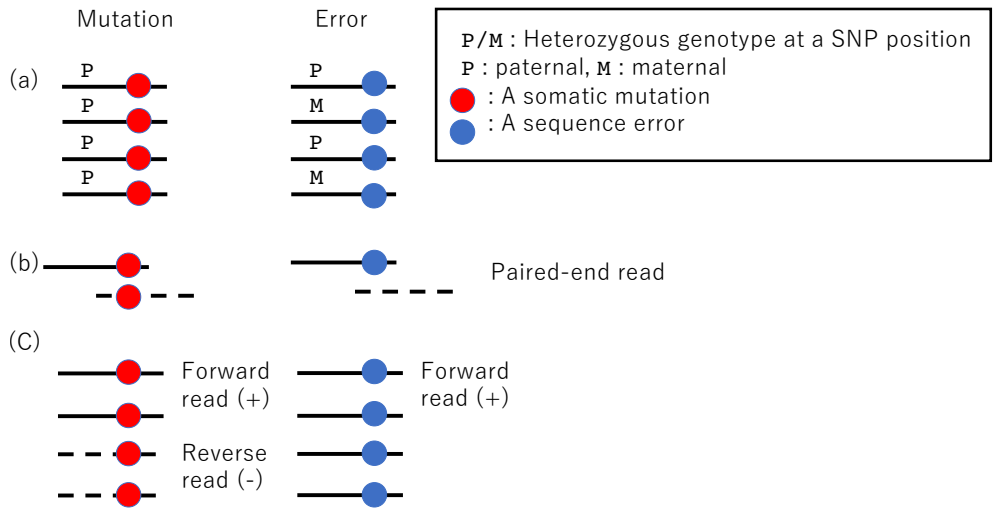


Figure 3.3: (a) The typical pattern of reads when heterozygous SNPs near the mutation candidate appear. (b) The typical pattern of paired-end reads when overlapping paired-end reads cover the mutation candidate. (c) The typical pattern of reads when the strand bias appears in variant supporting reads.

Heterozygous SNPs Covered by Paired-End Reads

The first additional information source in somatic mutation calling is heterozygous SNPs near somatic mutation candidates. The human genome is a diploid

set of haplotypes, i. e., the maternal haplotype and paternal haplotype. Each somatic mutation is known to occur typically only on one side of the haplotypes, i. e., heterozygous mutation. Therefore, variant supporting reads that cover heterozygous SNPs are generated from only one side of the haplotypes as shown in the left side of Fig. 3.3(a). However, when sequence errors occur on the mutation candidate position, variant supporting reads covering heterozygous SNPs probably have both heterozygous SNPs as in the right side of Fig. 3.3(a). This information source was used in HapMuC [77].

Overlaps of Paired-End Reads

The second additional information source is overlaps of paired-end reads. Through Illumina's sequencing, a pair of paired-end reads, i. e., forward and reverse reads, is sequenced from both sides of the same DNA fragment. If the DNA fragment is shorter than 2-fold the read length, the pair of reads has an overlapping region where the sequencing process is conducted twice from different directions independently.

If the both forward and reverse reads show the same alteration in the overlapping region as in the left side of Fig. 3.3(b), it is likely that the change is because of a mutation and not because of errors, as the occurrence probability of two errors at the same site in the overlapping region is expected to be very low, except for PCR errors in the sample preparation phase [9]. In contrast, an error case is probable when only one of the reads contains an alteration in the overlapping region as in the right side of Fig. 3.3(b). This information source has been used in OVarCall [55].

Strand Biases of Paired-End Reads

The third additional information source we considered is strand biases in variant supporting reads that cover a mutation candidate. If only forward (or reverse) reads contain a mutation candidate despite sufficient numbers of both forward and reverse reads, this phenomenon is known as strand bias as in the right side of Fig. 3.3(c). If a true somatic mutation exists, strand bias rarely occurs, and the proportion of variant supporting forward/reverse reads should be ideally similar as in the left side of Fig. 3.3(c). This information source is used for filtering in MuTect [10].

Representative Examples in Real Data Sets

We show examples from real data sets, in which we can find that given mutation candidates are only errors. Fig. 3.4 shows screenshots of IGV (`http://software.broadinstitute.org/software/igv/`).

The first erroneous case shown in Fig. 3.4(a) represents the variant supporting reads with both heterozygous SNPs. In this case, variant supporting reads have both heterozygous SNPs, as indicated by red and blue circles. This case corresponds with the erroneous case in Fig. 3.3(a).

The second erroneous case shown in Fig. 3.4(b) represents a paired-end reads with inconsistent bases at a mutation candidate position. In this case, a pair of paired-end reads that are highlighted in the red line have different bases at the mutation candidate position. This case corresponds with the erroneous case in Fig. 3.3(b).

Simpler methods, e. g., a Fisher's exact test-based method of VarScan2, evaluate these two types of errors as somatic mutations. In the case of Fig. 3.4(a),

VarScan2 showed a low p-value of 0.043, and in Fig. 3.4(a), VarScan2 also showed a low p-value of 0.0050. The main purpose of this paper is to construct a Bayesian method that discriminates these errors from somatic mutations.

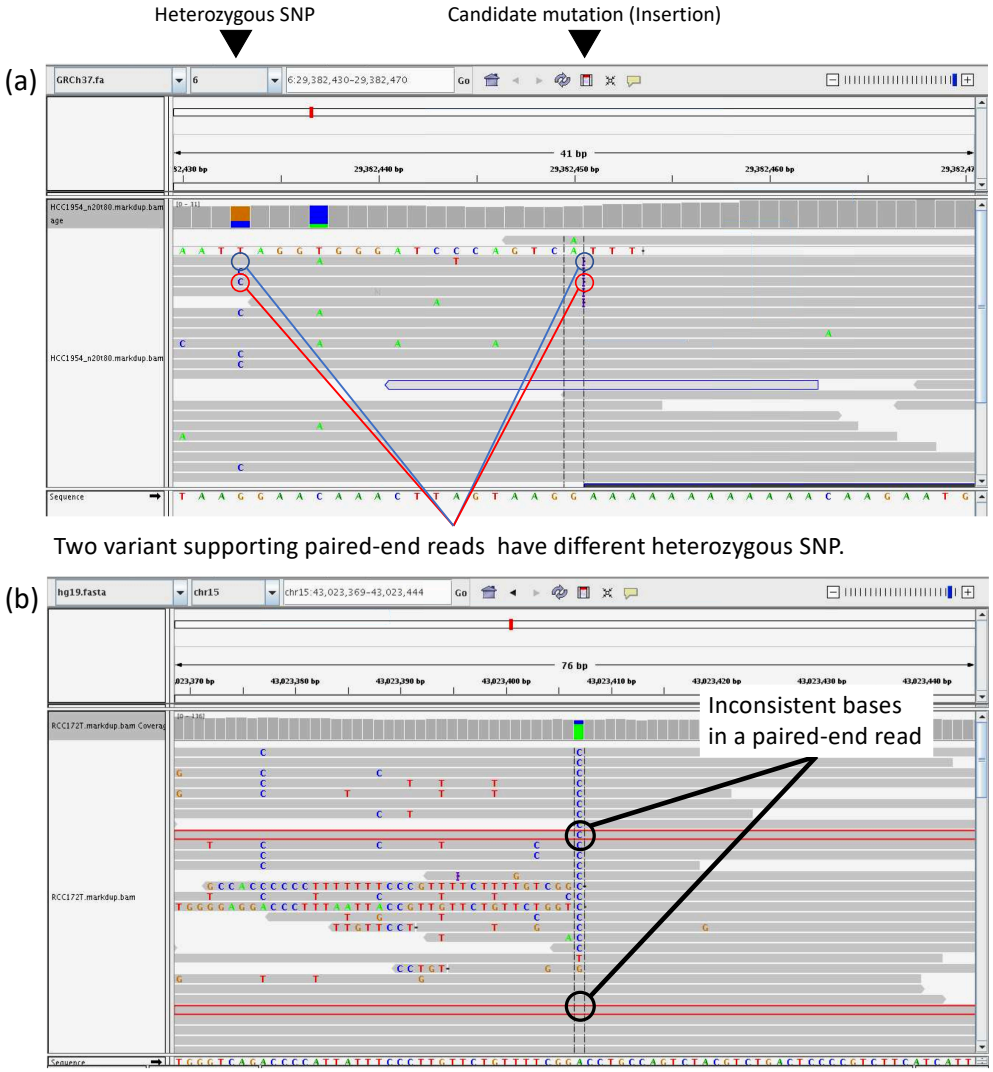


Figure 3.4: Typical cases of errors shown in the IGV screenshot. (a) In this case, both heterozygous SNPs near the mutation candidate appear in the variant supporting reads. See the erroneous case in Fig. 3.3(a). (b) One corresponding paired-end read is highlighted in the red line. In this case, inconsistent bases in a paired-end read occur at a mutation candidate position. See the erroneous case in Fig. 3.3(b). Our method successfully evaluates these errors with low Bayes factor scores, i. e., 0.000059 in (a) and 0.0000011 in (b).

3.4.2 Graphical Model of OHVarfinDer

We show the graphical model of OHVarfinDer in Fig. 3.5. We distinguish tumor and normal data and describe the n-th paired-end read from tumor sample as $\boldsymbol{r}_{T,n} := (r_{T,n,+}, r_{T,n,-})$, where $r_{T,n,+}$ and $r_{T,n,-}$ are string sequence of A, T, G, C. For reads from normal sample, we denote $\boldsymbol{r}_{N,n} := (r_{N,n,+}, r_{N,n,-})$. We also represent the n-th partition category as $t_{T,n}, t_{N,n}$ in tumor and normal sample. For the latent variables, we express the n-th latent variable in the tumor sample as $\boldsymbol{z}_{T,n}$, and describe the n-th latent variable in the normal sample as $\boldsymbol{z}_{N,n}$. $\boldsymbol{z}_{T,n}$ and $\boldsymbol{z}_{N,n}$ are one-hot encoding vectors indicating the original DNA sequence pair. $\mathcal{H}_k$ represents an array of DNA sequence pairs for k-th partition category from which each observed read pair is supposed to be generated. $\Theta_{S,k}$ represents a set of parameters that regulates the frequency of DNA sequences in $\mathcal{H}_k$.

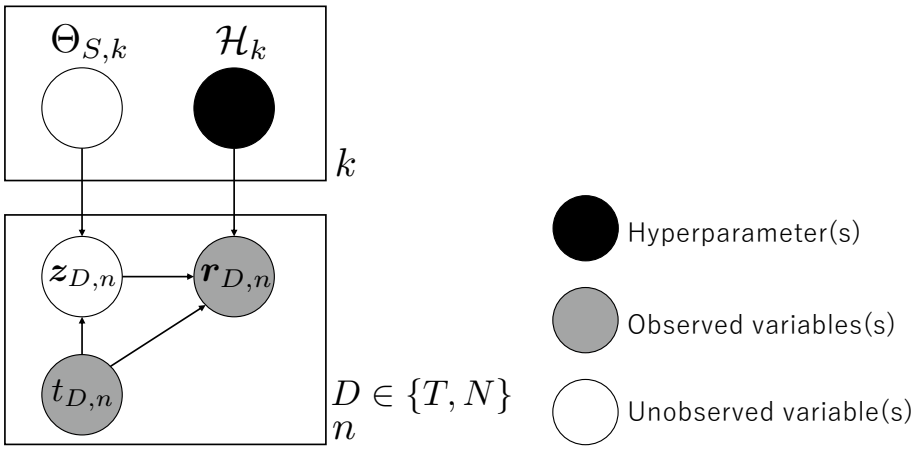


Figure 3.5: Graphical model of OHVarfinDer. $S \in \{M, E\}$ represents the hypothesis.

We can see that the probability of $\Pr(t_{T,n}|\mathcal{M}_M)$, $\Pr(t_{N,n}|\mathcal{M}_E)$, $\Pr(t_{T,n}|\mathcal{M}_M)$ and $\Pr(t_{N,n}|\mathcal{M}_E)$ is not required to compute Bayes factor in a similar manner with Section 3.3.3.

To simplify the notation, we define:

$$\begin{aligned}
\mathcal{R}_{\mathrm{NT}} &:= \{\boldsymbol{r}_{N,n}|n = 1, \cdots, d_N\} \cup \{\boldsymbol{r}_{T,n}|n = 1, \cdots, d_T\}, \\
\mathcal{T}_{\mathrm{NT}} &:= \{t_{N,n}|n = 1, \cdots, d_N\} \cup \{t_{T,n}|n = 1, \cdots, d_T\}, \\
\mathcal{Z}_{\mathrm{NT}} &:= \{\boldsymbol{z}_{N,n}|n = 1, \cdots, d_N\} \cup \{\boldsymbol{z}_{T,n}|n = 1, \cdots, d_T\}, \\
\Theta_{S,\mathrm{all}} &:= \bigcup_{k=1}^{K} \Theta_{S,k}, \ \ S \in \{M, E\},
\end{aligned}$$

where d_N and d_T represent the depth coverage in normal or tumor sequence data. The marginal likelihood for model $\mathcal{M}_S$ can be represented as follows:

$$\begin{aligned}
&\Pr(\mathcal{R}_{\mathrm{NT}}, \mathcal{T}_{\mathrm{NT}}|\mathcal{M}_S) \\
&= \int \Pr(\Theta_{S,\mathrm{all}}|\mathcal{M}_S) \\
&\quad \cdot \Pr(\mathcal{T}_{\mathrm{NT}}|\mathcal{M}_S)\Pr(\mathcal{Z}_{\mathrm{NT}}|\mathcal{T}_{\mathrm{NT}}, \Theta_{S,\mathrm{all}}, \mathcal{M}_S)\Pr(\mathcal{R}_{\mathrm{NT}}|\mathcal{Z}_{\mathrm{NT}}, \mathcal{T}_{\mathrm{NT}}, \mathcal{M}_S) d\mathcal{Z} d\Theta_{S,\mathrm{all}} \\
&= \int \Pr(\Theta_{S,\mathrm{all}}|\mathcal{M}_S) \\
&\quad \cdot \prod_{D \in \{N,T\}} \prod_{n}^{d_D} \Pr(t_{D,n}|\mathcal{M}_S)\Pr(\boldsymbol{z}_{D,n}|\Theta_{S,t_{D,n}}, \mathcal{M}_{S,t_{D,n}})\Pr(\boldsymbol{r}_{D,n}|\boldsymbol{z}_{D,n}, \mathcal{H}_{t_{D,n}}) d\mathcal{Z} d\Theta_{S,\mathrm{all}}
\end{aligned}$$

$$= \left\{ \prod_{D} \prod_{n}^{d_D} \Pr(t_{D,n}|\mathcal{M}_S) \right\} \cdot F_S,$$

where

$$F_S(\mathcal{R}_{\mathrm{NT}}) := \int \Pr(\Theta_{S,\mathrm{all}}|\mathcal{M}_S) \cdot \prod_{D\in\{N,T\}} \prod_{n=1}^{d_D} \Pr(\boldsymbol{z}_{D,n}|\Theta_{S,t_{D,n}}, \mathcal{M}_{S,t_{D,n}}) \Pr(\boldsymbol{r}_{D,n}|\boldsymbol{z}_{D,n}, \mathcal{H}_{t_{D,n}}) d\mathcal{Z} d\Theta_{S,\mathrm{all}}.$$

From this, if $\Pr(t_{D,n}|\mathcal{M}_M) = \Pr(t_{D,n}|\mathcal{M}_E)$ for any $D \in \{T, N\}$ and n, it is not required to set partitioning probabilities, like Section 3.3.3.

3.4.3 Partitioning Rules for Each Paired-End Read in OHVarfinDer

In our method, we split paired-end reads into 5 types. $t_{D,n} \in \{0, 1, 2, 3, 4\}$ is determined for each paired-end read by the following partitioning rule.

O(+)H(-) Category

A paired-end read in this category ($t_{D,n} = 0$) is overlapping between the forward read and reverse read at the mutation candidate position and covers no heterozygous SNPs nearby the candidate position.

O(-)H(+) Category

A paired-end read in this category ($t_{D,n} = 1$) is not overlapping between the forward read and reverse read at the mutation candidate position and covers heterozygous SNPs nearby the candidate position. Note that global haplotype phasing is not necessary and we only conduct haplotype phasing locally around the mutation candidate positions as previously conducted in [77].

O(+)H(+) Category

A paired-end read in this category ($t_{D,n} = 2$) is overlapping between the forward read and reverse read at the mutation candidate position and covers heterozygous SNPs nearby the candidate position.

O(-)H(-)S(+) Category

A paired-end read in this category ($t_{D,n} = 3$) is not overlapping between the forward read and reverse read at the mutation candidate position and covers no heterozygous SNPs nearby the candidate position. The mutation candidate position is covered by the forward read. (Forward/reverse is determined by the mapping direction compared to the reference sequence.)

O(-)H(-)S(-) Category

A paired-end read in this category ($t_{D,n} = 4$) is not overlapping between the forward read and reverse read at the mutation candidate position and covers no heterozygous SNPs nearby the candidate position. The mutation candidate position is covered by the reverse read.

Suitability of Partitioning-Based Model Integration

We should note that partitioning-based model integration is suited for this problem for two reasons. The first reason is that we can set partitioning rules on paired-end reads and construct generative models for each data set by referring to existing methods. The second reason is that partitioning probabilities $\Pr(t|\mathcal{M}_M), \Pr(t|\mathcal{M}_E)$ are thought to be the same, e. g., the existence of a mutation does not affect whether a paired-end read will cover a heterozygous SNP.

3.4.4 All the Parameters and Hyperparameters in Mutated Data Generation Model

Table 3.2: Notation summary of mutated data generation model

Notation	Type	Meaning
Δ^k	k dimensional non negative simplex	Definition of type
$\boldsymbol{\pi}_H$	Δ^3	Haplotype frequency with variant
π_F	Real number $\in [0,1]$	Reference allele frequency
ϵ_l	Real number $\in [0,1]$	Error rate in overlapping paired-end reads
ϵ_h	Real number $\in [0,1]$	Error rate in hetero SNP covering reads
ϵ_b	Real number $\in [0,1]$	Strand bias rate
π_{HE}	Real number $\in [0,1]$	The haplotype frequency without variant
ϵ_s	Real number $\in [0,1]$	An error rate for unpaired read
$\boldsymbol{\gamma}_H$	$(\mathbb{R}_+, \mathbb{R}_+, \mathbb{R}_+)$	A hyperparameter for $\boldsymbol{\pi}_H$
$\boldsymbol{\gamma}_F$	$(\mathbb{R}_+, \mathbb{R}_+)$	A hyperparameter for $\boldsymbol{\pi}_F$
$\boldsymbol{\alpha}_l$	$(\mathbb{R}_+, \mathbb{R}_+)$	A hyperparameter for ϵ_l
$\boldsymbol{\alpha}_h$	$(\mathbb{R}_+, \mathbb{R}_+)$	A hyperparameter for ϵ_h
$\boldsymbol{\alpha}_b$	$(\mathbb{R}_+, \mathbb{R}_+)$	A hyperparameter for ϵ_b
$\boldsymbol{\gamma}_{\mathrm{HE}}$	$(\mathbb{R}_+, \mathbb{R}_+)$	A hyperparameter for $\boldsymbol{\pi}_{\mathrm{HE}}$
$\boldsymbol{\alpha}_s$	$(\mathbb{R}_+, \mathbb{R}_+)$	A hyperparameter for ϵ_s

3.4.5 All the Parameters and Hyperparameters in Error Data Generation Model

Table 3.3: Notation summary of error data generation model

Notation	Type	Meaning
Δ^k	k dimensional non negative simplex	Definition of type
$\pi_{\mathrm{HE}}, \pi_{T,\mathrm{HE}}, \pi_{N,\mathrm{HE}}$	Real number $\in [0,1]$	Haplotype frequencies with variant
$\epsilon_l, \epsilon_{T,l}, \epsilon_{N,l}$	Real number $\in [0,1]$	Error rates in overlapping paired-end reads
$\epsilon_h, \epsilon_{T,h}, \epsilon_{N,h}$	Real number $\in [0,1]$	Error rates in hetero SNP covering reads
$\epsilon_b, \epsilon_{T,b}, \epsilon_{N,b}$	Real number $\in [0,1]$	Strand bias rate
$\epsilon_s, \epsilon_{T,s}, \epsilon_{N,s}$	Real number $\in [0,1]$	Error rates for unpaired read
$\boldsymbol{\gamma}_{\mathrm{HE}}, \boldsymbol{\gamma}_{T,\mathrm{HE}}, \boldsymbol{\gamma}_{N,\mathrm{HE}}$	$(\mathbb{R}_+, \mathbb{R}_+)$	Hyperparameters for $\boldsymbol{\pi}_{\mathrm{HE}}$
$\boldsymbol{\alpha}_l, \boldsymbol{\alpha}_{T,l}, \boldsymbol{\alpha}_{N,l}$	$(\mathbb{R}_+, \mathbb{R}_+)$	Hyperparameters for ϵ_l
$\boldsymbol{\alpha}_h, \boldsymbol{\alpha}_{T,h}, \boldsymbol{\alpha}_{N,h}$	$(\mathbb{R}_+, \mathbb{R}_+)$	Hyperparameters for ϵ_h
$\boldsymbol{\alpha}_b, \boldsymbol{\alpha}_{T,b}, \boldsymbol{\alpha}_{N,b}$	$(\mathbb{R}_+, \mathbb{R}_+)$	Hyperparameters for ϵ_b
$\boldsymbol{\alpha}_s, \boldsymbol{\alpha}_{T,s}, \boldsymbol{\alpha}_{N,s}$	$(\mathbb{R}_+, \mathbb{R}_+)$	Hyperparameters for ϵ_s

3.4.6 Distribution of Reads

$$\begin{aligned}&\Pr(\boldsymbol{r}_{D,n}|\boldsymbol{z}_{D,n}, \mathcal{H}_{t_{D,n}})\\ &\quad = P_{\mathrm{align}}(r_{D,n,+}|\mathcal{H}_{t_{D,n},\mathrm{idx}(z_{D,n}),+})P_{\mathrm{align}}(r_{D,n,-}|\mathcal{H}_{t_{D,n},\mathrm{idx}(z_{D,n}),-}),\end{aligned}$$

where $P_{\mathrm{align}}(\cdot)$ is the alignment probability which is formulated by profile HMM [1, 77] and $\mathrm{idx}(\cdot)$ is a function that returns the index where the value is 1 from a given one-hot encoding vector.

3.4.7 Distributions of Reads and Latent Variables for Each Partition

O(+)H(-) Category

We prepared $\mathcal{H}_0$ and its frequency as Fig. 3.6. Based on this, we define functions reprepenting the log probability of $\boldsymbol{r}_{D,n}$ and $\boldsymbol{z}_{D,n}$ when $t_{D,n} = 0$ given parameter sets of $\Theta_{M,0}$ and $\Theta_{E,0}$ as follows.

$$
\begin{aligned}
&L_{M,O}(\boldsymbol{r}_{D,n}, \boldsymbol{z}_{D,n}, \pi_F, \epsilon_l, \epsilon_b) \\
&:= \boldsymbol{z}_{D,n,0} \left\{ \ln \pi_F (1-\epsilon_l) + \ln \Pr(\boldsymbol{r}_{D,n} | \mathcal{H}_{0,\mathrm{idx}(\boldsymbol{z}_{D,n,0})}) \right\} \\
&\quad + \boldsymbol{z}_{D,n,1} \left\{ \ln (1-\pi_F) + \ln \Pr(\boldsymbol{r}_{D,n} | \mathcal{H}_{0,\mathrm{idx}(\boldsymbol{z}_{D,n,1})}) \right\} \\
&\quad + \boldsymbol{z}_{D,n,2} \left\{ \ln \pi_F \epsilon_l \epsilon_b + \ln \Pr(\boldsymbol{r}_{D,n} | \mathcal{H}_{0,\mathrm{idx}(\boldsymbol{z}_{D,n,2})}) \right\} \\
&\quad + \boldsymbol{z}_{D,n,3} \left\{ \ln \pi_F \epsilon_l (1-\epsilon_b) + \ln \Pr(\boldsymbol{r}_{D,n} | \mathcal{H}_{0,\mathrm{idx}(\boldsymbol{z}_{D,n,3})}) \right\}, \\
&L_{E,O}(\boldsymbol{r}_{D,n}, z_{D,n}, \epsilon_{\mathrm{le}}, \epsilon_{\mathrm{be}}) \\
&:= z_{D,n,0} \left\{ 2 \ln (1-\epsilon_{\mathrm{le}}) + \ln \Pr(\boldsymbol{r}_{D,n} | \mathcal{H}_{0,\mathrm{idx}(z_{D,n,0})}) \right\} \\
&\quad + z_{D,n,1} \left\{ 2 \ln \epsilon_{\mathrm{le}} + \ln \Pr(\boldsymbol{r}_{D,n} | \mathcal{H}_{0,\mathrm{idx}(z_{D,n,1})}) \right\} \\
&\quad + z_{D,n,2} \left\{ \ln 2\epsilon_{\mathrm{le}} (1-\epsilon_{\mathrm{le}}) \epsilon_{\mathrm{be}} + \ln \Pr(\boldsymbol{r}_{D,n} | \mathcal{H}_{0,\mathrm{idx}(z_{D,n,2})}) \right\} \\
&\quad + z_{D,n,3} \left\{ \ln 2\epsilon_{\mathrm{le}} (1-\epsilon_{\mathrm{le}}) (1-\epsilon_{\mathrm{be}}) + \ln \Pr(\boldsymbol{r}_{D,n} | \mathcal{H}_{0,\mathrm{idx}(z_{D,n,3})}) \right\}, \ D \in \{T, N\}.
\end{aligned}
$$

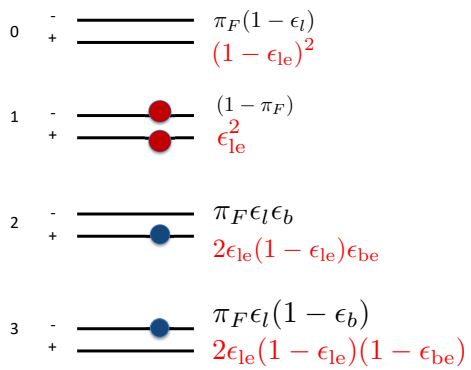


Figure 3.6: A set of paired-end reads in $\mathcal{H}_0$ and corresponding frequencies of $\boldsymbol{z}_{D,n}$ at $t_{D,n} = 0$ for the mutated and error data generation model. Frequencies of $\boldsymbol{z}_{D,n}$ represented by $\Theta_{M,0}$ and $\Theta_{E,0}$ are shown in black (red) letters are for mutated (error) model.

O(-)H(+) Category

We prepared $\mathcal{H}_1$ and its frequency as Fig. 3.7. Based on this, we define functions reprepenting the log probability of $\boldsymbol{r}_{D,n}$ and $\boldsymbol{z}_{D,n}$ when $t_{D,n} = 1$ given parameter sets of $\Theta_{M,1}$ and $\Theta_{E,1}$ as follows.

$$
\begin{aligned}
&L_{M,H}(\boldsymbol{r}_{D,n}, \boldsymbol{z}_{D,n}, \boldsymbol{\pi}_H, \epsilon_h) \\
&:= z_{D,n,0} \left\{ \ln \pi_{H,0} + \ln \Pr(\boldsymbol{r}_{D,n} | \mathcal{H}_{1,\mathrm{idx}(z_{D,n,0})}) \right\} \\
&\quad + z_{D,n,1} \left\{ \ln \pi_{H,1}(1-\epsilon_h) + \ln \Pr(\boldsymbol{r}_{D,n} | \mathcal{H}_{1,\mathrm{idx}(z_{D,n,1})}) \right\} \\
&\quad + z_{D,n,2} \left\{ \ln \pi_{H,2} + \ln \Pr(\boldsymbol{r}_{D,n} | \mathcal{H}_{1,\mathrm{idx}(z_{D,n,2})}) \right\} \\
&\quad + z_{D,n,3} \left\{ \ln \pi_{H,1}\epsilon_h + \ln \Pr(\boldsymbol{r}_{D,n} | \mathcal{H}_{1,\mathrm{idx}(z_{D,n,3})}) \right\}, \\
&L_{E,H}(\boldsymbol{r}_{D,n}, \boldsymbol{z}_{D,n}, \pi_{\mathrm{HE}}, \epsilon_{\mathrm{he}}) \\
&:= z_{D,n,0} \left\{ \ln \pi_{\mathrm{HE},0}(1-\epsilon_{\mathrm{he}}) + \ln \Pr(\boldsymbol{r}_{D,n} | \mathcal{H}_{1,\mathrm{idx}(z_{D,n,0})}) \right\} \\
&\quad + z_{D,n,1} \left\{ \ln \pi_{\mathrm{HE},1}(1-\epsilon_{\mathrm{he}}) + \ln \Pr(\boldsymbol{r}_{D,n} | \mathcal{H}_{1,\mathrm{idx}(z_{D,n,1})}) \right\} \\
&\quad + z_{D,n,2} \left\{ \ln \pi_{\mathrm{HE},0}\epsilon_{\mathrm{he}} + \ln \Pr(\boldsymbol{r}_{D,n} | \mathcal{H}_{1,\mathrm{idx}(z_{D,n,2})}) \right\} \\
&\quad + z_{D,n,3} \left\{ \ln \pi_{\mathrm{HE},1}\epsilon_{\mathrm{he}} + \ln \Pr(\boldsymbol{r}_{D,n} | \mathcal{H}_{1,\mathrm{idx}(z_{D,n,3})}) \right\}, \; D \in \{T, N\}.
\end{aligned}
$$

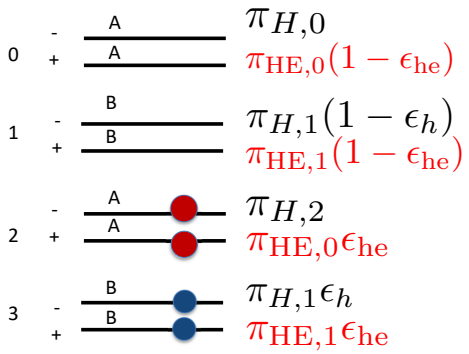


Figure 3.7: A set of paired-end reads in $\mathcal{H}_1$ and corresponding frequencies of $\boldsymbol{z}_{D,n}$ at $t_{D,n} = 1$ for the mutated and error data generation model. Frequencies of $\boldsymbol{z}_{D,n}$ represented by $\Theta_{M,1}$ and $\Theta_{E,1}$ are shown in black (red) letters are for mutated (error) model.

O(+)H(+) Category

We prepared $\mathcal{H}_2$ and its frequency as Fig. 3.8. Based on this, we define functions reprepenting the log probability of $\boldsymbol{r}_{D,n}$ and $\boldsymbol{z}_{D,n}$ when $t_{D,n} = 2$ given parameter sets of $\Theta_{M,2}$ and $\Theta_{E,2}$ as follows.

$$
\begin{aligned}
&L_{M,\mathrm{OH}}(\boldsymbol{r}_{D,n}, \boldsymbol{z}_{D,n}, \pi_H, \epsilon_l, \epsilon_b) \\
&:= z_{D,n,0} \left\{ \ln \pi_{H,0}(1-\epsilon_l) + \ln \Pr(\boldsymbol{r}_{D,n} | \mathcal{H}_{2,\mathrm{idx}(z_{D,n,0})}) \right\} \\
&\quad + z_{D,n,1} \left\{ \ln \pi_{H,1}(1-\epsilon_l)^2 + \ln \Pr(\boldsymbol{r}_{D,n} | \mathcal{H}_{2,\mathrm{idx}(z_{D,n,1})}) \right\} \\
&\quad + z_{D,n,2} \left\{ \ln \pi_{H,2} + \ln \Pr(\boldsymbol{r}_{D,n} | \mathcal{H}_{2,\mathrm{idx}(z_{D,n,2})}) \right\} \\
&\quad + z_{D,n,3} \left\{ \ln \pi_{H,1}\epsilon_l^2 + \ln \Pr(\boldsymbol{r}_{D,n} | \mathcal{H}_{2,\mathrm{idx}(z_{D,n,3})}) \right\} \\
&\quad + z_{D,n,4} \left\{ \ln \pi_{H,0}\epsilon_l\epsilon_b + \ln \Pr(\boldsymbol{r}_{D,n} | \mathcal{H}_{2,\mathrm{idx}(z_{D,n,4})}) \right\} \\
&\quad + z_{D,n,5} \left\{ \ln 2\pi_{H,1}(1-\epsilon_l)\epsilon_l\epsilon_b + \ln \Pr(\boldsymbol{r}_{D,n} | \mathcal{H}_{2,\mathrm{idx}(z_{D,n,5})}) \right\} \\
&\quad + z_{D,n,6} \left\{ \ln \pi_{H,0}\epsilon_l(1-\epsilon_b) + \ln \Pr(\boldsymbol{r}_{D,n} | \mathcal{H}_{2,\mathrm{idx}(z_{D,n,6})}) \right\} \\
&\quad + z_{D,n,7} \left\{ \ln 2\pi_{H,1}\epsilon_l(1-\epsilon_l)(1-\epsilon_b) + \ln \Pr(\boldsymbol{r}_{D,n} | \mathcal{H}_{2,\mathrm{idx}(z_{D,n,7})}) \right\}, \\
&L_{E,\mathrm{OH}}(\boldsymbol{r}_{D,n}, \boldsymbol{z}_{D,n}, \pi_{\mathrm{HE}}, \epsilon_{\mathrm{le}}, \epsilon_{\mathrm{be}}) \\
&:= z_{D,n,0} \left\{ \ln \pi_{\mathrm{HE},0}(1-\epsilon_{\mathrm{le}})^2 + \ln \Pr(\boldsymbol{r}_{D,n} | \mathcal{H}_{2,\mathrm{idx}(z_{D,n,0})}) \right\} \\
&\quad + z_{D,n,1} \left\{ \ln \pi_{\mathrm{HE},1}(1-\epsilon_{\mathrm{le}})^2 + \ln \Pr(\boldsymbol{r}_{D,n} | \mathcal{H}_{2,\mathrm{idx}(z_{D,n,1})}) \right\} \\
&\quad + z_{D,n,2} \left\{ \ln \pi_{\mathrm{HE},0}\epsilon_{\mathrm{le}}^2 + \ln \Pr(\boldsymbol{r}_{D,n} | \mathcal{H}_{2,\mathrm{idx}(z_{D,n,2})}) \right\} \\
&\quad + z_{D,n,3} \left\{ \ln \pi_{\mathrm{HE},1}\epsilon_{\mathrm{le}}^2 + \ln \Pr(\boldsymbol{r}_{D,n} | \mathcal{H}_{2,\mathrm{idx}(z_{D,n,3})}) \right\} \\
&\quad + z_{D,n,4} \left\{ \ln 2\pi_{\mathrm{HE},0}(1-\epsilon_{\mathrm{le}})\epsilon_{\mathrm{le}}\epsilon_{\mathrm{be}} + \ln \Pr(\boldsymbol{r}_{D,n} | \mathcal{H}_{2,\mathrm{idx}(z_{D,n,4})}) \right\} \\
&\quad + z_{D,n,5} \left\{ \ln 2\pi_{\mathrm{HE},1}(1-\epsilon_{\mathrm{le}})\epsilon_{\mathrm{le}}\epsilon_{\mathrm{be}} + \ln \Pr(\boldsymbol{r}_{D,n} | \mathcal{H}_{2,\mathrm{idx}(z_{D,n,5})}) \right\} \\
&\quad + z_{D,n,6} \left\{ \ln 2\pi_{\mathrm{HE},0}\epsilon_{\mathrm{le}}(1-\epsilon_{\mathrm{le}})(1-\epsilon_{\mathrm{be}}) + \ln \Pr(\boldsymbol{r}_{D,n} | \mathcal{H}_{2,\mathrm{idx}(z_{D,n,6})}) \right\} \\
&\quad + z_{D,n,7} \left\{ \ln 2\pi_{\mathrm{HE},1}\epsilon_{\mathrm{le}}(1-\epsilon_{\mathrm{le}})(1-\epsilon_{\mathrm{be}}) + \ln \Pr(\boldsymbol{r}_{D,n} | \mathcal{H}_{2,\mathrm{idx}(z_{D,n,7})}) \right\}, \; D \in \{T, N\}.
\end{aligned}
$$

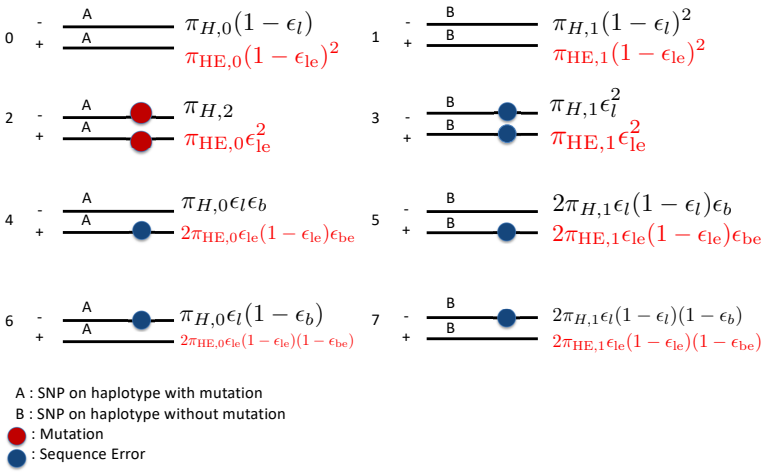


Figure 3.8: A set of paired-end reads in $\mathcal{H}_2$ and corresponding frequencies of $\boldsymbol{z}_{D,n}$ at $t_{D,n} = 2$ for the mutated and error data generation model. Frequencies of $\boldsymbol{z}_{D,n}$ represented by $\Theta_{M,2}$ and $\Theta_{E,2}$ are shown in black (red) letters are for mutated (error) model.

O(-)H(-)S(+) Category

We prepared $\mathcal{H}_3$ and its frequency as Fig. 3.9. Based on this, we define functions reprepenting the log probability of $\boldsymbol{r}_{D,n}$ and $\boldsymbol{z}_{D,n}$ when $t_{D,n} = 3$ given parameter sets of $\Theta_{M,3}$ and $\Theta_{E,3}$ as follows.

$$
\begin{aligned}
&L_{M,P}(\boldsymbol{r}_{D,n}, \boldsymbol{z}_{D,n}, \pi_F, \epsilon_b) \\
&:= z_{D,n,0} \left\{ \ln \pi_F + \ln \Pr(\boldsymbol{r}_{D,n} | \mathcal{H}_{3,\mathrm{idx}(z_{D,n,0})}) \right\} \\
&\quad + z_{D,n,1} \left\{ \ln(1-\pi_F)(1-\epsilon_b) + \ln \Pr(\boldsymbol{r}_{D,n} | \mathcal{H}_{3,\mathrm{idx}(z_{D,n,1})}) \right\} \\
&\quad + z_{D,n,2} \left\{ \ln(1-\pi_F)\epsilon_b + \ln \Pr(\boldsymbol{r}_{D,n} | \mathcal{H}_{3,\mathrm{idx}(z_{D,n,2})}) \right\}, \\
&L_{E,P}(r_{D,n}, \boldsymbol{z}_{D,n}, \epsilon_S, \epsilon_{\mathrm{be}}) \\
&:= z_{D,n,0} \left\{ \ln(1-\epsilon_S) + \ln \Pr(\boldsymbol{r}_{D,n} | \mathcal{H}_{3,\mathrm{idx}(z_{D,n,0})}) \right\} \\
&\quad + z_{D,n,1} \left\{ \ln \epsilon_S(1-\epsilon_{\mathrm{be}}) + \ln \Pr(\boldsymbol{r}_{D,n} | \mathcal{H}_{3,\mathrm{idx}(z_{D,n,1})}) \right\} \\
&\quad + z_{D,n,2} \left\{ \ln \epsilon_S \epsilon_{\mathrm{be}} + \ln \Pr(\boldsymbol{r}_{D,n} | \mathcal{H}_{3,\mathrm{idx}(z_{D,n,2})}) \right\}, \; D \in \{T, N\}.
\end{aligned}
$$

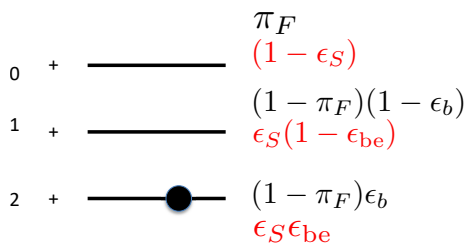


Figure 3.9: A set of paired-end reads in $\mathcal{H}_3$ and corresponding frequencies of $\boldsymbol{z}_{D,n}$ at $t_{D,n} = 3$ for the mutated and error data generation model. Frequencies of $\boldsymbol{z}_{D,n}$ represented by $\Theta_{M,3}$ and $\Theta_{E,3}$ are shown in black (red) letters are for mutated (error) model.

O(-)H(-)S(-) Category

We prepared $\mathcal{H}_4$ and its frequency as Fig. 3.10. Based on this, we define functions reprepenting the log probability of $\boldsymbol{r}_{D,n}$ and $\boldsymbol{z}_{D,n}$ when $t_{D,n} = 4$ given parameter sets of $\Theta_{M,4}$ and $\Theta_{E,4}$ as follows.

$$
\begin{aligned}
&L_{M,M}(\boldsymbol{r}_{D,n}, \boldsymbol{z}_{D,n}, \pi_F, \epsilon_b) \\
&:= z_{D,n,0} \left\{ \ln \pi_F + \ln \Pr(\boldsymbol{r}_{D,n} | \mathcal{H}_{4,\mathrm{idx}(z_{D,n,0})}) \right\} \\
&\quad + z_{D,n,1} \left\{ \ln(1-\pi_F)\epsilon_b + \ln \Pr(\boldsymbol{r}_{D,n} | \mathcal{H}_{4,\mathrm{idx}(z_{D,n,1})}) \right\} \\
&\quad + z_{D,n,2} \left\{ \ln(1-\pi_F)(1-\epsilon_b) + \ln \Pr(\boldsymbol{r}_{D,n} | \mathcal{H}_{4,\mathrm{idx}(z_{D,n,2})}) \right\}, \\
&L_{E,M}(\boldsymbol{r}_{D,n}, \boldsymbol{z}_{D,n}, \epsilon_S, \epsilon_{\mathrm{be}}) \\
&:= z_{D,n,0} \left\{ \ln(1-\epsilon_S) + \ln \Pr(\boldsymbol{r}_{D,n} | \mathcal{H}_{4,\mathrm{idx}(z_{D,n,0})}) \right\} \\
&\quad + z_{D,n,1} \left\{ \ln \epsilon_S \epsilon_{\mathrm{be}} + \ln \Pr(\boldsymbol{r}_{D,n} | \mathcal{H}_{4,\mathrm{idx}(z_{D,n,1})}) \right\} \\
&\quad + z_{D,n,2} \left\{ \ln \epsilon_S(1-\epsilon_{\mathrm{be}}) + \ln \Pr(\boldsymbol{r}_{D,n} | \mathcal{H}_{4,\mathrm{idx}(z_{D,n,2})}) \right\}, \ D \in \{T, N\}.
\end{aligned}
$$

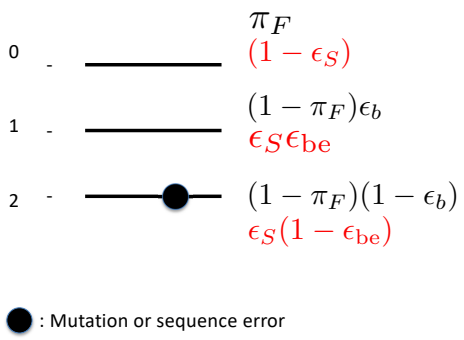


Figure 3.10: A set of paired-end reads in $\mathcal{H}_4$ and corresponding frequencies of $\boldsymbol{z}_{D,n}$ at $t_{D,n} = 4$ for the mutated and error data generation model. Frequencies of $\boldsymbol{z}_{D,n}$ represented by $\Theta_{M,4}$ and $\Theta_{E,4}$ are shown in black (red) letters are for mutated (error) model.

3.4.8 Joint Probability for Mutated Data Generation Model

We set the joint probability for the mutated data generation model as follows.

$$
\begin{aligned}
&\ln \Pr(\mathcal{R}_{\mathrm{NT}}, \mathcal{T}_{\mathrm{NT}}, \mathcal{Z}_{\mathrm{NT}}, \Theta_{M,\mathrm{all}} | \mathcal{M}_M) \\
&= \ln \Pr(\mathcal{T}_{\mathrm{NT}} | \mathcal{M}_M) + \ln \Pr(\mathcal{R}_{\mathrm{NT}}, \mathcal{Z}_{\mathrm{NT}}, \Theta_{M,\mathrm{all}} | \mathcal{T}_{\mathrm{NT}}, \mathcal{M}_M) \\
&= \ln \Pr(\mathcal{T}_{\mathrm{NT}} | \mathcal{M}_M) \\
&\quad + \ln P_{\mathrm{beta}}(\boldsymbol{\pi}_{\mathrm{HE}} | \boldsymbol{\gamma}_{\mathrm{HE}}) + \ln P_{\mathrm{dir}}(\boldsymbol{\pi}_H | \boldsymbol{\gamma}_H) + \ln P_{\mathrm{beta}}(\boldsymbol{\pi}_F | \boldsymbol{\gamma}_F) \\
&\quad + \ln P_{\mathrm{beta}}(\epsilon_s | \boldsymbol{\alpha}_s) + \ln P_{\mathrm{beta}}(\epsilon_l | \boldsymbol{\alpha}_l) + \ln P_{\mathrm{beta}}(\epsilon_h | \boldsymbol{\alpha}_h) + \ln P_{\mathrm{beta}}(\epsilon_b | \boldsymbol{\alpha}_b) \\
&\quad + \sum_{n | t_{T,n}=0} L_{M,O}(\boldsymbol{r}_{T,n}, \boldsymbol{z}_{T,n}, \pi_F, \epsilon_l, \epsilon_b) \\
&\quad + \sum_{n | t_{T,n}=1} L_{M,H}(\boldsymbol{r}_{T,n}, \boldsymbol{z}_{T,n}, \boldsymbol{\pi}_H, \epsilon_h) \\
&\quad + \sum_{n | t_{T,n}=2} L_{M,\mathrm{OH}}(\boldsymbol{r}_{T,n}, \boldsymbol{z}_{T,n}, \pi_H, \epsilon_l, \epsilon_b) \\
&\quad + \sum_{n | t_{T,n}=3} L_{M,P}(\boldsymbol{r}_{T,n}, \boldsymbol{z}_{T,n}, \pi_F, \epsilon_b) \\
&\quad + \sum_{n | t_{T,n}=4} L_{M,M}(\boldsymbol{r}_{T,n}, \boldsymbol{z}_{T,n}, \pi_F, \epsilon_b) \\
&\quad + \sum_{n | t_{N,n}=0} L_{E,O}(\boldsymbol{r}_{N,n}, \boldsymbol{z}_{N,n}, \epsilon_l, \epsilon_b) \\
&\quad + \sum_{n | t_{N,n}=1} L_{E,H}(\boldsymbol{r}_{N,n}, \boldsymbol{z}_{N,n}, \pi_{\mathrm{HE}}, \epsilon_h) \\
&\quad + \sum_{n | t_{N,n}=2} L_{E,\mathrm{OH}}(\boldsymbol{r}_{N,n}, \boldsymbol{z}_{N,n}, \pi_{\mathrm{HE}}, \epsilon_l, \epsilon_b) \\
&\quad + \sum_{n | t_{N,n}=3} L_{E,P}(\boldsymbol{r}_{N,n}, \boldsymbol{z}_{N,n}, \epsilon_s, \epsilon_b) \\
&\quad + \sum_{n | t_{N,n}=4} L_{E,M}(\boldsymbol{r}_{N,n}, \boldsymbol{z}_{N,n}, \epsilon_s, \epsilon_b).
\end{aligned}
$$

3.4.9 Lower Bound for Marginal Likelihood in Mutated Data Generation Model

The marginal likelihood required for computing the Bayes factor is $\Pr(\mathcal{R}_{\mathrm{NT}} | \mathcal{M}_M, \mathcal{T}_{\mathrm{NT}})$ because we set $\Pr(\mathcal{T}_{\mathrm{NT}} | \mathcal{M}_M) = \Pr(\mathcal{T}_{\mathrm{NT}} | \mathcal{M}_E)$. This marginal likelihood can be lower bounded by Jensens' inequality in the same manner as variational Bayes.

$$
\begin{aligned}
&\ln \Pr(\mathcal{R}_{\mathrm{NT}} | \mathcal{M}_M, \mathcal{T}_{\mathrm{NT}}) \\
&\geq E_q \left[\ln \frac{\Pr(\mathcal{R}_{\mathrm{NT}}, \mathcal{Z}_{\mathrm{NT}}, \Theta_{M,\mathrm{all}} | \mathcal{M}_S, \mathcal{T}_{\mathrm{NT}})}{q(\mathcal{Z}_{\mathrm{NT}}, \Theta_{M,\mathrm{all}})} \right] =: \mathcal{L}_M(q),
\end{aligned}
$$

where $q(\Theta_{M,\mathrm{all}}, \mathcal{Z}_{\mathrm{NT}})$ is a free distribution for $\Theta_{M,\mathrm{all}}, \mathcal{Z}_{\mathrm{NT}}$.

We approximately evaluate the marginal likelihood by $\mathcal{L}_M(q)$, which is maximized through the following procedures of variational Bayes.

3.4.10 Assumptions on Free Distributions

We assume the following form of free distributions as follows.

$$
\begin{aligned}
q(\mathcal{Z}_{\mathrm{NT}}, \Theta_{M,\mathrm{all}}) &:= q(\mathcal{Z}_{\mathrm{NT}}) q(\Theta_{M,\mathrm{all}}), \\
q(\mathcal{Z}_{\mathrm{NT}}) &:= \prod_{D \in \{T,N\}} \prod_{n=1}^{d_D} q(z_{D,n}), \\
q(\Theta_{M,\mathrm{all}}) &:= q(\boldsymbol{\pi}_H) q(\pi_F) q(\epsilon_l) q(\epsilon_h) q(\epsilon_b) q(\pi_{\mathrm{HE}}) q(\epsilon_s).
\end{aligned}
$$

3.4.11 Maximize $\mathcal{L}_M(q)$ w.r.t. $q(\boldsymbol{\pi}_H)$

We would like to get optimal $q^*(\boldsymbol{\pi}_H)$ which maximize $\mathcal{L}_T(q)$ with respect to $q(\boldsymbol{\pi}_H)$. For simplicity, we define $z^{(k)}_{D,n,i}$ for $D \in \{T, N\}$ as follows:

$$
z^{(k)}_{D,n,i} := \begin{cases} z_{D,n,i} & (t_{D,n} = k) \\ 0 & (\text{otherwise}) \end{cases}.
$$

Then, the lower bound can be written as follows.

$$
\begin{aligned}
&\mathcal{L}_M(q) \\
&= E_q \left[\left\{ (\gamma_{H,0} - 1) + \sum_n \left\{ z^{(1)}_{T,n,0} + z^{(2)}_{T,n,0} + z^{(2)}_{T,n,4} + z^{(2)}_{T,n,6} \right\} \right\} \ln \pi_{H,0} \right] \\
&\quad + E_q \left[\left\{ (\gamma_{H,1} - 1) + \sum_n \left\{ z^{(1)}_{T,n,5} + z^{(1)}_{T,n,4} + z^{(2)}_{T,n,1} + z^{(2)}_{T,n,3} + z^{(2)}_{T,n,5} + z^{(2)}_{T,n,7} \right\} \right\} \ln \pi_{H,1} \right] \\
&\quad + E_q \left[\left\{ (\gamma_{H,2} - 1) + \sum_n \left\{ z^{(1)}_{T,n,2} + z^{(2)}_{T,n,2} \right\} \right\} \ln \pi_{H,2} \right] \\
&\quad - E_q \left[\ln q(\boldsymbol{\pi}_H) \right] + \mathrm{const} \\
&= -\mathrm{KL}[q(\boldsymbol{\pi}_H) || P_{\mathrm{dir}}(\boldsymbol{\pi}_H | \boldsymbol{\gamma}^*_H)] + \mathrm{const},
\end{aligned}
$$

where

$$
\begin{aligned}
\gamma^*_{H,0} &= E_q \left[(\gamma_{H,0} - 1) + \sum_n \left\{ z^{(1)}_{T,n,0} + z^{(2)}_{T,n,0} + z^{(2)}_{T,n,4} + z^{(2)}_{T,n,6} \right\} \right], \\
\gamma^*_{H,1} &= E_q \left[(\gamma_{H,1} - 1) + \sum_n \left\{ z^{(1)}_{T,n,5} + z^{(1)}_{T,n,4} + z^{(2)}_{T,n,1} + z^{(2)}_{T,n,3} + z^{(2)}_{T,n,5} + z^{(2)}_{T,n,7} \right\} \right], \\
\gamma^*_{H,2} &= E_q \left[(\gamma_{H,2} - 1) + \sum_n \left\{ z^{(1)}_{T,n,2} + z^{(2)}_{T,n,2} \right\} \right].
\end{aligned}
$$

Therefore, we can maximize the lower bound by minimization of KL divergence of $\mathrm{KL}[q(\boldsymbol{\pi}_H) || P_{\mathrm{dir}}(\boldsymbol{\pi}_H | \boldsymbol{\gamma}^*_H)] \geq 0$. The optimal form distribution is

$$
q(\boldsymbol{\pi}_H) = P_{\mathrm{dir}}(\boldsymbol{\pi}_H | \boldsymbol{\gamma}^*_H).
$$

3.4.12 Maximize $\mathcal{L}_M(q)$ w.r.t. $q(\boldsymbol{\pi}_F)$

We would like to get optimal $q^*(\pi_F)$ which maximize $\mathcal{L}_T(q)$ with respect to $q(\pi_F)$.

$$
\begin{aligned}
&\mathcal{L}_M(q) \\
&= E_q\left[\left\{(\gamma_{F,0}-1)+\sum_n\left\{z^{(0)}_{T,n,0}+z^{(0)}_{T,n,2}+z^{(0)}_{T,n,3}+z^{(3)}_{T,n,0}+z^{(4)}_{T,n,0}\right\}\right\}\ln\pi_F\right] \\
&\quad + E_q\left[\left\{(\gamma_{F,1}-1)+\sum_n\left\{z^{(0)}_{T,n,1}+z^{(3)}_{T,n,1}+z^{(3)}_{T,n,2}+z^{(4)}_{T,n,1}+z^{(4)}_{T,n,2}\right\}\right\}\ln(1-\pi_F)\right] \\
&\quad - E_q\left[\ln q(\boldsymbol{\pi}_F)\right] + \text{const} \\
&= -\mathrm{KL}[q(\pi_F)||P_{\text{beta}}(\pi_F|\boldsymbol{\gamma}^*_F)] + \text{const},
\end{aligned}
$$

where

$$
\begin{aligned}
\gamma^*_{F,0} &= E_q\left[(\gamma_{F,0}-1)+\sum_n\left\{z^{(0)}_{T,n,0}+z^{(0)}_{T,n,2}+z^{(0)}_{T,n,3}+z^{(3)}_{T,n,0}+z^{(4)}_{T,n,0}\right\}\right], \\
\gamma^*_{F,1} &= E_q\left[(\gamma_{F,1}-1)+\sum_n\left\{z^{(0)}_{T,n,1}+z^{(3)}_{T,n,1}+z^{(3)}_{T,n,2}+z^{(4)}_{T,n,1}+z^{(4)}_{T,n,2}\right\}\right].
\end{aligned}
$$

Therefore, we can maximize the lower bound by minimization of KL divergence of $\mathrm{KL}[q(\pi_H)||P_{\text{beta}}(\pi_F|\boldsymbol{\gamma}^*_F)] \geq 0$. The optimal form distribution is

$$
q^*(\pi_F) = P_{\text{beta}}(\pi_F|\boldsymbol{\gamma}^*_F).
$$

3.4.13 Maximize $\mathcal{L}_M(q)$ w.r.t. $q(\epsilon_l)$

We would like to get optimal $q^*(\epsilon_l)$ which maximize $\mathcal{L}_T(q)$ with respect to $q(\epsilon_l)$.

$$
\begin{aligned}
&\mathcal{L}_M(q) \\
&= E_q\left[\{(\alpha_{l,0}-1)\}\ln\epsilon_l\right] \\
&\quad + E_q\left[\left\{\sum_n\left\{z^{(0)}_{T,n,2}+z^{(0)}_{T,n,3}+2z^{(2)}_{T,n,3}+z^{(2)}_{T,n,4}+z^{(2)}_{T,n,5}+z^{(2)}_{T,n,6}+z^{(2)}_{T,n,7}\right\}\right\}\ln\epsilon_l\right] \\
&\quad + E_q\left[\sum_n\left\{2z^{(0)}_{N,n,1}+z^{(0)}_{N,n,2}+z^{(0)}_{N,n,3}\right.\right. \\
&\qquad \left.\left.+2z^{(2)}_{N,n,2}+2z^{(2)}_{N,n,3}+z^{(2)}_{N,n,4}+z^{(2)}_{N,n,5}+z^{(2)}_{N,n,6}+z^{(2)}_{N,n,7}\right\}\ln\epsilon_l\right] \\
&\quad + E_q\left[\{(\alpha_{l,1}-1)\}\ln(1-\epsilon_l)\right] \\
&\quad + E_q\left[\sum_n\left\{z^{(0)}_{T,n,0}+z^{(2)}_{T,n,0}+2z^{(2)}_{T,n,1}+z^{(2)}_{T,n,5}+z^{(2)}_{T,n,7}\right\}\ln(1-\epsilon_l)\right] \\
&\quad + E_q\left[\sum_n\left\{2z^{(0)}_{N,n,0}+z^{(0)}_{N,n,2}+z^{(0)}_{N,n,3}\right.\right. \\
&\qquad \left.\left.+2z^{(2)}_{N,n,0}+2z^{(2)}_{N,n,1}+z^{(2)}_{N,n,4}+z^{(2)}_{N,n,5}+z^{(2)}_{N,n,6}+z^{(2)}_{N,n,7}\right\}\ln(1-\epsilon_l)\right] \\
&\quad - E_q\left[\ln q(\boldsymbol{\epsilon}_l)\right] + \text{const} \\
&= -\mathrm{KL}[q(\epsilon_l)||P_{\text{beta}}(\epsilon_l|\boldsymbol{\alpha}^*_l)] + \text{const},
\end{aligned}
$$

where

$$\begin{aligned}
\alpha^*_{l,0} &= (\alpha_{h,0}-1) + \sum_n E_q\left[z^{(0)}_{T,n,2} + z^{(0)}_{T,n,3} + 2z^{(2)}_{T,n,3}\right] \\
&\quad + \sum_n E_q\left[z^{(2)}_{T,n,4} + z^{(2)}_{T,n,5} + z^{(2)}_{T,n,6} + z^{(2)}_{T,n,7}\right] \\
&\quad + \sum_n E_q\left[2z^{(0)}_{N,n,1} + z^{(0)}_{N,n,2} + z^{(0)}_{N,n,3}\right] \\
&\quad + \sum_n E_q\left[2z^{(2)}_{N,n,2} + 2z^{(2)}_{N,n,3} + z^{(2)}_{N,n,4} + z^{(2)}_{N,n,5} + z^{(2)}_{N,n,6}\right], \\
\alpha^*_{l,1} &= (\alpha_{h,1}-1) + \sum_n E_q\left[z^{(0)}_{T,n,0} + z^{(2)}_{T,n,0} + 2z^{(2)}_{T,n,1} + z^{(2)}_{T,n,5} + z^{(2)}_{T,n,7}\right] \\
&\quad + \sum_n E_q\left[z^{(2)}_{T,n,4} + z^{(2)}_{T,n,5} + z^{(2)}_{T,n,6} + z^{(2)}_{T,n,7}\right] \\
&\quad + \sum_n E_q\left[2z^{(0)}_{N,n,0} + z^{(0)}_{N,n,2} + z^{(0)}_{N,n,3}\right] \\
&\quad + \sum_n E_q\left[2z^{(2)}_{N,n,0} + 2z^{(2)}_{N,n,1} + z^{(2)}_{N,n,4} + z^{(2)}_{N,n,5} + z^{(2)}_{N,n,6} + z^{(2)}_{N,n,7}\right].
\end{aligned}$$

Therefore, we can maximize the lower bound by minimization of KL divergence of $\mathrm{KL}[q(\epsilon_l)||P_{\mathrm{beta}}(\epsilon_l|\boldsymbol{\alpha}^*_l)] \geq 0$. The optimal form distribution is

$$q^*(\epsilon_l) = P_{\mathrm{beta}}(\epsilon_l|\boldsymbol{\alpha}^*_l).$$

3.4.14 Maximize $\mathcal{L}_M(q)$ w.r.t. $q(\epsilon_h)$

We would like to get optimal $q^*(\epsilon_h)$ which maximize $\mathcal{L}_M(q)$ with respect to $q(\epsilon_h)$.

$$\begin{aligned}
\mathcal{L}_M(q) &= E_q\left[\left\{(\alpha_{l,0}-1) + \sum_n \left\{z^{(1)}_{T,n,3}\right\}\right\}\ln\epsilon_h\right] \\
&\quad + E_q\left[\sum_n \left\{z^{(1)}_{N,n,2} + z^{(1)}_{N,n,3}\right\}\ln\epsilon_h\right] \\
&\quad + E_q\left[\left\{(\alpha_{l,1}-1) + \sum_n z^{(1)}_{T,n,1}\right\}\ln(1-\epsilon_h)\right] \\
&\quad + E_q\left[\sum_n \left\{z^{(1)}_{N,n,0} + z^{(1)}_{N,n,1}\right\}\ln(1-\epsilon_h)\right] \\
&\quad - E_q\left[\ln q(\boldsymbol{\epsilon}_h)\right] + \mathrm{const} \\
&= -\mathrm{KL}[q(\epsilon_h)||P_{\mathrm{beta}}(\epsilon_h|\boldsymbol{\alpha}^*_h)] + \mathrm{const},
\end{aligned}$$

where

$$\begin{aligned}
\alpha^*_{h,0} &= E_q\left[(\alpha_{h,0}-1) + \sum_n z^{(1)}_{T,n,3} + \sum_n \left\{z^{(1)}_{N,n,2} + z^{(1)}_{N,n,3}\right\}\right], \\
\alpha^*_{h,1} &= E_q\left[(\alpha_{h,1}-1) + \sum_n z^{(1)}_{T,n,1} + \sum_n \left\{z^{(1)}_{N,n,0} + z^{(1)}_{N,n,1}\right\}\right].
\end{aligned}$$

Therefore, we can maximize the lower bound by minimization of KL divergence of $\mathrm{KL}[q(\epsilon_h)||P_{\mathrm{beta}}(\epsilon_h|\boldsymbol{\alpha}^*_h)]$. The optimal form distribution is

$$q^*(\epsilon_h) = P_{\mathrm{beta}}(\epsilon_h|\boldsymbol{\alpha}^*_h).$$

3.4.15 Maximize $\mathcal{L}_M(q)$ w.r.t. $q(\epsilon_b)$

We would like to get optimal $q^*(\epsilon_b)$ which maximize $\mathcal{L}_M(q)$ with respect to $q(\epsilon_b)$.

$$\begin{aligned}
\mathcal{L}_M(q) &= E_q\left[\left\{(\alpha_{b,0}-1)+\sum_D\sum_n\left\{z^{(0)}_{D,n,2}+z^{(2)}_{D,n,4}+z^{(2)}_{D,n,5}+z^{(3)}_{D,n,2}+z^{(4)}_{D,n,1}\right\}\right\}\ln\epsilon_b\right] \\
&\quad + E_q\left[\left\{(\alpha_{b,1}-1)+\sum_D\sum_n\left\{z^{(0)}_{D,n,3}+z^{(2)}_{D,n,6}+z^{(2)}_{D,n,7}+z^{(3)}_{D,n,1}+z^{(4)}_{D,n,2}\right\}\right\}\ln(1-\epsilon_b)\right] \\
&\quad - E_q\left[\ln q(\boldsymbol{\epsilon}_b)\right]+\text{const} \\
&= -\text{KL}[q(\epsilon_b)||P_{\text{beta}}(\epsilon_b|\boldsymbol{\alpha}^*_b)]+\text{const},
\end{aligned}$$

where

$$\begin{aligned}
\alpha^*_{b,0} &= E_q\left[(\alpha_{b,0}-1)+\sum_D\sum_n\left\{z^{(0)}_{D,n,2}+z^{(2)}_{D,n,4}+z^{(2)}_{D,n,5}+z^{(3)}_{D,n,2}+z^{(4)}_{D,n,1}\right\}\right], \\
\alpha^*_{b,1} &= E_q\left[(\alpha_{b,1}-1)+\sum_D\sum_n\left\{z^{(0)}_{D,n,3}+z^{(2)}_{D,n,6}+z^{(2)}_{D,n,7}+z^{(3)}_{D,n,1}+z^{(4)}_{D,n,2}\right\}\right].
\end{aligned}$$

Therefore, we can maximize the lower bound by minimization of KL divergence of $\text{KL}[q(\epsilon_b)||P_{\text{beta}}(\epsilon_b|\boldsymbol{\alpha}^*_b)] \geq 0$. The optimal form distribution is

$$q^*(\epsilon_b) = P_{\text{beta}}(\epsilon_b|\boldsymbol{\alpha}^*_b).$$

3.4.16 Maximize $\mathcal{L}_M(q)$ w.r.t. $q(\boldsymbol{z}_{D,n})$

Updating procedure for $q(\boldsymbol{z}_{D,n})$ is dependent on the value of $t_{D,n}$ and $D \in \{T, N\}$. We only show the updating procedure for $t_{D,n} = 0$ and $D = T$.

$$\begin{aligned}
\mathcal{L}_M(q) &= E_q\left[z_{T,n,0}\right]E_q\left[\ln\pi_F(1-\epsilon_l)+\ln\Pr(\boldsymbol{r}_{T,n}|\mathcal{H}_{0,\text{idx}(z_{T,n,0})})\right] \\
&\quad + E_q\left[z_{T,n,1}\right]E_q\left[\ln(1-\pi_F)+\ln\Pr(\boldsymbol{r}_{T,n}|\mathcal{H}_{0,\text{idx}(z_{T,n,1})})\right] \\
&\quad + E_q\left[z_{T,n,2}\right]E_q\left[\ln\pi_F\epsilon_l\epsilon_b+\ln\Pr(\boldsymbol{r}_{T,n}|\mathcal{H}_{0,\text{idx}(z_{T,n,2})})\right] \\
&\quad + E_q\left[z_{T,n,3}\right]E_q\left[\ln\pi_F\epsilon_l(1-\epsilon_b)+\ln\Pr(\boldsymbol{r}_{T,n}|\mathcal{H}_{0,\text{idx}(z_{T,n,3})})\right] \\
&\quad - E_q\left[\ln q(\boldsymbol{z}_{T,n})\right]+\text{const} \\
&= E_q\left[z_{T,n,0}\right]E_q\left[\ln\rho^*_{T,n,0}\right]+E_q\left[z_{T,n,1}\right]E_q\left[\ln\rho^*_{T,n,1}\right] \\
&\quad + E_q\left[z_{T,n,2}\right]E_q\left[\ln\rho^*_{T,n,2}\right]+E_q\left[z_{T,n,3}\right]E_q\left[\ln\rho^*_{T,n,3}\right] \\
&\quad - E_q\left[\ln q(\boldsymbol{z}_{T,n})\right]+\text{const} \\
&= -\text{KL}[q(\boldsymbol{z}_{T,n})||P_{\text{multi}}(\boldsymbol{z}_{T,n}|\boldsymbol{\zeta}^*_{T,n})]+\text{const},
\end{aligned}$$

where

$$\begin{aligned}
\rho^*_{T,n,0} &= E_q\left[\ln \pi_F(1-\epsilon_l) + \ln \Pr(\boldsymbol{r}_{T,n}|\mathcal{H}_{0,\mathrm{idx}(z_{T,n,0})})\right], \\
\rho^*_{T,n,1} &= E_q\left[\ln(1-\pi_F) + \ln \Pr(\boldsymbol{r}_{T,n}|\mathcal{H}_{0,\mathrm{idx}(z_{T,n,1})})\right], \\
\rho^*_{T,n,2} &= E_q\left[\ln \pi_F\epsilon_l\epsilon_b + \ln \Pr(\boldsymbol{r}_{T,n}|\mathcal{H}_{0,\mathrm{idx}(z_{T,n,2})})\right], \\
\rho^*_{T,n,3} &= E_q\left[\ln \pi_F\epsilon_l(1-\epsilon_b) + \ln \Pr(\boldsymbol{r}_{T,n}|\mathcal{H}_{0,\mathrm{idx}(z_{T,n,3})})\right], \\
\zeta^*_{T,n,j} &\propto \rho^*_{T,n,j}, \\
\sum_{j=0}^{3} \zeta^*_{T,n,j} &= 1.
\end{aligned}$$

Therefore, we can maximize the lower bound by minimization of KL divergence of $\mathrm{KL}[q(\boldsymbol{z}_{T,n})||P_{\mathrm{multi}}(\boldsymbol{z}_{T,n}|\boldsymbol{\zeta}^*_{T,n})] \geq 0$. When $t_{T,n} = 0$, the optimal form distribution is

$$q(\boldsymbol{z}_{T,n}) = P_{\mathrm{multi}}(\boldsymbol{z}_{T,n}|\boldsymbol{\zeta}^*_{T,n}).$$

3.4.17 Joint Probability for Error Data Generation Model

We set the joint probability for the error data generation model as follows.

$$\begin{aligned}
&\ln \Pr(\mathcal{R}_{\mathrm{NT}}, \mathcal{Z}_{\mathrm{NT}}, \Theta_{E,\mathrm{all}}, \mathcal{T}_{\mathrm{NT}}|\mathcal{M}_E) \\
&= \ln \Pr(\mathcal{T}_{\mathrm{NT}}|\mathcal{M}_E) + \ln \Pr(\mathcal{R}_{\mathrm{NT}}, \mathcal{Z}_{\mathrm{NT}}|\mathcal{T}_{\mathrm{NT}}, \mathcal{M}_E) \\
&= \ln \Pr(\mathcal{T}_{\mathrm{NT}}|\mathcal{M}_E) \\
&\quad + \ln P_{\mathrm{beta}}(\boldsymbol{\pi}_{\mathrm{HE}}|\boldsymbol{\gamma}_{\mathrm{HE}}) + \ln P_{\mathrm{beta}}(\epsilon_s|\boldsymbol{\alpha}_s) \\
&\quad + \ln P_{\mathrm{beta}}(\epsilon_l|\boldsymbol{\alpha}_l) + \ln P_{\mathrm{beta}}(\epsilon_h|\boldsymbol{\alpha}_h) + \ln P_{\mathrm{beta}}(\epsilon_b|\boldsymbol{\alpha}_b) \\
&\quad + \sum_{n|t_{T,n}=0} L_{E,O}(\boldsymbol{r}_{T,n}, \boldsymbol{z}_{T,n}, \epsilon_l, \epsilon_b) + \sum_{n|t_{T,n}=1} L_{E,H}(\boldsymbol{r}_{T,n}, \boldsymbol{z}_{T,n}, \pi_{\mathrm{HE}}, \epsilon_h) \\
&\quad + \sum_{n|t_{T,n}=2} L_{E,\mathrm{OH}}(\boldsymbol{r}_{T,n}, \boldsymbol{z}_{T,n}, \pi_{\mathrm{HE}}, \epsilon_l, \epsilon_b) + \sum_{n|t_{T,n}=3} L_{E,P}(\boldsymbol{r}_{T,n}, \boldsymbol{z}_{T,n}, \epsilon_s, \epsilon_b) \\
&\quad + \sum_{n|t_{T,n}=4} L_{E,M}(\boldsymbol{r}_{T,n}, \boldsymbol{z}_{T,n}, \epsilon_s, \epsilon_b) \\
&\quad + \sum_{n|t_{N,n}=0} L_{E,O}(\boldsymbol{r}_{N,n}, \boldsymbol{z}_{N,n}, \epsilon_l, \epsilon_b) + \sum_{n|t_{N,n}=1} L_{E,H}(\boldsymbol{r}_{N,n}, \boldsymbol{z}_{N,n}, \pi_{\mathrm{HE}}, \epsilon_h) \\
&\quad + \sum_{n|t_{N,n}=2} L_{E,\mathrm{OH}}(\boldsymbol{r}_{N,n}, \boldsymbol{z}_{N,n}, \pi_{\mathrm{HE}}, \epsilon_l, \epsilon_b) + \sum_{n|t_{N,n}=3} L_{E,P}(\boldsymbol{r}_{N,n}, \boldsymbol{z}_{N,n}, \epsilon_s, \epsilon_b) \\
&\quad + \sum_{n|t_{N,n}=4} L_{E,M}(\boldsymbol{r}_{N,n}, \boldsymbol{z}_{N,n}, \epsilon_s, \epsilon_b).
\end{aligned}$$

High Depth Coverage Case

Here we would like to explain the joint probability for error data generation model in the higher depth case. If depth coverage is high, i.e., depth ≥ 100, we rarely collect erroneous candidates at which reads in both normal and tumor samples have reads with lower base qualities. The reason is the filter condition. In general case, we filter out a candidate if the candidate has variant supporting reads in normal samples. Therefore, if the majority of reads have lower base qualities in the normal sample, the position is more likely to be filtered out as the number

of depth coverage increases. Therefore, if depth coverage is high, then reads with lower base qualities do not appear in both the tumor and normal sample after filtering. To incorporate this phenomenon in high depth case, we prepare distinct parameters for tumor and normal samples and set the joint probability in an independent form.

For simplicity, we set the following notations.

$$
\begin{aligned}
\Theta^{(N)}_{E,\mathrm{all}} &:= \left\{\boldsymbol{\pi}_{N,\mathrm{HE}}, \epsilon_{N,s}, \epsilon_{N,l}, \epsilon_{N,h}, \epsilon_{N,b}\right\}, \\
\Theta^{(T)}_{E,\mathrm{all}} &:= \left\{\boldsymbol{\pi}_{T,\mathrm{HE}}, \epsilon_{T,s}, \epsilon_{T,l}, \epsilon_{T,h}, \epsilon_{T,b}\right\}, \\
\mathcal{R}_N &:= \{\boldsymbol{r}_{N,n} | n = 1, \cdots, d_N\}, \mathcal{R}_T := \{\boldsymbol{r}_{T,n} | n = 1, \cdots, d_T\}, \\
\mathcal{T}_N &:= \{t_{N,n} | n = 1, \cdots, d_N\}, \ \mathcal{T}_T := \{t_{T,n} | n = 1, \cdots, d_T\}, \\
\mathcal{Z}_N &:= \{\boldsymbol{z}_{N,n} | n = 1, \cdots, d_N\}, \ \mathcal{Z}_T := \{\boldsymbol{z}_{T,n} | n = 1, \cdots, d_T\}.
\end{aligned}
$$

By using the above notations, the joint probability of error data generation model in this higher coverage case can be represented as follows.

$$
\begin{aligned}
&\ln \Pr(\mathcal{R}_{\mathrm{NT}}, \mathcal{Z}_{\mathrm{NT}}, \Theta^{(N)}_{E,\mathrm{all}}, \Theta^{(T)}_{E,\mathrm{all}}, \mathcal{T}_{\mathrm{NT}} | \mathcal{M}_E) \\
&= \ln \Pr(\mathcal{R}_N, \mathcal{Z}_N, \Theta^{(N)}_{E,\mathrm{all}}, \mathcal{T}_N | \mathcal{M}_E) + \ln \Pr(\mathcal{R}_T, \mathcal{Z}_T, \Theta^{(T)}_{E,\mathrm{all}}, \mathcal{T}_T | \mathcal{M}_E),
\end{aligned}
$$

where

$$
\begin{aligned}
&\ln \Pr(\mathcal{R}_D, \mathcal{Z}_D, \Theta^{(D)}_{E,\mathrm{all}}, \mathcal{T}_D | \mathcal{M}_E) \\
&= \ln \Pr(\mathcal{T}_D | \mathcal{M}_E) \\
&\quad + \ln P_{\mathrm{beta}}(\boldsymbol{\pi}_{D,\mathrm{HE}} | \boldsymbol{\gamma}_{D,\mathrm{HE}}) + \ln P_{\mathrm{beta}}(\epsilon_{D,s} | \boldsymbol{\alpha}_{D,s}) \\
&\quad + \ln P_{\mathrm{beta}}(\epsilon_{D,l} | \boldsymbol{\alpha}_{D,l}) + \ln P_{\mathrm{beta}}(\epsilon_{D,h} | \boldsymbol{\alpha}_{D,h}) + \ln P_{\mathrm{beta}}(\epsilon_{D,b} | \boldsymbol{\alpha}_{D,b}) \\
&\quad + \sum_{n | t_{D,n}=0} L_{E,O}(\boldsymbol{r}_{D,n}, \boldsymbol{z}_{D,n}, \epsilon_{D,l}, \epsilon_{D,b}) + \sum_{n | t_{D,n}=1} L_{E,H}(\boldsymbol{r}_{D,n}, \boldsymbol{z}_{D,n}, \pi_{D,\mathrm{HE}}, \epsilon_{D,h}) \\
&\quad + \sum_{n | t_{D,n}=2} L_{E,\mathrm{OH}}(\boldsymbol{r}_{D,n}, \boldsymbol{z}_{D,n}, \pi_{D,\mathrm{HE}}, \epsilon_{D,l}, \epsilon_{D,b}) + \sum_{n | t_{D,n}=3} L_{E,P}(\boldsymbol{r}_{D,n}, \boldsymbol{z}_{D,n}, \epsilon_{D,s}, \epsilon_{D,b}) \\
&\quad + \sum_{n | t_{D,n}=4} L_{E,M}(\boldsymbol{r}_{D,n}, \boldsymbol{z}_{D,n}, \epsilon_{D,s}, \epsilon_{D,b}), \\
&D \in \{T, N\}.
\end{aligned}
$$

Obtaining the lower bound of the marginal likelihoods can be conducted in the variational Bayes procedures which are similar to that for the mutated data generation model. For the efficacy of setting different joint probabilities for error data generation model, see Section C.

3.5 Results

3.5.1 Performance Evaluation of OHVarfinDer Using Simulation Data Sets

Simulation Data Generation Procedures

We tested OHVarfinDer using simulation data sets. The simulation procedure is described as follows. In the following procedure, we prepared two types of errors. The first type of errors are position-specific ones, and known as error prone sites [72, 55]. The second type of errors are non position-specific ones.

1) Generate a random reference DNA sequence.

2) Generate a heterozygous germ line variant in a random location, as well as two haplotypes (h1 and h2)

3) Generate a somatic mutation randomly around a heterozygous germ line variant, according to an empirical distribution of whole genome data, as well as two haplotypes (h3 and h4)

4) Randomly generate paired-end reads around 900 somatic mutations and 2100 error prone sites randomly.

   2-a) Determine the number of paired-end reads covering the position, by generating a random value d from a norm distribution of $N(\cdot|50, 2)$, and round d to the nearest integer value.

   2-b) Randomly determine the haplotype of the original DNA fragment. We set the frequency of haplotypes as h1: 50-v%, h2: 50%, h3: v%, h4: 0% if a somatic mutation truly exists. We set the frequency of haplotypes as h1: 50%, h2: 50%, h3: 0%, h4: 0% otherwise.

   2-c) For each paired-end read, determine the DNA fragment size by generating a random value l from $N(\cdot|\mu_l, \sigma_l)$, and round l to the nearest integer value.

   2-d) Generate the 100-bp length read sequence on forward strand. Each observed base flips with the sequence error probability of p_{error}. If the position of each observed base is the error prone site, p_{error} is generated from a beta distribution of Beta$(\cdot|2, 30)$. If the position of each observed base is not the error prone site, p_{error} is generated from Beta$(\cdot|10, 1000)$.

   2-e) Generate the read sequence on the reverse strand like 2-d).

Performance Evaluation of OHVarfinDer Using Simulation Data

As a counterpart method, we prepared OVarCall, HapMuC, and a simple Fisher's exact test [19] method, which uses a 2 × 2 contingency table of read counts, tumor and normal samples/variant and reference alleles. We calculated the area under the curve (AUC) values from the plotted ROC curve [5] for each simulation condition as shown in Table 3.4.

In the simulation data set under the condition of **B**, only overlapping paired-end read information was available. In this case, our method performs comparable with OVarCall. In the simulation data set under the condition of **C**, only heterozygous SNP information was available. In this case, our method performed comparably well with HapMuC that can utilize this information source. In the

simulation data set under the condition of **A**, neither of the above types of information was available. In this case, our method performed comparably well with Fisher's exact test. In the simulation data set under the condition of **D**, both overlapping paired-end read information and heterozygous SNP information were available. In this case, our method outperformed both OVarCall and HapMuC. We also show the additional experiments based on this simulation data sets at Sections A.2 and B at which we compared our method with Bayesian model averaging-based method and several supervised learning-based methods.

3.5.2 Performance Evaluation of OHVarfinDer Using Real Data

SNVs in Exome Sequence Data Set

We confirmed whether the performance of our method could be improved by using overlapping information using real exome data sets, as shown in Table 3.5. For the real data sets, we used exome sequence data from renal clear-cell carcinoma, which has already been used for performance evaluation of OVarCall [55]. In these data sets, approximately 40% of paired-end reads overlapped, and thus the use of overlapping paired-end reads is expected to affect the performance. In this data set, true somatic SNVs were validated by deep sequencing [72]. Both in the case of lower variant allele frequency of 2%-7% and the case of moderate variant allele frequency above 7%, OHVarfinDer performed comparably well with OVarCall and outperformed HapMuC. Furthermore, we observed that our method returned a low Bayes factor of 0.0000011 in the false-positive case in Fig. 3.4(b). Therefore, we confirmed that our method can incorporate overlapping information and improve its performance.

SNVs and InDels in Whole Genome Data Set at TCGA Mutation Calling Benchmark 4 Datasets

We examined whether we could improve the performance of our method by using heterozygous SNP and strand bias information using whole genome sequence data. The results are summarized in Table 3.6. For the data set, we used whole genome sequence data sets from breast cancer cell lines, which are publicly available as a part of The Cancer Genome Atlas (TCGA) Mutation Calling Benchmark 4 datasets (`https://gdc.cancer.gov/resources-tcga-users/tcga-mutation-calling-benchmark-4-files`) and have been used for performance evaluation of HapMuC.

In these data sets, pure cell line sequence data sets of normal and tumor cell line and computational mixtures of these sequence data sets are prepared, e. g., HCC1143_n40t60 represents that 40% of pure normal and 60% of pure tumor sequence data are mixed. In this experiment, we obtained answers of true mutations from these pure cell line data sets, and we conducted performance evaluations for tumor sequence data sets with several mixture rates, i. e., n20t80, n40t60, n60t40, n80t20. For these data sets, the use of heterozygous SNPs information and strand bias information is important for improving performance because the average proportion of overlapping paired-end reads was approximately 3% within these data sets.

For the performance of OHVarfinDer, OHVarfinDer performed better than any other mutation caller, except for HCC1954_n80t20. We also observed that our method returned low Bayes factor of 0.000059 in the false-positive case in Fig. 3.4(a). Therefore, we confirmed that our method can incorporate heterozygous SNP and strand bias information and improve its performance.

Table 3.4: Summary of AUC in simulation data sets

	$v(\%)$	HeteroSNPs	Overlap	Distance to SNP	μ_l	σ_l	OHVarfinDer	OVarCall	HapMuC	Fisher	#SNV	#Error
A	5	-	-	500-5000	300	30	0.828	0.750	0.828	0.810	341	822
	10	-	-				0.891	0.867	0.880	0.891	713	871
	20	-	-				0.967	0.978	0.950	0.983	896	872
B	5	-	+	500-5000	180	30	0.938	0.917	0.786	0.817	407	1394
	10	-	+				0.958	0.954	0.843	0.899	763	1413
	20	-	+				0.989	0.991	0.947	0.988	897	1411
C	5	+	-	1-100	300	30	0.880	0.765	0.882	0.825	301	851
	10	+	-				0.916	0.877	0.907	0.886	733	871
	20	+	-				0.986	0.984	0.977	0.983	896	925
D	5	+	+	1-100	180	30	0.943	0.923	0.838	0.803	388	1356
	10	+	+				0.975	0.952	0.918	0.914	757	1398
	20	+	+				0.994	0.991	0.977	0.990	896	1354

Table 3.5: Summary of AUC in exome sequence data sets

SNV/InDel	VAF	OVarCall	OHVarfinDer	HapMuC	Strelka	MuTect	VarScan2	#SNV	#Error
SNV	2-7%	0.982	0.990	0.965	0.933	0.875	0.625	52	2422
SNV	7%-	0.991	0.988	0.955	0.995	0.994	0.900	184	1982

Table 3.6: Summary of AUC in TCGA mutation calling benchmark 4 datasets

Sample	SNV/InDel	OVarCall	OHVarfinDer	HapMuC	Strelka	MuTect	VarScan2	#SNV	#Error
HCC1143_n20t80	SNV	0.869	0.906	0.827	0.873	0.848	0.801	10618	2327
HCC1143_n40t60		0.870	0.901	0.824	0.877	0.855	0.799	8517	2049
HCC1143_n60t40		0.884	0.912	0.843	0.901	0.876	0.814	5450	1684
HCC1143_n80t20		0.901	0.941	0.870	0.938	0.918	0.830	1874	1451
HCC1954_n20t80		0.882	0.934	0.852	0.903	0.869	0.862	10653	2854
HCC1954_n40t60		0.893	0.941	0.852	0.917	0.880	0.858	7969	2327
HCC1954_n60t40		0.917	0.949	0.865	0.937	0.905	0.852	4638	1770
HCC1954_n80t20		0.941	0.970	0.880	0.972	0.942	0.848	1389	1404
total		0.895	0.935	0.860	0.913	0.886	0.852	51108	15866
HCC1143_n20t80	InDel	0.707	0.796	0.678	0.713	-	0.722	926	4951
HCC1143_n40t60		0.733	0.814	0.700	0.755	-	0.748	617	4761
HCC1143_n60t40		0.760	0.834	0.723	0.784	-	0.778	328	4562
HCC1143_n80t20		0.809	0.855	0.770	0.816	-	0.800	94	4899
HCC1954_n20t80		0.800	0.860	0.771	0.822	-	0.825	1771	5219
HCC1954_n40t60		0.821	0.866	0.778	0.843	-	0.835	1172	5215
HCC1954_n60t40		0.819	0.863	0.770	0.848	-	0.831	607	5200
HCC1954_n80t20		0.815	0.887	0.777	0.864	-	0.823	159	5053
total		0.777	0.838	0.774	0.794	-	0.792	5674	39861

3.6 Discussion

Some mutation calling methods, e. g., HapMuC and OVarCall, can incorporate a characteristic information source, e. g., heterozygous SNPs and overlapped paired-end reads, in their mutation calling process. However, no existing Bayesian methods utilize multiple types of such characteristic information sources simultaneously in an explicit manner.

In this chapter, we first introduced a framework for Bayesian model integration named as partitioning-based model integration, which differs from the Bayesian model averaging [29, 64]. In this framework, we first set a partitioning rule for data and augmented the data with indicator variables that show the category of partitioning. Second, we constructed a generative model for each category of the partitioned data set. This framework requires two assumptions. The first assumption is that we can set a partitioning rule and construct corresponding generative models. The second assumption is that partitioning probabilities are common among the mutated data generation model and error data generation model. If the above assumptions hold true, we can compute the Bayes factor without a careful setting of prior partitioning probabilities. In our problem setting of mutation calling, the above two assumptions seem natural, and thus we constructed a Bayesian mutation calling method, OHVarfinDer, based on this framework.

We conducted performance evaluations with simulation and real data sets. In the simulation data sets, we showed that our method could utilize multiple information sources, particularly overlapping paired-end read information and heterozygous SNP information. If only one information source was given, our method performed comparably well with other existing methods. If both information sources were given, our method performed better than other existing methods. In the real data sets, e. g., The Cancer Genome Atlas (TCGA) Mutation Calling Benchmark 4 datasets, we also demonstrated the better performance of our method compared to other existing methods.

We have demonstrated how to integrate known multiple information sources for mutation calling by our framework. We note that mapping quality and base quality of reads are also used in our method by incorporating the profile HMM modeling [1, 77]. Although our framework is practically useful for mutation calling, there is at least one limitation for this framework, i. e., our framework does not assume inference over the parameter distributions, e. g., prior distributions for the error parameters. Such inference is important if we consider using multiple sequence data sets simultaneously. For example, if we can use pooled normal sequence data sets, we can infer the error distributions depending on the genomic positions. For the future work, we plan to extend our framework to infer the form of the parameter distributions, e. g., incorporating predictive distributions for the error parameters.

Chapter 4

Flexible Bayesian Modeling for Accurate Mutation Calling from Multi-Regional Tumor Samples

4.1 Overview

The process of genomic alteration is one of the most important factors for carcinogenesis. Acquired somatic mutations, together with individual germline variations, have a large effect on cancer evolution. By obtaining accurate genomic alteration profiles, we can estimate the cause of cancer for individual patients and search for optimal therapies. Thus, mutation calling from sequence data sets has become a fundamental analysis in cancer therapy and research. An enormous number of studies [40, 71, 10, 72, 77, 36, 53, 67] have been conducted to improve the performance of single-tumor-based mutation call, i.e., mutation call from a tumor and a matched normal sequence data set, and the performance of mutation call is updated annually by modeling properties of raw sequence data sets in more sophisticated manners. OHVarfinDer constructs Bayesian models to utilize sequence data specific properties. DeepVariant [62] is a convolutional neural network (CNN) based method for detecting germline mutations and able to learn the properties in any sequence data platform. NeuSomatic is also a CNN based method for somatic mutation call, which is motivated by DeepVariant.

Mutation profiles from multi-regional tumor sequencing data sets give helpful information to understand the tumor evolutionary process and the intratumoral heterogeneity. In order to detect subclonal mutations with lower variant allele frequencies, researchers have developed mutation calling methods that are suitable for multi-regional tumor data sets. There are mainly two types of approaches for multi-regional mutation call. The first type of the methods [65, 15, 78, 68] consider the property of tumor phylogenetic tree and clonal populations. The second type [32] focused on the sharing assumption of a mutation across multiple samples, defined in Section 4.3.1. For these multi-regional mutation calling methods, comprehensive performance evaluations were conducted in recent reports [13].

Although one of the existing methods of multiSNV is based on the sharing assumption of mutation and improved the performance of mutation call, there are still two drawbacks. First, multiSNV does not consider the "No-TP case": even if we could detect mutation candidates in multiple regions, no true mutations exist, unfortunately. We will define No-TP case in Section 4.3.3. Second, detection of a mutation for each tumor region in multiSNV is based on scores from a set of predefined generative models and cannot leverage scores from other state-of-the-art mutation calling methods for a single-regional tumor.

Here, we propose a Bayesian method of MultiMuC for multi-regional muta-

tion call. Our method has two defining characteristics. First, our method avoids the No-TP case by leveraging the specificity of detection and the number of detected candidates. We evaluate the probability of the No-TP case and investigate that the probability decreases as the specificity of detection or the number of detected candidates increases. Second, our method can incorporate scores from state-of-the-art mutation calling methods as long as these scores are based on probabilities, i.e, Bayes factors [33] or posterior probabilities. We investigate that Bayes factors provide sufficient information for obtaining the consistent posterior distribution and maximum a posteriori (MAP) state even if data generation probabilities for each data set are not available. We demonstrate that our method improves the original detection performance in state-of-the-art mutation calling methods for a single-regional tumor through real-data-based (TCGA 4 mutation calling benchmark datasets) sequence data simulation and outperforms existing multi-regional mutation calling methods.

The organization of this chapter is as follows. First, we explain the related works. Second, we explain the mutation sharing assumption used in multiSNV [32]. Third, we define the probability of the No-TP case and evaluate the probability. Forth, we elucidate the manner to use scores from existing single-tumor-based mutation call for multiple-tumor-based mutation call by introducing a simple Bayesian hierarchical model as a toy-example. Fifth, we describe the Bayesian statistical model of MultiMuC and MCMC procedures for MAP inference. Finally, we show experimental results to evaluate the performance of our method.

Contents of this chapter are mainly related to the published work of [54].

4.2 Related Work

4.2.1 multiSNV

multiSNV is the first multiple-tumor-based mutation calling method. multiSNV constructs a stochastic model in which the frequency of mutation genotype is shared among multiple samples. multiSNV uses the mutation sharing property; if at least one tumor sequence data has a mutation at the candidate position, then the method can expect the other tumor samples to have the mutation with higher confidence. We introduce a simplified stochastic model of multiSNV (Fig. 4.1) and explain how to model the mutation sharing property by setting common mutation frequency. This simplified model is not completely the same as multiSNV but enough to explain the key concept of the method.

We assume the distribution of mutation frequency of g and the genotype in the i-th tumor sample $G_{T,i}$ are set as follows.

$$\begin{aligned} g &\sim \mathrm{Beta}(\cdot|\alpha, \beta), \\ G_{T,i} &\sim \mathrm{Ber}(\cdot|g). \end{aligned}$$

By using the allele frequency f and sequence error rate p_{err}, we set the probability of the i-th tumor data $D_{T,i}$ as follows.

$$\begin{aligned} \Pr(D_{T,i}|G_{T,i} = 0) &= (p_{\mathrm{err}})^{d_i - r_i} (1 - p_{\mathrm{err}})^{r_i}, \\ \Pr(D_{T,i}|G_{T,i} = 1) &= f^{a_i} (1 - f)^{d_i - a_i}, \end{aligned}$$

where d_i is the depth coverage, r_i is the number of reference-supporting reads, and a_i is the number of variant-supporting reads in the i-th tumor sample.

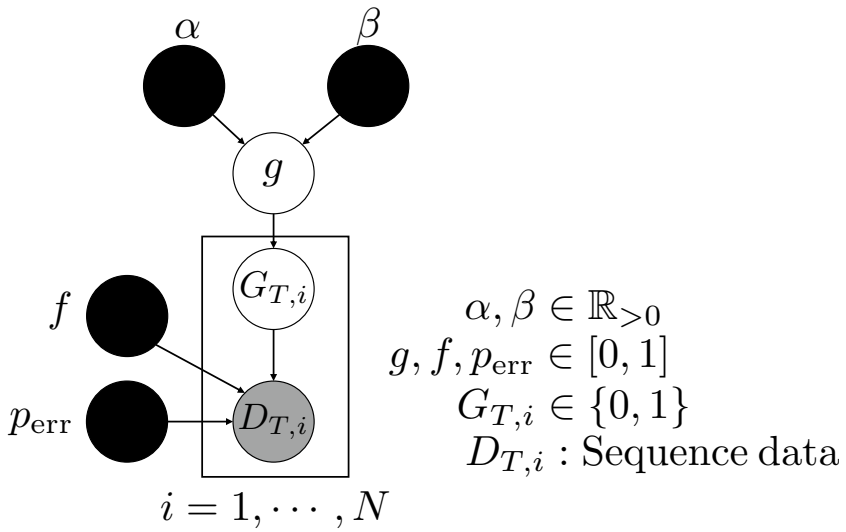


Figure 4.1: Simplified model of multiSNV. In this model, we ignore several settings of multiSNV; we ignore the genotype for the normal sample and assume only two genotypes of reference ($G_{T,i} = 0$) and tumor ($G_{T,i} = 1$).

We focus on the distribution of $G_{T,i}$. If no sequence data is given, the prior of $G_{T,i}$ depends on g.

$$\Pr(G_{T,i}) = g^{G_{T,i}}(1-g)^{1-G_{T,i}}.$$

If no sequence data is given but the genotype of $G_{T,1}$ is observed, then the posterior distribution of $G_{T,i}$ ($\neq 1$) is as follows.

$$\begin{aligned}
\Pr(G_{T,i}|G_{T,1}) &\propto \Pr(G_{T,i}, G_{T,1}) \\
&= \int \Pr(G_{T,i}, G_{T,1}|g)\Pr(g)dg \\
&\propto \int g^{G_{T,i}}(1-g)^{1-G_{T,i}} g^{G_{T,1}}(1-g)^{1-G_{T,1}} g^{\alpha-1}(1-g)^{\beta-1}dg \\
&\propto \{B(\alpha + G_{T,1} + 1, \beta + (1 - G_{T,1}))\}^{G_{T,i}} \\
&\quad \cdot \{B(\alpha + G_{T,1}, \beta + (1 - G_{T,1}) + 1)\}^{1-G_{T,i}} \\
&\propto (\alpha + G_{T,1})^{G_{T,i}}(\beta + (1 - G_{T,1}))^{1-G_{T,i}}, \\
\therefore \Pr(G_{T,i}|G_{T,1}) &= \left(\frac{\alpha + G_{T,1}}{\alpha + \beta + 1}\right)^{G_{T,i}} \cdot \left(\frac{\beta + 1 - G_{T,1}}{\alpha + \beta + 1}\right)^{1-G_{T,i}}.
\end{aligned}$$

From this, if the observed genotype is mutation ($G_{T,1} = 1$), then $G_{T,i}$ is also mutation ($= 1$) with higher probability. By setting $\alpha \ll 1$, we can change the posterior probability drastically; if $G_{T,1} = 1$ is observed, then $G_{T,i} = 1$ occurs with much higher probability than the prior probability.

4.2.2 NeuSomatic

NeuSomatic [67] is a somatic mutation calling method for single-regional tumor data set, which is motivated by a germline variant detection method of DeepVariant [62]. The network architecture of NeuSomatic is based on the convolutional neural network [41]. This method outputs the posterior somatic event probability as a mutation calling score.

4.2.3 Strelka2

Strelka2 [36] is a succeeding version of Strelka [71]. Strelka2 can use training data set to estimate the error probabilities and incorporate read generation probabilities for computing the posterior probabilities. Strelka2 returns the posterior error event probabilities in the form of a Phred quality score.

4.2.4 MuTect2

MuTect2 is a succeeding version of MuTect [10]. The biggest difference between MuTect and MuTect2 is that MuTect2 can detect somatic insertions and deletions. MuTect2 outputs the likelihood ratio in a similar manner to MuTect.

4.3 Methods

4.3.1 The Mutation Sharing Assumption

Here, we explain the mutation sharing assumption that is leveraged to improve the performance of multi-regional mutation call. We assume that there are N sequence data sets $\{D_i\}_{i=1,\cdots,N}$ and latent variables $\{X_i\}_{i=1,\cdots,N}$ ($X_i \in \{0,1\}$) express the existence of a mutation at i-th data set and $C \in \{0,1\}$ represents the existence of the mutation at least one data set and $\{V_i\}_{i=1,\cdots,N}$ ($V_i \in \mathbb{R}$) are the scores from single-tumor-based mutation call. The concept of the mutation sharing assumption for mutation call can be summarized in the following assumption.

Assumption 4.3.1 (The mutation sharing assumption)**.**

$$\forall v \in \mathbb{R}, \exists w < v \ \ s.t. \ e(w|C=1) = e(v),\ r(w) > r(v),$$

where

$$e(v) := \frac{1}{N}\sum_{i=1}^{N} \Pr(X_i = 1 | V_i > v) \ \ \text{(Precision on average)},$$

$$e(v|C=c) := \frac{1}{N}\sum_{i=1}^{N} \Pr(X_i = 1 | V_i > v, C = c) \ \ \text{(Precision given C)},$$

$$r(v) := \frac{1}{N}\sum_{i=1}^{N} \Pr(V_i > v | X_i = 1) \ \ \text{(Recall on average)}.$$

According to this assumption, if we know (or predict with high confidence) the existence of a mutation in at least one tumor data set ($C = 1$), then we can improve recall from $r(v)$ up to $r(w)$ by lowering the threshold from v down to w with constant precision $e(w|C = 1) = e(v)$. Based on this idea, multiSNV [32] has succeeded in performance improvement.

4.3.2 Increasing Posterior Odds Score of Mutation Call Given $C = 1$.

Here, we show that the assumption is based on an increase of posterior odds for mutation call. We assume that each score is represented by posterior odds form $V_i := \Pr(X_i = 1|D_i)/\Pr(X_i = 0|D_i)$. If we observe $C = 1$ in addition to the

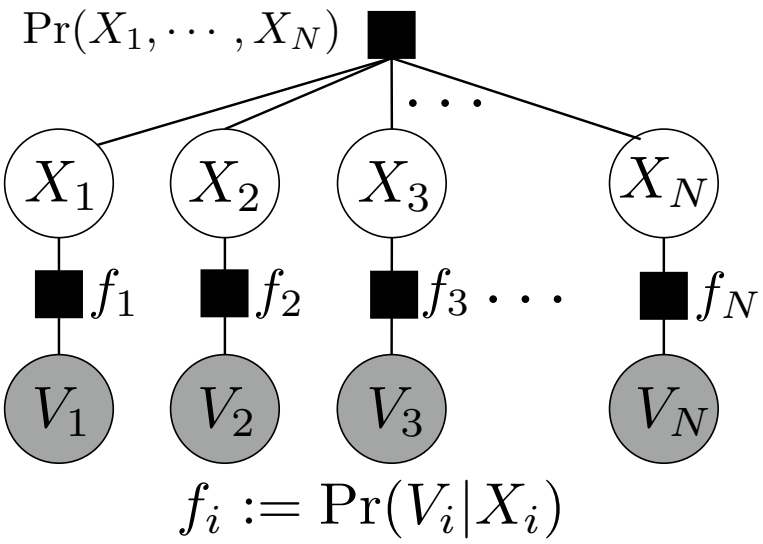


Figure 4.2: Graphical representation of the assumed stochastic dependence between $\{X_i\}_{i=1,\cdots,N}$ and $\{V_i\}_{i=1,\cdots,N}$. In this assumed stochastic dependence, we do not set any independence of prior distribution in $\Pr(\{X_i\}_{i=1,\cdots,N})$ and only assume that each V_i is dependent on the corresponding X_i.

observed sequence data set, the true posterior odds can be represented as follows.

$$V_i' := \frac{\Pr(X_i = 1|C = 1, D_i)}{\Pr(X_i = 0|C = 1, D_i)} = \frac{\Pr(X_i = 1, C = 1, D_i)}{\Pr(X_i = 0, C = 1, D_i)}$$
$$= \frac{\Pr(X_i = 1|C = 1)}{\Pr(X_i = 0|C = 1)} \frac{\Pr(D_i|X_i = 1, C = 1)}{\Pr(D_i|X_i = 0, C = 1)}.$$

The true posterior odds are greater than the original posterior odds as shown in the following theorem.

Theorem 4.3.1 (Increasing posterior odds)**.**

If $\Pr(D_i|X_i, C) = \Pr(D_i|X_i)$, $0 < \Pr(C = 0) < 1$, and $V_i, V_i' \in \mathbb{R}$, then $V_i' > V_i$.

Proof.

It is sufficient to show that the following condition holds true.

$$\cdot \frac{\Pr(X_i = 1|C = 1)}{\Pr(X_i = 0|C = 1)} > \frac{\Pr(X_i = 1)}{\Pr(X_i = 0)}.$$

The condition can be proved by evaluating $\Pr(X_i = 1)$ and $\Pr(X_i = 0)$ as follows.

$\Pr(X_i = 1)$
$= \Pr(X_i = 1|C = 1)\Pr(C = 1) + \Pr(X_i = 1|C = 0)\Pr(C = 0)$
$= \Pr(X_i = 1|C = 1)\Pr(C = 1)$ ($\because$ $\Pr(X_i = 1|C = 0) = 0$),
$\Pr(X_i = 0)$
$= \Pr(X_i = 0|C = 1)\Pr(C = 1) + \Pr(X_i = 0|C = 0)\Pr(C = 0)$
$= \Pr(X_i = 0|C = 1)\Pr(C = 1) + \Pr(C = 0)$ ($\because$ $\Pr(X_i = 0|C = 0) = 1$)
$> \Pr(X_i = 0|C = 1)\Pr(C = 1)$ ($\because$ $0 < \Pr(C = 0) < 1$).

By using the above evaluations, we can show $\frac{\Pr(X_i = 1|C = 1)}{\Pr(X_i = 0|C = 1)} > \frac{\Pr(X_i = 1)}{\Pr(X_i = 0)}$.

From this condition and the given hypothesis,

$$V_i' = \frac{\Pr(X_i = 1|C = 1)}{\Pr(X_i = 0|C = 1)} \frac{\Pr(D_i|X_i = 1, C = 1)}{\Pr(D_i|X_i = 0, C = 1)} > \frac{\Pr(X_i = 1)}{\Pr(X_i = 0)} \frac{\Pr(D_i|X_i = 1)}{\Pr(D_i|X_i = 0)} = V_i.$$

□

4.3.3 The Probability of No-TP (True Positive) Case

We focus on the No-TP cases that cause performance degradation. We define the No-TP case for v detection threshold and M candidate number as the case that the mutation does not truly exist in any region at all even if M candidate mutations are found by the threshold value of v. Under the No-TP case, we cannot obtain any true mutations by lowering the threshold but only obtain false positives, and then we only degrade the performance of detection. We assume for simplicity that $V_1 \geq V_2 \cdots \geq V_N$ and we define the probability of the No-TP case as follows.

$$\Pr(X_1 = 0, \cdots, X_N = 0 | V_1 > v, \cdots, V_M > v, V_{M+1} \leq v, \cdots, V_N \leq v). \quad (4.1)$$

To evaluate the probability, we assume the stochastic dependence as shown in the graphical model of Fig. 4.2. In this setting, we do not set any restriction for stochastic dependence between $X_1, \cdots, X_N$ and only assumes the following conditional independence between $V_1, \cdots, V_N$.

$$\Pr(V_1, \cdots, V_N | X_1, \cdots, X_N) = \prod_{i=1}^{N} \Pr(V_i | X_i). \quad (4.2)$$

The Probability of No-TP Case when $M = N$

We evaluate the probability of No-TP case when $M = N$.

$$\begin{aligned}
&\Pr(X_1 = 0, \cdots, X_N = 0 | V_1 > v, \cdots, V_N > v) \\
&\propto \Pr(X_1 = 0, \cdots, X_N = 0, V_1 > v, \cdots, V_N > v) \\
&= \Pr(X_1 = 0, \cdots, X_N = 0) \Pr(V_1 > v, \cdots, V_N > v | X_1 = 0, \cdots, X_N = 0) \\
&= \Pr(X_1 = 0, \cdots, X_N = 0) \prod_{i=1}^{N} \Pr(V_i > v | X_i = 0) \ \because) \textit{Eq. (4.2)} \\
&= \Pr(X_1 = 0, \cdots, X_N = 0) \prod_{i=1}^{N} (1 - s_i(v)), \quad (4.3)
\end{aligned}$$

where $s_i(v) := \Pr(V_i \leq v | X_i = 0)$ corresponds to the specificity. Therefore from Eq. (4.3), we will decrease the probability of the No-TP case by increasing the number of candidate $M(= N)$ or improving the specificity $s_i(v)$.

The Probability of No-TP Case when $M < N$

Here, we also evaluate the probability of the No-TP case when $M < N$. For simplicity, we define variables and relational operators between vector and scalar. For Eq. (4.4), we also define similar relational operators for $\geq, <, \leq, =$ between vector and scalar.

$$\begin{aligned}
&\boldsymbol{V} := (V_1, \cdots, V_M), \ \widetilde{\boldsymbol{V}} := (V_{M+1}, \cdots, V_N), \boldsymbol{X} := (X_1, \cdots, X_M), \ \widetilde{\boldsymbol{X}} := (X_{M+1}, \cdots, X_N), \\
&\boldsymbol{u} > v \iff u_i > v \ (\forall i), \quad (4.4) \\
&\boldsymbol{u} \neq v \iff u_i \neq v \ (\exists i).
\end{aligned}$$

The probability of the No-TP case can be represented as follows.

$$\Pr(\boldsymbol{X} = 0, \widetilde{\boldsymbol{X}} = 0 | \boldsymbol{V} > v, \widetilde{\boldsymbol{V}} \leq v). \quad (4.5)$$

For $\Pr(\boldsymbol{V} > v, \widetilde{\boldsymbol{V}} \leq v)$, we can obtain a lower bound as follows.

$$
\begin{aligned}
&\Pr(\boldsymbol{V} > v, \widetilde{\boldsymbol{V}} \leq v) \\
&= \sum_{\boldsymbol{X}, \widetilde{\boldsymbol{X}}} \Pr(\boldsymbol{X}, \widetilde{\boldsymbol{X}}) \Pr(\boldsymbol{V} > v, \widetilde{\boldsymbol{V}} \leq v | \boldsymbol{X}, \widetilde{\boldsymbol{X}}) \\
&= \sum_{\boldsymbol{X}, \widetilde{\boldsymbol{X}}} \Pr(\boldsymbol{X}, \widetilde{\boldsymbol{X}}) \prod_{i=1}^{M} \Pr(V_i > v | X_i) \prod_{k=M+1}^{N} \Pr(V_k \leq v | X_k) \ (\because \textit{Eq. (4.2)}) \\
&= \sum_{\boldsymbol{X}, \widetilde{\boldsymbol{X}}} \Pr(\boldsymbol{X}, \widetilde{\boldsymbol{X}}) \prod_{i=1}^{M} (1 - s_i(v))^{1 - X_i} R_i(v)^{X_i} \prod_{k=M+1}^{N} s_k(v)^{1 - X_k} (1 - R_k(v))^{X_k} \\
&\geq \Pr(\boldsymbol{X} = 1, \widetilde{\boldsymbol{X}} = 0) \prod_{i=1}^{M} R_i(v) \prod_{k=M+1}^{N} s_k(v) =: A, \qquad (4.6)
\end{aligned}
$$

where $R_i(v) := \Pr(V_i > v | X_i = 1)$ corresponds to recall.
From Eq. (4.6), if $A > 0$, we can derive an upper bound for Eq. (4.5) as follows.

$$
\begin{aligned}
&\Pr(\boldsymbol{X} = 0, \widetilde{\boldsymbol{X}} = 0 | \boldsymbol{V} > v, \widetilde{\boldsymbol{V}} \leq v) = \frac{\Pr(\boldsymbol{X} = 0, \widetilde{\boldsymbol{X}} = 0, \boldsymbol{V} > v, \widetilde{\boldsymbol{V}} \leq v)}{\Pr(\boldsymbol{V} > v, \widetilde{\boldsymbol{V}} \leq v)} \\
&\leq \min \left(1, \frac{\Pr(\boldsymbol{X} = 0, \widetilde{\boldsymbol{X}} = 0) \prod_{i=1}^{M} (1 - s_i(v)) \prod_{k=M+1}^{N} s_k(v)}{\Pr(\boldsymbol{X} = 1, \widetilde{\boldsymbol{X}} = 0) \prod_{i=1}^{M} R_i(v) \prod_{k=M+1}^{N} s_k(v)} \right) \\
&= \min \left(1, \frac{\Pr(\boldsymbol{X} = 0, \widetilde{\boldsymbol{X}} = 0)}{\Pr(\boldsymbol{X} = 1, \widetilde{\boldsymbol{X}} = 0)} \prod_{i=1}^{M} \frac{1 - s_i(v)}{R_i(v)} \right). \qquad (4.7)
\end{aligned}
$$

From Eq. (4.7), as the specificity $s_i(v)$ increases, the upper bound will decrease when $M < N$. Furthermore, if recall ($R_i(v)$) is larger than false positive rate ($1 - s_i(v)$), we will also decrease the upper bound by increasing the number of mutation candidates M.

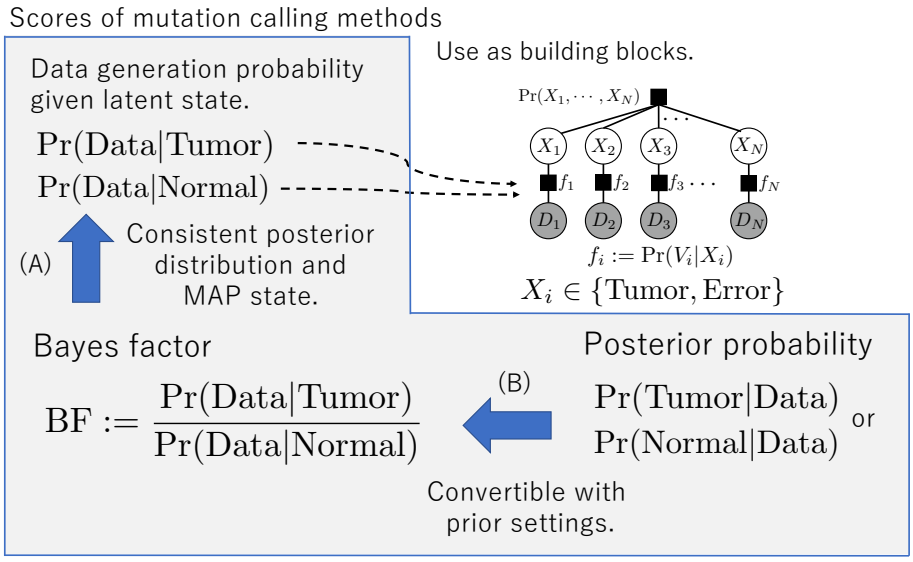


Figure 4.3: Summary of the Bayes factor based model construction.

4.3.4 Leveraging Scores from Other Methods for Bayesian Models

We propose an idea to leverage probabilistic scores from other state-of-the-art mutation calling methods for a single-regional tumor to construct a Bayesian

hierarchical model for multi-regional tumors.

We can see that data generation probabilities given dependent latent variables can be used as building blocks to construct a Bayesian hierarchical model. For example of Fig. 4.3, if we can borrow $\Pr(D_i|X_i = \text{Tumor})$ and $\Pr(D_i|X_i = \text{Error})$ as building blocks, then we only need to additionally build the stochastic dependence of latent variables $\{X_i\}_{i=1,\cdots,N}$ to construct the Bayesian models.

Although we would like to use the data generation probabilities given dependent latent variables from this idea, e.g., Pr(Data|Error) and Pr(Data|Tumor) defined in mutation calling methods for each region of tumor, such probabilities are not available in most cases. On the other hand, alternative values, e.g., Bayes factors or posterior probabilities are available as mutation calling scores from state-of-the-art methods, e.g., Strelka2 and NeuSomatic. In the following sections, we will demonstrate how Bayes factors and posterior probabilities can be used as building blocks to construct a Bayesian model (Fig. 4.3). First, we will show how to extract equivalent information to the data generation probabilities from Bayes factors by considering the posterior distributions and maximum a posteriori (MAP) state (Fig. 4.3-A) through introducing a toy example model. Next, we will show how to convert posterior probabilities to Bayes factors (Fig. 4.3-B).

Data Generation Probabilities from Bayes Factors

We show that Bayes factors are sufficient for MAP estimate for a toy example stochastic model even when full data generation probabilities of Pr(Data|Tumor) and Pr(Data|Error) are not given.

For this example, we assume the stochastic model as shown in Fig. 4.4. $S \in \{0, 1\}$ represents the existence of tumor cells and $Y_i \in \{0, 1\}$ represents the existence of mutation at the i-th data set D_i. The Bayes factor for the i-th

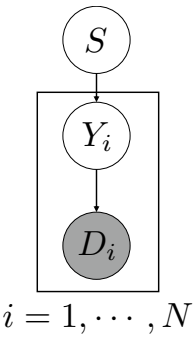


Figure 4.4: A toy example model for multiple tumor samples. $S \in \{0, 1\}$ represents the existence of tumor cells and $Y_i \in \{0, 1\}$ represents the existence of mutation at i-th data set D_i.

data set is defined as the ratio of the marginal likelihood and P_{call} is defined in a single-tumor-based mutation calling method.

$$\text{BF}_i := \frac{P_{\text{call}}(D_i|Y_i = 1)}{P_{\text{call}}(D_i|Y_i = 0)}. \tag{4.8}$$

We assume that each data generation probability is a positive value for any observed data point D_i.

$$P_{\text{call}}(D_i|Y_i = 1) > 0, P_{\text{call}}(D_i|Y_i = 0) > 0. \tag{4.9}$$

We consider two settings of probability distributions for this toy example model and denote the first setting as $\Pr^{(1)}(\cdot)$ and the second setting as $\Pr^{(2)}(\cdot)$. For both settings, we assume common distributions for S and Y_i.

$$\Pr^{(1)}(S) = \Pr^{(2)}(S) = P_{\text{ber}}(S|f_1),$$
$$\Pr^{(1)}(Y_i|S) = \Pr^{(2)}(Y_i|S) = P_{\text{ber}}(Y_i|f_2)^S \cdot P_{\text{ber}}(Y_i|f_3)^{(1-S)},$$

where $P_{\text{ber}}(\cdot|f)$ means the probability mass function of Bernoulli distribution with a frequency of f and we set $0 \leq f_1, f_2, f_3 \leq 1$. In the first setting at Eq. (4.10), we use both the numerator and denominator in each Bayes factor to define the distributions. In the second setting at Eq. (4.11), we only use Bayes factors and supplement the distributions with a pre-defined positive constant p for all the data index i.

$$\Pr^{(1)}(D_i|Y_i = 0) = P_{\text{call}}(D_i|Y_i = 0), \Pr^{(1)}(D_i|Y_i = 1) = P_{\text{call}}(D_i|Y_i = 1), \quad (4.10)$$
$$\Pr^{(2)}(D_i|Y_i = 0) = p,\ \Pr^{(2)}(D_i|Y_i = 1) = p \cdot \text{BF}_i,\ \ (0 < p). \quad (4.11)$$

As shown in the following lemma and corollary, this difference in setting the probability distribution does not affect the posterior distribution and MAP state of the latent variable S and Y_i. Therefore, Bayes factors give sufficient information on data generation probabilities for MAP inference of the latent state for some set of stochastic models.

Lemma 4.3.1 (Unchanged posterior)**.**

$$\forall S, \boldsymbol{Y}\ s.t.\ \Pr^{(1)}(S, \boldsymbol{Y}|\boldsymbol{D}) = \Pr^{(2)}(S, \boldsymbol{Y}|\boldsymbol{D}),$$
where
$$\boldsymbol{Y} := (Y_1, \cdots, Y_N),\ \boldsymbol{D} := (D_1, \cdots, D_N)\ .$$

Proof. It is sufficient if we can show that the following conditions hold true.

$$\cdot \Pr^{(1)}(S, \{D_i, Y_i\}_i) > 0 \iff \Pr^{(2)}(S, \{D_i, Y_i\}_i) > 0 \text{ (Same support region)},$$
$$\cdot \Pr^{(1)}(S', \{D_i, Y_i'\}_i) > 0, \Pr^{(1)}(S, \{D_i, Y_i\}_i) > 0$$
$$\Rightarrow \frac{\Pr^{(1)}(S', \{D_i, Y_i'\}_i)}{\Pr^{(1)}(S, \{D_i, Y_i\}_i)} = \frac{\Pr^{(2)}(S', \{D_i, Y_i'\}_i)}{\Pr^{(2)}(S, \{D_i, Y_i\}_i)} \text{ (Same probability ratio)}.$$

The first condition is satisfied from Eq. (4.9) and $0 < p$. For the second condition, we can show the condition by substitution. From these two conditions, we can prove the consistency of the posterior as follows.

(i) $\forall S, \boldsymbol{Y}\ s.t.\ \Pr^{(1)}(S, \boldsymbol{Y}, \boldsymbol{D}) = \Pr^{(2)}(S, \boldsymbol{Y}, \boldsymbol{D}) = 0$

The posterior probabilities are consistent due to the same joint probabilities.

(ii) $\exists S', \boldsymbol{Y'}\ s.t.\ \Pr^{(1)}(S', \boldsymbol{Y'}, \boldsymbol{D}) > 0, \Pr^{(2)}(S', \boldsymbol{Y'}, \boldsymbol{D}) > 0$

From the second condition,

$$\forall S, \boldsymbol{Y} (\text{in the support regions})\ \ s.t.\ \frac{\Pr^{(1)}(S, \boldsymbol{Y}, \boldsymbol{D})}{\Pr^{(1)}(S', \boldsymbol{Y'}, \boldsymbol{D})} = \frac{\Pr^{(2)}(S, \boldsymbol{Y}, \boldsymbol{D})}{\Pr^{(2)}(S', \boldsymbol{Y'}, \boldsymbol{D})}$$
$$\Rightarrow \Pr^{(2)}(S, \boldsymbol{Y}, \boldsymbol{D}) = \frac{\Pr^{(2)}(S', \boldsymbol{Y'}, \boldsymbol{D})}{\Pr^{(1)}(S', \boldsymbol{Y'}, \boldsymbol{D})} \Pr^{(1)}(S, \boldsymbol{Y}, \boldsymbol{D})$$
$$\Rightarrow \Pr^{(2)}(S, \boldsymbol{Y}, \boldsymbol{D}) \propto \Pr^{(1)}(S, \boldsymbol{Y}, \boldsymbol{D})\ \text{(w.r.t. } S, \boldsymbol{Y}).$$

From this, the posterior probabilities are also consistent.

□

From Lemma 4.3.1, we obtain the following corollary.

Corollary 4.3.1 (Unchanged MAP state)**.**

$$\arg\max_{S,\boldsymbol{Y}} \Pr^{(1)}(S, \boldsymbol{Y}|\boldsymbol{D}) = \arg\max_{S,\boldsymbol{Y}} \Pr^{(2)}(S, \boldsymbol{Y}|\boldsymbol{D}),$$

where

$$\boldsymbol{Y} := (Y_1, \cdots, Y_N),\ \boldsymbol{D} := (D_1, \cdots, D_N)\,.$$

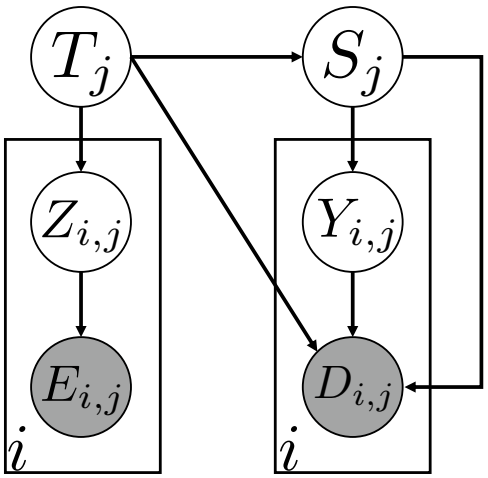
Bayes Factors from Posterior Probabilities

Some methods, e.g., Strelka2 and NeuSomatic return the output of posterior event probabilities. We can convert the posterior event probabilities to Bayes factors by setting the prior event ratio, e.g., Pr(tumor)/Pr(error) = 1 used in this paper.

$$\mathrm{BF} = \frac{P_{\mathrm{call}}(\mathrm{tumor}|D)}{P_{\mathrm{call}}(\mathrm{error}|D)} \frac{\Pr(\mathrm{error})}{\Pr(\mathrm{tumor})} = \frac{P_{\mathrm{call}}(\mathrm{tumor}|D)}{1 - P_{\mathrm{call}}(\mathrm{tumor}|D)} \frac{\Pr(\mathrm{error})}{\Pr(\mathrm{tumor})}. \tag{4.12}$$

4.3.5 Bayesian Statistical Model in MultiMuC

Based on the ideas shown above, we constructed the Bayesian statistical method termed MultiMuC and the graphical summary of MultiMuC is shown in Fig. 4.5.



$i = 1, \cdots, N$: Tumor data index.

$j = 1, \cdots, G$: Mutation candidate index.

Figure 4.5: Graphical summary of MultiMuC. i represents the location index of tumor sequence data and j represents the index of mutation candidate. The left side of the figure shows the evidence generation model and the right side of the figure shows the data generation model.

Our method is composed of an evidence generation model and a data generation model as shown in the left part and the right part in Fig. 4.5 respectively. For the evidence generation model, $E_{i,j}$ represents the i-th evidence around the j-th somatic mutation, $Z_{i,j} \in \{0, 1\}$ represents the existence of the j-th somatic mutation at the i-th evidence, and $T_j \in \{0, 1\}$ represents the existence of the j-th somatic mutation for at least one evidence. The distributions of these random variables are set as follows.

$$\Pr(T_j) = P_{\mathrm{ber}}(\cdot|0.5),$$
$$\Pr(Z_{i,j}|T_j) = P_{\mathrm{ber}}(\cdot|\epsilon)^{1-T_j} \cdot P_{\mathrm{ber}}(\cdot|0.5)^{T_j},$$
$$\Pr(E_{i,j}|Z_{i,j}) = 1^{1-Z_{i,j}} \cdot H_{i,j}^{Z_{i,j}}.$$

$H_{i,j}$ is the Bayes factor and we can detect a mutation with high specificity if $H_{i,j} > 1$. $\epsilon(\approx 0)$ corresponds to the false-positive rate (equal to $1-$specificity) for this Bayes factor of $H_{i,j}$. We note that we can regulate the distribution so that $T_j = 1$ only if we observe enough number of candidates by increasing ϵ.

For the data generation model, $D_{i,j}$ represents the i-th data set around the j-th mutation candidate, $Y_{i,j} \in \{0, 1\}$ represents the existence of the j-th somatic mutation at the i-th data set and $S_j \in \{0, 1\}$ represents the existence of the j-th somatic mutation for at least one data set. The distributions of these random variables are set as follows depending on T_j.

$$\Pr(S_j|T_j) = P_{\mathrm{ber}}(\cdot|0.5)^{1-T_j} P_{\mathrm{ber}}(\cdot|p_{\mathrm{con}})^{T_j},$$
$$\Pr(Y_{i,j}|S_j) = P_{\mathrm{ber}}(\cdot|\delta)^{1-S_j} \cdot P_{\mathrm{ber}}(\cdot|0.5)^{S_j},$$
$$\Pr(D_{i,j}|Y_{i,j}, S_j, T_j) = 1^{1-Y_{i,j}} \left(L_{i,j} \cdot 10^{\theta T_j} \cdot 10^{\rho S_j}\right)^{Y_{i,j}}.$$

$L_{i,j}$ is the Bayes factor that is generally used and δ corresponds to its false positive rate. $10^{\theta}(> 1)$ lowers the threshold of Bayes factors when the presence of a mutation can be predicted with high specificity ($T_j = 1$). $10^{\rho}(> 1)$ also lowers the threshold of Bayes factors when the presence of a mutation can be predicted from the usual result ($S_j = 1$). $p_{\mathrm{con}}(\approx 1)$ is the consistency rate from $T_j = 1$ to $S_j = 1$. In this paper, we used the following setting of hyperparameters: $\epsilon = 0.2$, $\delta = 0.02$, $\theta = 0.5$, $\rho = 0.1$ and $p_{\mathrm{con}} = 0.999$.

In this method we estimate the MAP state by MCMC [59] for each position j and use $\hat{Y}_{i,j}$ for mutation call.

$$\hat{\boldsymbol{Y}}_j, \hat{\boldsymbol{Z}}_j, \hat{S}_j, \hat{T}_j = \underset{\boldsymbol{Y}_j, \boldsymbol{Z}_j, S_j, T_j}{\arg\max} \Pr(\boldsymbol{Y}_j, \boldsymbol{Z}_j, S_j, T_j | D_{\cdot,j}, E_{\cdot,j}),$$
$$(\hat{\boldsymbol{Y}}_j := (\hat{Y}_{1,j}, \cdots, \hat{Y}_{N,j}),\ \hat{\boldsymbol{Z}}_j := (\hat{Z}_{1,j}, \cdots, \hat{Z}_{N,j})).$$

Preparation of Inputs

This method requires Bayes factors with high specificity in addition to the usual Bayes factor results. For preparation of these Bayes factors, we set threshold values and multiplied the original Bayes factor by the inverse number of the threshold as follows.

$$H_{i,j} = \mathrm{BF}_{i,j} \cdot 10^{-1.5},\ L_{i,j} = \mathrm{BF}_{i,j} \cdot 10^{a}, \tag{4.13}$$

where $\mathrm{BF}_{i,j}$ is the original Bayes factor outputs of the single-tumor-based method and 10^{-a} corresponds to the threshold value for the Bayes factor in general usage. For mutation calling with high specificity, we set $10^{1.5}$ as the threshold value. For MuTect2, we conducted $a = a - 6.3$ because of the default threshold setting in MuTect2.

4.3.6 MAP Inference in MultiMuC by MCMC

To estimate the MAP state, we sample random variables from the posterior distribution by Gibbs sampling. We obtain a set of random variables from the sampled sequence with the maximum posterior probability after the burn-in period. We show the conditional probabilities used for Gibbs sampling.

Gibbs Sampling for T_j

$$
\begin{aligned}
\Pr(T_j|S,Z,Y,E,D) &\propto \Pr(T_j,S,Z,Y,E,D) \\
&= \Pr(T_j)\Pr(S_j|T_j)\prod_j \Pr(Z_{i,j}|T_j)\Pr(D_{i,j}|Y_{i,j},S_j,T_j) \\
&= \left(\frac{1}{2}\right)^{1-T_j} \cdot \left(\frac{1}{2}\right)^{T_j} \\
&\quad \cdot P_{\mathrm{ber}}\left(S_j \middle| \frac{1}{2}\right)^{1-T_j} \cdot P_{\mathrm{ber}}\left(S_j \,|p_{\mathrm{con}}\right)^{T_j} \\
&\quad \cdot \prod_i P_{\mathrm{ber}}\left(Z_{i,j} \middle| \frac{1}{2}\right)^{T_j} \cdot P_{\mathrm{ber}}\left(Z_{i,j} \,|\epsilon\right)^{1-T_j} \\
&\quad \cdot \prod_i 1^{1-Y_{i,j}} \left(L_{i,j}10^{\theta\cdot T_j+\rho\cdot S_j}\right)^{Y_{i,j}} \\
&\propto \left[\frac{1}{2}\prod_i \left\{\epsilon^{Z_{i,j}}(1-\epsilon)^{1-Z_{i,j}}\right\}\right]^{1-T_j} \\
&\quad \cdot \left[p_{\mathrm{con}}^{S_j}(1-p_{\mathrm{con}})^{S_j}\prod_i \left\{\frac{1}{2}\cdot 10^{\theta\cdot y_{i,j}}\right\}\right]^{T_j}.
\end{aligned}
$$

Gibbs Sampling for $Z_{i,j}$

$$
\begin{aligned}
\Pr(Z_{i,j}|T,E) &\propto \Pr(Z_{i,j},T,E) \\
&= \Pr(Z_{i,j}|T_j)\Pr(E_{i,j}|Z_{i,j}) \\
&= \left[\epsilon^{Z_{i,j}}(1-\epsilon)^{1-Z_{i,j}}\right]^{(1-T_j)} \\
&\quad \cdot \left[\left(\frac{1}{2}\right)^{Z_{i,j}} \cdot \left(\frac{1}{2}\right)^{1-Z_{i,j}}\right]^{T_j} \\
&\quad \cdot 1^{1-Z_{i,j}} H_{i,j}^{Z_{i,j}} \\
&\propto \left[(1-\epsilon)^{1-T_j}\right]^{1-Z_{i,j}} \left[\epsilon^{1-T_j}H_{i,j}\right]^{Z_{i,j}}.
\end{aligned}
$$

Gibbs Sampling for S_j

$$
\begin{aligned}
\Pr(S_j|T,Y,D) &\propto \Pr(S_j|T_j)\prod_i \Pr(Y_{i,j}|S_j)\Pr(D_{i,j}|S_j,T_j,Y_{i,j}) \\
&= P_{\mathrm{ber}}\left(S_j \middle| \frac{1}{2}\right)^{1-T_j} \cdot P_{\mathrm{ber}}\left(S_j \,|p_{\mathrm{con}}\right)^{T_j} \\
&\quad \cdot \prod_i \left[\delta^{Y_{i,j}}(1-\delta)^{1-Y_{i,j}}\right]^{1-S_j} \left[\left(\frac{1}{2}\right)^{Y_{i,j}} \left(\frac{1}{2}\right)^{1-Y_{i,j}}\right]^{1-S_j} \\
&\quad \cdot \prod_i 1^{1-Y_{i,j}} \left(L_{i,j}10^{\theta T_j+\rho S_j}\right)^{Y_{i,j}} \\
&\propto \left[(1-p_{\mathrm{con}})^{T_j} \cdot \prod_i \delta^{Y_{i,j}}(1-\delta)^{1-Y_{i,j}}\right]^{1-S_j}
\end{aligned}
$$

$$\cdot \left[p_{\text{con}}^{T_j} \cdot \prod_i \frac{1}{2} \cdot \left(L_{i,j} 10^{\theta T_j + \rho} \right)^{Y_{i,j}} \right]^{S_{i,j}}.$$

Gibbs Sampling for $Y_{i,j}$

$$\begin{aligned}
\Pr(Y_{i,j}|S_j, D_{\cdot,j}) &\propto \Pr(Y_{i,j}|S_j)\Pr(D_{i,j}|S_j, T_j, Y_{i,j}) \\
&= \left[\delta^{Y_{i,j}} \cdot (1-\delta)^{1-Y_{i,j}}\right]^{1-S_j} \cdot \left[\left(\frac{1}{2}\right)^{Y_{i,j}} \cdot \left(\frac{1}{2}\right)^{1-Y_{i,j}} \right]^{S_j} \\
&\quad \cdot 1^{1-Y_{i,j}} \cdot \left[10^{\theta T_j + \rho S_j}\right]^{Y_{i,j}} \\
&\propto \left[(1-\delta)^{1-S_j}\right]^{1-Y_{i,j}} \left[\delta^{1-S_j} \cdot L_{i,j} \cdot 10^{\theta T_j + \rho S_j}\right]^{Y_{i,j}}.
\end{aligned}$$

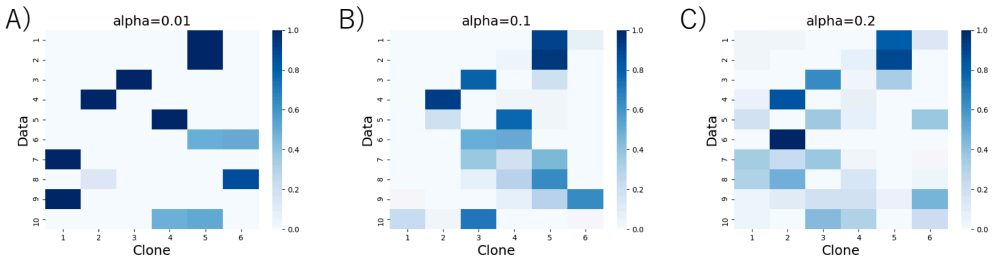


Figure 4.6: Examples of simulated clonal mixture rates. A) illustrates the case of $\alpha = 0.01$ and B) illustrates the case of $\alpha = 0.1$, and C) illustrates the case of $\alpha = 0.2$.

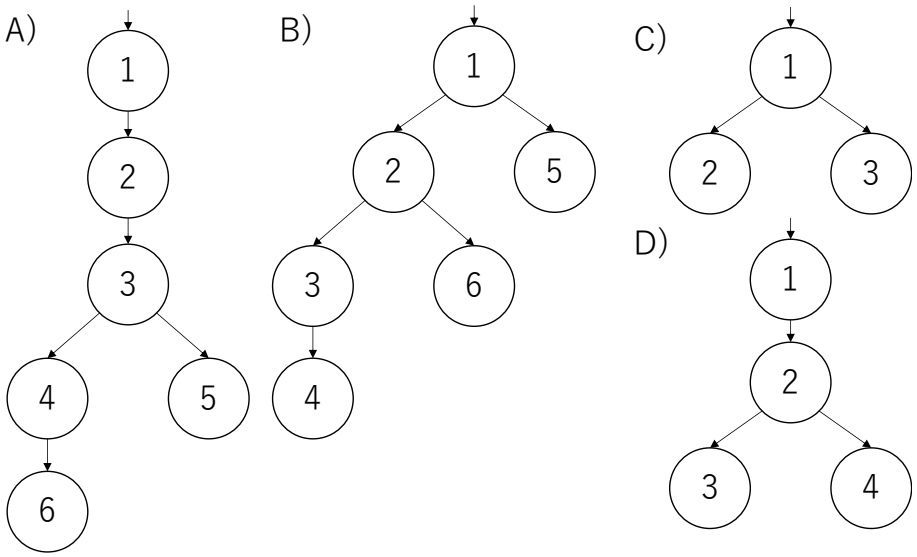


Figure 4.7: Simulated trees used for evaluations. Each numbered node corresponds to a clone and each edge corresponds to a non-empty set of somatic mutations. For each simulated data, we sampled a mixture rate of clones and simulated bulk sample data sets.

4.4 Results

4.4.1 Simulation Experiments Based on Real Data Sets

We evaluated MultiMuC performance by simulating multiple tumor sequence datasets. To do this, we used multiple settings for both the tumor phylogenetic tree and the mixture rate of clones, where a clone means a type of tumor cell population. These datasets were prepared in 24 different configurations. Fig. 4.6 and Fig. 4.7 show the examples of the mixture composition rates and tumor phylogenetic trees that were used. The simulation procedures were as follows.

1) Collect true somatic mutations and sequence errors from a single pure tumor ($= t_{\text{original}}$) and a matched pure normal ($= n_{\text{original}}$) data set.

2) Filter out true mutations with allele frequencies of <30% or >70% for allele frequencies to decrease from $\sim 50\%$ following the phylogenetic tree.

3) Generate a random phylogenetic tree $\mathcal{T}$.

4) Randomly relate each somatic mutation with an edge of the tree $\mathcal{T}$.

5) For each tumor simulation data set, we generated reads as follows for 10 tumor data sets.

   5-a) Sample a mixture rate of clones $\boldsymbol{p}_{\text{mix}} \sim \text{Dirichlet}(\cdot|(\alpha, \cdots, \alpha))$.

   5-b) For each true somatic mutation s, calculate the total population of clone $p_{\text{tumor}} = \sum_{i \in A} p_{\text{mix},i}$, $A := \{i | i\text{-th clone has mutation } s\}$.

   5-c) Collect reads around the true somatic mutation of s from t_{original} at the down sampling rate of p_{tumor} and from n_{original} at the rate of $1 - p_{\text{tumor}}$.

   5-d) For each error position e, sample an error rate $p_{\text{error}} \sim \text{Beta}(\cdot|0.1, 0.1)$.

   5-e) Collect reads around the error position of e from t_{original} at the rate of p_{error} and from n_{original} at the rate of $1 - p_{\text{error}}$.

For t_{original} and n_{original}, we used real data sets from TCGA 4 mutation calling benchmark datasets. For the datasets, see `https://gdc.cancer.gov/resources-tcga-users/tcga-mutation-calling-benchmark-4-files`.

Performance Comparison

We conducted a performance comparison based on F-measure. We used Strelka2, MuTect2, NeuSomatic and OHVarfinDer as Bayes factor inputs. For the counterpart method, we prepared multiSNV and Treeomics. We summarized the F-measures of these methods at $a = 0.0$ in Fig. 4.8. In this figure, +M indicates that our method was used. Our method steadily contributes to performance improvement for Strelka2, NeuSomatic and OHVarfinDer and does not cause performance degradation for MuTect2 (Fig. 4.8-B). Furthermore, the combined output of our method and single-tumor-based methods outperformed both multiSNV and Treeomics (Fig. 4.8-A). To see the performance of original mutation calling methods, we summarized original recalls, precisions and F-measures in Figs. 4.9 to 4.11. From this, the reason for no statistically significant performance gain in MuTect2 may be due to the increase of recalls at MuTect2 being smaller than that of the other methods used, as shown in Fig. 4.9.

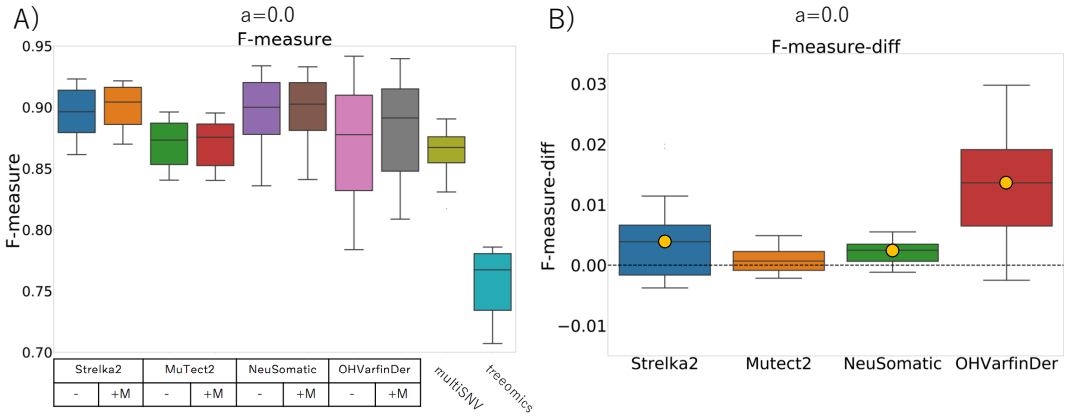


Figure 4.8: The summary of F-measure at $a = 0.0$. 10^{-a} is the Bayes factor threshold for mutation call as shown in Eq. (4.13). +M represents the use of MultiMuC, and an orange-colored circle represents a positive difference of F-measure on average with a P-value less than 0.01 (two-sided paired t-test). A) represents the summary of F-measure with the threshold at $a = 0.0$. B) represents the difference of F-measure by applying MultiMuC with the threshold at $a = 0.0$.

Effects of MultiMuC at Different Thresholding of 10^{-a}

To confirm that our method does not degrade the performance of detection for different thresholding values of 10^{-a}, we evaluated the difference in F-measure, recall, and precision by applying MultiMuC as shown in Figs. 4.12 to 4.14. MultiMuC can achieve statistically significant improvement on F-measures in most of the cases, except for MuTect2. The reason for no statistically significant performance improvement in MuTect2 may also be because we cannot achive higher recall in MuTect2 by only increasing the detection threshold at any a (Fig. 4.9).

Effects of Evidence Generation Model in MultiMuC at Different Thresholding of 10^{-a}

MultiMuC leverages the specificity of detection and the number of detected candidates through the evidence generation model. Here, we investigate the effects of evidence generation model in MultiMuC at different threshold value of 10^{-a}. In (+E), we sets $\epsilon = 0.2$, $\delta = 0.02$, $\theta = 0.5$, $\rho = 0.1$ and $p_{\text{con}} = 0.999$. In (-E), we sets $\epsilon = 0.2$, $\delta = 0.02$, $\theta = 0.6$, $\rho = 0.0$ and $p_{\text{con}} = 0.5$. In (-E), evidence generation model does not affect the inference of $\boldsymbol{Y}_j$ due to $\rho = 0.0$ and $p_{\text{con}} = 0.5$. Under these settings, we evaluated the difference in F-measure, recall, and precision by applying MultiMuC as shown in Figs. 4.15 to 4.17.

The most striking difference between (+E) and (-E) is the effect on precisions. As we can see from the Fig. 4.16, evidence generation model can suppress the degradation of precisions. In spite of the suppressed degradation of precision, MultiMuC with (+E) setting still improves the recall (Fig. 4.15) in most cases. For the difference of F-measures, (+E) model can improve F-measures while suppressing the drastic performance degradation which appears in the case of Strelka2 (Fig. 4.17). From this, we can see that the evidence generation model can avoid performance degradations by suppressing the degradation of precisions.

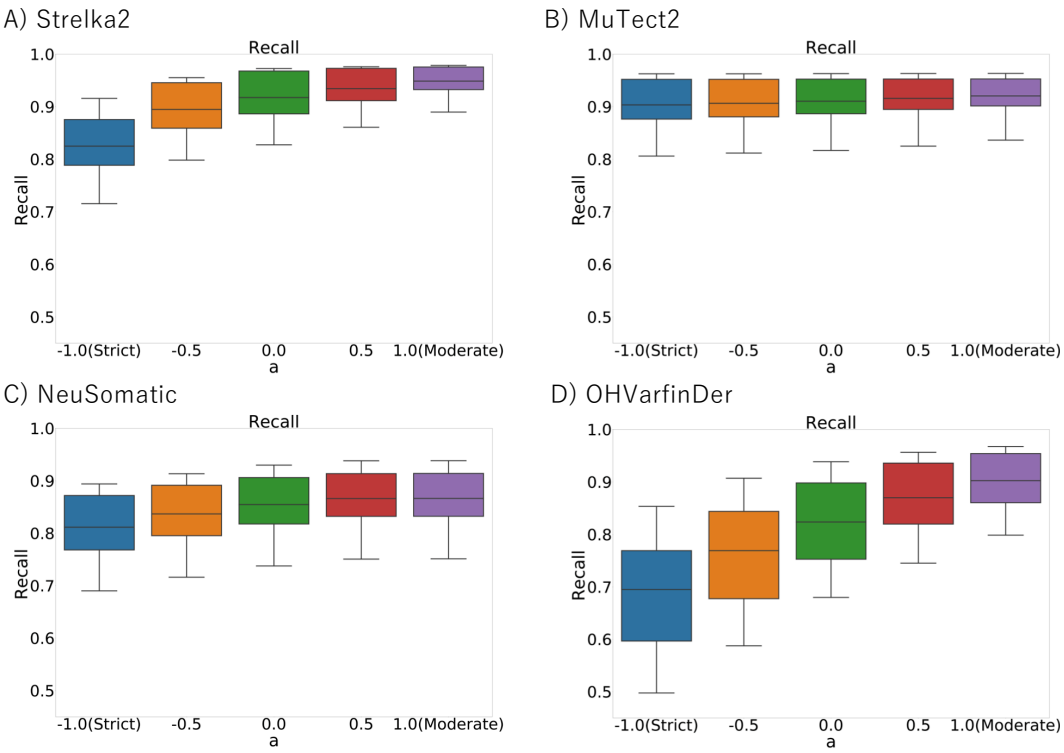


Figure 4.9: Summary of recalls in the original mutation calling methods at different default threshold values of 10^{-a}, where $a \in \{-1.0, -0.5, 0.0, 0.5, 1.0\}$. A) at Strelka2. B) at MuTect2. C) at NeuSomatic. D) at OHVarfinDer.

4.5 Discussion

In this chapter, we propose a Bayesian method for multi-regional mutation call based on the mutation sharing assumption with two characteristics. First, our method avoids the No-TP case by considering both the specificity of detection and the number of detected candidates to avoid performance degradation. Second, our method can incorporate scores from state-of-the-art mutation calling methods for a single-regional tumor if scores are based on probabilities except for P-values. This performance improvement of mutation call will contribute to an improved inference of tumor phylogeny.

For future work, we would like extend our method to handle the mutation calling results which are based on P-values. With this method, we can use the outputs of single-tumor-based methods if the posterior event probability or Bayes factor is available. However, our method cannot handle P-value-based outputs of some single-tumor-based methods [39, 40, 72, 81, 58] although P-value is a useful measure for decision making.

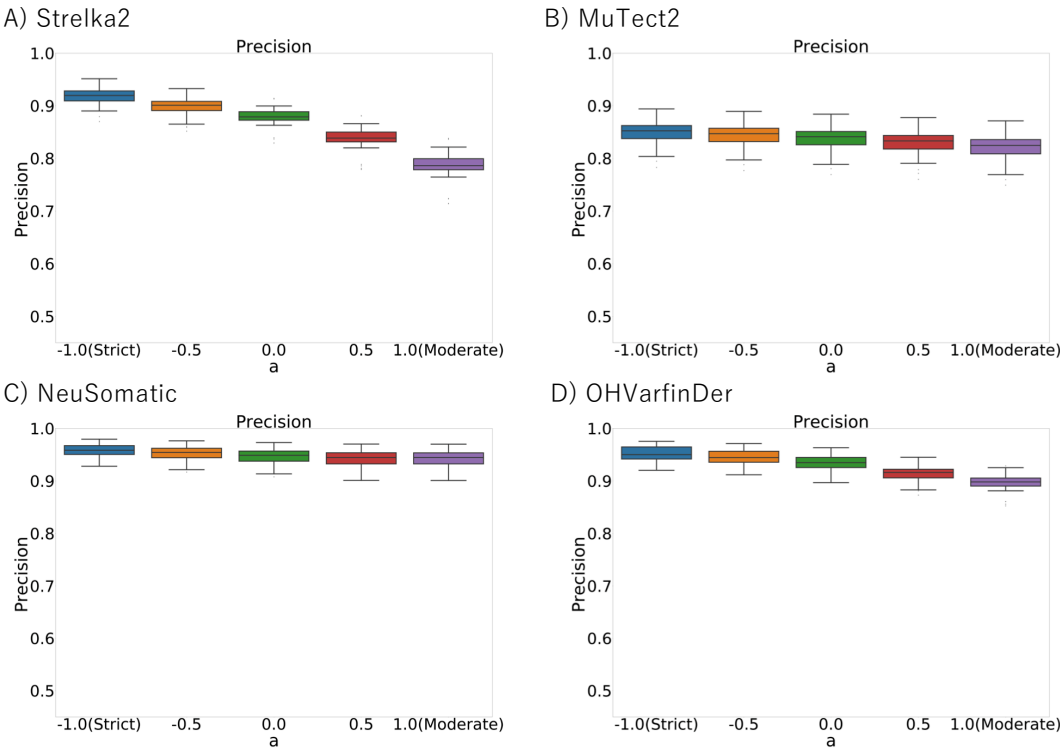


Figure 4.10: Summary of precisions in the original mutation calling methods at different default threshold values of 10^{-a}, where $a \in \{-1.0, -0.5, 0.0, 0.5, 1.0\}$. A) at Strelka2. B) at MuTect2. C) at NeuSomatic. D) at OHVarfinDer.

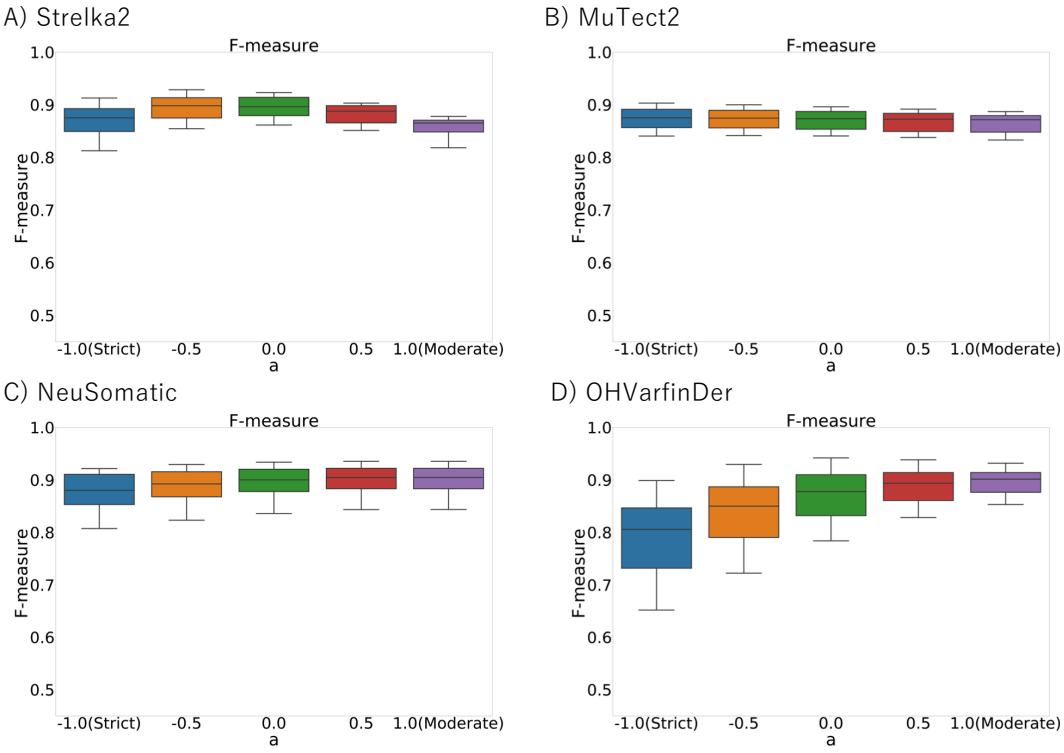


Figure 4.11: Summary of F-measures in the original mutation calling methods at different default threshold values of 10^{-a}, where $a \in \{-1.0, -0.5, 0.0, 0.5, 1.0\}$. A) at Strelka2. B) at MuTect2. C) at NeuSomatic. D) at OHVarfinDer.

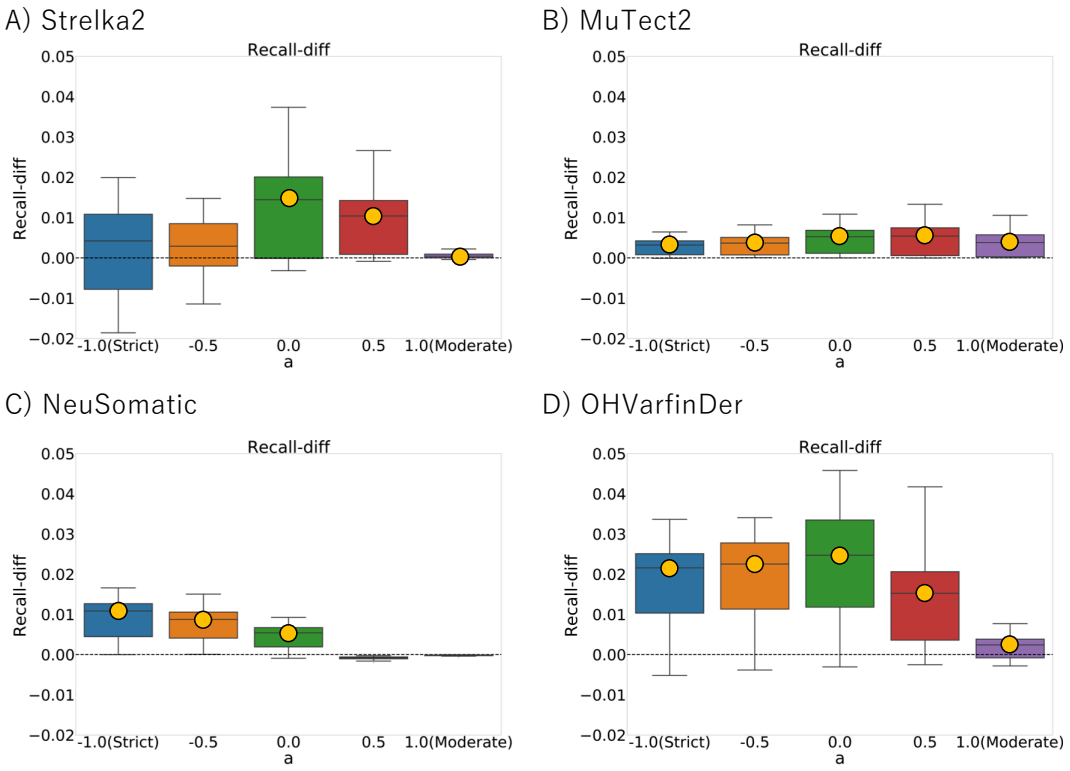


Figure 4.12: Summary of the difference in recall by applying MultiMuC in the different thresholding values of 10^{-a}, where $a \in \{-1.0, -0.5, 0.0, 0.5, 1.0\}$. An orange-colored circle shows that the average recall difference is positive and the P-value of the two-sided paired t-test is less than 0.01. A) at Strelka2. B) at MuTect2. C) at NeuSomatic. D) at OHVarfinDer.

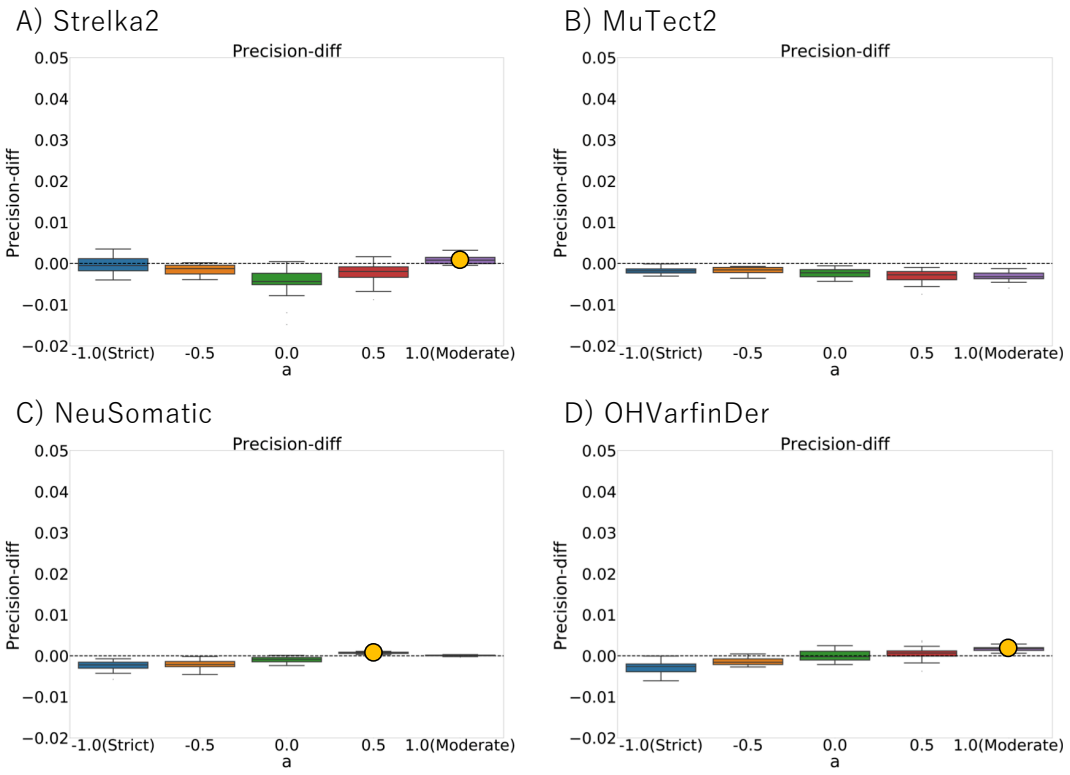


Figure 4.13: Summary of the difference in precision by applying MultiMuC in the different thresholding values of 10^{-a}, where $a \in \{-1.0, -0.5, 0.0, 0.5, 1.0\}$. An orange-colored circle shows that the average precision difference is positive and the P-value of the two-sided paired t-test is less than 0.01. A) at Strelka2. B) at MuTect2. C) at NeuSomatic. D) at OHVarfinDer.

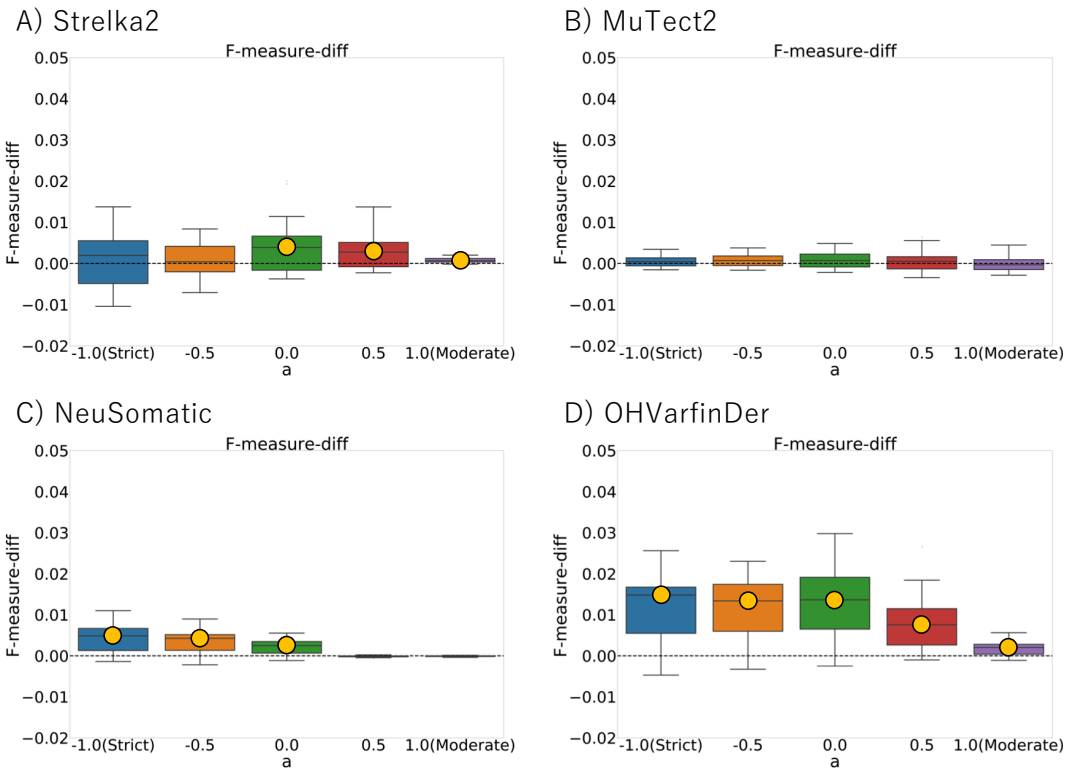


Figure 4.14: Summary of the difference in F-measure by applying MultiMuC in the different thresholding values of 10^{-a}, where $a \in \{-1.0, -0.5, 0.0, 0.5, 1.0\}$. An orange-colored circle shows that the average F-measure difference is positive and the P-value of the two-sided paired t-test is less than 0.01. A) at Strelka2. B) at MuTect2. C) at NeuSomatic. D) at OHVarfinDer.

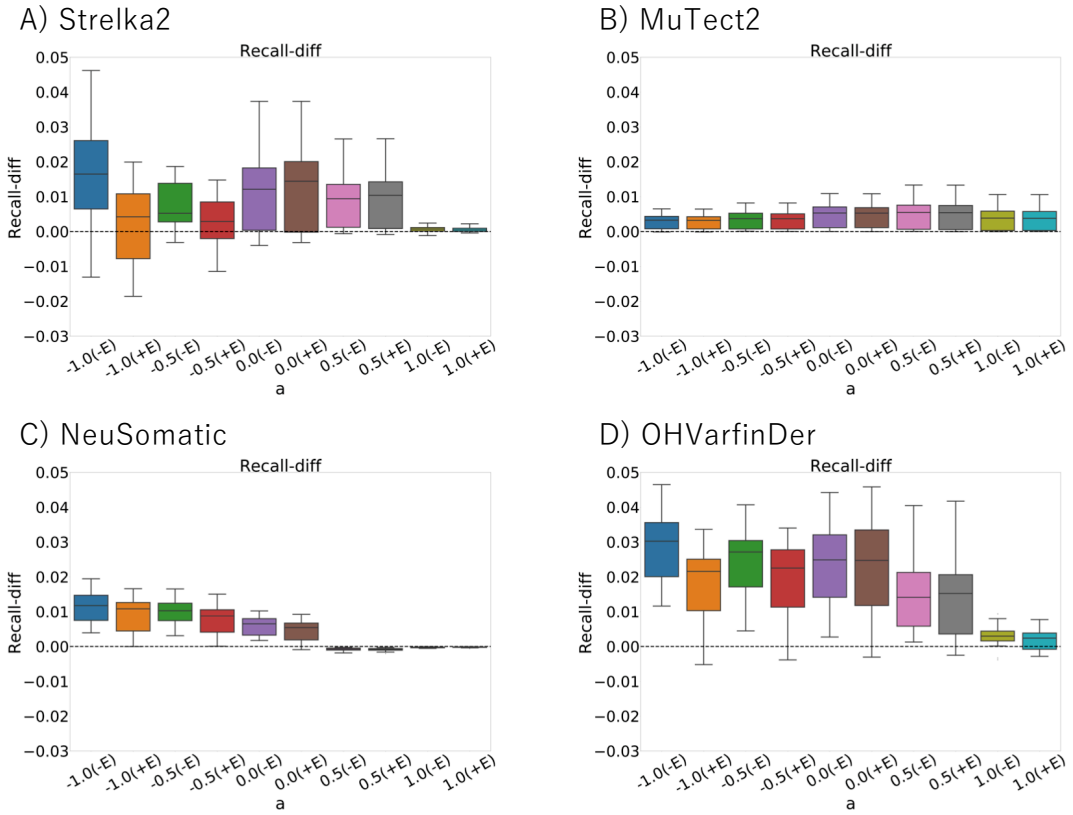


Figure 4.15: Summary of the difference in recall by applying MultiMuC in two different settings of (+E) and (-E) at the different thresholding values of 10^{-a}, where $a \in \{-1.0, -0.5, 0.0, 0.5, 1.0\}$. (-E) express that the evidence generation model is removed from the MultiMuC model and (+E) represents that the evidence generation model is incorporated in the MultiMuC model. The asterisk represents a positive difference ((+E) minus (-E)) of recalls on average with P-value less than 0.01 (two-sided paired t-test).

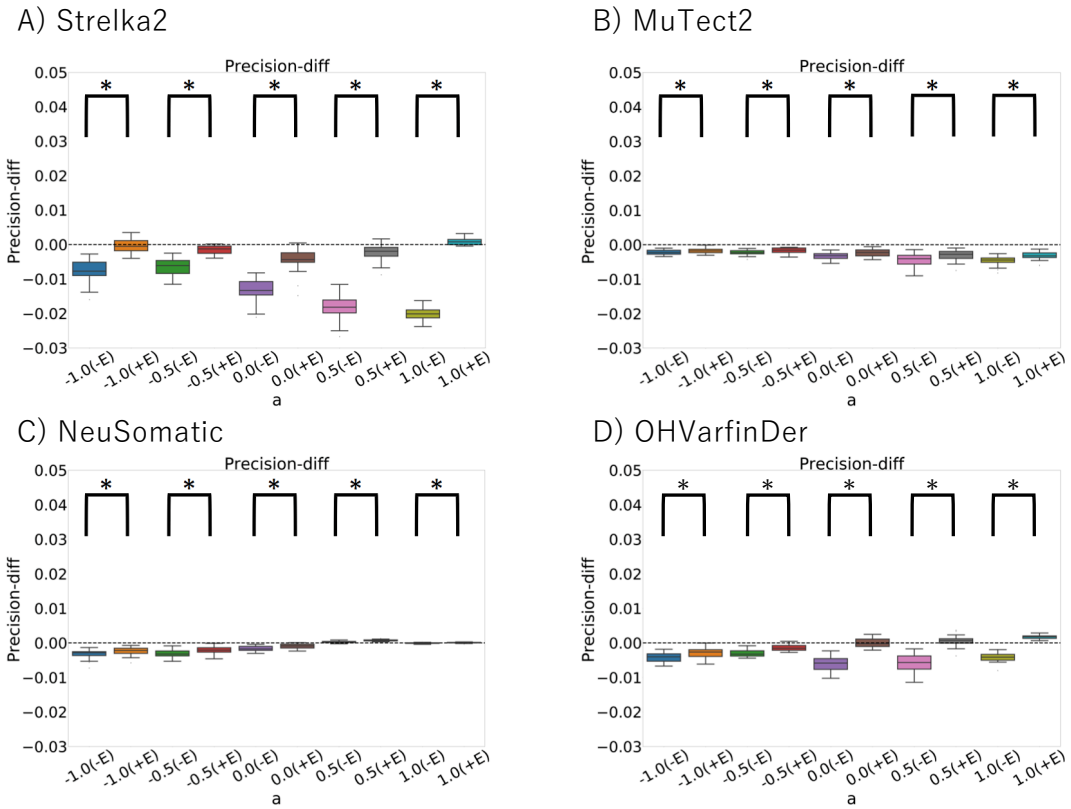


Figure 4.16: Summary of the difference in precision by applying MultiMuC in two different settings of (+E) and (-E) at the different thresholding values of 10^{-a}, where $a \in \{-1.0, -0.5, 0.0, 0.5, 1.0\}$. (-E) express that the evidence generation model is removed from the MultiMuC model and (+E) represents that the evidence generation model is incorporated in the MultiMuC model. The asterisk represents a positive difference ((+E) minus (-E)) of precisions on average with P-value less than 0.01 (two-sided paired t-test).

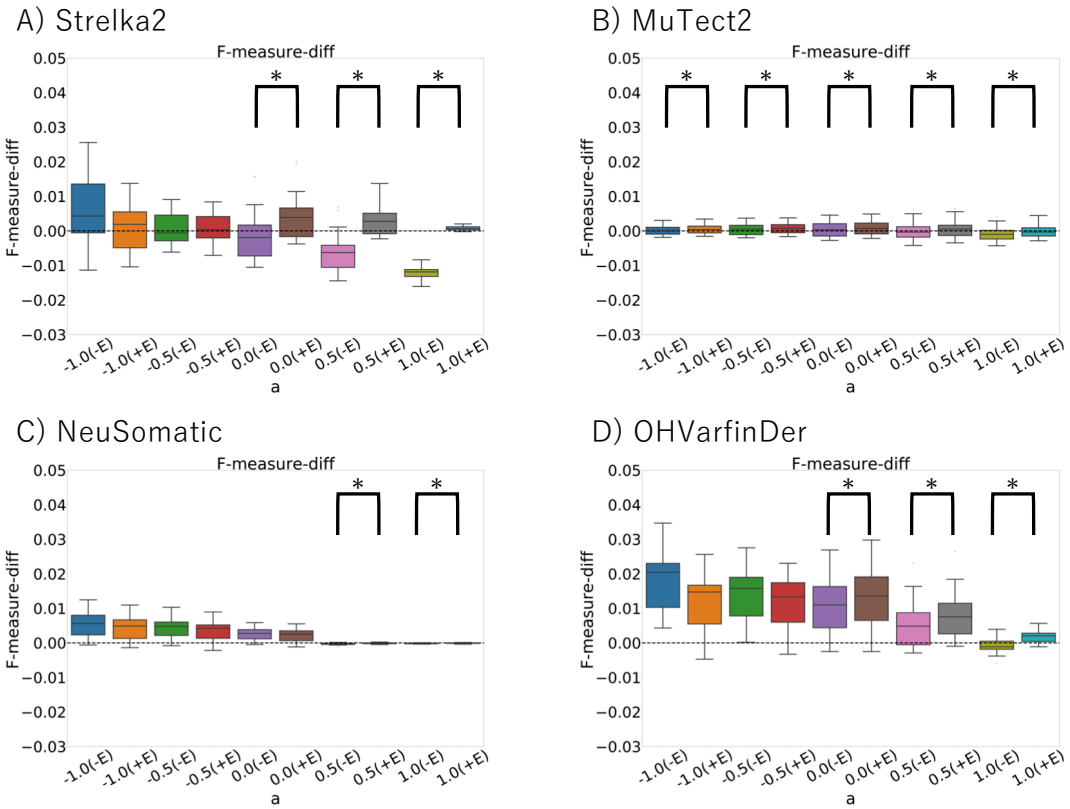


Figure 4.17: Summary of the difference in F-measure by applying MultiMuC in two different settings of (+E) and (-E) at the different thresholding values of 10^{-a}, where $a \in \{-1.0, -0.5, 0.0, 0.5, 1.0\}$. (-E) express that the evidence generation model is removed from the MultiMuC model and (+E) represents that the evidence generation model is incorporated in the MultiMuC model. The asterisk represents a positive difference ((+E) minus (-E)) of F-measures on average with P-value less than 0.01 (two-sided paired t-test).

Chapter 5

Properties of Tumor Phylogeny for Accurate Mutation Call

5.1 Overview

Cancer is known as genomic disease, and tumor tissue is a heterogeneous population of cancer cells in general cases. Because this heterogeneity leads to drug resistance, assessment of the heterogeneity is important in cancer therapies. To understand the heterogeneity, multi-regional mutation call is conducted in general.

There exist mainly two types of methods that are designed particularly for multi-regional mutation call. The first type of method uses the assumption of mutation sharing [32], which is mentioned in Chapter 4. The second type of method is based on the property of tumor phylogeny [65, 15, 78, 68].

Comprehensive performance comparison to these multi-regional mutation calling methods was recently conducted [13]. Within this study, a comparison against single-tumor-based mutation calling methods is also conducted. In several experimental settings, phylogeny-based methods show poor performance even to single-tumor-based mutation calling methods. However, these results are inconsistent with the previous reports [65, 15, 78, 68].

From these studies, we can expect two cases in which tumor phylogeny works and does not work for performance improvement. However, existing studies do not reveal when and how much tumor phylogeny works for detection performance, and it remains ambiguous whether or not we can improve detection performance by leveraging tumor phylogeny.

In this chapter, we assume that we can predict a somatic mutation at each tumor region with 100% sensitivity, and under this assumption, we evaluate the performance of predicting a mutation in a patient (not in each tumor region). This chapter is organized as follows. First, we show problem settings, including the assumption, and then, we evaluate the specificity and sensitivity in two cases: tumor phylogeny is used and not used. Finally, we consider other assumptions about insufficient depth coverage and also evaluate the specificity and sensitivity.

5.2 Related Works

Here, we introduce existing methods of Treeomics, SNV-PPILP, and MuClone for multi-regional mutation call. These methods consider the existence of tumor phylogeny and hence use the property that the total pattern of column vectors in the mutation profiles is limited. This is because that the total pattern of column vectors in the mutation profiles is limited if the observed mutation profile has a corresponding phylogenetic tree.

To demonstrate this point, we show the property that the patterns of column vectors are limited. We should note that this property was not shown clearly in the original reports.

5.2.1 Basic Ideas of Leveraging Phylogeny

Lemma 5.2.1 (Existence of a full binary phylogenetic tree)**.**

If $T \in \{0,1\}^{c\times k}$ *has a phylogenetic tree, then*
$\exists \mathcal{T} = (V, E)$ *s.t.* $\mathcal{T}$ *satisfies the following conditions,*
a) $\mathcal{T}$ *is a phylogenetic tree for* T*,*
b) $|F_{\mathcal{T}}| \leq c$*,*
c) The root node has only one outgoing edge,
d) Any node except for the root has zero or two outgoing edges,
where V *is a set of vertices,* E *is a set of edges, and*
$F_{\mathcal{T}}$ *is a set of leaves in* $\mathcal{T}$*.*

Proof. T has a phylogenetic tree, hence we can choose a phylogenetic tree $\mathcal{T}$. We can assume $|F_{\mathcal{T}}| \leq c$ by removing leaves in $\mathcal{T}$ if no cell corresponds to the leaf in $f : R \to F_{\mathcal{T}}$. We can also assume that the root node has only one outgoing edge by adding new root node and connect the novel root and the previous root node.

For the last condition, we remove the following two types of internal nodes: i) the internal node having only one outgoing edge, and ii) the internal node having more than 2 outgoing edges. It is sufficient to show the operation to remove nodes that satisfy i) or ii) from T while keeping conditions of a)-c).

For i), just remove the nodes as in Fig. 5.1. We can easily check a)-c) still holds true after this operation. For ii) just remove the node as in Fig. 5.2. If the number of outgoing edges is more than three, apply this operation recursively. We can also check that a)-c) still hold true after these operations. □

Theorem 5.2.1 (Patterns of column vectors in phylogenetic matrix)**.**

If $T \in \{0,1\}^{c\times k}$ *has a phylogenetic tree, then*
$|\{\boldsymbol{t}_i | i = 1, \cdots k\}| \leq 2c - 1$*, where* $\boldsymbol{t}_i$ *is the* i*-th column vector of* T*.*

Proof. Because of the definition of the phylogenetic tree under the infinite sites assumption, any mutation in T corresponds to an edge in $\mathcal{T}$. If one mutation corresponds to an edge in $\mathcal{T}$, we find one pattern from column vectors in T. If no mutation corresponds, we find no patterns from the column vectors. Therefore, $|\{\boldsymbol{t}_i | i = 1, \cdots k\}| \leq |E|$. Because of Lemma 5.2.1, we can assume that the root of $\mathcal{T}$ is connected to the root of a full binary tree at which the number of leaves is $\leq c$. From this, $|E| \leq 2c - 1$. □

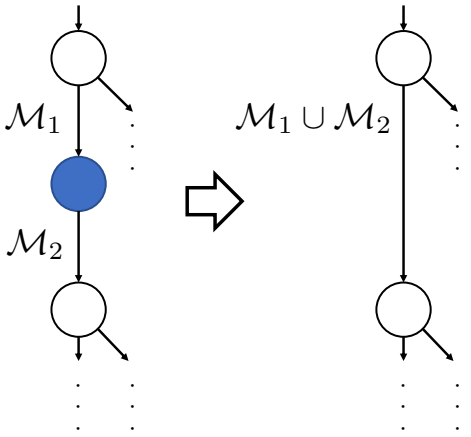


Figure 5.1: A procedure of removing a node having only one outgoing edge.

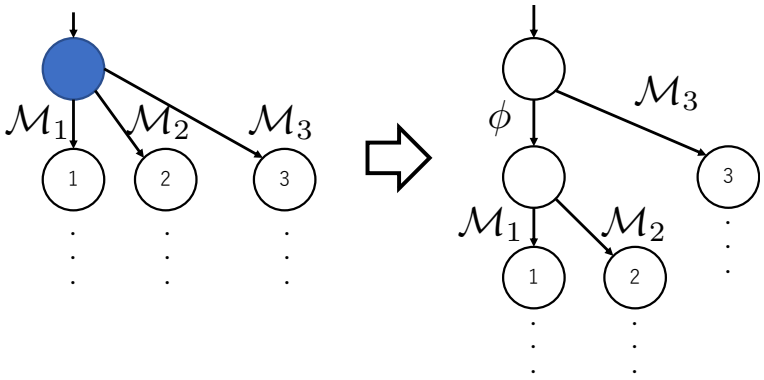


Figure 5.2: A procedure of removing a node having more than two outgoing edges.

Corollary 5.2.1 (Observable cell types and column vectors)**.** *Let $T \in \{0,1\}^{c\times k}$ have a phylogenetic tree, and $U \in \mathbb{R}_{\geq 0}^{n\times c}, U_1 \in \mathbb{R}_{\geq 0}^{n\times c_1}$ have a non-negative simplex for every row vector, where $U = \begin{bmatrix} U_1 & O \end{bmatrix}$. Then, the pattern of column vectors of $(\boldsymbol{y}_1, \cdots \boldsymbol{y}_k)$ in $Y := \frac{1}{2}UT$ is limited as follows.*

$$|\{\boldsymbol{y}_i | i = 1, \cdots k\}| \leq 2c_1 - 1.$$

Proof. We split T by row as follows.

$$T = \begin{bmatrix} T_1 \\ T_2 \end{bmatrix},$$

where $T_1 \in \{0,1\}^{c_1\times k}$,$T_2 \in \{0,1\}^{(c-c_1)\times k}$. Then, Y is as follows.

$$Y = \frac{1}{2}UT = \frac{1}{2}U_1T_1 = \frac{1}{2}(U_1\tilde{\boldsymbol{t}_1} \cdots U_1\tilde{\boldsymbol{t}_k}),$$

where $\tilde{\boldsymbol{t}_1}, \cdots, \tilde{\boldsymbol{t}_k}$ is the column vectors in T_1. Because the equivalent conditions listed in Lemma 2.7.1 hold true for the subset of rows, both T_1 and T_2 have a phylogenetic tree. From Theorem 5.2.1, total patterns of column vectors in T_1 is limited by $2c_1 - 1$. Then,

$$|\{\boldsymbol{y}_i | i = 1, \cdots k\}| \leq 2c_1 - 1.$$

□

From Corollary 5.2.1, the total pattern of column vectors in the matrix of variant allele frequencies are limited by the number of observable cell types.

5.2.2 MuClone

MuClone [15] relies on the limited observation of tumor cell types. MuClone constructs a Bayesian clustering model for somatic mutations. Their Bayesian modeling restricts the pattern of column vectors in the mutation profile for true mutations by setting mutation clusters and candidates that do not correspond to the mutation clusters are assigned to a unique error cluster. Within each mutation cluster, a specific set of samples can have the mutation more likely. On the other hand, within the error cluster, any sample can have the error uniformly at random.

5.2.3 Treeomics and SNV-PPILP

Treeomics [65] and SNV-PPILP [78] rely on the idea of [68] that considers the existence of tumor phylogeny within a set of true mutations. They retrieve a maximum evolutionary compatible set of mutations from a given mutation profile and guarantee that all the retrieved mutations truly correspond to the edge of the tumor phylogenetic tree under the infinite sites assumption. Based on the retrieved evolutionary compatible set, they also detect mutations with low allele frequency.

Let $X \in \{0,1\}^{n \times k}$ be a mutation profile, $\mathcal{M}$ be a set of indices for mutations in X, $\mathcal{S}$ be a set of indices for samples in X. We prepare the notation for an array slicing-like matrix of X as $X[\mathcal{S}_1, \mathcal{M}_1] \in \{0,1\}^{|\mathcal{S}_1| \times |\mathcal{M}_1|}$, where $\mathcal{S}_1 \subseteq \mathcal{S}$, $\mathcal{M}_1 \subseteq \mathcal{M}$, and $X[\mathcal{S}_1, \mathcal{M}_1]$ is a matrix made by removing rows at $\mathcal{S} \setminus \mathcal{S}_1$ and colums at $\mathcal{M} \setminus \mathcal{M}_1$. The problem formulation can be represented as follows.

$$\max_{\mathcal{M}_1 \subseteq \mathcal{M}} |\mathcal{M}_1|$$

subject to:

$$X[\mathcal{S}, \mathcal{M}_1] \text{ has a phylogenetic tree.}$$

To solve this problem, existing methods retrieve a maximum evolutionary compatible set by mixed-integer linear programming because this problem is known to be NP-hard.

5.3 Problem Settings and Assumptions

5.3.1 Given Mutation Profiles

We assume that two types of mutation profiles are given as shown in Fig. 5.3. The first mutation profile is a reliable profile, e.g., a mutation profile estimated by a short-read sequencer in multi-regional tumors, and express the first profile as $A \in \{0,1\}^{n \times k}$, where n is the number of sequenced samples and k is the number of mutations. $A_{n',k'} = 1$ means that the mutation candidate exists at the k'-th genomic position in the n'-th data set. For simplicity, we assume that each column vector of A is sorted in descending order, and represent the i-th column vector as $\boldsymbol{a}_i$.

$$A = (\boldsymbol{a}_1 \cdots \boldsymbol{a}_k).$$

The second profile is an unreliable profile, e.g., a mutation profile estimated by a long-read sequencer, and this profile contains erroneous positions at which no tumor regions have the true mutation. We describe the second profile as $B \in \{0,1\}^{n \times (k_1 + k_2)}$, where k_1 is the number of non-erroneous positions and k_2 is

the number of erroneous positions. We make a mutation profile $C \in \{0,1\}^{n \times k_1}$ by collecting only non-erroneous positions from B and make error profile $Z \in \{0,1\}^{n \times k_2}$ by collecting erroneous positions. For simplicity, we assume that each column vector of B,C,Z is sorted in descending order, and represent the j-th column vector as $\boldsymbol{b}_j, \boldsymbol{c}_j, \boldsymbol{z}_j$.

$$
\begin{aligned}
B &= (\boldsymbol{b}_1 \cdots \boldsymbol{b}_{k_1+k_2}), \\
C &= (\boldsymbol{c}_1 \cdots \boldsymbol{c}_{k_1}), \\
Z &= (\boldsymbol{z}_1 \cdots \boldsymbol{z}_{k_2}).
\end{aligned}
$$

The purpose here is to label the mutations of B by using A. That is, we judge whether each j-th column vector $\boldsymbol{b}_j$ belongs to C or Z.

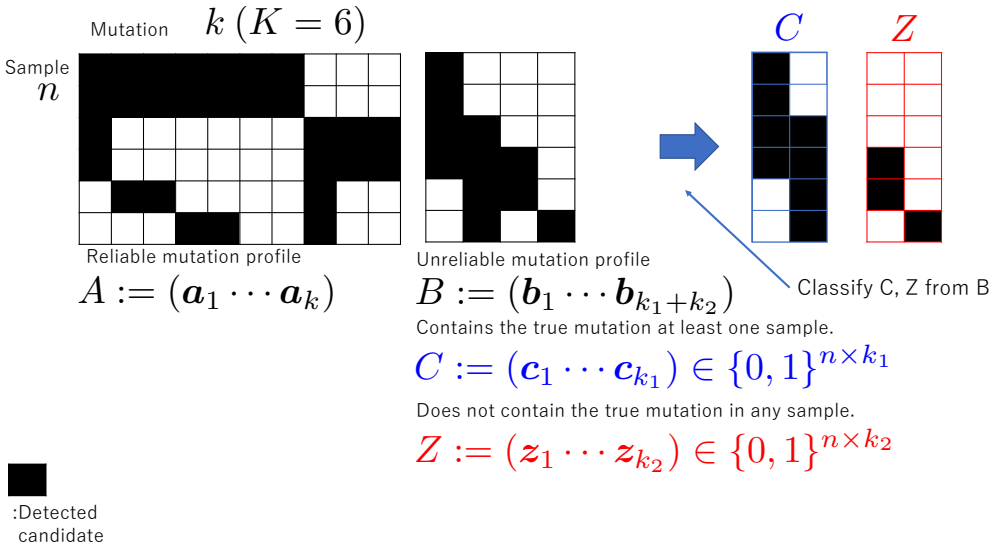


Figure 5.3: A graphical summary of the problem settings. In this problem setting, we have a reliable mutation profile A and an unreliable mutation profile B. Within column vectors in B, two types of column vectors exist: column vectors with at least one true mutations (those in C) and those without any mutations (those in Z). The purpose in this problem setting is to label each column vectors of B: from C or from Z.

5.3.2 Assumptions for Given Profiles

We assume a binary matrix $T \in \{0,1\}^{c \times k}$ and clonal mixture matrix $U \in \mathbb{R}_{\geq 0}^{n \times c}$, where c is the number of leaves in the phylogenetic tree. For T, we assume that the tumor phylogeny satisfies the infinite sites assumption and T has a corresponding phylogenetic tree with c leaves (Section 2.7). We also assume that row vectors are disjoint. For U, every row vector of U is a simplex vector. From T and U, we assume that A is generated. That is,

$$
A_{i,j} = h\left(\left(\frac{1}{2}UT\right)_{i,j}\right),
$$

where

$$
h(x) = \begin{cases} 0 & (x \leq 0) \\ 1 & (x > 0) \end{cases}.
$$

For $C \in \{0,1\}^{n \times k_1}$ and $Z \in \{0,1\}^{n \times k_2}$, we assume that each column vector is independently generated by the stochastic models below.

$$
\begin{aligned}
I_j &\sim \mathrm{Unif}(\cdot|1,k), \\
\xi_{j,i} &\sim \mathrm{Ber}(\cdot|f_1), \\
c_{j,i} &= \max(a_{I_j,i} + \xi_{j,i}, 1), \\
z_{l,i} &\sim \mathrm{Ber}(\cdot|f_2),
\end{aligned}
$$

where $\mathrm{Unif}(\cdot|a,b)$ represents the discrete uniform distribution with the range from $a \in \mathbb{Z}$ up to $b \in \mathbb{Z}$, $\mathrm{Ber}(\cdot|f)$ is the Bernoulli distribution with frequency of f, $0 < f_1 < 1$, $0 < f_2 < 1$, $j \in \{1, \cdots, k_1\}$, $l \in \{1, \cdots, k_2\}$, and $i \in \{1, \cdots, n\}$. From the above stochastic models, we can see that each column vector $\boldsymbol{c}_j$ has an original template vector $\boldsymbol{a}_{I_j}$ with additive noise $\boldsymbol{\xi}_j$ as shown in Fig. 5.4, and that each column vector $\boldsymbol{z}_j$ does not have an original template vector as shown in Fig. 5.5.

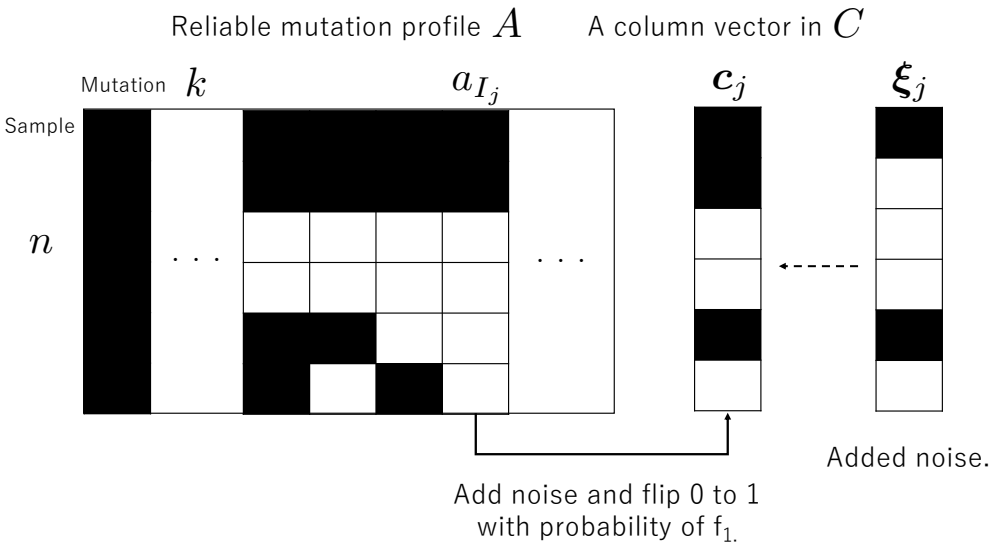


Figure 5.4: The assumed generative model of each column vector in C. Letting $\boldsymbol{c}_j$ be the j-th column vector in C, $\boldsymbol{c}_j$ has the original column vector $\boldsymbol{a}_{I_j}$ in A. By adding a noise $\boldsymbol{\xi}_j$ to $\boldsymbol{a}_{I_j}$, $\boldsymbol{c}_j$ is obtained.

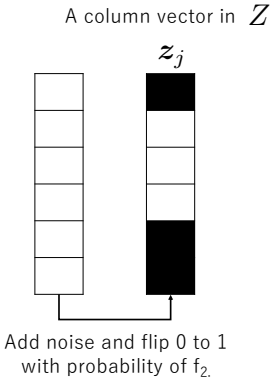


Figure 5.5: The assumed generative model of each column vector in Z. Each column vector in Z is simply obtained by adding noise.

5.3.3 Labeling Methods

We set two labeling functions of $L, R_r : \{0,1\}^n \times \{0,1\}^{n\times k} \to \{0,1\}$ as follows,

$$L(\boldsymbol{b}, A) = \begin{cases} 1 & (\exists j \in \{1, \cdots, k\} \ s.t. \ \boldsymbol{b} = \boldsymbol{a}_j) \\ 0 & (\text{Otherwise}) \end{cases}, \tag{5.1}$$

$$R_r(\boldsymbol{b}, A) = \begin{cases} 1 & (\sum_{i=1}^{n} b_i \geq r) \\ 0 & (\text{Otherwise}) \end{cases}. \tag{5.2}$$

As we can see from the definition, L sets the label by using A, while R_r sets the label by ignoring A and only use $\boldsymbol{b}$. In other words, L leverages the limited patterns of column vectors of A and R_r leverages the number of detected candidates. Fig. 5.6 shows a graphical summary of labeling methods of L and R_r.

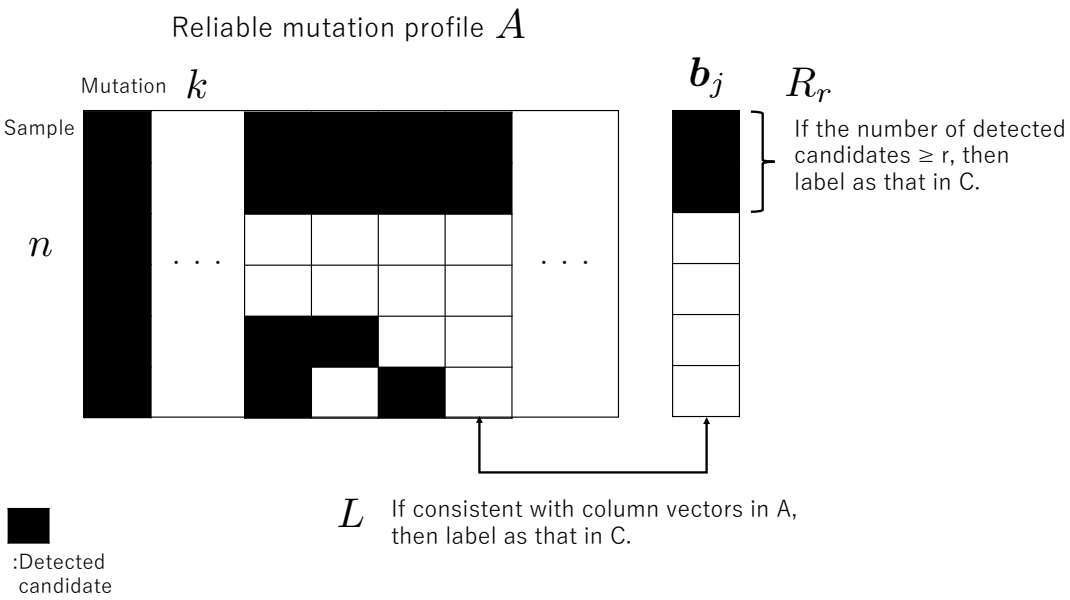


Figure 5.6: A graphical summary for L and R_r. L checks the existence of a consistent column vector in A. R_r only checks the number of detected candidate in a given column vector from B.

5.3.4 Sensitivity and Specificity

We introduce several notations for evaluating the performance of classification as follows.

$$\begin{aligned}
\mathrm{TP}(F, A, B) &:= |\{j | j \in \{1, \cdots, k_1 + k_2\}, F(\boldsymbol{b}_j, A) = 1, \boldsymbol{b}_j \text{ belongs to } C\}|, \\
\mathrm{FP}(F, A, B) &:= |\{j | j \in \{1, \cdots, k_1 + k_2\}, F(\boldsymbol{b}_j, A) = 1, \boldsymbol{b}_j \text{ belongs to } Z\}|, \\
\mathrm{TN}(F, A, B) &:= |\{j | j \in \{1, \cdots, k_1 + k_2\}, F(\boldsymbol{b}_j, A) = 0, \boldsymbol{b}_j \text{ belongs to } Z\}|, \\
\mathrm{FN}(F, A, B) &:= |\{j | j \in \{1, \cdots, k_1 + k_2\}, F(\boldsymbol{b}_j, A) = 0, \boldsymbol{b}_j \text{ belongs to } C\}|,
\end{aligned}$$

where $A \in \{0,1\}^{n\times k}$, $B \in \{0,1\}^{n\times(k_1+k_2)}$, and $F : \{0,1\}^n \times \{0,1\}^{n\times k} \to \{0,1\}$.

We will evaluate the expectation of sensitivity and specificity of L and R_r in

the following sections.

$$\mathbb{E}_B\left[\frac{\mathrm{TP}(F,A,B)}{\mathrm{TP}(F,A,B)+\mathrm{FN}(F,A,B)}\right]=\frac{\mathbb{E}_B[\mathrm{TP}(F,A,B)]}{k_1}\ \text{(Sensitivity)},$$
$$\mathbb{E}_B\left[\frac{\mathrm{TN}(F,A,B)}{\mathrm{FP}(F,A,B)+\mathrm{TN}(F,A,B)}\right]=\frac{\mathbb{E}_B[\mathrm{TN}(F,A,B)]}{k_2}\ \text{(Specificity)},$$

where expectation $\mathbb{E}_B$ is taken with respect to all the generated unreliable mutation profile B.

5.4 Performance Evaluation

5.4.1 Performance Evaluation of L

Evaluation of $\frac{\mathbb{E}_B[\mathrm{TN}(L,A,B)]}{k_2}$

We evaluate the upper bound and lower bound for $\mathbb{E}_B[\mathrm{FP}(L,A,B)]$. Letting K be the number of unique columns in A, the lower bound can be derived as follows.

$$\begin{aligned}\mathbb{E}_B[\mathrm{FP}(L,A,B)] &= k_2\sum_{j=1}^{K}\left(\prod_{i=1}^{n} f_2^{a_{I_j,i}}(1-f_2)^{1-a_{I_j,i}}\right)\\ &\geq k_2K\min_{j\in\{1,\cdots,K\}}\left(\prod_{i=1}^{n} f_2^{a_{I_j,i}}(1-f_2)^{1-a_{I_j,i}}\right)\\ &\geq k_2K\min(f_2,1-f_2)^n\\ &= k_2K\underline{f_2}^n,\end{aligned}$$

where $\underline{f_2} := \min(f_2, 1-f_2)$. The upper bound can also be derived as follows.

$$\begin{aligned}\mathbb{E}_B[\mathrm{FP}(L,A,B)] &= k_2\sum_{j=1}^{K}\left(\prod_{i=1}^{n} f_2^{a_{I_j,i}}(1-f_2)^{1-a_{I_j,i}}\right)\\ &\leq k_2K\max_{j\in\{1,\cdots,K\}}\left(\prod_{i=1}^{n} f_2^{a_{I_j,i}}(1-f_2)^{1-a_{I_j,i}}\right)\\ &\leq k_2K\max(f_2,1-f_2)^n\\ &= k_2K\overline{f_2}^n,\end{aligned}$$

where $\overline{f_2} := \max(f_2, 1-f_2)$. From this, we can evaluate $\mathbb{E}_B[\mathrm{TN}(L,A,B)]$ as follows.

$$(1-K\overline{f_2}^n)\leq\frac{\mathbb{E}_B[\mathrm{TN}(L,A,B)]}{k_2}\leq(1-K\underline{f_2}^n). \tag{5.3}$$

Evaluation of $\frac{\mathbb{E}_B[\mathrm{TP}(L,A,B)]}{k_1}$

From the linearity of the expectation, the expected number of true positives can be written as follows.

$$\mathbb{E}_B[\mathrm{TP}(L,A,B)]=\mathbb{E}_B\left[\sum_{j=1}^{k_1}L(\boldsymbol{c}_j,A)\right]=\sum_{j=1}^{k_1}\Pr(L(\boldsymbol{c}_j,A)=1).$$

The lower bound for $\Pr(L(\boldsymbol{c}_j, A) = 1)$ is as follows.

$$
\begin{aligned}
&\Pr(L(\boldsymbol{c}_j, A) = 1) \\
&= \sum_{i=1}^{n} \Pr\left(L(\boldsymbol{c}_j, A) = 1, \boldsymbol{c}_j \ \ s.t. \ \sum_{n'=1}^{n} a_{I_j,n'} = i \right) \\
&= \sum_{i=1}^{n} \Pr\left(L(\boldsymbol{c}_j, A) = 1 \middle| \boldsymbol{c}_j \ \ s.t. \ \sum_{n'=1}^{n} a_{I_j,n'} = i \right) \Pr\left(\boldsymbol{c}_j \ \ s.t. \ \sum_{n'=1}^{n} a_{I_j,n'} = i \right) \\
&= \sum_{i=1}^{n} w_i \Pr\left(L(\boldsymbol{c}_j, A) = 1 \middle| \boldsymbol{c}_j \ \ s.t. \ \sum_{n'=1}^{n} a_{I_j,n'} = i \right) \\
&\geq \sum_{i=1}^{n} w_i \Pr\left(\boldsymbol{a}_{I_j} = \boldsymbol{c}_j \middle| \boldsymbol{c}_j \ \ s.t. \ \sum_{n'=1}^{n} a_{I_j,n'} = i \right) = \sum_{i=1}^{n} w_i (1 - f_1)^{(n-i)}, \\
&\quad \because) \ \boldsymbol{a}_{I_j} = \boldsymbol{c}_j \Rightarrow L(\boldsymbol{c}_j, A) = 1,
\end{aligned}
$$

where $w_i := \Pr\left(\boldsymbol{c}_j \ \ s.t. \ \sum_{n'=1}^{n} a_{I_j,n'} = i\right)$. From this,

$$
\mathbb{E}_B[\mathrm{TP}(L, A, B)] \geq k_1 \sum_{i=1}^{n} w_i (1 - f_1)^{(n-i)}.
$$

For obtaining the upper bound of $\mathbb{E}_B[\mathrm{TP}(L, A, B)]$, we focus on two things as shown in Fig. 5.7. First, the number of column vectors in A that each $\boldsymbol{c}_j$ can correspond is at most K. Second, the probability for each $\boldsymbol{c}_j$ corresponding to one column vector is at most $\overline{f_1}^{n-i}$, where $\overline{f_1} := \max(f_1, 1 - f_1)$, and $i = \sum_{n'=1}^{n} a_{I_j,n'}$. From this, we can obtain the upper bound for the conditional probability as follows.

$$
\Pr\left(L(\boldsymbol{b}_j, A) = 1 \middle| \boldsymbol{c}_j \ \ s.t. \ \sum_{n'=1}^{n} a_{I_j,n'} = i \right) \leq K \overline{f_1}^{(n-i)}.
$$

Then, the upper bound of $\mathbb{E}_B[\mathrm{TP}(L, A, B)]$ is as follows.

$$
\mathbb{E}_B[\mathrm{TP}(L, A, B)] \leq k_1 \sum_{i=1}^{n} w_i K \overline{f_1}^{(n-i)} = k_1 K \sum_{i=1}^{n} w_i \overline{f_1}^{(n-i)}.
$$

Therefore,

$$
G_n(\boldsymbol{w}, (1 - f_1)) \leq \frac{\mathbb{E}_B[\mathrm{TP}(L, A, B)]}{k_1} \leq K G_n(\boldsymbol{w}, \overline{f_1}), \tag{5.4}
$$

where

$$
\begin{aligned}
\boldsymbol{w} &:= (w_1, \cdots, w_n), \\
G_n(\boldsymbol{x}, f) &:= \sum_{i=1}^{n} x_i f^{(n-i)}.
\end{aligned}
$$

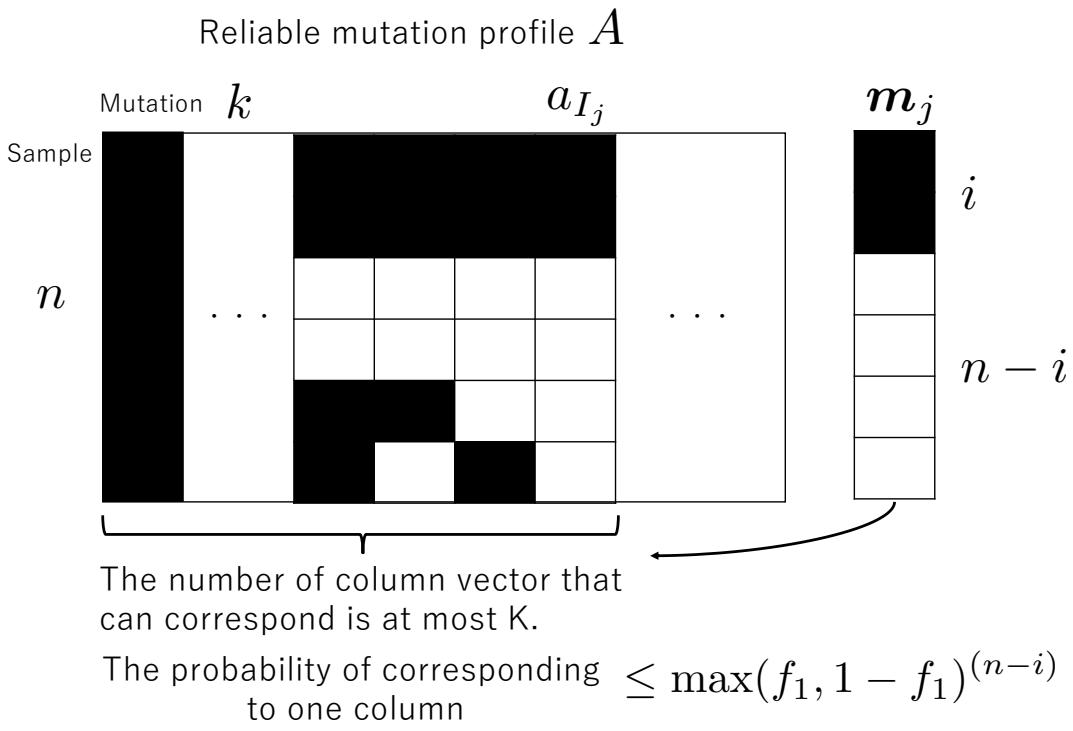


Figure 5.7: The key idea for obtaining the upper bound of $\mathbb{E}_B[\mathrm{TP}(L, A, B)]$.

5.4.2 Performance Evaluation of R_r

We can evaluate the specificity and sensitivity for R_r.

$$\begin{aligned}
\frac{\mathbb{E}_B[\mathrm{TP}(R_r, A, B)]}{k_1} &= (k_1)^{-1}\mathbb{E}_B\left[\sum_{j=1}^{k_1}\sum_{q=r}^{n}\mathbb{I}_{\{\sum_{n'=1}^{n} c_{j,n'}=q\}}\right] \\
&= (k_1)^{-1}\sum_{j=1}^{k_1}\sum_{q=r}^{n}\Pr\left(\sum_{n'=1}^{n} c_{j,n'} = q\right) \\
&= (k_1)^{-1}\sum_{j=1}^{k_1}\sum_{q=r}^{n}\sum_{x=1}^{q}\Pr\left(\sum_{n'=1}^{n} b_{j,n'} = q, \boldsymbol{c}_j \ s.t. \ \sum_{n''=1}^{n} a_{I_j,n''} = x\right) \\
&= (k_1)^{-1}\sum_{j=1}^{k_1}\sum_{q=r}^{n}\sum_{x=1}^{q} w_x \ {}_{n-x}\mathrm{C}_{q-x} \ f_1^{q-x}(1-f_1)^{n-q} \\
&= \sum_{q=r}^{n}\sum_{x=1}^{q} w_x \ {}_{n-x}\mathrm{C}_{q-x} \ f_1^{q-x}(1-f_1)^{n-q}, \qquad (5.5) \\
\frac{\mathbb{E}_B[\mathrm{TN}(R_r, A, B)]}{k_2} &= \sum_{x=0}^{r-1} {}_n\mathrm{C}_x \ (1-f_2)^{n-x} \ f_2^x \\
&= 1 - \sum_{x=r}^{n} {}_n\mathrm{C}_x \ (1-f_2)^{n-x} \ f_2^x. \qquad (5.6)
\end{aligned}$$

5.4.3 Performance Evaluation Summary of L, R_r

Under the assumption described in Section 5.3.2, the expected value of specificity and sensitivity for L, R_r can be summarized as follows.

$$\frac{\mathbb{E}_B[\mathrm{TN}(R_r, A, B)]}{k_2} = 1 - \sum_{x=r}^{n} {}_n\mathrm{C}_x \, (1-f_2)^{n-x} \, f_2^x, \tag{5.7}$$

$$\frac{\mathbb{E}_B[\mathrm{TP}(R_r, A, B)]}{k_1} = \sum_{q=r}^{n} \sum_{x=1}^{q} w_x \, {}_{n-x}\mathrm{C}_{q-x} \, f_1^{q-x} (1-f_1)^{n-q}, \tag{5.8}$$

$$(1 - K\overline{f_2}^n) \leq \frac{\mathbb{E}_B[\mathrm{TN}(L, A, B)]}{k_2} \leq (1 - K\underline{f_2}^n), \tag{5.9}$$

$$G_n(\boldsymbol{w}, (1-f_1)) \leq \frac{\mathbb{E}_B[\mathrm{TP}(L, A, B)]}{k_1} \leq K G_n(\boldsymbol{w}, \overline{f_1}), \tag{5.10}$$

where

$$\begin{aligned}
&K : \text{The number of disjoint columns in } A, \\
&\boldsymbol{w} := (w_1, \cdots, w_n), \\
&w_i := \Pr\left(\boldsymbol{c}_j \;\; s.t. \sum_{n'=1}^{n} a_{I_j, n'} = i \right), \\
G_n(\boldsymbol{x}, f) &:= \sum_{i=1}^{n} x_i f^{(n-i)}, \\
&\overline{f_1} := \max(f_1, 1-f_1), \\
&\overline{f_2} := \max(f_2, 1-f_2), \\
&\underline{f_2} := \min(f_2, 1-f_2).
\end{aligned}$$

From the evaluated values of specificity and sensitivity for L, we can expect that we can expect a higher detection specificity because the number of K is expected to be limited from Corollary 5.2.1. This does not hold true for R_r because $\sum_{x=r}^{n} {}_n\mathrm{C}_x$ will increase drastically as we decrease r.

5.5 Examples for $G_n(\boldsymbol{x}, f)$

From the previous sections, we evaluate the expected values of specificity and sensitivity for L. Instead of considering the mixture matrix U and phylogeny matrix T, we evaluated them by considering $w_i = \Pr\left(\boldsymbol{c}_j \;\; s.t. \; \sum_{n'=1}^{n} a_{I_j, n'} = i\right)$. In this section, we will give several settings for w_i, and calculate its corresponding specificity and sensitivity.

5.5.1 Several Examples of w_i

Uniform case

Here we consider the uniform case as follows.

$$w_i = \frac{1}{n} \; (\forall i).$$

In this case, $G_n(\boldsymbol{w}, f)$ for $0 < f < 1$ can be evaluated as follows.

$$\begin{aligned} G_n(\boldsymbol{w}, f) &= \frac{1}{n}\sum_{i=1}^{n} f^{(n-i)} \\ &= \frac{1}{n} f^n \sum_{i=1}^{n} f^{-i} \\ &= \frac{1}{n} f^n \frac{f^{-1}(1-f^{-n})}{1-f^{-1}} \\ &= \frac{1}{n} f^n \frac{(1-f^{-n})}{f-1} \\ &= \frac{1}{n}\frac{(f^n-1)}{f-1} = \frac{1}{n}\frac{(1-f^n)}{1-f}. \end{aligned}$$

Therefore, when we can evaluate the expected values of specificity and sensitivity as follows.

$$\begin{aligned} &\frac{\mathbb{E}_B[\mathrm{TP}(L,A,B)]}{k_1} \geq G_n(\boldsymbol{w}, (1-f_1)) = \frac{1}{n}\frac{(1-(1-f_1)^n)}{f_1} > \frac{1}{n}, \\ &\frac{\mathbb{E}_B[\mathrm{TN}(L,A,B)]}{k_2} \geq (1-K\overline{f_2}^n). \end{aligned}$$

Exponential Case

Here we consider the exponential case as follows.

$$w_i = \frac{p^i}{\sum_{j=1}^{n} p^j} = \frac{p^i}{\frac{p(1-p^n)}{1-p}},$$

where we assume $p > 0, p \neq 1$. In this case, $G_n(\boldsymbol{w}, f)$ for $0 < f < 1, p \neq f$ can be evaluated as follows.

$$\begin{aligned} G_n(\boldsymbol{w}, f) &= \frac{1}{n}\sum_{i=1}^{n} w_i f^{(n-i)} \\ &= \frac{(1-p)}{p(1-p^n)} f^n \frac{\frac{p}{f}(1-\left(\frac{p}{f}\right)^n)}{1-\frac{p}{f}} \\ &= \frac{(1-p)}{p(1-p^n)}\frac{p(f^n-p^n)}{f-p} \\ &= \frac{(1-p)}{(1-p^n)}\frac{(f^n-p^n)}{f-p}. \end{aligned}$$

$G_n(\boldsymbol{w}, f)$ for $0 < f < 1, p = f$ can be evaluated as follows.

$$G_n(\boldsymbol{w}, f) = \sum_{i=1}^{n} w_i f^{(n-i)} = n\frac{f^n(1-p)}{p(1-p^n)}.$$

From this, we can evaluate the specificity and sensitivity when $p = 1 - f_1$ as follows.

$$\frac{\mathbb{E}_B[\mathrm{TP}(L,A,B)]}{k_1} \geq n\frac{(1-f_1)^n(1-p)}{p(1-p^n)}, \tag{5.11}$$

$$\frac{\mathbb{E}_B[\mathrm{TN}(L,A,B)]}{k_2} \geq (1-K\overline{f_2}^n). \tag{5.12}$$

When $p \neq 1 - f_1$,

$$\frac{\mathbb{E}_B[\mathrm{TP}(L,A,B)]}{k_1} \geq \frac{(1-p)}{(1-p^n)} \frac{((1-f_1)^n - p^n)}{(1-f_1)-p}, \tag{5.13}$$

$$\frac{\mathbb{E}_B[\mathrm{TN}(L,A,B)]}{k_2} \geq (1 - K\overline{f_2}^n). \tag{5.14}$$

5.6 Comparison of Specificity between L and R_r

Here we compare the specificity of L and R_r and consider when the phylogeny-based labeling method L has a higher detection specificity than R_r. Because we cannot exactly evaluate the expected specificity of L, we discuss the sufficient condition as follows.

$$\begin{aligned}
&\frac{\mathbb{E}_B[\mathrm{TN}(R_r,A,B)]}{k_2} \leq \frac{\mathbb{E}_B[\mathrm{TN}(L,A,B)]}{k_2} \\
&\Leftarrow \frac{\mathbb{E}_B[\mathrm{TN}(R_r,A,B)]}{k_2} \leq 1 - K\overline{f_2}^n \quad \because) \textit{Eq.}\ (5.32) \\
&\Leftrightarrow 1 - \sum_{x=r}^{n} {}_n\mathrm{C}_x\, (1-f_2)^{n-x}\, f_2^x \leq 1 - K\overline{f_2}^n. \qquad (5.15)
\end{aligned}$$

As a result, we can obtain the following theorems of Theorem 5.6.1 and Theorem 5.6.2. From these theorems, if n is a large number and r is a smaller number, we can expect that L has a higher detection specificity than R_r.

Theorem 5.6.1. *If* $\frac{1}{2} \leq f_2 < 1$ *and* $K \leq 2 \cdot 2^{\min(\lfloor \frac{n}{2} \cdot \frac{1-f_2}{f_2} \rfloor, n-r)} - 1$, *then*

$$\frac{\mathbb{E}_B[\mathrm{TN}(R_r,A,B)]}{k_2} \leq \frac{\mathbb{E}_B[\mathrm{TN}(L,A,B)]}{k_2}.$$

Proof. If $\frac{1}{2} \leq f_2 < 1$, then $\overline{f_2} = f_2$. Therefore,

$$\begin{aligned}
&\textit{Eq.}\ (5.15) \\
&\Leftrightarrow 1 - \sum_{x=r}^{n} {}_n\mathrm{C}_x\, (1-f_2)^{n-x}\, f_2^x \leq 1 - K\overline{f_2}^n \\
&\Leftrightarrow K\overline{f_2}^n \leq \sum_{x=r}^{n} {}_n\mathrm{C}_x\, (1-f_2)^{n-x}\, f_2^x \\
&\Leftrightarrow K \leq \sum_{x=r}^{n} {}_n\mathrm{C}_x \left(\frac{1-f_2}{f_2}\right)^{n-x} = \sum_{y=0}^{n-r} {}_n\mathrm{C}_{n-y} \left(\frac{1-f_2}{f_2}\right)^{y} = \sum_{y=0}^{n-r} {}_n\mathrm{C}_y \left(\frac{1-f_2}{f_2}\right)^{y} \\
&\Leftrightarrow K \leq 1 + \sum_{y=1}^{n-r} {}_n\mathrm{C}_y \left(\frac{1-f_2}{f_2}\right)^{y}.
\end{aligned}$$

The right-hand side can be lower bounded as follows.

$$1 + \sum_{y=1}^{n-r} {}_n\mathrm{C}_y \left(\frac{1-f_2}{f_2}\right)^{y} \geq 1 + \sum_{y=1}^{n-r} \left(\frac{n}{y} \cdot \frac{1-f_2}{f_2}\right)^{y}.$$

Therefore,

$$\textit{Eq.}\ (5.15) \Leftarrow K \leq 1 + \sum_{y=1}^{n-r} \left(\frac{n}{y} \cdot \frac{1-f_2}{f_2}\right)^{y}.$$

Furthermore, we can obtain a lower bound for $\sum_{y=1}^{n-r}\left(\frac{n}{y}\cdot\frac{1-f_2}{f_2}\right)^y$ as follows.

$$\sum_{y=1}^{n-r}\left(\frac{n}{y}\cdot\frac{1-f_2}{f_2}\right)^y \geq \sum_{y=1}^{\min(\lfloor\frac{n}{2}\cdot\frac{1-f_2}{f_2}\rfloor,n-r)} 2^y = 2\cdot 2^{\min(\lfloor\frac{n}{2}\cdot\frac{1-f_2}{f_2}\rfloor,n-r)} - 2$$
$$\because) \ \frac{n}{y}\cdot\frac{1-f_2}{f_2} \geq 2 \Leftrightarrow y \leq \frac{n}{2}\cdot\frac{1-f_2}{f_2}.$$

From this,

$$Eq.\ (5.15) \Leftarrow K \leq 2\cdot 2^{\min(\lfloor\frac{n}{2}\cdot\frac{1-f_2}{f_2}\rfloor,n-r)} - 1.$$

□

Theorem 5.6.2. *If* $0 < f_2 < \frac{1}{2}$ *and* $K \leq 2^{\lfloor\frac{n}{2}\cdot\frac{f_2}{1-f_2}\rfloor+1} - 2^r$, *then*

$$\frac{\mathbb{E}_B[\mathrm{TN}(R_r,A,B)]}{k_2} \leq \frac{\mathbb{E}_B[\mathrm{TN}(L,A,B)]}{k_2}.$$

Proof. If $0 < f_2 < \frac{1}{2}$, then $\overline{f_2} = 1 - f_2$. Therefore,

$$\begin{aligned} Eq.\ (5.15) &\Leftrightarrow K \leq \sum_{x=r}^{n} {}_n\mathrm{C}_x \left(\frac{f_2}{1-f_2}\right)^x \\ &\Leftarrow K \leq \sum_{x=r}^{n}\left(\frac{n}{x}\cdot\frac{f_2}{1-f_2}\right)^x. \end{aligned}$$

The sufficient condition is obtained by evaluating the lower bound for $\sum_{x=r}^{n} {}_n\mathrm{C}_x \left(\frac{f_2}{1-f_2}\right)^x$. Furthermore, we can obtain a lower bound for $\sum_{x=r}^{n}\left(\frac{n}{x}\cdot\frac{f_2}{1-f_2}\right)^x$ as follows.

$$\sum_{x=r}^{n}\left(\frac{n}{x}\cdot\frac{f_2}{1-f_2}\right)^x \geq \sum_{x=r}^{\lfloor\frac{n}{2}\cdot\frac{f_2}{1-f_2}\rfloor} 2^x,$$
$$\because) \ \frac{n}{x}\cdot\frac{f_2}{1-f_2} \geq 2 \Leftrightarrow x \leq \frac{n}{2}\cdot\frac{f_2}{1-f_2}.$$

From this,

$$Eq.\ (5.15) \Leftarrow K \leq \sum_{x=r}^{\lfloor\frac{n}{2}\cdot\frac{f_2}{1-f_2}\rfloor} 2^x = 2^{\lfloor\frac{n}{2}\cdot\frac{f_2}{1-f_2}\rfloor+1} - 2^r.$$

□

5.6.1 Examples of Performance

Here, we would like to evaluate the performance in several settings of f_1, f_2, $\boldsymbol{w}$, n, K, r. For $\boldsymbol{w}$, we sample $\boldsymbol{w}$ from Dirichlet$(\cdot|(\alpha,\cdots,\alpha))$, where $\alpha = 1.0$, and take the average of the performance. The following procedure is used for evaluation.

1) Conduct the following procedures 100 times and take the average of expected specificity and sensitivity for each f_1, f_2, n, K, r (If we cannot evaluate the exact value, we take the average of the upper or lower bound).

1-a) Sample $\boldsymbol{w} \sim \mathrm{Dirichlet}(\cdot|(\alpha, \cdots, \alpha))$, where $\alpha = 1.0$.

1-b) Evaluate the lower (upper) bound of $\frac{\mathbb{E}_B[\mathrm{TP}(L,A,B)]}{k_1}$ and $\frac{\mathbb{E}_B[\mathrm{TN}(L,A,B)]}{k_2}$.

1-c) If the lower (upper) bound > 1.0, substitute 1.0 for the bound.

1-d) If the lower (upper) bound < 0.0, substitute 0.0 for the bound.

1-e) Evaluate $\frac{\mathbb{E}_B[\mathrm{TP}(R_r,A,B)]}{k_1}$ and $\frac{\mathbb{E}_B[\mathrm{TN}(R_r,A,B)]}{k_2}$.

For the case of $n = 20$ and $r \in \{1, 3, 5\}$, we summarized the results in the following figures of Figs. 5.8 to 5.16. For the case of $n = 10$ and $r \in \{1, 3, 5\}$, see Figs. D.5 to D.13.

We show the performance evaluation results of R_r in Figs. 5.8 to 5.10. For the performance evaluation of R_r, K does not affect the performance and we only examined the case of $K = 30$. For the specificity of R_1, R_3, R_5, only when f_2 is close to 0.0, detection specificity is high. However the detection specificity drastically decreases as f_2 increases. For the sensitivity of R_1, R_3, R_5, they have enough high detection sensitivity for almost all the cases. From this, when f_2 is close to 0.0, R_r is a useful detection method but R_r can be a meaningless method when f_2 is relatively high due to the drastic decrease of specificity.

The performance evaluation results of L are shown in Figs. 5.11 to 5.16. For the specificity of L, when f_2 is around 0.5, detection specificity is high and the evaluated bounds are meaningless when f_2 is close to 0 or 1. For the sensitivity of L, detection sensitivity is from 5% to 40%. From this, we can detect a somatic mutation in a patient with high specificity and moderate (but not ignorable) sensitivity.

Performance of R_1 at $n = 20$, $K = 30$

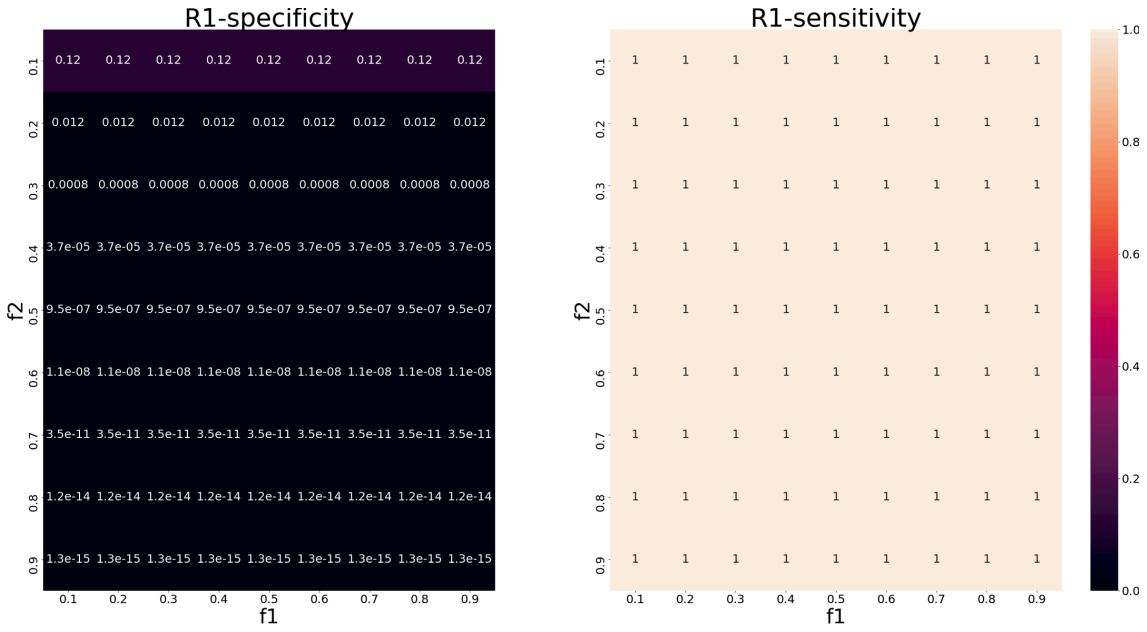


Figure 5.8: Specificity and sensitivity of R_1 at $n = 20$, $K = 30$.

Performance of R_3 at $n = 20$, $K = 30$

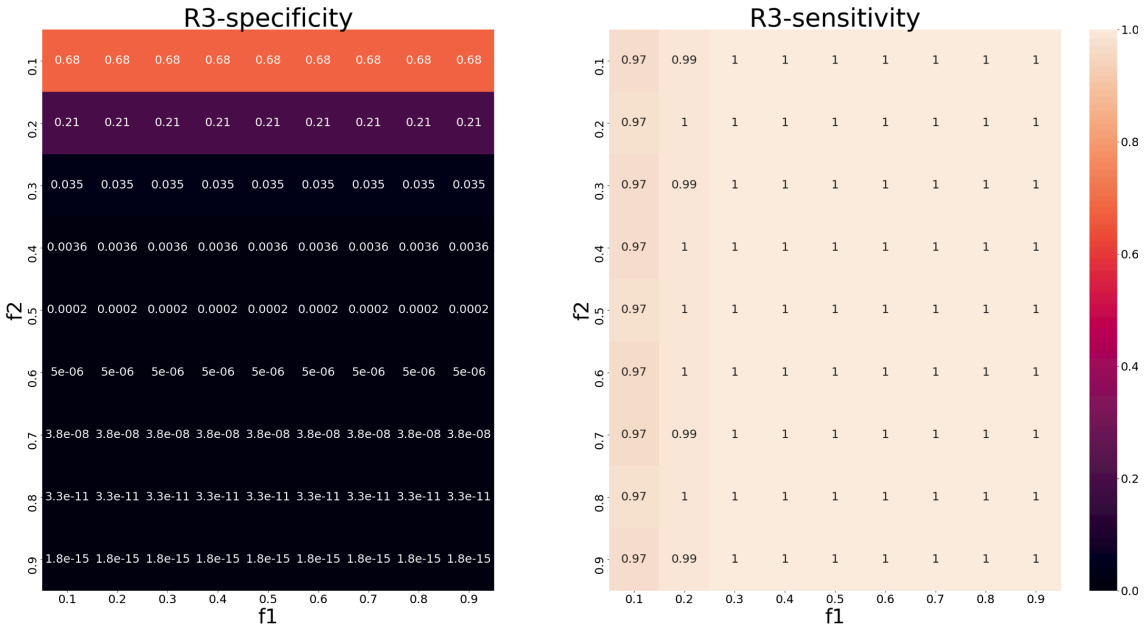


Figure 5.9: Specificity and sensitivity of R_3 at $n = 20$, $K = 30$.

Performance of R_5 at $n = 20$, $K = 30$

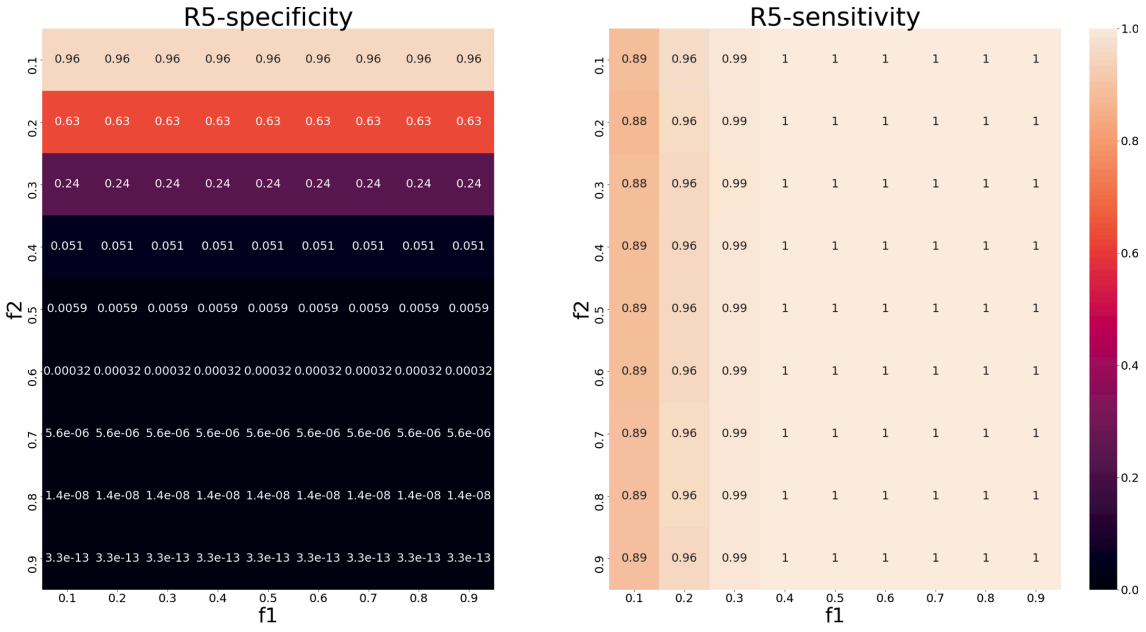


Figure 5.10: Specificity and sensitivity of R_5 at $n = 20$, $K = 30$.

Performance of L at $n = 20$, $K = 30$

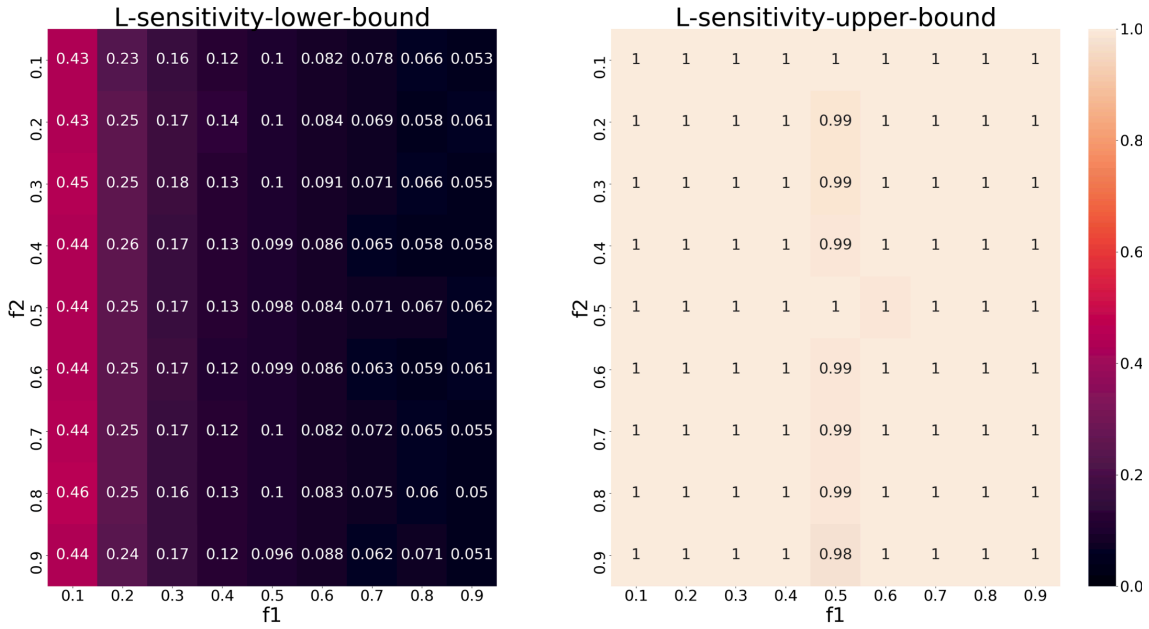


Figure 5.11: Lower and upper bounds for the sensitivity of L at $n = 20$, $K = 30$.

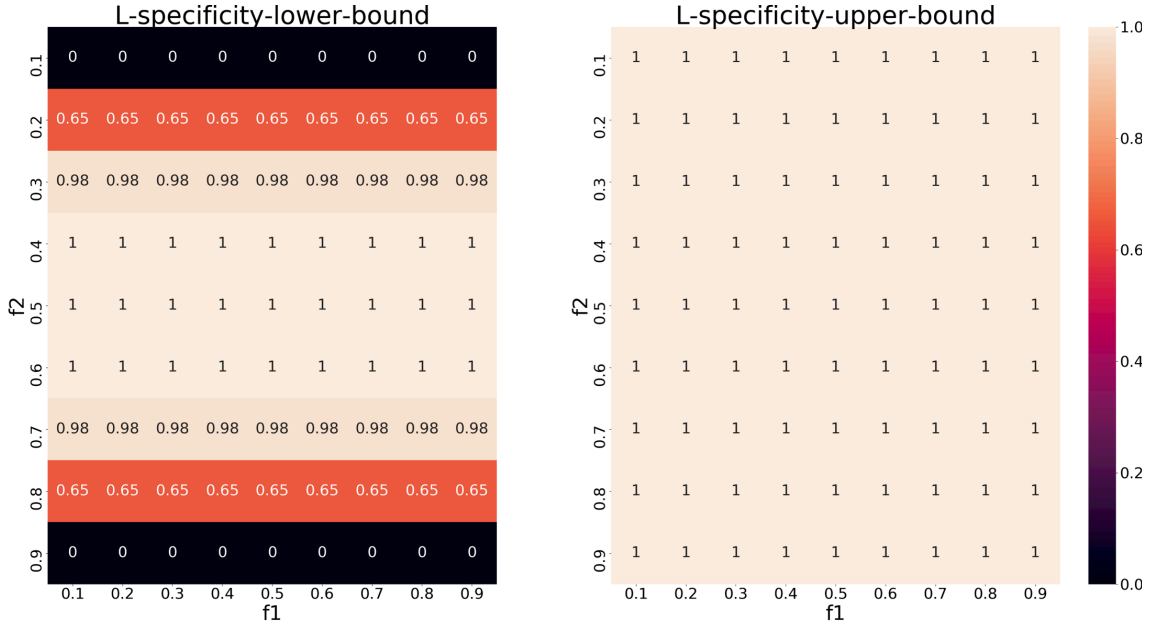


Figure 5.12: Lower and upper bounds for the specificity of L at $n = 20$, $K = 30$.

Performance of L at $n = 20$, $K = 50$

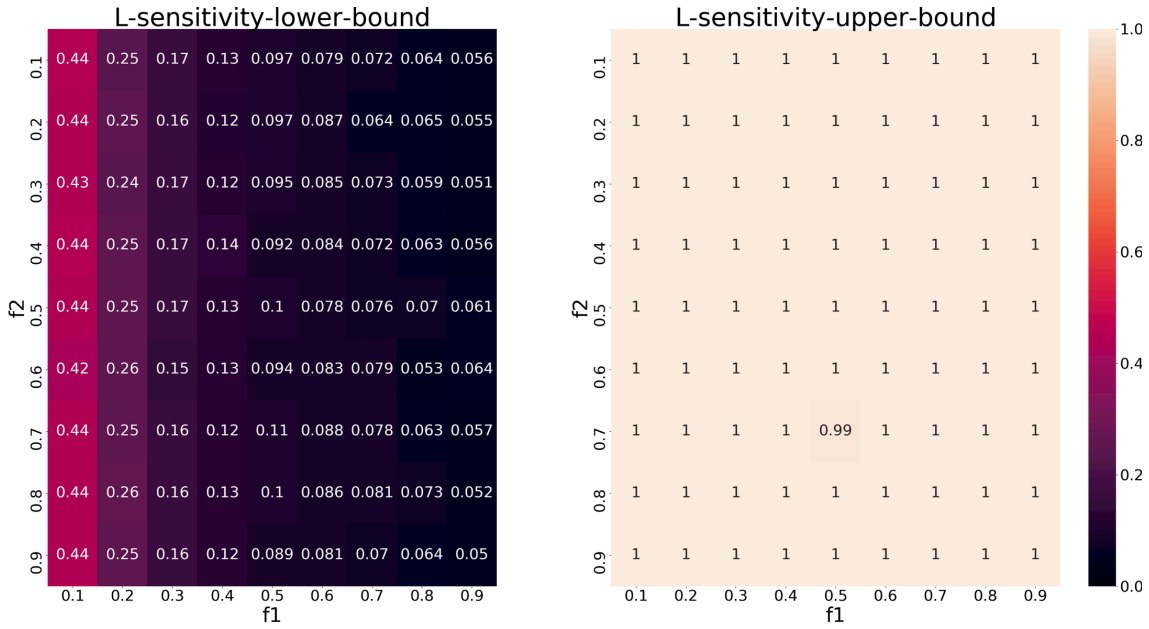


Figure 5.13: Lower and upper bounds for the sensitivity of L at $n = 20$, $K = 50$.

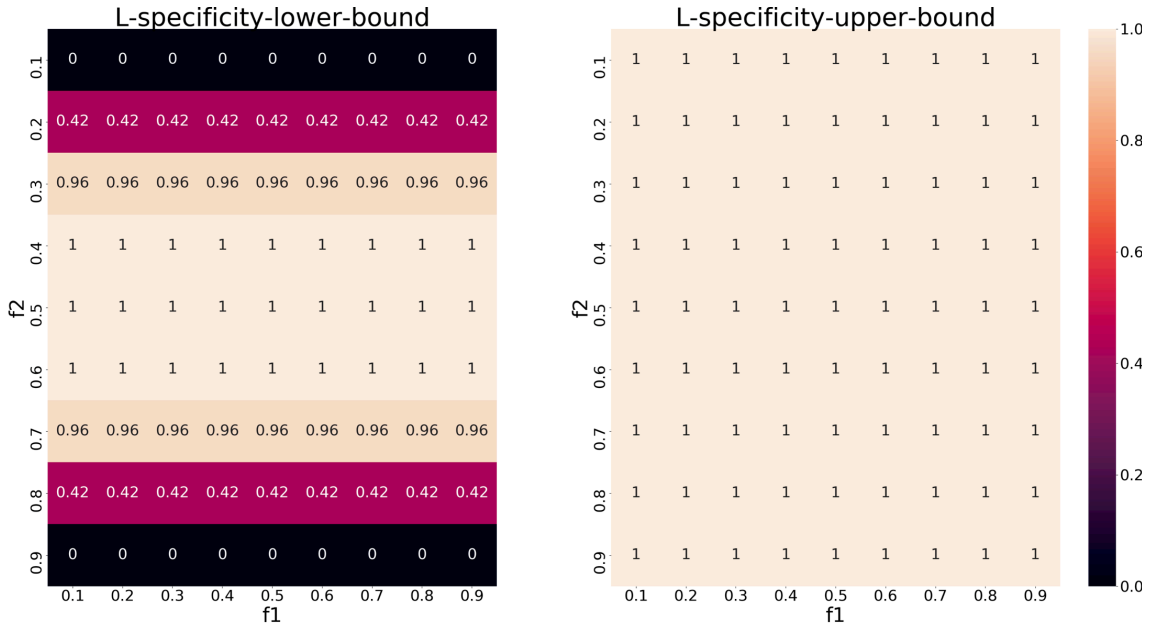


Figure 5.14: Lower and upper bounds for the specificity of L at $n = 20$, $K = 50$.

Performance of L at $n = 20$, $K = 100$

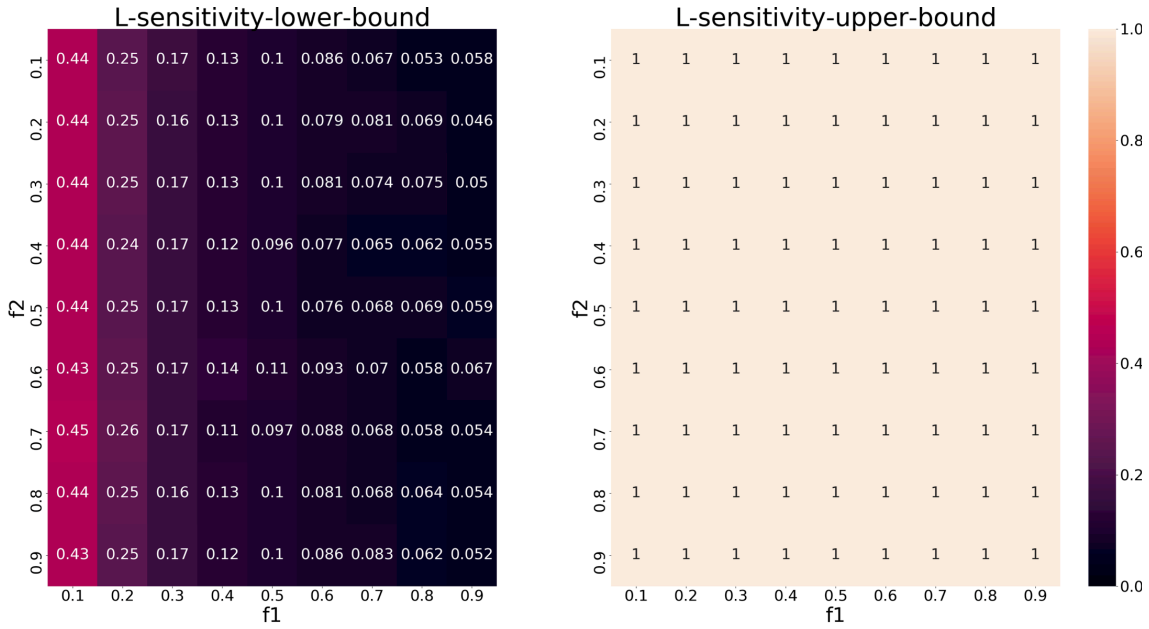


Figure 5.15: Lower and upper bounds for the sensitivity of L at $n = 20$, $K = 100$.

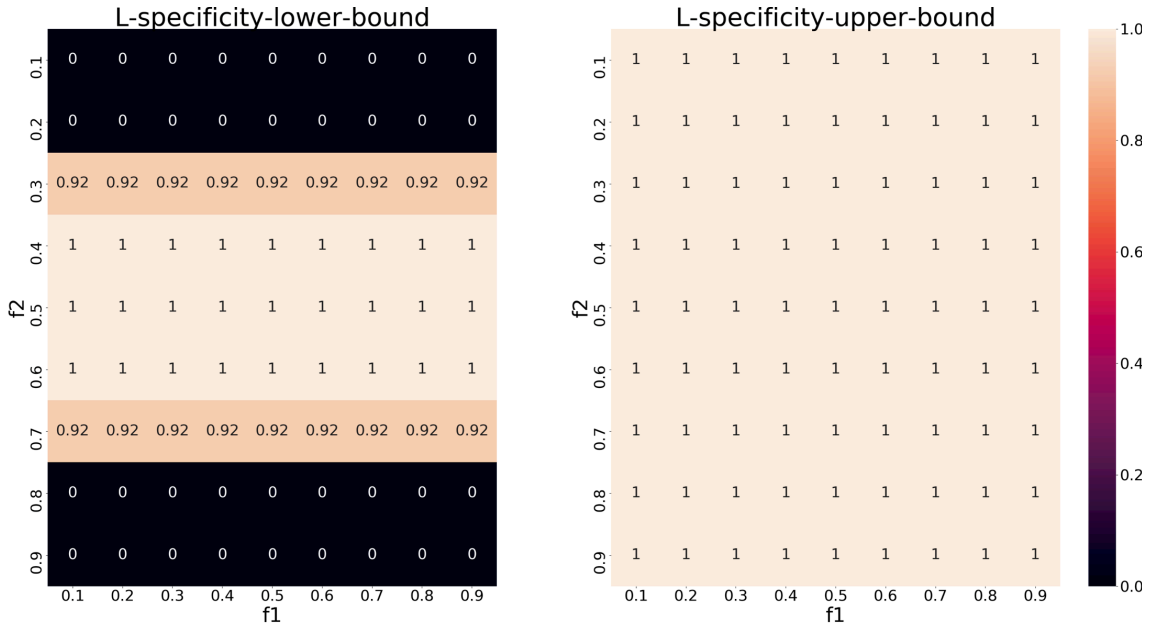


Figure 5.16: Lower and upper bounds for the specificity of L at $n = 20$, $K = 100$.

5.7 Performance Evaluation with Dropout Events

In the actual sequencing data sets, depth coverage is not enough for all the genomic positions and tumor regions. For example, the dropout event is well known in single-cell RNA sequencing data sets, and zero coverage is observed in many samples. We consider such dropout events, in which we could not obtain enough sequencing depth and we cannot expect enough detection sensitivity. In this section, we show the problem settings for dropout events and labeling methods in this case, and then we will evaluate the expected specificity and

sensitivity with observed dropout events.

5.7.1 Given Dropout Profile

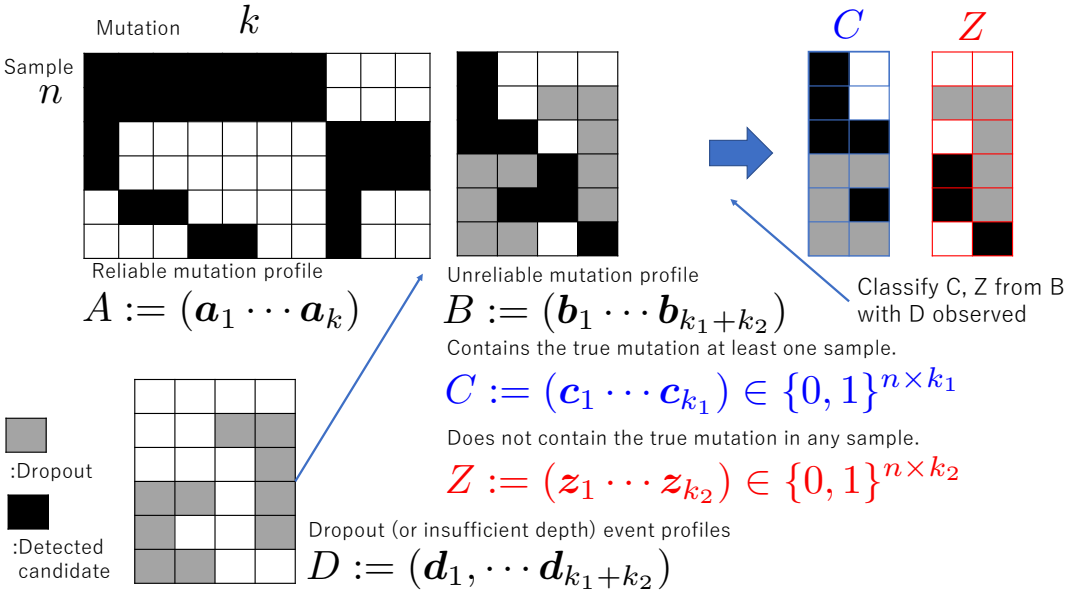


Figure 5.17: A graphical summary for the problem setting with dropout events.

We additionally defined the dropout events profile as follows (Fig. 5.17).

$$
\begin{aligned}
&D := (\boldsymbol{d}_1, \cdots, \boldsymbol{d}_k) \in \{0,1\}^{n \times (k_1+k_2)}, \\
&\boldsymbol{d}_j \in \{0,1\}^n, \\
&s_j := \sum_{n'=1}^{n} d_{j,n'}, \\
&k_1(s) = |\{\boldsymbol{c}_j | s_j = s\}|, \\
&k_2(s) = |\{\boldsymbol{z}_j | s_j = s\}|,
\end{aligned}
$$

where $\boldsymbol{d}_j \in \{0,1\}^n$ represents the j-th column vector of D and $D_{n',k'} = 1$ means that sequence depth is sufficient and $D_{n',k'} = 0$ means that dropout event occurs in the n'-th data set at k'-th genomic position. s_j is the number of dropout events in the j-th genomic position. $k_1(s)$ is the number of column vectors in C such that $s_j = s$. $k_2(s)$ is the number of column vectors in Z such that $s_j = s$.

We assume that the profile of dropout events D are independent of B, i.e., I_j,$\boldsymbol{c}_j$,$\boldsymbol{\xi}_j$, and $\boldsymbol{z}_j$. Therefore, even after we observed D, we can evaluate the expected specificity and sensitivity similarly to the previous section.

5.7.2 Labeling Functions Given Dropout Events

Given the observation of dropout events, we set two labeling functions of $L^*, R_r^* : \{0,1\}^n \times \{0,1\}^{n \times k} \times \{0,1\}^n \to \{0,1\}$ as follows,

$$
L^*(\boldsymbol{b}, A, \boldsymbol{d}) = \begin{cases} 1 & (\exists j \in \{1, \cdots, k\} \ s.t. \ \boldsymbol{d} \odot \boldsymbol{b} = \boldsymbol{d} \odot \boldsymbol{a}_j) \\ 0 & (\text{Otherwise}) \end{cases}, \tag{5.16}
$$

$$
R_r^*(\boldsymbol{b}, A, \boldsymbol{d}) = \begin{cases} 1 & (\sum_{i=1}^{n} b_i d_i \geq r) \\ 0 & (\text{Otherwise}) \end{cases}, \tag{5.17}
$$

where $\odot$ represents the Hadamard product as follows.

$$\begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} \odot \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix} = \begin{pmatrix} x_1 y_1 \\ x_2 y_2 \\ \vdots \\ x_n y_n \end{pmatrix}.$$

As we can see from the definitions of L^*, R_r^*, these labeling function is a version of L, R_r to consider the dropout events.

5.7.3 Performance Given Dropout Profile

We define the following values for evaluating the performance as follows.

$$\begin{aligned}
&\mathrm{TP}(F, A, B, s|D) := |\{j | j \in \{1, \cdots, k_1 + k_2\}, F(\boldsymbol{b}_j, A, \boldsymbol{d}_j) = 1, \boldsymbol{b}_j \text{ belongs to } C, s_j = s\}|, \\
&\mathrm{TN}(F, A, B, s|D) := |\{j | j \in \{1, \cdots, k_1 + k_2\}, F(\boldsymbol{b}_j, A, \boldsymbol{d}_j) = 0, \boldsymbol{b}_j \text{ belongs to } Z, s_j = s\}|, \\
&\mathrm{FP}(F, A, B, s|D) := |\{j | j \in \{1, \cdots, k_1 + k_2\}, F(\boldsymbol{b}_j, A, \boldsymbol{d}_j) = 1, \boldsymbol{b}_j \text{ belongs to } Z, s_j = s\}|, \\
&\mathrm{FN}(F, A, B, s|D) := |\{j | j \in \{1, \cdots, k_1 + k_2\}, F(\boldsymbol{b}_j, A, \boldsymbol{d}_j) = 0, \boldsymbol{b}_j \text{ belongs to } C, s_j = s\}|.
\end{aligned}$$

The total evaluation for the expected number of specificity and sensitivity can be obtained as follows.

$$\begin{aligned}
\frac{\mathbb{E}_{B|D}\left[\sum_{s=0}^{n} \mathrm{TP}(F, A, B, s|D)\right]}{k_1} &= \sum_{s=0}^{n} \frac{k_1(s)}{k_1} \mathbb{E}_{B|D}\left[\mathrm{TPR}(F, A, B, s|D)\right] \text{ (Sensitivity)}, \\
\frac{\mathbb{E}_{B|D}\left[\sum_{s=0}^{n} \mathrm{TN}(F, A, B, s|D)\right]}{k_2} &= \sum_{s=0}^{n} \frac{k_2(s)}{k_2} \mathbb{E}_{B|D}\left[\mathrm{TNR}(F, A, B, s|D)\right] \text{ (Specificity)},
\end{aligned}$$

where

$$\begin{aligned}
\mathrm{TPR}(F, A, B, s|D) &:= \begin{cases} \frac{\mathrm{TP}(F, A, B, s|D)}{k_1(s)} & (k_1(s) > 0) \\ 0 & (k_1(s) = 0) \end{cases}, \\
\mathrm{TNR}(F, A, B, s|D) &:= \begin{cases} \frac{\mathrm{TN}(F, A, B, s|D)}{k_2(s)} & (k_2(s) > 0) \\ 0 & (k_2(s) = 0) \end{cases},
\end{aligned}$$

and $\mathbb{E}_{B|D}$ is taken with respect to all the B after the dropout profile is observed.

5.7.4 Performance for Each s

The performance for each $s = 0, 1, \cdots, n$ can be evaluated similarly. For $s = 0$, the performance is as follows when $k(0) > 0$.

$$\frac{\mathbb{E}_{B|D}\left[\mathrm{TN}(R_r^*, A, B, 0|D)\right]}{k_2(0)} = 1, \tag{5.18}$$

$$\frac{\mathbb{E}_{B|D}\left[\mathrm{TP}(R_r^*, A, B, 0|D)\right]}{k_1(0)} = 0, \ (\because \text{we set } r \geq 1) \tag{5.19}$$

$$\frac{\mathbb{E}_{B|D}\left[\mathrm{TN}(L^*, A, B, 0|D)\right]}{k_2(0)} = 1, \tag{5.20}$$

$$\frac{\mathbb{E}_{B|D}\left[\mathrm{TP}(L^*, A, B, 0|D)\right]}{k_1(0)} = 0. \tag{5.21}$$

For $s > 0$, the performance is as follows when $k(s) > 0$.

$$\frac{\mathbb{E}_{B|D}[\mathrm{TN}(R_r^*, A, B, s|D)]}{k_2(s)} = 1 - \sum_{x=r}^{s} {}_s\mathrm{C}_x (1-f_2)^{s-x} f_2^x, \tag{5.22}$$

$$\frac{\mathbb{E}_{B|D}[\mathrm{TP}(R_r^*, A, B, s|D)]}{k_1(s)} = \sum_{q=r}^{s} \sum_{x=1}^{q} w_{x\ s-x}\mathrm{C}_{q-x} f_1^{q-x} (1-f_1)^{s-q}, \tag{5.23}$$

$$(1 - \overline{K}(s)\overline{f_2}^s) \leq \frac{\mathbb{E}_{B|D}\left[\mathrm{TN}(L^*, A, B, s|D)\right]}{k_2(s)} \leq (1 - \underline{K}(s)\underline{f_2}^s), \tag{5.24}$$

$$G_s(\boldsymbol{w}_s, (1-f_1)) \leq \frac{\mathbb{E}_{B|D}\left[\mathrm{TP}(L^*, A, B, s|D)\right]}{k_1(s)} \leq \overline{K}(s) G_s(\boldsymbol{w}_s, \overline{f_1}), \tag{5.25}$$

where

$$\begin{aligned}
w_{s,i} &:= \Pr(\boldsymbol{c}_j \ \ s.t. \ \sum_{n'=1}^{n} a_{I_j,n'} d_{j,n'} = i | s_j = s), \\
\boldsymbol{w}_s &:= (w_{s,1}, \cdots w_{s,s}), \\
K(\boldsymbol{d}) &: \#\text{ of disjoint vectors in } \{\boldsymbol{a}_j | \boldsymbol{d}_j = \boldsymbol{d}\}, \\
\overline{K}(s) &:= \max_{\boldsymbol{d}_j \ s.t. \ \sum_{n'=1}^{n} \boldsymbol{d}_{j,n'} = s} K(\boldsymbol{d}_j), \\
\underline{K}(s) &:= \max_{\boldsymbol{d}_j \ s.t. \ \sum_{n'=1}^{n} \boldsymbol{d}_{j,n'} = s} K(\boldsymbol{d}_j).
\end{aligned}$$

5.8 Evaluation with Insufficient Coverage

Here, we also evaluate the performance of detection in insufficient coverage. In this case, we assume that original detection sensitivity is less than 100% due to the insufficient depth coverage. For example, Oxford nanopore long-read sequencer can generate around 30Gb˜50Gb per one flow cell of MinION sequencer (500$˜1000$). Therefore, if we want to guarantee 100 depth coverage for 10 sequenced tumor samples, we require about 50,000$˜100,000$. Due to the limitation of the budget, we would like to assume that depth coverage is not enough for all the positions and the original detection sensitivity is less than 100%.

5.9 Insufficient Coverage Assumptions for Given Profiles

We assume the following stochastic models for each column.

$$\begin{aligned}
I_j &\sim \mathrm{Unif}(\cdot|1, k), \\
\xi_{j,i} &\sim \mathrm{Ber}(\cdot|f_{\mathrm{tp}}), \\
\epsilon_{j,i} &\sim \mathrm{Ber}(\cdot|f_{\mathrm{fp}}), \\
c_{j,i} &= a_{I_j,i}\xi_{j,i} + (1 - a_{I_j,i})\epsilon_{j,i}, \\
z_{l,i} &\sim \mathrm{Ber}(\cdot|f_{\mathrm{err}}),
\end{aligned}$$

where $0 < f_{\mathrm{tp}} < 1$, $0 < f_{\mathrm{fp}} < 1$, $0 < f_{\mathrm{err}} < 1$, $j \in \{1, \cdots, k_1\}$, $l \in \{1, \cdots, k_2\}$, and $i \in \{1, \cdots, n\}$. From the above stochastic models, we can see that each column vector $\boldsymbol{c}_j$ has an original template vector $\boldsymbol{a}_{I_j}$ with noises of $\boldsymbol{\xi}_j$ and $\boldsymbol{\epsilon}_j$, and that

each column vector $\boldsymbol{z}_j$ has no original template vectors. Under this assumption, we assume false negative events and $c_{j,i} = 0$ can happen even if the corresponding template have the muattion and $a_{I_j,i} = 1$.

5.10 Performance Evaluation

5.10.1 Performance Evaluation of L

Evaluation of $\frac{\mathbb{E}_B[\mathrm{TN}(L,A,B)]}{k_2}$

We evaluate the upper bound and lower bound for $\mathbb{E}_B[\mathrm{FP}(L, A, B)]$. The lower bound can be derived as follows.

$$\begin{aligned}
\mathbb{E}_B[\mathrm{FP}(L, A, B)] &= k_2 \sum_{j=1}^{K} \left(\prod_{i=1}^{n} f_{\mathrm{err}}^{a_{I_j,i}} (1 - f_{\mathrm{err}})^{1-a_{I_j,i}} \right) \\
&\geq k_2 K \min_{j \in \{1,\cdots,K\}} \left(\prod_{i=1}^{n} f_{\mathrm{err}}^{a_{I_j,i}} (1 - f_{\mathrm{err}})^{1-a_{I_j,i}} \right) \\
&\geq k_2 K \min(f_{\mathrm{err}}, 1 - f_{\mathrm{err}})^n \\
&= k_2 K \underline{f_{\mathrm{err}}}^n,
\end{aligned}$$

where $\underline{f_{\mathrm{err}}} := \min(f_{\mathrm{err}}, 1 - f_{\mathrm{err}})$. The upper bound can also be derived as follows.

$$\begin{aligned}
\mathbb{E}_B[\mathrm{FP}(L, A, B)] &= k_2 \sum_{j=1}^{K} \left(\prod_{i=1}^{n} f_{\mathrm{err}}^{a_{I_j,i}} (1 - f_{\mathrm{err}})^{1-a_{I_j,i}} \right) \\
&\leq k_2 K \max_{j \in \{1,\cdots,K\}} \left(\prod_{i=1}^{n} f_{\mathrm{err}}^{a_{I_j,i}} (1 - f_{\mathrm{err}})^{1-a_{I_j,i}} \right) \\
&\leq k_2 K \max(f_{\mathrm{err}}, 1 - f_{\mathrm{err}})^n \\
&= k_2 K \overline{f_{\mathrm{err}}}^n,
\end{aligned}$$

where $\overline{f_{\mathrm{err}}} := \max(f_{\mathrm{err}}, 1 - f_{\mathrm{err}})$. From this, we can estimate $\mathbb{E}_B[\mathrm{TN}(L, A, B)]$ as follows.

$$(1 - K\overline{f_{\mathrm{err}}}^n) \leq \frac{\mathbb{E}_B[\mathrm{TN}(L, A, B)]}{k_2} \leq (1 - K\underline{f_{\mathrm{err}}}^n). \tag{5.26}$$

Evaluation of $\frac{\mathbb{E}_B[\mathrm{TP}(L,A,B)]}{k_1}$

From the linearity of the expectation, the expected number of true positives can also be written as follows.

$$\mathbb{E}_B[\mathrm{TP}(L, A, B)] = \mathbb{E}_B \left[\sum_{j=1}^{k_1} L(\boldsymbol{c}_j, A) \right] = \sum_{j=1}^{k_1} \Pr(L(\boldsymbol{c}_j, A) = 1).$$

The lower bound of $\Pr(L(\boldsymbol{c}_j, A) = 1)$ is as follows.

$$
\begin{aligned}
&\Pr(L(\boldsymbol{c}_j, A) = 1) \\
&= \sum_{i=1}^{n} \Pr\left(L(\boldsymbol{c}_j, A) = 1, \boldsymbol{c}_j \ \ s.t. \ \sum_{n'=1}^{n} a_{I_j,n'} = i \right) \\
&= \sum_{i=1}^{n} \Pr\left(L(\boldsymbol{c}_j, A) = 1 \middle| \boldsymbol{c}_j \ \ s.t. \ \sum_{n'=1}^{n} a_{I_j,n'} = i \right) \Pr\left(\boldsymbol{c}_j \ \ s.t. \ \sum_{n'=1}^{n} a_{I_j,n'} = i \right) \\
&= \sum_{i=1}^{n} w_i \Pr\left(L(\boldsymbol{c}_j, A) = 1 \middle| \boldsymbol{c}_j \ \ s.t. \ \sum_{n'=1}^{n} a_{I_j,n'} = i \right) \\
&\geq \sum_{i=1}^{n} w_i \Pr\left(\boldsymbol{a}_{I_j} = \boldsymbol{c}_j \middle| \boldsymbol{c}_j \ \ s.t. \ \sum_{n'=1}^{n} a_{I_j,n'} = i \right) = \sum_{i=1}^{n} w_i f_{\mathrm{tp}}^{i} (1 - f_{\mathrm{fp}})^{(n-i)}.
\end{aligned}
$$

From this,

$$
\mathbb{E}_B[\mathrm{TP}(L, A, B)] \geq k_1 (1 - f_{\mathrm{fp}})^n \sum_{i=1}^{n} w_i \left(\frac{f_{\mathrm{tp}}}{1 - f_{\mathrm{fp}}} \right)^i .
$$

For obtaining the upper bound of $\mathbb{E}_B[\mathrm{TP}(L, A, B)]$, we also focus on two things. First, the number of column vectors in A that each $\boldsymbol{c}_j$ can correspond is at most K. Second, the probability for each $\boldsymbol{c}_j$ corresponding to one column vector is at most $\overline{f_{\mathrm{tp}}}^{i} \overline{f_{\mathrm{fp}}}^{n-i}$, where $\overline{f_{\mathrm{tp}}} := \max(f_{\mathrm{tp}}, 1 - f_{\mathrm{tp}})$, $\overline{f_{\mathrm{fp}}} := \max(f_{\mathrm{fp}}, 1 - f_{\mathrm{fp}})$, and $i = \sum_{n'=1}^{n} a_{I_j,n'}$. From this, we can obtain the upper bound for the conditional probability as follows.

$$
\Pr\left(L(\boldsymbol{b}_j, A) = 1 \middle| \boldsymbol{c}_j \ \ s.t. \ \sum_{n'=1}^{n} a_{I_j,n'} = i \right) \leq K \overline{f_{\mathrm{tp}}}^{i} \cdot \overline{f_{\mathrm{fp}}}^{(n-i)}.
$$

Then, the upper bound of $\mathbb{E}_B[\mathrm{TP}(L, A, B)]$ is as follows.

$$
\mathbb{E}_B[\mathrm{TP}(L, A, B)] \leq k_1 \sum_{i=1}^{n} w_i K \overline{f_{\mathrm{tp}}}^{i} \cdot \overline{f_{\mathrm{fp}}}^{(n-i)} = k_1 K \overline{f_{\mathrm{fp}}}^{n} \sum_{i=1}^{n} w_j \left(\overline{f_{\mathrm{tp}}} / \overline{f_{\mathrm{fp}}} \right)^i .
$$

Therefore,

$$
f_{\mathrm{tp}}^{n} \cdot G_n \left(\boldsymbol{w}, \frac{1 - f_{\mathrm{fp}}}{f_{\mathrm{tp}}} \right) \leq \frac{\mathbb{E}_B[\mathrm{TP}(L, A, B)]}{k_1} \leq K \overline{f_{\mathrm{tp}}}^{n} \cdot G_n \left(\boldsymbol{w}, \overline{f_{\mathrm{fp}}} / \overline{f_{\mathrm{tp}}} \right). \quad (5.27)
$$

5.10.2 Performance Evaluation of R_r

Here, we evaluate the specificity and sensitivity for R_r.

$$
\begin{aligned}
&\frac{\mathbb{E}_B[\mathrm{TP}(R_r, A, B)]}{k_1} \\
&= (k_1)^{-1} \mathbb{E}_B \left[\sum_{j=1}^{k_1} \sum_{q=r}^{n} \mathbb{I}_{\left\{ \sum_{n'=1}^{n} c_{j,n'} = q \right\}} \right] \\
&= (k_1)^{-1} \sum_{j=1}^{k_1} \sum_{q=r}^{n} \Pr\left(\sum_{n'=1}^{n} c_{j,n'} = q \right) \\
&= (k_1)^{-1} \sum_{j=1}^{k_1} \sum_{q=r}^{n} \sum_{x=1}^{q} \Pr\left(\sum_{n'=1}^{n} b_{j,n'} = q, \boldsymbol{c}_j \ \ s.t. \ \sum_{n''=1}^{n} a_{I_j,n''} = x \right)
\end{aligned}
$$

$$= (k_1)^{-1} \sum_{j=1}^{k_1} \sum_{q=r}^{n} \sum_{x=1}^{q} \Pr\left(\sum_{n'=1}^{n} b_{j,n'} = q, \boldsymbol{c}_j \ \ s.t. \ \sum_{n''=1}^{n} a_{I_j,n''} = x \right)$$

$$= (k_1)^{-1} \sum_{j=1}^{k_1} \sum_{q=r}^{n} \sum_{x=1}^{q} w_x \Pr\left(\sum_{n'=1}^{n} b_{j,n'} = q \,\middle|\, \boldsymbol{c}_j \ \ s.t. \ \sum_{n''=1}^{n} a_{I_j,n''} = x \right)$$

$$= \sum_{q=r}^{n} \sum_{x=1}^{q} w_x \sum_{y=\max(0,q-n+x)}^{\min(x,q)} {}_x\mathrm{C}_y \, f_{\mathrm{tp}}^{y} f_{\mathrm{fn}}^{(x-y)} \cdot {}_{n-x}\mathrm{C}_{q-y} f_{\mathrm{fp}}^{(q-y)} (1 - f_{\mathrm{fp}})^{(n-x-q+y)}, \tag{5.28}$$

$$\frac{\mathbb{E}_B[\mathrm{TN}(R_r, A, B)]}{k_2} = 1 - \sum_{x=r}^{n} {}_n\mathrm{C}_x \, (1 - f_{\mathrm{err}})^{n-x} \, f_{\mathrm{err}}^{x}. \tag{5.29}$$

5.10.3 Performance Evaluation Summary of L, R_r

Under the insufficient coverage assumption described in Section 5.9, the expected value of specificity and sensitivity for L, R_r can be summarized as follows.

$$\frac{\mathbb{E}_B[\mathrm{TN}(R_r, A, B)]}{k_2} = 1 - \sum_{x=r}^{n} {}_n\mathrm{C}_x \, (1 - f_{\mathrm{err}})^{n-x} \, f_{\mathrm{err}}^{x}, \tag{5.30}$$

$$\frac{\mathbb{E}_B[\mathrm{TP}(R_r, A, B)]}{k_1}$$

$$= \sum_{q=r}^{n} \sum_{x=1}^{q} \sum_{y=\max(0,q-n+x)}^{\min(x,q)} w_x \cdot \ {}_x\mathrm{C}_y \, f_{\mathrm{tp}}^{y} f_{\mathrm{fn}}^{(x-y)} \cdot {}_{n-x}\mathrm{C}_{q-y} f_{\mathrm{fp}}^{(q-y)} (1 - f_{\mathrm{fp}})^{(n-x-q+y)}, \tag{5.31}$$

$$(1 - K\overline{f_{\mathrm{err}}}^{n}) \leq \frac{\mathbb{E}_B[\mathrm{TN}(L, A, B)]}{k_2} \leq (1 - K\underline{f_{\mathrm{err}}}^{n}), \tag{5.32}$$

$$f_{\mathrm{tp}}^{n} \cdot G_n\left(\boldsymbol{w}, \frac{1 - f_{\mathrm{fp}}}{f_{\mathrm{tp}}}\right) \leq \frac{\mathbb{E}_B[\mathrm{TP}(L, A, B)]}{k_1} \leq K\overline{f_{\mathrm{tp}}}^{n} \cdot G_n\left(\boldsymbol{w}, \overline{f_{\mathrm{fp}}}/\overline{f_{\mathrm{tp}}}\right), \tag{5.33}$$

where

$$\begin{aligned}
K &: \text{The number of disjoint columns in } A, \\
\boldsymbol{w} &:= (w_1, \cdots, w_n), \\
w_i &:= \Pr\left(\boldsymbol{c}_j \ \ s.t. \ \sum_{n'=1}^{n} a_{I_j,n'} = i \right), \\
G_n(\boldsymbol{x}, f) &:= \sum_{i=1}^{n} x_i f^{(n-i)}, \\
\overline{f_{\mathrm{tp}}} &:= \max(f_{\mathrm{tp}}, 1 - f_{\mathrm{tp}}), \\
\overline{f_{\mathrm{fp}}} &:= \max(f_{\mathrm{fp}}, 1 - f_{\mathrm{fp}}), \\
\overline{f_{\mathrm{err}}} &:= \max(f_{\mathrm{err}}, 1 - f_{\mathrm{err}}), \\
\underline{f_{\mathrm{err}}} &:= \min(f_{\mathrm{err}}, 1 - f_{\mathrm{err}}).
\end{aligned}$$

5.11 Discussion

In this chapter, we consider whether or not tumor phylogeny is useful for predicting the somatic mutations in at least one tumor region under two different

assumptions.

First, we assume that sensitivity is 100% for predicting each somatic mutation in each tumor region. Under this setting, we evaluate the expected specificity and sensitivity of two prediction methods: L which leverages the property of tumor phylogeny and R_r which does not use the tumor phylogeny. By comparing the lower bound of the expected specificity of L and the expected specificity of R_r, we derive a sufficient condition for L to have a higher detection specificity than R_r. From the sufficient condition, L is expected to have a higher detection specificity than R_r when the number of samples is large and r is small. Second, we additionally assume insufficient depth coverage. This additional assumption considers the practical settings of sequencing technologies: dropout events in single-cell sequencing and insufficient depth coverage in Oxford nanopore sequencing. Under this additional assumption, we also evaluated the expected specificity and sensitivity of L and R_r.

Chapter 6

Conclusion

6.1 Summary

Cancer is driven by genomic alterations. Profiles of genomic alterations provide the most important information in cancer genomics, and almost all of the analysis in this field is based on the profiles of genomic alterations. For example, from these profiles, researchers infer the origin of the tumor evolution and medical doctors search the optimal therapy for the individual cancer patient. Therefore, the detection method of somatic mutations is one of the most important analysis methods in this field, and the improvement of the accuracy is expected to affect all the other analyses. To achieve better detection accuracy, incorporating NGS data specific properties or biological prior knowledge is expected to be important. However, due to the deficiency in incorporating these properties or prior knowledge in existing methods, there remains room for performance improvement in such fundamental and important analysis methods. In this thesis, we found a design of the Bayesian hierarchical model to incorporate these information sources, evaluate the effectiveness of biological prior knowledge, and constructed an accurate detection method of somatic mutations.

In chapter 3, we proposed a somatic mutation calling method named as OHVarfinDer for the single-tumor-based approach. Our method incorporates the multiple NGS data specific properties and improves detection performance by integrating multiple Bayesian hierarchical models into one model by partitioning-based model integration. Unlike the Bayesian model averaging, our design of the Bayesian model does not require additional hyperparameter settings, which simplifies the construction of the Bayesian models. This is an advantage of partitioning-based model integration because we can ignore the additional hyperparameters.

In chapter 4, we presented a multiple-tumor-based mutation calling method termed MultiMuC. Within MultiMuC, we reflected two ideas for performance improvement in multiple-tumor-based mutation call in which the mutation sharing assumption is applied. First, we focused on the No-TP case: we could expect mutation candidates in multiple regions, but actually, no true mutations exist. In the No-TP case, reflecting the mutation sharing assumption only degrade the detection performance. Hence, we evaluated the probability of No-TP case and found that high detection specificity and the existence of enough number of detected candidates can decrease the probability of the No-TP case. From this, MultiMuC incorporates the specificity of detection and the number of detected candidate to avoid the No-TP case. Second, we considered the manner of integrating the NGS data specific properties in multiple-tumor-based mutation call. To incorporate NGS data specific properties, we try to use data genera-

tion probabilities within existing single-tumor-based mutation calling methods. For using these data generation probabilities, we proposed Bayes-factor-based model construction. Through Bayes-factor-based model construction, we guaranteed that Bayes factors are sufficient for obtaining the consistent maximum a posteriori (MAP) state even when the data generation probabilities are not directly available and Bayes factors are available instead. Based on this idea, the Bayesian model of MultiMuC is constructed based on the Bayes factors from another existing single-tumor-based mutation calling method.

In chapter 5, we examined the effectiveness of tumor phylogeny for patient-wise mutation call (detecting each mutation for a patient not for each tumor region) from multi-regional tumor sequence data sets. To consider this, we set several assumptions for generating the results of mutation calling. Under the assumptions, we evaluated the expected specificity and sensitivity of the tumor-phylogeny-based detection method and the non-tumor-phylogeny-based detection method. From these evaluations, we found that tumor phylogeny is effective for predicting each mutation in a patient in particular situations.

6.2 Future Work

6.2.1 Application of Mutation Sharing Assumption for Copy Number Alterations or Structural Variations

In this thesis, we proposed two ideas for improving multiple-tumor-based mutation call: avoiding the No-TP case and Bayes-factor-based model construction. Based on these two ideas, we showed that the improvement of the detection performance is possible. In this thesis, we only focused on the single nucleotide variations (SNVs) and short insertions and deletions (InDels), but did not consider the copy number alterations (CNAs) or structural variations (SVs). For the future direction, it is valuable to check the effectiveness of mutation sharing assumption for detection of CNAs or SVs and it is also valuable to implement an extension of MultiMuC to detect CNAs or SVs.

6.2.2 Application of Tumor Phylogeny for Mutation Call in Multi-Regional Tumor Sequence Data Sets

In this thesis, we considered the effectiveness of tumor phylogeny in patient-wise mutation call (detecting each somatic mutation in a patient) from multi-regional tumor sequence data sets under several assumptions. For the future direction, there remain two types of research work.

First, by applying this idea to the real data sets, it is possible to detect somatic mutations in repeat regions or pseudogenes, which cannot be detected by Illumina sequencer previously. For example, we prepare two types of mutation profiles from multi-regional tumors. For the first mutation profile, we use Illumina sequencer with enough depth. For the second mutation profile, we use Oxford nanopore sequencer with enough depth. From these two profiles, we can use the property of tumor phylogeny and may predict a mutation (which cannot be detected by Illumina sequencer previously) in a patient. Therefore, the application of our idea may break through the current limitations of Illumina-sequencer-based mutation call and high error rates of the Oxford nanopore sequencers. Although it may require a lot of budgets to apply our idea in current technologies, the application of our idea may lead to treasure-like discovery in the field of cancer genomics, and the application of our idea can be valuable research work.

Second, it is still ambiguous whether or not tumor phylogeny is effective for region-wise mutation call (detecting each somatic mutation in each tumor region). For this problem, we privately developed and tested about 15 Bayesian models, but none of them cannot successfully improve the detection performance, unfortunately. It may also be a helpful research work to consider the effectiveness of tumor phylogeny for the region-wise mutation call.

Appendix

A Comparison of Partitioning-based Model Integration and Bayesian Model Averaging

A.1 Generative Model in Bayesian Model Averaging

Here, we explain the details of the generative model that extends the existing methods of HapMuC and OVarCall based on Bayesian model averaging. To avoid confusion, we express the probability to which we apply Bayesian model averaging as $\Pr^{(\mathrm{BMA})}(\cdot)$ and express the probability defined in OHVarfinDer (to which we apply partitioning-based model integration) as $\Pr^{(\mathrm{PBMI})}(\cdot)$.

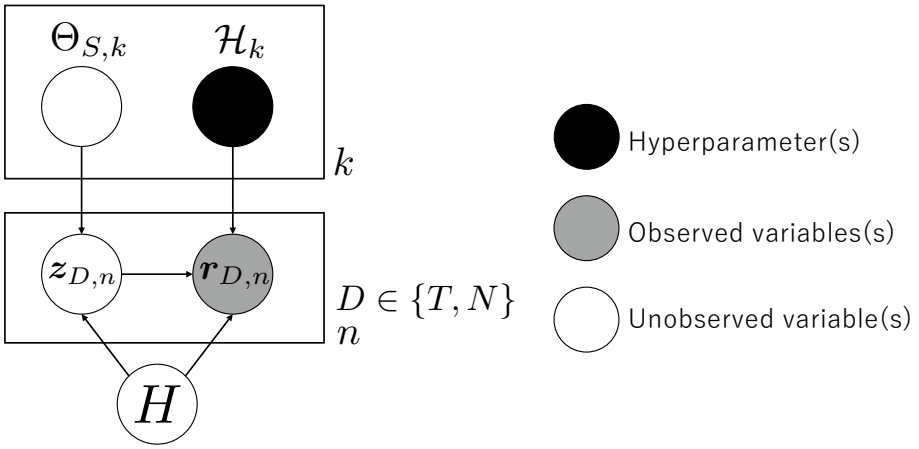


Figure A.1: Graphical model for the generative models to which we apply Bayesian model averaging. $S \in \{M, E\}$ represents the hypothesis.

We show the graphical summary of the generative model to which we apply Bayesian model averaging in Fig. A.1. By using $\Pr^{(\mathrm{PBMI})}(\cdot)$, $\Pr^{(\mathrm{BMA})}(\cdot)$ can be expressed as follows.

$$\begin{aligned}
&\Pr^{(\mathrm{BMA})}(\mathcal{R}_{\mathrm{NT}}|\mathcal{M}_S) \\
&= \sum_{k=0}^{4} \Pr^{(\mathrm{BMA})}(H=k) \int \Pr^{(\mathrm{PBMI})}(\Theta_{S,k}|\mathcal{M}_S) \Pr^{(\mathrm{BMA})}(\mathcal{R}_{\mathrm{NT}}, \mathcal{Z}_{\mathrm{NT}}|\Theta_{S,k}, H=k) d\mathcal{Z}_{\mathrm{NT}} d\Theta_{S,k},
\end{aligned}$$

where

$$\begin{aligned}
&\Pr^{(\mathrm{BMA})}(\mathcal{R}_{\mathrm{NT}}, \mathcal{Z}_{\mathrm{NT}}|\Theta_{S,k}, H=k) \\
&= \prod_{D\in\{N,T\}} \prod_{n=1}^{d_D} Pr^{(\mathrm{PBMI})}(\boldsymbol{z}_{D,n}|\Theta_{S,k}, \mathcal{M}_{S,k}) \Pr^{(\mathrm{PBMI})}(\boldsymbol{r}_{D,n}|\boldsymbol{z}_{D,n}, \mathcal{H}_k), \\
&\Pr^{(\mathrm{BMA})}(H=k) = \frac{1}{5}.
\end{aligned}$$

A.2 Experimental Results

Based on the simulation data sets prepared in Section 3.5.1, we compared the performance of OHVarfinDer and the Bayesian model averaging-based methods in Section A.1 as summarized by the following table of Table A.1. When no properties are available, the Bayesian model averaging-based method performs comparably with OHVarfinDer. However, when at least one property is available, OHVarfinDer performs better than the Bayesian model averaging-based method. From this, partitioning-based model integration is more suited to the incorporation of multiple sequence data specific properties than Bayesian model averaging.

Table A.1: Comparison of AUC in simulation data sets

	$v(\%)$	HeteroSNPs	Overlap	Distance to SNP	μ_l	σ_l	OHVarfinDer	BMA
A	5	-	-	500-5000	300	30	0.828	0.826
	10	-	-				0.891	0.889
	20	-	-				0.967	0.965
B	5	-	+	500-5000	180	30	0.938	0.911
	10	-	+				0.958	0.945
	20	-	+				0.989	0.989
C	5	+	-	1-100	300	30	0.880	0.852
	10	+	-				0.916	0.885
	20	+	-				0.986	0.973
D	5	+	+	1-100	180	30	0.943	0.908
	10	+	+				0.975	0.951
	20	+	+				0.994	0.989

B Comparison of Partitioning-based Model Integration and Supervised Learning Methods

In the main text at Chapter 3, we only considered an extension of existing mutation calling methods of OVarCall and HapMuC which are based on Bayesian statistics, and proposed the partitioning-based model integration. However, we did not consider a method based on supervised learning methods. Here, we would like to compare our methods of OHVarfinDer and other supervised learning methods based on the simulation data sets prepared in Section 3.5.1.

As a counterpart method, we prepared random forest [7], AdaBoost [20], and XGBoost [8], and all of these methods use the P-value of Fisher's exact test and the Bayes factors of OVarCall and HapMuC. In this experiment, we collected the training data sets from all the 12 settings of simulation data sets and trained the model, and measured the AUC of the ROC curve for each setting. Figs. B.2 to B.4 summarized the difference of AUC values (the supervised method minus OHVarfinDer) in 12 simulation settings at the different proportion of training data sets, and orange-colored box shows the P-value of the paired t-test is less than 0.01.

From these results, partitioning-based model integration performs better than the supervised learning methods when the proportion of the training data set is extremely small around 0.1% and 0.5%. When the proportion of the training data set is around 1% and 10%, partitioning-based model integration performs comparably with the supervised learning methods. When the proportion of the training data set is $\geq$ 30%, XGBoost performs better than partitioning-based model integration. Therefore, partitioning-based model integration is suited for incorporating multiple sequence data specific properties when we cannot use the sufficient number of training data sets.

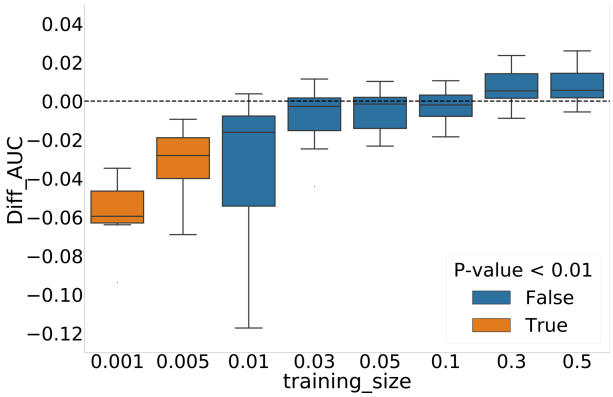


Figure B.2: Comparison of partitioning-based model integration and AdaBoost in different size of training data sets based on 12 settings of simulation data sets in Section 3.5.1.

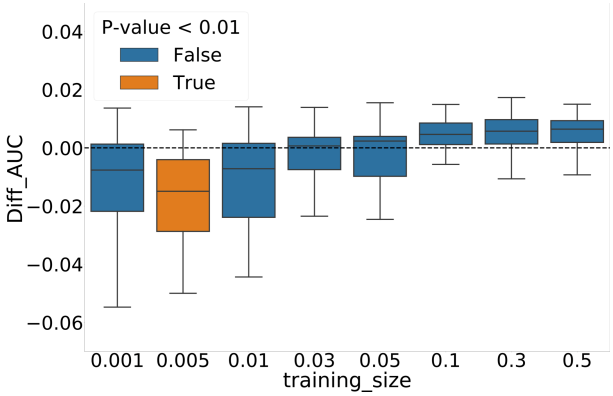


Figure B.3: Comparison of partitioning-based model integration and random forest in different size of training data sets based on 12 settings of simulation data sets in Section 3.5.1.

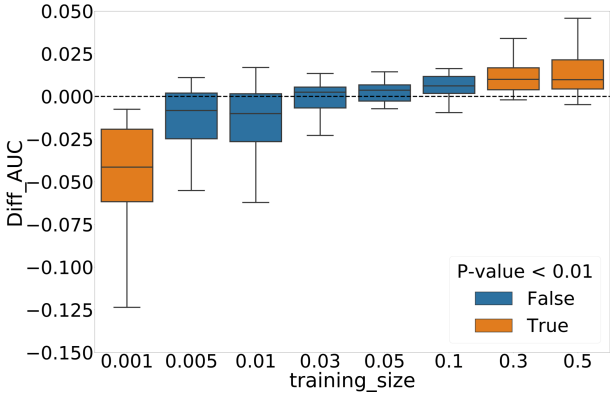


Figure B.4: Comparison of partitioning-based model integration and XGBoost in different size of training data sets based on 12 settings of simulation data sets in Section 3.5.1.

C Effects of Error Data Generation Model in Higher Depth

In our method of OHVarfinDer, we set the different joint probability in higher depth condition in which depth coverage ≥ 100. Here, we would like to show that this setting of joint probability is effective for detecting somatic mutations in exome sequence data sets with higher depth coverage. For the data set, we used the exome sequence data sets of renal clear-cell carcinoma which are introduced in Section 3.5.2. We summarized the results in Table C.2. Within this table, OHVarfinDer(LD) does not change the joint probability of error data generation model and OHVarfinDer(HD) changes the joint probability when depth coverage ≥ 100. As we can see from this table, changing the joint probability in the error data generation model improves the detection performance.

Table C.2: Comparison of AUC in exome sequence data sets

SNV/InDel	VAF	OHVarfinDer(LD)	OHVarfinDer(HD)	#SNV	#Error
SNV	2-7%	0.935	0.990	52	2422
SNV	7%-	0.979	0.988	184	1982

D Performance Evaluation Summary of L and R_r at $n = 10$

In this section, we show the performance evaluation summaries of L and R_r when the detection sensitivity of region-wise mutation call is 100% and $n = 10$.

Performance of R_1 at $n = 10$, $K = 30$

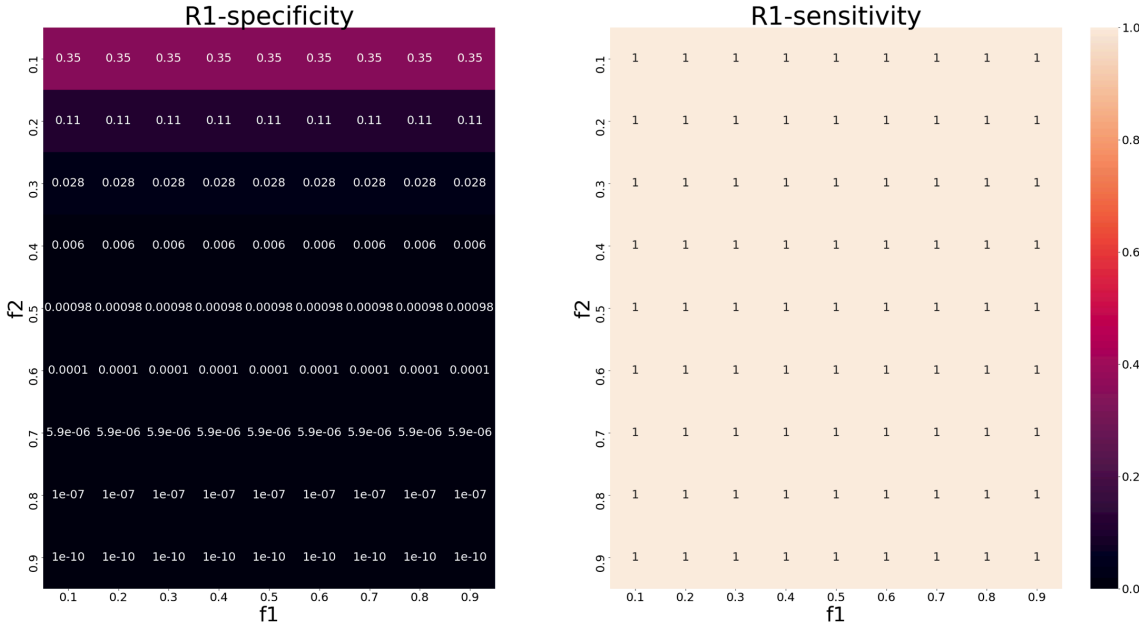


Figure D.5: Specificity and sensitivity of R_1 at $n = 10$, $K = 30$.

Performance of R_3 at $n = 10$, $K = 30$

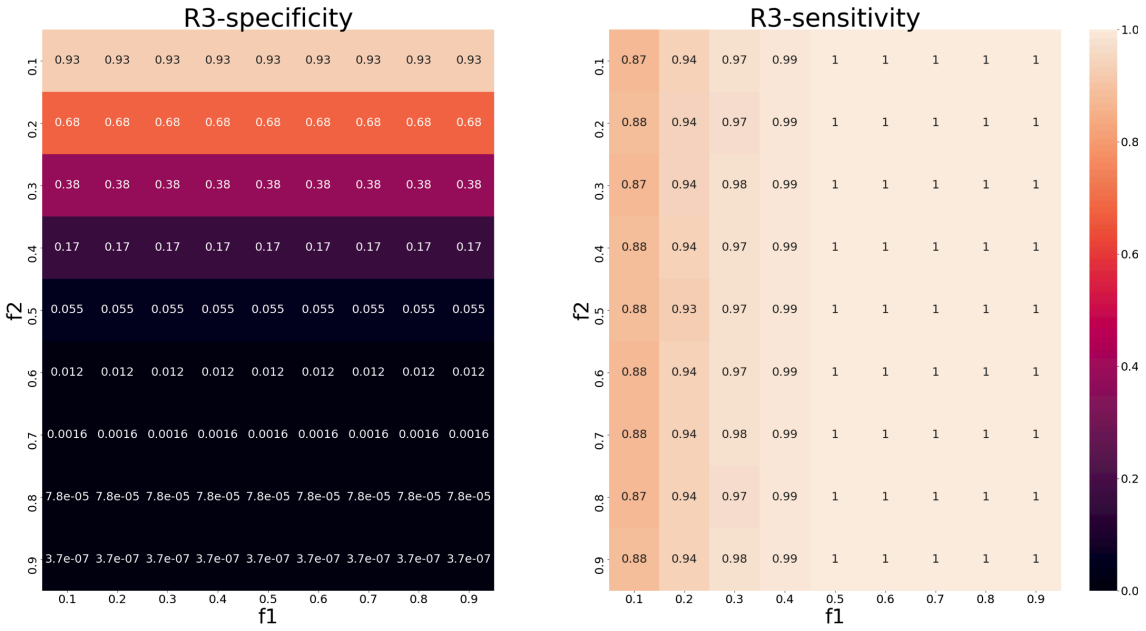


Figure D.6: Specificity and sensitivity of R_3 at $n = 10$, $K = 30$.

Performance of R_5 at $n = 10$, $K = 30$

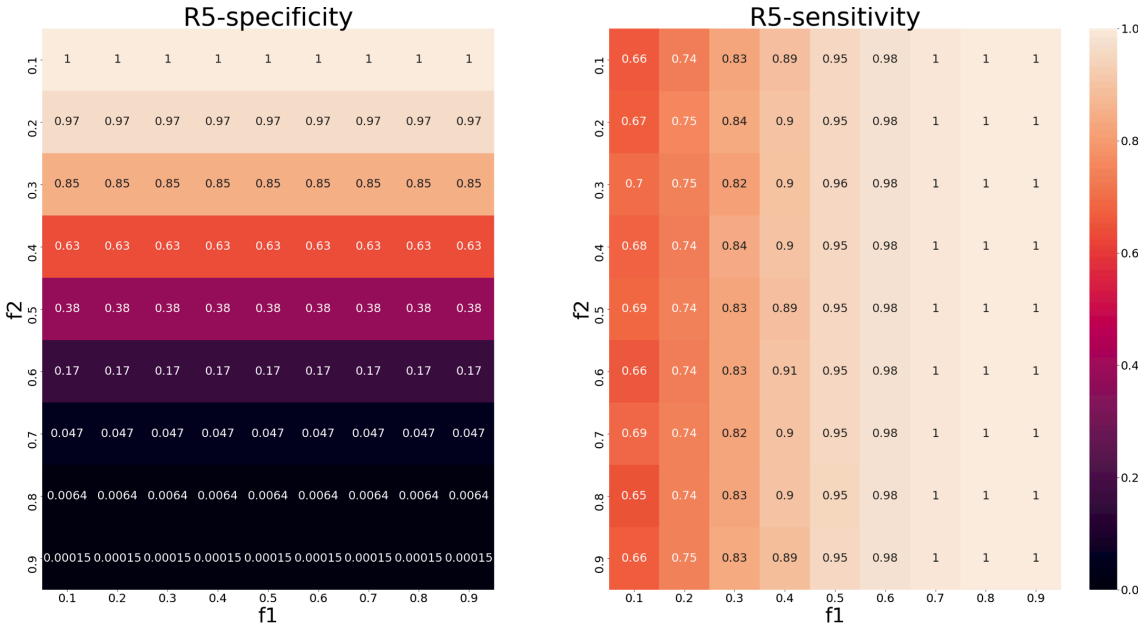


Figure D.7: Specificity and sensitivity of R_5 at $n = 10$, $K = 30$.

Performance of L at $n = 10$, $K = 30$

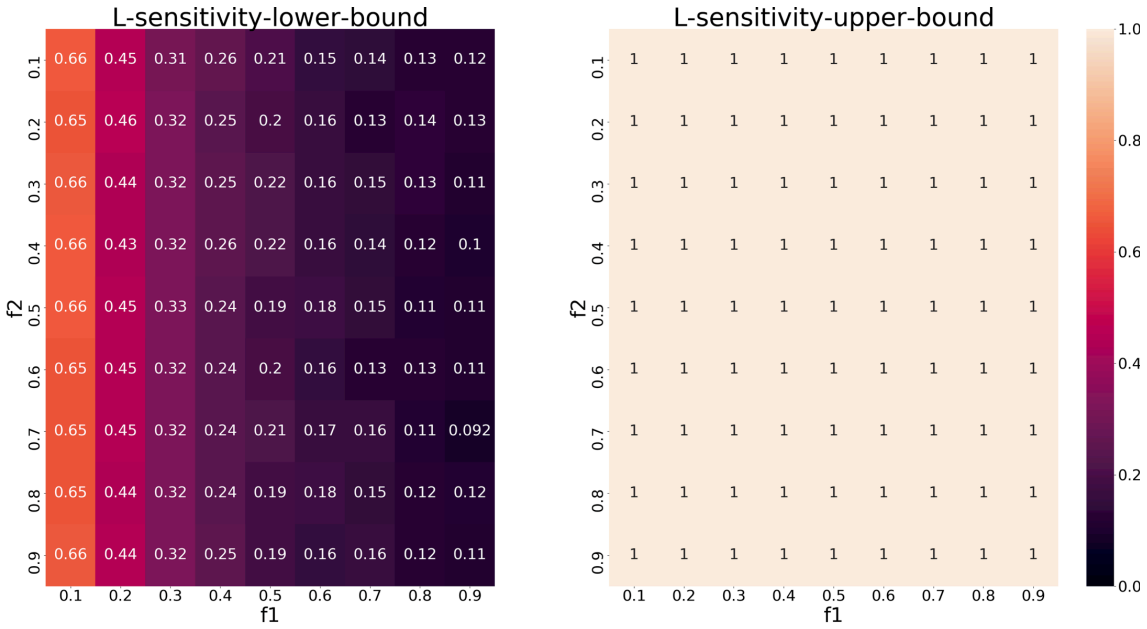


Figure D.8: Lower and upper bounds for the sensitivity of L at $n = 10$, $K = 30$.

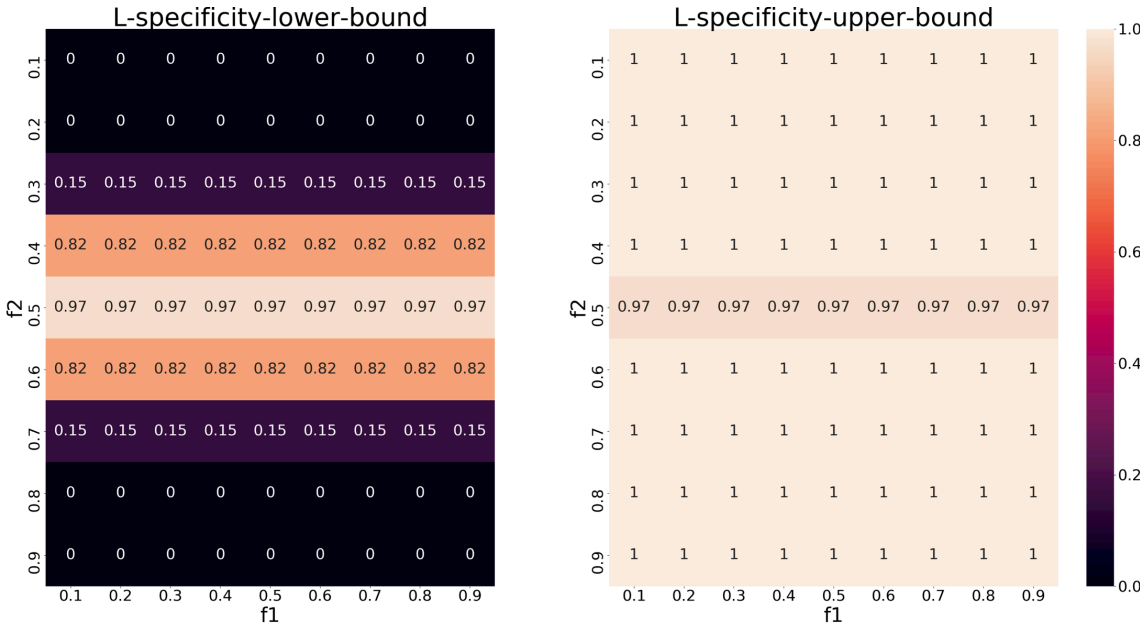


Figure D.9: Lower and upper bounds for the specificity of L at $n = 10$, $K = 30$.

Performance of L at $n = 10$, $K = 50$

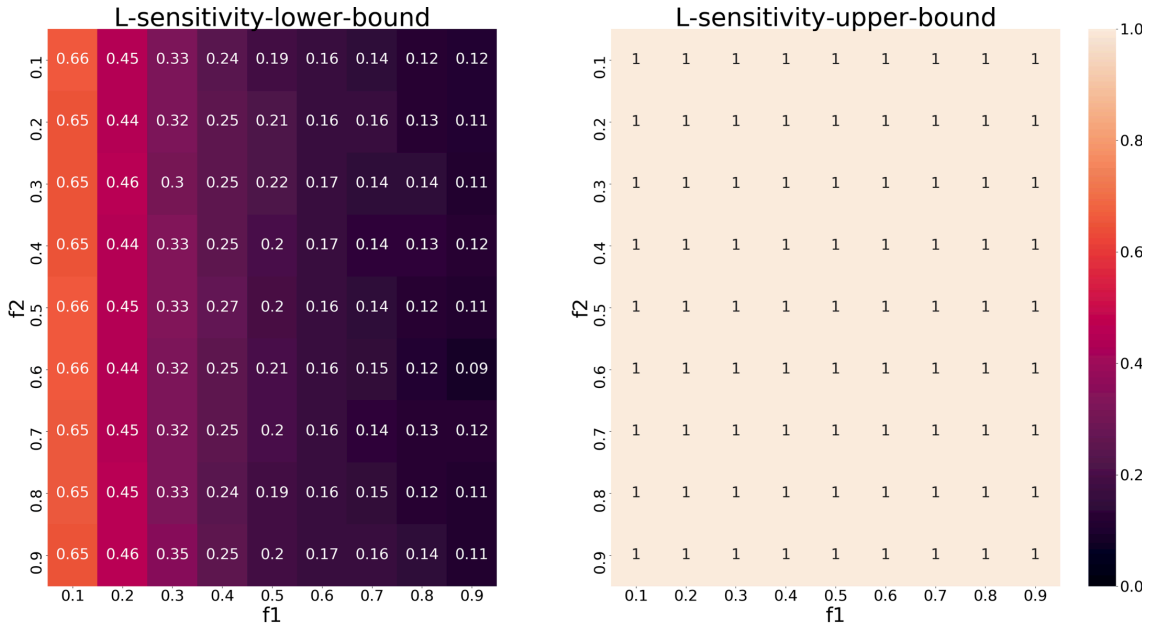


Figure D.10: Lower and upper bounds for the sensitivity of L at $n = 10$, $K = 50$.

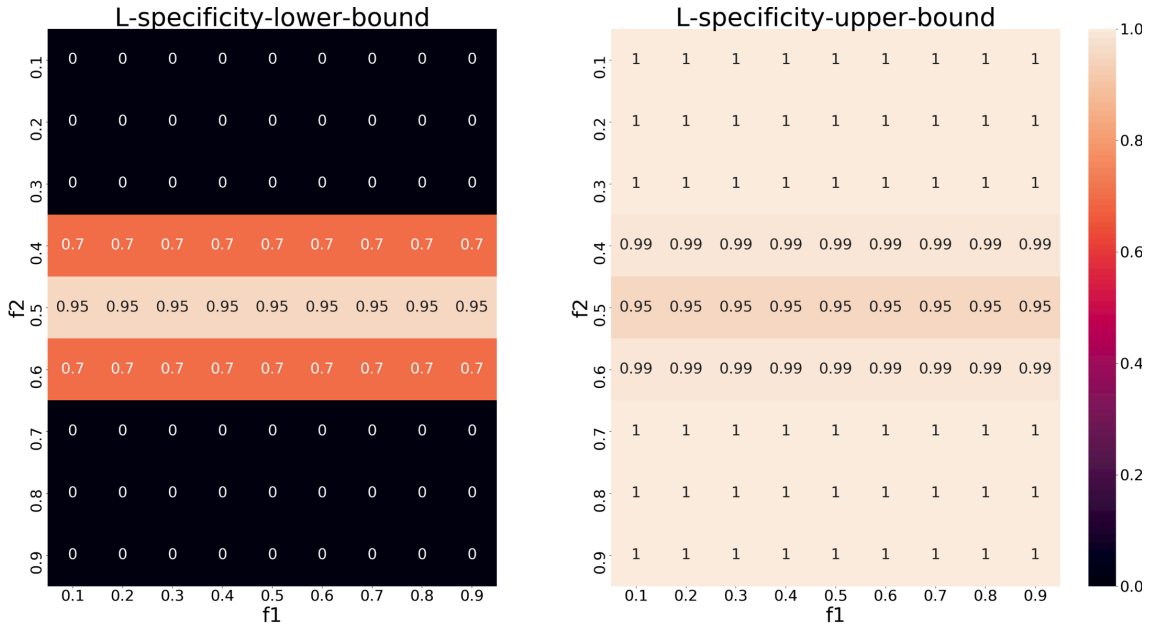


Figure D.11: Lower and upper bounds for the specificity of L at $n = 10$, $K = 50$.

Performance of L at $n = 10$, $K = 100$

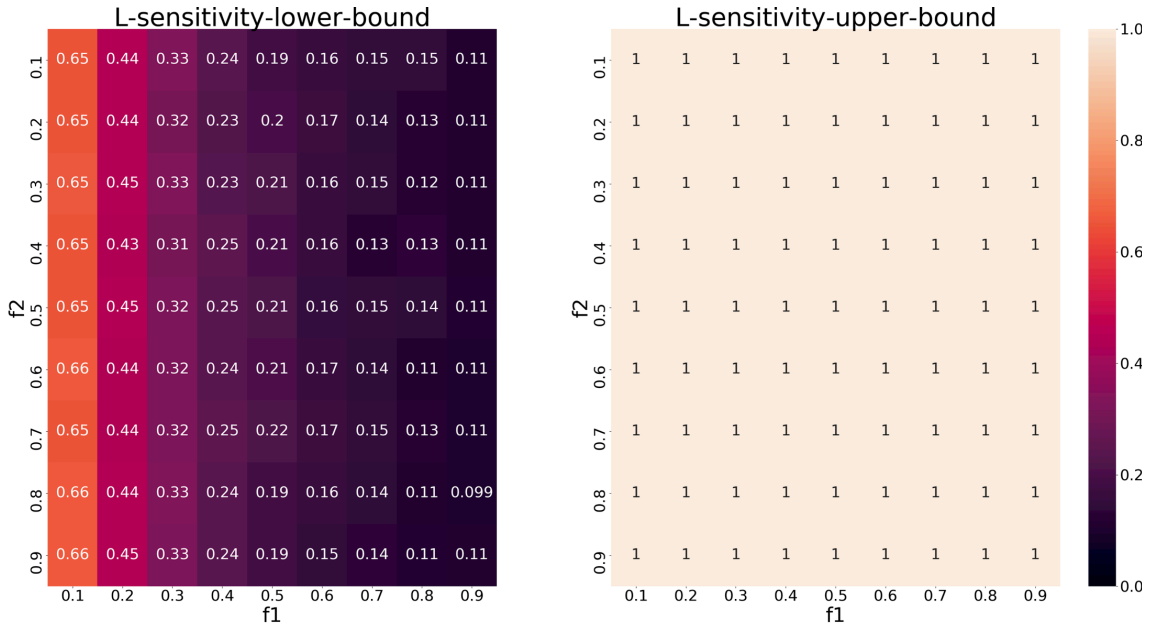


Figure D.12: Lower and upper bounds for the sensitivity of L at $n = 10$, $K = 100$.

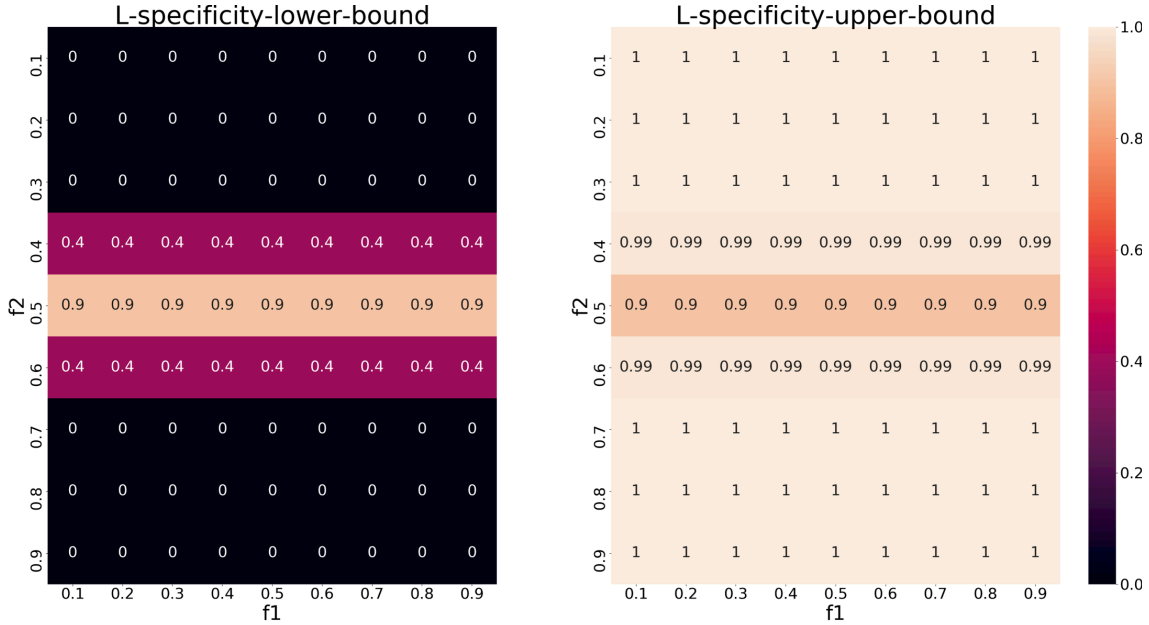


Figure D.13: Lower and upper bounds for the specificity of L at $n = 10$, $K = 100$.

Bibliography

[1] C. A. Albers, G. Lunter, D. G. MacArthur, G. McVean, W. H. Ouwehand, and R. Durbin. Dindel: accurate indel calls from short-read data. *Genome Research*, 21(6):961–973, 2011.

[2] L. B. Alexandrov, S. Nik-Zainal, D. C. Wedge, S. A. J. R. Aparicio, S. Behjati, A. V. Biankin, G. R. Bignell, N. Bolli, A. Borg, A.-L. Børresen-Dale, S. Boyault, B. Burkhardt, A. P. Butler, C. Caldas, H. R. Davies, C. Desmedt, R. Eils, J. E. Eyfjörd, J. A. Foekens, M. Greaves, F. Hosoda, B. Hutter, T. Ilicic, S. Imbeaud, M. Imielinski, N. Jäger, D. T. W. Jones, D. Jones, S. Knappskog, M. Kool, S. R. Lakhani, C. López-Otín, S. Martin, N. C. Munshi, H. Nakamura, P. A. Northcott, M. Pajic, E. Papaemmanuil, A. Paradiso, J. V. Pearson, X. S. Puente, K. Raine, M. Ramakrishna, A. L. Richardson, J. Richter, P. Rosenstiel, M. Schlesner, T. N. Schumacher, P. N. Span, J. W. Teague, Y. Totoki, A. N. J. Tutt, R. Valdés-Mas, M. M. van Buuren, L. van 't Veer, A. Vincent-Salomon, N. Waddell, L. R. Yates, J. Zucman-Rossi, P. Andrew Futreal, U. McDermott, P. Lichter, M. Meyerson, S. M. Grimmond, R. Siebert, E. Campo, T. Shibata, S. M. Pfister, P. J. Campbell, M. R. Stratton, T. Shibata, S. M. Pfister, P. J. Campbell, and M. R. Stratton. Signatures of mutational processes in human cancer. *Nature*, 500(7463):415–421, 2013.

[3] M. J. Beal. Variational Algorithms for Approximate Bayesian Inference. *PhD Thesis*, 2003.

[4] M. J. Beal and Z. Ghahramani. Variational bayesian learning of directed graphical models with hidden variables. *Bayesian Analysis*, 1(4):793–831, 2006.

[5] A. P. Bradley. The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern Recognition*, 30(7):1145–1159, 1997.

[6] F. Bray, J. Ferlay, I. Soerjomataram, R. L. Siegel, L. A. Torre, and A. Jemal. Global cancer statistics 2018: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries. *CA: A Cancer Journal for Clinicians*, 68(6):394–424, 2018.

[7] L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, Oct 2001.

[8] T. Chen and C. Guestrin. XGBoost: a scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '16*, pages 785–794, New York, New York, USA, 2016. ACM Press.

[9] H. Chen-Harris, M. K. Borucki, C. Torres, T. R. Slezak, and J. E. Allen. Ultra-deep mutant spectrum profiling: improving sequencing accuracy using overlapping read pairs. *BMC Genomics*, 14(1):96, 2013.

[10] K. Cibulskis, M. S. Lawrence, S. L. Carter, A. Sivachenko, D. Jaffe, C. Sougnez, S. Gabriel, M. Meyerson, E. S. Lander, and G. Getz. Sensitive detection of somatic point mutations in impure and heterogeneous cancer samples. *Nature Biotechnology*, 31(3):213–219, 2013.

[11] F. S. Collins. *The Language of Life*. HarperCollins Publishers Inc., 2011.

[12] A. G. Deshwar, S. Vembu, C. K. Yung, G. H. Jang, L. Stein, and Q. Morris. PhyloWGS: reconstructing subclonal composition and evolution from whole-genome sequencing of tumors. *Genome Biology*, 16(1):35, 2015.

[13] H. Detering, L. Tomás, T. Prieto, and D. Posada. Accuracy of somatic variant detection in multiregional tumor sequencing data. *bioRxiv*, page 655605, 2019.

[14] A. Dobin, C. A. Davis, F. Schlesinger, J. Drenkow, C. Zaleski, S. Jha, P. Batut, M. Chaisson, and T. R. Gingeras. STAR: ultrafast universal RNA-seq aligner. *Bioinformatics*, 29(1):15–21, 2013.

[15] F. Dorri, S. Jewell, A. Bouchard-Côté, and S. P. Shah. Somatic mutation detection and classification through probabilistic integration of clonal population information. *Communications Biology*, 2(1):44, 2019.

[16] M. El-Kebir. SPhyR: tumor phylogeny estimation from single-cell sequencing data under loss and error. *Bioinformatics*, 34(17):i671–i679, 2018.

[17] M. El-Kebir, L. Oesper, H. Acheson-Field, and B. J. Raphael. Reconstruction of clonal trees and tumor composition from multi-sample sequencing data. *Bioinformatics*, 31(12):i62–70, 2015.

[18] M. El-Kebir, G. Satas, L. Oesper, and B. J. Raphael. Inferring the Mutational History of a Tumor Using Multi-state Perfect Phylogeny Mixtures. *Cell Systems*, 3(1):43–53, 2016.

[19] R. A. Fisher. Statistical methods for research workers. 1934.

[20] Y. Freund and R. E. Schapire. A Decision-theoretic generalization of on-Line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119–139, aug 1997.

[21] S. Geman and D. Geman. Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-6(6):721–741, 1984.

[22] M. Gerlinger, A. J. Rowan, S. Horswell, J. Larkin, D. Endesfelder, E. Gronroos, P. Martinez, N. Matthews, A. Stewart, P. Tarpey, I. Varela, B. Phillimore, S. Begum, N. Q. McDonald, A. Butler, D. Jones, K. Raine, C. Latimer, C. R. Santos, M. Nohadani, A. C. Eklund, B. Spencer-Dene, G. Clark, L. Pickering, G. Stamp, M. Gore, Z. Szallasi, J. Downward, P. A. Futreal, and C. Swanton. Intratumor Heterogeneity and Branched Evolution Revealed by Multiregion Sequencing. *New England Journal of Medicine*, 366(10):883–892, 2012.

[23] Z. Ghahramani and M. J. Beal. Variational inference for bayesian mixtures of factor analysers. In *Advances in Neural Information Processing Systems 12*, pages 449–455. MIT Press, 2000.

[24] Z. Ghahramani and M. J. Beal. Propagation algorithms for variational bayesian learning. In *Advances in Neural Information Processing Systems 13*, pages 507–513. MIT Press, 2001.

[25] G. R. Grant, M. H. Farkas, A. D. Pizarro, N. F. Lahens, J. Schug, B. P. Brunk, C. J. Stoeckert, J. B. Hogenesch, and E. A. Pierce. Comparative analysis of RNA-Seq alignment algorithms and the RNA-Seq unified mapper (RUM). *Bioinformatics*, 27(18):2518–2528, 2011.

[26] D. Gusfield. Efficient algorithms for inferring evolutionary trees. *Networks*, 21(1):19–28, 1991.

[27] I. Hajirasouliha, A. Mahmoody, and B. J. Raphael. A combinatorial approach for analyzing intra-tumor heterogeneity from high-throughput sequencing data. *Bioinformatics*, 30(12):i78–i86, 2014.

[28] W. K. Hastings. Monte Carlo Sampling Methods Using Markov Chains and Their Applications. *Biometrika*, 57(1):97, 1970.

[29] J. A. Hoeting, D. Madigan, A. E. Raftery, and C. T. Volinsky. Bayesian Model Averaging: A Tutorial. *Statistical Science*, 14(4):382–417, 1999.

[30] K. Itahashi, S. Kondo, T. Kubo, Y. Fujiwara, M. Kato, H. Ichikawa, T. Koyama, R. Tokumasu, J. Xu, C. S. Huettner, V. V. Michelini, L. Parida, T. Kohno, and N. Yamamoto. Evaluating Clinical Genome Sequence Analysis by Watson for Genomics. *Frontiers in Medicine*, 5:305, 2018.

[31] H. Jeffreys. Some Tests of Significance, Treated by the Theory of Probability. *Mathematical Proceedings of the Cambridge Philosophical Society*, 31(2):203–222, 1935.

[32] M. Josephidou, A. G. Lynch, and S. Tavaré. multiSNV: a probabilistic approach for improving detection of somatic point mutations from multiple related tumour samples. *Nucleic Acids Research*, 43(9):e61–e61, 2015.

[33] R. E. Kass and A. E. Raftery. Bayes Factors. *Source Journal of the American Statistical Association*, 90(430):773–795, 1995.

[34] D. Kim, B. Langmead, and S. L. Salzberg. HISAT: a fast spliced aligner with low memory requirements. *Nature Methods*, 12(4):357–360, 2015.

[35] D. Kim, G. Pertea, C. Trapnell, H. Pimentel, R. Kelley, and S. L. Salzberg. TopHat2: accurate alignment of transcriptomes in the presence of insertions, deletions and gene fusions. *Genome Biology*, 14(4):R36, 2013.

[36] S. Kim, K. Scheffler, A. L. Halpern, M. A. Bekritsky, E. Noh, M. Källberg, X. Chen, D. Beyter, P. Krusche, and C. T. Saunders. Strelka2: fast and accurate calling of germline and somatic variants. *Nature Methods*, 15(8):591–594, 2018.

[37] M. Kimura. The number of heterozygous nucleotide sites maintained in a finite population due to steady flux of mutations. *Genetics*, 61(4), 1969.

[38] K. Kinzler, M. Nilbert, B. Vogelstein, T. Bryan, D. Levy, K. Smith, A. Preisinger, S. R. Hamilton, P. Hedge, A. Markham, M. Carlson, G. Joslyn, J. Groden, R. White, Y. Miki, Y. Miyoshi, I. Nishisho, and Y. Nakamura. Identification of a gene located at chromosome 5q21 that is mutated in colorectal cancers. *Science*, 251(4999):1366–1370, 1991.

[39] D. C. Koboldt, K. Chen, T. Wylie, D. E. Larson, M. D. McLellan, E. R. Mardis, G. M. Weinstock, R. K. Wilson, and L. Ding. VarScan: variant detection in massively parallel sequencing of individual and pooled samples. *Bioinformatics*, 25(17):2283–2285, 2009.

[40] D. C. Koboldt, Q. Zhang, D. E. Larson, D. Shen, M. D. McLellan, L. Lin, C. A. Miller, E. R. Mardis, L. Ding, and R. K. Wilson. VarScan 2: somatic mutation and copy number alteration discovery in cancer by exome sequencing. *Genome Research*, 22(3):568–576, 2012.

[41] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012.

[42] B. Langmead and S. L. Salzberg. Fast gapped-read alignment with Bowtie 2. *Nature Methods*, 9(4):357–359, 2012.

[43] H. Li and R. Durbin. Fast and accurate short read alignment with Burrows-Wheeler transform. *Bioinformatics*, 25(14):1754–60, 2009.

[44] H. Li and R. Durbin. Fast and accurate short read alignment with Burrows-Wheeler transform. *Bioinformatics*, 25(14):1754–1760, 2009.

[45] H. Li, J. Ruan, and R. Durbin. Mapping short DNA sequencing reads and calling variants using mapping quality scores. *Genome Research*, 18(11):1851–8, 2008.

[46] Y. Liao, G. K. Smyth, and W. Shi. The Subread aligner: fast, accurate and scalable read mapping by seed-and-vote. *Nucleic Acids Research*, 41(10):e108–e108, 2013.

[47] J. G. Lohr, P. Stojanov, S. L. Carter, P. Cruz-Gordillo, M. S. Lawrence, D. Auclair, C. Sougnez, B. Knoechel, J. Gould, G. Saksena, K. Cibulskis, A. McKenna, M. A. Chapman, R. Straussman, J. Levy, L. M. Perkins, J. J. Keats, S. E. Schumacher, M. Rosenberg, K. C. Anderson, P. Richardson, A. Krishnan, S. Lonial, J. Kaufman, D. S. Siegel, D. H. Vesole, V. Roy, C. E. Rivera, S. V. Rajkumar, S. Kumar, R. Fonseca, G. J. Ahmann, P. L. Bergsagel, A. K. Stewart, C. C. Hofmeister, Y. A. Efebera, S. Jagannath, A. Chari, S. Trudel, D. Reece, J. Wolf, T. Martin, T. Zimmerman, C. Rosenbaum, A. J. Jakubowiak, D. Lebovic, R. Vij, K. Stockerl-Goldstein, G. Getz, and T. R. Golub. Widespread Genetic Heterogeneity in Multiple Myeloma: Implications for Targeted Therapy. *Cancer Cell*, 25(1):91–101, 2014.

[48] F. Marass, F. Mouliere, K. Yuan, N. Rosenfeld, and F. Markowetz. A phylogenetic latent feature model for clonal deconvolution. *The Annals of Applied Statistics*, 10(4):2377–2404, 2016.

[49] E. R. Mardis. DNA sequencing technologies: 2006–2016. *Nature Protocols*, 12(2):213–218, 2017.

[50] J. Meienberg, R. Bruggmann, K. Oexle, and G. Matyas. Clinical sequencing: is WGS the better WES? *Human Genetics*, 135(3):359–362, 2016.

[51] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller. Equation of State Calculations by Fast Computing Machines. *The Journal of Chemical Physics*, 21(6):1087–1092, 1953.

[52] M. L. Metzker. Sequencing technologies — the next generation. *Nature Reviews Genetics*, 11(1):31–46, 2010.

[53] T. Moriyama, S. Imoto, S. Hayashi, Y. Shiraishi, S. Miyano, and R. Yamaguchi. A Bayesian model integration for mutation calling through data partitioning. *Bioinformatics*, 2019.

[54] T. Moriyama, S. Imoto, S. Miyano, and R. Yamaguchi. Accurate and flexible bayesian mutation call from multi-regional tumor samples. In *Mathematical and Computational Oncology*, pages 47–61. Springer International Publishing, 2019.

[55] T. Moriyama, Y. Shiraishi, K. Chiba, R. Yamaguchi, S. Imoto, and S. Miyano. OVarCall: Bayesian Mutation Calling Method Utilizing Overlapping Paired-End Reads. *IEEE Transactions on NanoBioscience*, 16(2):116–122, 2017.

[56] A. Mortazavi, B. A. Williams, K. McCue, L. Schaeffer, and B. Wold. Mapping and quantifying mammalian transcriptomes by RNA-Seq. *Nature Methods*, 5(7):621–628, 2008.

[57] K. Nakamura, T. Oshima, T. Morimoto, S. Ikeda, H. Yoshikawa, Y. Shiwa, S. Ishikawa, M. C. Linak, A. Hirai, H. Takahashi, M. Altaf-Ul-Amin, N. Ogasawara, and S. Kanaya. Sequence-specific error profile of Illumina sequencers. *Nucleic Acids Research*, 39(13):e90, 2011.

[58] G. Narzisi, A. Corvelo, K. Arora, E. A. Bergmann, M. Shah, R. Musunuri, A.-K. Emde, N. Robine, V. Vacic, and M. C. Zody. Genome-wide somatic variant calling using localized colored de Bruijn graphs. *Communications Biology*, 1(1):20, 2018.

[59] R. M Neal. Probabilistic Inference Using Markov Chain Monte Carlo Methods. Technical report, 1993.

[60] S. Nik-Zainal, L. B. Alexandrov, D. C. Wedge, P. Van Loo, C. D. Greenman, K. Raine, D. Jones, J. Hinton, J. Marshall, L. A. Stebbings, A. Menzies, S. Martin, K. Leung, L. Chen, C. Leroy, M. Ramakrishna, R. Rance, K. W. Lau, L. J. Mudie, I. Varela, D. J. McBride, G. R. Bignell, S. L. Cooke, A. Shlien, J. Gamble, I. Whitmore, M. Maddison, P. S. Tarpey, H. R. Davies, E. Papaemmanuil, P. J. Stephens, S. McLaren, A. P. Butler, J. W. Teague, G. Jönsson, J. E. Garber, D. Silver, P. Miron, A. Fatima, S. Boyault, A. Langerød, A. Tutt, J. W.M. Martens, S. A.J.R. Aparicio, Å. Borg, A. V. Salomon, G. Thomas, A.-L. Børresen-Dale, A. L. Richardson, M. S. Neuberger, P. A. Futreal, P. J. Campbell, and M. R. Stratton. Mutational Processes Molding the Genomes of 21 Breast Cancers. *Cell*, 149(5):979–993, 2012.

[61] B. J. Pope, T. Nguyen-Dumont, F. Hammet, and D. J. Park. ROVER variant caller: read-pair overlap considerate variant-calling software applied to PCR-based massively parallel sequencing datasets. *Source Code for Biology and Medicine*, 9(1):3, 2014.

[62] R. Poplin, P.-C. Chang, D. Alexander, S. Schwartz, T. Colthurst, A. Ku, D. Newburger, J. Dijamco, N. Nguyen, P. T. Afshar, S. S. Gross, L. Dorfman, C. Y. McLean, and M. A. DePristo. A universal SNP and small-indel variant

caller using deep neural networks. *Nature Biotechnology*, 36(10):983–987, 2018.

[63] R. C. Poulos, Y. T. Wong, R. Ryan, H. Pang, and J. W. H. Wong. Analysis of 7,815 cancer exomes reveals associations between mutational processes and somatic driver mutations. *PLOS Genetics*, 14(11):e1007779, 2018.

[64] A. E. Raftery, D. Madigan, and J. A. Hoeting. Bayesian Model Averaging for Linear Regression Models. Technical Report 437, 1997.

[65] J. G. Reiter, A. P. Makohon-Moore, J. M. Gerold, I. Bozic, K. Chatterjee, C. A. Iacobuzio-Donahue, B. Vogelstein, and M. A. Nowak. Reconstructing metastatic seeding patterns of human cancers. *Nature Communications*, 8:14114, 2017.

[66] G. O. Roberts and J. S. Rosenthal. General state space Markov chains and MCMC algorithms. *Probability Surveys*, 1(0):20–71, 2004.

[67] S. M. E. Sahraeian, R. Liu, B. Lau, K. Podesta, M. Mohiyuddin, and H. Y. K. Lam. Deep convolutional neural networks for accurate somatic mutation detection. *Nature Communications*, 10(1):1041, 2019.

[68] R. Salari, S. S. Saleh, D. Kashef-Haghighi, D. Khavari, D. E Newburger, R. B. West, A. Sidow, and S. Batzoglou. Inference of tumor phylogenies with improved somatic mutation discovery. *Journal of Computational Biology*, 20(11):933–44, 2013.

[69] F. Sanger and A. R. Coulson. A rapid method for determining sequences in DNA by primed synthesis with DNA polymerase. *Journal of Molecular Biology*, 94(3):441–448, 1975.

[70] G. Satas and B. J. Raphael. Tumor phylogeny inference using tree-constrained importance sampling. *Bioinformatics*, 33(14):i152–i160, 2017.

[71] C. T. Saunders, W. S. W. Wong, S. Swamy, J. Becq, L. J. Murray, and R. K. Cheetham. Strelka: accurate somatic small-variant calling from sequenced tumor-normal sample pairs. *Bioinformatics*, 28(14):1811–1817, 2012.

[72] Y. Shiraishi, Y. Sato, K. Chiba, Y. Okuno, Y. Nagata, K. Yoshida, N. Shiba, Y. Hayashi, H. Kume, Y. Homma, M. Sanada, S. Ogawa, and S. Miyano. An empirical Bayesian framework for somatic mutation detection from cancer genome sequencing data. *Nucleic Acids Research*, 41(7):e89, 2013.

[73] E. M. Southern. Detection of specific sequences among DNA fragments separated by gel electrophoresis. *Journal of Molecular Biology*, 98(3):503–517, 1975.

[74] G. Stanta and S. Bonin. Overview on Clinical Relevance of Intra-Tumor Heterogeneity. *Frontiers in Medicine*, 5:85, 2018.

[75] F. Strino, F. Parisi, M. Micsinai, and Y. Kluger. TrAp: a tree approach for fingerprinting subclonal tumor composition. *Nucleic Acids Research*, 41(17):e165–e165, 2013.

[76] L. Tierney. Markov Chains for Exploring Posterior Distributions. *The Annals of Statistics*, 22(4):1701–1728, 1994.

[77] N. Usuyama, Y. Shiraishi, Y. Sato, H. Kume, Y. Homma, S. Ogawa, S. Miyano, and S. Imoto. HapMuC: somatic mutation calling using heterozygous germ line variants near candidate mutations. *Bioinformatics*, 30(23):3302–3309, 2014.

[78] K. E. van Rens, V. Mäkinen, and A. I. Tomescu. SNV-PPILP: refined SNV calling for tumor data using perfect phylogenies and ILP. *Bioinformatics*, 31(7):1133–1135, 2015.

[79] D. Weese, M. Holtgrewe, and K. Reinert. RazerS 3: Faster, fully sensitive read mapping. *Bioinformatics*, 28(20):2592–2599, 2012.

[80] R. A. Weinberg. *The Biology of Cancer*. Garland Science, 2013.

[81] A. Wilm, P. P. K. Aw, D. Bertrand, G. H. T. Yeo, S. H. Ong, C. H. Wong, C. C. Khor, R. Petric, M. L. Hibberd, and N. Nagarajan. LoFreq: a sequence-quality aware, ultra-sensitive variant caller for uncovering cell-population heterogeneity from high-throughput sequencing datasets. *Nucleic Acids Research*, 40(22):11189–11201, 2012.

[82] T. D. Wu and S. Nacu. Fast and SNP-tolerant detection of complex variants and splicing in short reads. *Bioinformatics*, 26(7):873–881, 2010.

[83] K. Yoshida, M. Sanada, Y. Shiraishi, D. Nowak, Y. Nagata, R. Yamamoto, Y. Sato, A. Sato-Otsubo, A. Kon, M. Nagasaki, G. Chalkidis, Y. Suzuki, M. Shiosaka, R. Kawahata, T. Yamaguchi, M. Otsu, N. Obara, M. Sakata-Yanagimoto, K. Ishiyama, H. Mori, F. Nolte, W.-K. Hofmann, S. Miyawaki, S. Sugano, C. Haferlach, H. P. Koeffler, L.-Y. Shih, T. Haferlach, S. Chiba, H. Nakauchi, S. Miyano, and S. Ogawa. Frequent pathway mutations of splicing machinery in myelodysplasia. *Nature*, 478(7367):64–69, 2011.

[84] K. Yuan, T. Sakoparnig, F. Markowetz, and N. Beerenwinkel. BitPhylogeny: a probabilistic framework for reconstructing intra-tumor phylogenies. *Genome Biology*, 16(1):36, 2015.

[85] H. Zafar, N. Navin, K. Chen, and L. Nakhleh. SiCloneFit: Bayesian inference of population structure, genotype, and phylogeny of tumor clones from single-cell genome sequencing data. *Genome Research*, 2019.

[86] H. Zafar, A. Tzen, N. Navin, K. Chen, and L. Nakhleh. SiFit: inferring tumor trees from single-cell sequencing data under finite-sites models. *Genome Biology*, 18(1):178, 2017.