BIOINFORMATICS ORIGINAL PAPER

Vol. 30 no. 23 2014, pages 3302–3309
doi:10.1093/bioinformatics/btu537

Sequence analysis

Advance Access publication August 14, 2014

HapMuC: somatic mutation calling using heterozygous germ line variants near candidate mutations

Naoto Usuyama[1], Yuichi Shiraishi[1], Yusuke Sato[2,3], Haruki Kume[2], Yukio Homma[2], Seishi Ogawa[3], Satoru Miyano[1] and Seiya Imoto[1,*]

[1]Human Genome Center, Institute of Medical Science, The University of Tokyo, Tokyo 108-8639, [2]Department of Urology, Graduate School of Medicine, The University of Tokyo, Tokyo 113-8655 and [3]Department of Pathology and Tumor Biology, Graduate School of Medicine, Kyoto University, Kyoto 606-8501, Japan

Associate Editor: Michael Brudno

ABSTRACT

Motivation: Identifying somatic changes from tumor and matched normal sequences has become a standard approach in cancer research. More specifically, this requires accurate detection of somatic point mutations with low allele frequencies in impure and heterogeneous cancer samples. Although haplotype phasing information derived by using heterozygous germ line variants near candidate mutations would improve accuracy, no somatic mutation caller that uses such information is currently available.

Results: We propose a Bayesian hierarchical method, termed HapMuC, in which power is increased by using available information on heterozygous germ line variants located near candidate mutations. We first constructed two generative models (the mutation model and the error model). In the generative models, we prepared candidate haplotypes, considering a heterozygous germ line variant if available, and the observed reads were realigned to the haplotypes. We then inferred the haplotype frequencies and computed the marginal likelihoods using a variational Bayesian algorithm. Finally, we derived a Bayes factor for evaluating the possibility of the existence of somatic mutations. We also demonstrated that our algorithm has superior specificity and sensitivity compared with existing methods, as determined based on a simulation, the TCGA Mutation Calling Benchmark 4 datasets and data from the COLO-829 cell line.

Availability and implementation: The HapMuC source code is available from http://github.com/usuyama/hapmuc.

Contact: imoto@ims.u-tokyo.ac.jp

Supplementary information: Supplementary data are available at *Bioinformatics* online.

Received on December 12, 2013; revised on July 28, 2014; accepted on August 4, 2014

1 INTRODUCTION

Cancer is caused by genomic alterations. In particular, somatic point mutations play a clear role in cancer development (Forbes *et al.*, 2011; Pleasance *et al.*, 2010). Driven by the recent advancement of high-throughput DNA sequencing technologies, mutation calling from tumor and matched normal sequences is becoming a standard approach in cancer research (Meyerson *et al.*, 2010). The simple goal of this approach is to detect genetic alterations that are unique to the tumor sample. For this purpose, statistical approaches that allow distinction of true somatic mutations from relatively numerous germ line variants, sequencing errors and alignment errors are required. Moreover, mutation calling methods need to be sufficiently sensitive to detect targets with low allele frequencies in cancer because of low tumor contents, copy number variations and tumor heterogeneity. In particular, it is important to capture subclones demonstrating tumor heterogeneity because this is thought to be one of the main causes of therapeutic resistance and cancer recurrence (Ding *et al.*, 2012; Nik-Zainal *et al.*, 2012; Shah *et al.*, 2009).

There are mainly two approaches to detect somatic mutations using sequence data of a tumor–normal pair. The first approach is to use Fisher's exact test on read counts for evaluating the differences in allele frequencies between tumor and matched normal tissue; VarScan 2 (Koboldt *et al.*, 2012) and Genomon (Yoshida *et al.*, 2011) are categorized in this group. By ignoring complex biological structures, this approach achieves a reasonable computational speed and robustness. The second approach is based on Bayesian modeling, which calculates the posterior probability that tumor and normal samples would have different genotypes. SomaticSniper (Larson *et al.*, 2012), jointSNVMix (Roth *et al.*, 2012), Strelka (Saunders *et al.*, 2012) and MuTect (Cibulskis *et al.*, 2013) are representatives of this second group. Comparative concurrent analysis of a pair of tumor and normal sequences achieved a performance superior to the previous independent type of analysis (Goya *et al.*, 2010; Ley *et al.*, 2008; Pleasance *et al.*, 2010) in which tumor and normal samples were genotyped independently and results were subtracted.

Here, we focused on heterozygous germ line variants located near candidate somatic mutations. In real data, we observed that somatic mutations occurred mostly on one of two normal haplotypes; however, sequencing errors occurred randomly on normal haplotypes. The data are shown in Section 3.1. Based on this observation, we propose a novel statistical method, HapMuC, which can use the information of heterozygous germ line variants near candidate mutations. First, we constructed two generative models under a Bayesian statistical framework: one represents true somatic mutations and the other regards candidate somatic mutations as errors. In our generative models, we prepared four candidate haplotypes by combining a candidate mutation and a heterozygous germ line variant, if available. The alignment probabilities of the observed reads given each candidate haplotype

*To whom correspondence should be addressed

were then computed by using profile hidden Markov models (profile HMM; Albers *et al.*, 2011; Eddy, 1998). Next, we inferred the haplotype frequencies and calculated the marginal likelihoods by using a variational Bayesian algorithm (Beal, 2003; Blei *et al.*, 2003). Finally, we derived a Bayes factor (BF), which is the ratio of the marginal likelihoods of these two models, to evaluate the possibility of the presence of somatic mutations.

We first demonstrated the effectiveness of HapMuC in a simulation study. Second, we also confirmed its efficiency based on a real sequence dataset obtained from breast cancer cell line samples, which forms part of the The Cancer Genome Atlas (TCGA) Mutation Calling Benchmark 4 datasets. Third, we tested HapMuC in the context of a whole genome using the TCGA datasets, and the analysis suggested that our algorithm performed better than the existing methods, including VarScan 2, SomaticSniper, Strelka and MuTect. In addition, we confirmed that the sensitivity of our algorithm was superior or comparable with the existing methods using the COLO-829 dataset with its list of validated somatic mutations (Pleasance *et al.*, 2010). Finally, we discuss limitations and propose future research directions for accurate mutation calling.

2 METHODS

2.1 Bayesian latent Dirichlet models

We constructed a generative model of observed reads. Suppose that x is an observed read and generated from one of four haplotypes. Let z_{ij} be an indicator variable that is equal to 1 if the ith read x_i was generated from the jth haplotype or, alternatively, is equal to 0 if x_i was generated from other haplotypes. We also define $\boldsymbol{z_i}$ as the ith row of $\boldsymbol{Z}$. A likelihood of the vector of reads $\boldsymbol{x}$ is formulated by a conditional density $p(\boldsymbol{x}|\boldsymbol{Z})$, where $\boldsymbol{Z}$ is a haplotype matrix whose (i, j)th element is z_{ij}. Fixing the haplotype matrix $\boldsymbol{Z}$, we can compute the value of the likelihood by a profile HMM (Albers *et al.*, 2011). The profile HMM formulates alignment probabilities of a read given a reference haplotype, considering mapping qualities, base qualities and homopolymer length:

$$p(x_i, \mathbf{X}_i, \mathbf{I}_i|\mathcal{H}_j, \boldsymbol{\theta}),$$

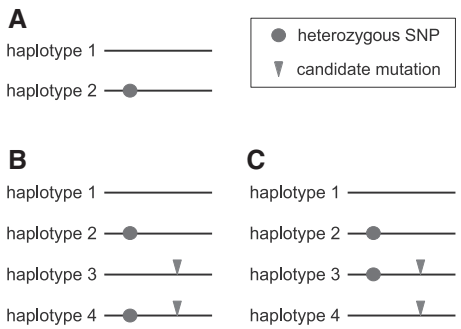

Fig. 1. Possible types of haplotypes around a candidate mutation. (**A**) Normal haplotypes around a heterozygous germ line variant. Under the assumption that a somatic mutation occurs within one of two normal haplotypes, haplotype 3 was generated by mutating the first or second haplotype. (**B**) Haplotype 1 was mutated, yielding haplotype 3. (**C**) Mutated haplotype 2 yields haplotype 3. In both cases, haplotype 4 was generated only by sequencing errors

where $\mathbf{X}_i$ is a hidden variable for alignment states, $\mathbf{I}_i$ is a hidden variable for insertion states and $\boldsymbol{\theta}$ represents the profile HMM model parameters. Albers *et al.* also provided a Viterbi algorithm to infer the maximum-likelihood alignment efficiently:

$$(\hat{\mathbf{X}}_i, \hat{\mathbf{I}}_i) = \underset{\mathbf{X}_i, \mathbf{I}_i}{\operatorname{argmax}}\, p(x_i, \mathbf{X}_i, \mathbf{I}_i|\mathcal{H}_j, \boldsymbol{\theta}).$$

We used their modeling and algorithm to infer the probability of observing a short read given a reference haplotype, that is,

$$p(x_i|\boldsymbol{z}_i : z_{ij} = 1) = p(x_i|\mathcal{H}_j) = p(x_i, \hat{\mathbf{X}}_i, \hat{\mathbf{I}}_i|\mathcal{H}_j, \boldsymbol{\theta})$$

In addition, when read-pair information is available, we approximate the likelihood by multiplying the likelihoods of the first read and the second read, as a paired-read is considered to arise from the same haplotype.

The haplotype matrix is, however, usually unknown, and we regard the elements in $\boldsymbol{Z}$ as latent variables that follow a multinomial distribution with density:

$$p(\boldsymbol{Z}|\boldsymbol{\pi}) = \prod_i \prod_j \pi_j^{z_{ij}},$$

where $\boldsymbol{\pi}$ is a vector of haplotype frequencies.

In this article, we also aimed to use information of a heterozygous germ line variant located near a candidate somatic mutation. Here, we focus on a candidate somatic mutation, which is located near a heterozygous germ line mutation. Therefore, around the targeted candidate mutation, there are two normal haplotypes, as shown in Figure 1. We should note that because we assume that a somatic mutation occurs independently on one of two normal haplotypes, the fourth haplotype is not observed in the ideal case, i.e. $\pi_4 = 0$ and $\boldsymbol{\pi} = (\pi_1, \pi_2, \pi_3)'$. However, in practice, the observed reads contain sequencing errors; thus, to achieve robust statistical inference, we added a variable ε for representing haplotype error, which achieves an extension of $p(\boldsymbol{Z}|\boldsymbol{\pi})$ to $p(\boldsymbol{Z}|\boldsymbol{\pi}, \varepsilon)$. Because tumor and normal samples have different haplotypes, it is necessary to construct statistical models of $p(\boldsymbol{Z}|\boldsymbol{\pi}, \varepsilon)$ in tumor and normal samples, separately. The concrete modelings are shown in later sections.

Because the haplotype frequencies, $\boldsymbol{\pi}$, are also parameters, we built a Bayesian hierarchical structure as $p(\boldsymbol{\pi}|\boldsymbol{\alpha})$, using Dirichlet distribution with a parameter vector $\boldsymbol{\alpha}$. We also assumed that the haplotype error, ε, follows the beta distribution, $p(\varepsilon|\boldsymbol{\gamma})$, with parameter $\boldsymbol{\gamma}$.

Using the probabilistic distributions described above, we define the decomposition of the joint distribution of the form:

$$p(\boldsymbol{x}, \boldsymbol{Z}, \boldsymbol{\pi}, \varepsilon|\boldsymbol{\alpha}, \boldsymbol{\gamma}) = p(\boldsymbol{x}|\boldsymbol{Z})p(\boldsymbol{Z}|\boldsymbol{\pi}, \varepsilon)p(\boldsymbol{\pi}|\boldsymbol{\alpha})p(\varepsilon|\boldsymbol{\gamma}). \quad (1)$$

In our case, there are two types of sequencing data: one is from a tumor sample, denoted by $\boldsymbol{x}_T$, and the other is from a matched normal sample, $\boldsymbol{x}_N$. Therefore, the haplotype matrix $\boldsymbol{Z}$ and the vector of haplotype frequencies $\boldsymbol{\pi}$ should be considered for each of tumor (T) and normal (N) tissue. Hence, the joint distribution is extended to

$$p(\mathcal{X}, \mathcal{Z}|\boldsymbol{\alpha}_N, \boldsymbol{\alpha}_T, \boldsymbol{\gamma}) = p(\boldsymbol{x}_N, \boldsymbol{Z}_N, \boldsymbol{\pi}_N, \varepsilon|\boldsymbol{\alpha}_N, \boldsymbol{\gamma})p(\boldsymbol{x}_T, \boldsymbol{Z}_T, \boldsymbol{\pi}_T, \varepsilon|\boldsymbol{\alpha}_T, \boldsymbol{\gamma})p(\varepsilon|\boldsymbol{\gamma}),$$

where $\mathcal{X} = (\boldsymbol{x}_N, \boldsymbol{x}_T)$ and $\mathcal{Z} = (\boldsymbol{Z}_N, \boldsymbol{Z}_T, \boldsymbol{\pi}_N, \boldsymbol{\pi}_T, \varepsilon)$. We can decompose $p(\boldsymbol{x}_N, \boldsymbol{Z}_N, \boldsymbol{\pi}_N, \varepsilon|\boldsymbol{\alpha}_N, \boldsymbol{\gamma})$ and $p(\boldsymbol{x}_T, \boldsymbol{Z}_T, \boldsymbol{\pi}_T, \varepsilon|\boldsymbol{\alpha}_T, \boldsymbol{\gamma})$ by the same way as Equation (1). A key concept in our method involves the modelings of $p(\boldsymbol{Z}|\boldsymbol{\pi}, \varepsilon)$ in tumor and normal tissues and is described in the following sections.

2.2 Modeling of sequencing data from a normal sample

First, we consider modeling of sequencing data of a normal sample. Because a normal sample contains two haplotypes, the frequencies of these haplotypes can be represented by one parameter π_N, and the third and fourth haplotypes can be regarded as the noise generated

from the first and second haplotypes, respectively. Therefore, for haplotype matrix $\boldsymbol{Z}_N$, $p(\boldsymbol{Z}_N|\pi_N, \varepsilon)$ is modeled by

$$p(\boldsymbol{Z}_N|\pi_N, \varepsilon) = \prod_{i=1}^{N_N} \{\pi_N(1-\varepsilon)\}^{z_{Ni1}} \cdot \{(1-\pi_N)(1-\varepsilon)\}^{z_{Ni2}} \cdot (\pi_N\varepsilon)^{z_{Ni3}} \cdot \{(1-\pi_N)\varepsilon\}^{z_{Ni4}},$$

where N_N is the number of normal reads. Because the haplotype frequency of a normal sample can be represented by a scalar parameter π_N, $p(\pi_N|\boldsymbol{\alpha}_N)$ yields the density of beta distribution.

2.3 Modeling of sequencing data from a tumor sample

As mentioned above, as we assumed that one of two haplotypes is mutated in the tumor sample, the fourth haplotype does not exist in the ideal situation. Let the first haplotype be the mutated haplotype. The fourth haplotype is then produced by the second haplotype with error rate ε; the haplotype frequencies of the second and fourth haplotypes are $\pi_{T2} \cdot (1-\varepsilon)$ and $\pi_{T2} \cdot \varepsilon$, respectively. Therefore, we obtain

$$p(\boldsymbol{Z}_T|\boldsymbol{\pi}_T, \varepsilon) = \prod_{i=1}^{N_T} \pi_{T1}^{z_{Ti1}} \cdot \{\pi_{T2}(1-\varepsilon)\}^{z_{Ti2}} \cdot \pi_{T3}^{z_{Ti3}} \cdot (\pi_{T2}\varepsilon)^{z_{Ti4}},$$

where N_T is the number of tumor reads and $\boldsymbol{\pi}_T = (\pi_{T1}, \pi_{T2}, \pi_{T3})'$ with $\pi_{T1} + \pi_{T2} + \pi_{T3} = 1$. The haplotype frequencies $\boldsymbol{\pi}_T = (\pi_{T1}, \pi_{T2}, \pi_{T3})'$ then follows multinomial distribution $p(\boldsymbol{\pi}_T|\boldsymbol{\alpha}_T)$, with parameter $\boldsymbol{\alpha}_T = (\alpha_{T1}, \alpha_{T2}, \alpha_{T3})'$.

2.4 BF for finding somatic mutations

Although we can fit the Bayesian latent Dirichlet model to the sequencing data of tumor and matched normal samples using some statistical methods, for evaluating the possibility of a variant being a true somatic mutation, we need to define a model that can be considered as a control against the model constructed under the premise of the existence of the mutation. In a Bayesian framework, we can compare the likelihoods of these two models by the BF given by

$$\mathrm{BF} = \frac{p_M(\boldsymbol{x}_N, \boldsymbol{x}_T|\boldsymbol{\alpha}_N, \boldsymbol{\alpha}_T, \boldsymbol{\gamma})}{p_E(\boldsymbol{x}_N, \boldsymbol{x}_T|\boldsymbol{\alpha}_N, \boldsymbol{\alpha}_T, \boldsymbol{\gamma})},$$

where p_M and p_E represent the models constructed under the premise of the existence of a mutation and without a mutation, respectively, and $p_M(\boldsymbol{x}_N, \boldsymbol{x}_T|\boldsymbol{\alpha}_N, \boldsymbol{\alpha}_T, \boldsymbol{\gamma})$ is the marginal likelihood defined by

$$p_M(\boldsymbol{x}_N, \boldsymbol{x}_T|\boldsymbol{\alpha}_N, \boldsymbol{\alpha}_T, \boldsymbol{\gamma}) = \sum_{\boldsymbol{Z}_N, \boldsymbol{Z}_T} \int p_M(\mathcal{X}, \mathcal{Z}|\boldsymbol{\alpha}_N, \boldsymbol{\alpha}_T, \boldsymbol{\gamma}) d\pi_N d\boldsymbol{\pi}_T d\varepsilon.$$

If the value of the BF is significantly large, we regard the candidate mutation as likely to be a true somatic mutation, with high probability.

As for the two models, p_M and p_E, p_M is the model already described in the previous sections, which we designate a mutation model. The latter model, p_E, can be constructed by replacing the haplotype frequencies in $p(\boldsymbol{Z}_T|\boldsymbol{\pi}_T, \varepsilon)$ with those in $p(\boldsymbol{Z}_T|\pi_N, \varepsilon)$, i.e. only two haplotypes ideally exist, and the third and fourth haplotypes are simply errors. We designate p_E as an error model. It is not an easy task to compute the marginal likelihood $p_M(\boldsymbol{x}_N, \boldsymbol{x}_T|\boldsymbol{\alpha}_N, \boldsymbol{\alpha}_T, \boldsymbol{\gamma})$ because it contains high-dimensional integrals and sums. To achieve efficient computation of the marginal likelihood, we derived a variational Bayes procedure for our proposed model.

2.5 Variational Bayes method for computing marginal likelihood

Suppose that $q(\mathcal{Z})$ is the variational distribution that yields the decomposition

$$q(\mathcal{Z}) = q(\boldsymbol{Z}_N)q(\boldsymbol{Z}_T)q(\pi_N)q(\boldsymbol{\pi}_T)q(\varepsilon). \quad (2)$$

Although the variational distribution should be prepared for each of the mutation and error models, we omit the index M or E for simplifying the representation. Using the variational distribution, we obtain a lower bound of the marginal likelihood

$$\log p(\mathcal{X}|\boldsymbol{\alpha}_N, \boldsymbol{\alpha}_T, \boldsymbol{\gamma}) \geq E_q\left[\log \frac{p(\mathcal{X}, \mathcal{Z}|\boldsymbol{\alpha}_N, \boldsymbol{\alpha}_T, \boldsymbol{\gamma})}{q(\mathcal{Z})}\right], \quad (3)$$

where E_q represents the expectation under the distribution q, as defined in Equation (2). In the variational Bayes framework, a functional of q in the right side of Equation (3) is called free energy and an iterative update of q, which maximizes the free energy achieved to an estimated posterior density of $\boldsymbol{Z}_N, \boldsymbol{Z}_T, \pi_N, \boldsymbol{\pi}_T$ and ε. The update formula of q in the right side of Equation (2) can be represented by

$$q^*(\boldsymbol{Z}_N) = \prod_{i=1}^{N_N} \prod_{j=1}^{4} (r^*_{Nij})^{z_{Nij}},$$

$$q^*(\boldsymbol{Z}_T) = \prod_{i=1}^{N_T} \prod_{j=1}^{4} (r^*_{Tij})^{z_{Tij}},$$

$$q^*(\pi_N) = p_{\mathrm{beta}}(\pi_N|\boldsymbol{\alpha}^*_N),$$

$$q^*(\boldsymbol{\pi}_T) = p_{\mathrm{Dir}}(\boldsymbol{\pi}_T|\boldsymbol{\alpha}^*_T),$$

$$q^*(\varepsilon) = p_{\mathrm{beta}}(\varepsilon|\boldsymbol{\gamma}^*),$$

where p_{beta} and p_{Dir} are the densities of beta and Dirichlet distributions, respectively, and the updated parameters can be obtained by

$$r^*_{Nij} = \frac{\rho^*_{Nij}}{\sum_{j'=1}^{4} \rho^*_{Nij'}},$$

$$r^*_{Tij} = \frac{\rho^*_{Tij}}{\sum_{j'=1}^{4} \rho^*_{Tij'}},$$

$$\alpha^*_{N1} = \sum_{i=1}^{N_N} (z_{Ni1} + z_{Ni3}) + \alpha_{N1}, \alpha^*_{N2} = \sum_{i=1}^{N_N} (z_{Ni2} + z_{Ni4}) + \alpha_{N2},$$

$$\alpha^*_{T1} = \sum_{i=1}^{N_T} z_{Ti1} + \alpha_{T1}, \alpha^*_{T2} = \sum_{i=1}^{N_T} (z_{Ti2} + z_{Ti4}) + \alpha_{T2},$$

$$\alpha^*_{T3} = \sum_{i=1}^{N_T} z_{Ti3} + \alpha_{T3},$$

$$\gamma^*_1 = \sum_{i=1}^{N_T} z_{Ti4} + \sum_{i'=1}^{N_N} (z_{Ni'3} + z_{Ni'4}) + \gamma_1,$$

$$\gamma^*_2 = \sum_{i=1}^{N_T} z_{Ti2} + \sum_{i'=1}^{N_N} (z_{Ni'1} + z_{Ni'2}) + \gamma_2,$$

with

$$\log \rho^*_{Ni1} = \log \frac{\alpha^*_{N1}}{\alpha^*_{N1}+\alpha^*_{N2}} + \log \frac{\gamma^*_2}{\gamma^*_1+\gamma^*_2} + \log p(x_{Ni}|\mathcal{H}_1),$$

$$\log \rho^*_{Ni2} = \log \frac{\alpha^*_{N2}}{\alpha^*_{N1}+\alpha^*_{N2}} + \log \frac{\gamma^*_2}{\gamma^*_1+\gamma^*_2} + \log p(x_{Ni}|\mathcal{H}_2),$$

$$\log \rho^*_{Ni3} = \log \frac{\alpha^*_{N1}}{\alpha^*_{N1}+\alpha^*_{N2}} + \log \frac{\gamma^*_1}{\gamma^*_1+\gamma^*_2} + \log p(x_{Ni}|\mathcal{H}_3),$$

$$\log \rho^*_{Ni4} = \log \frac{\alpha^*_{N2}}{\alpha^*_{N1}+\alpha^*_{N2}} + \log \frac{\gamma^*_1}{\gamma^*_1+\gamma^*_2} + \log p(x_{Ni}|\mathcal{H}_4),$$

$$\log \rho^*_{Ti1} = \log \frac{\alpha^*_{T1}}{\sum_{k=1}^{3}\alpha^*_{Tk}} + \log p(x_{Ti}|\mathcal{H}_1),$$

$$\log \rho^*_{Ti2} = \log \frac{\alpha^*_{T2}}{\sum_{k=1}^{3}\alpha^*_{Tk}} + \log \frac{\gamma^*_{N2}}{\gamma^*_{N1}+\gamma^*_{N2}} + \log p(x_{Ti}|\mathcal{H}_2),$$

$$\log \rho^*_{Ti3} = \log \frac{\alpha^*_{T3}}{\sum_{k=1}^{3}\alpha^*_{Tk}} + \log p(x_{Ti}|\mathcal{H}_3),$$

$$\log \rho^*_{Ti4} = \log \frac{\alpha^*_{T2}}{\sum_{k=1}^{3}\alpha^*_{Tk}} + \log \frac{\gamma^*_1}{\gamma^*_1+\gamma^*_2} + \log p(x_{Ti}|\mathcal{H}_4).$$

Here, $\mathcal{H}_j$ is the jth haplotype, and x_{Ni} and x_{Ti} are the ith read of normal and tumor samples, respectively.

We describe the derivation of the variational procedure in Supplementary Method S1.

2.6 Inferring local normal haplotypes when there exist multiple heterozygous germ line variants

When multiple heterozygous germ line variants are located nearby, the number of candidate haplotypes in the normal sample is two to the power of the number of the germ line variants. We use a haplotype assembly algorithm (He *et al.*, 2010) to infer the two normal local haplotypes from the set of possible haplotypes. The method constitutes a dynamic programming algorithm, which will find an optimal solution with respect to the minimum error correction (MEC) score. The MEC score is the number of conflicts between the reads and the two constructed haplotypes. The problem of minimizing the MEC score is Non-deterministic Polynomial-time hard (Cilibrasi *et al.*, 2005; Genovese *et al.*, 2008); however, the optimal algorithm is computationally practical for up to reads of length 15, where the length is the number of heterozygous germ line variants.

3 RESULTS

3.1 Sequence errors occur randomly on normal haplotypes in real data

We investigated the relationships between somatic mutations and heterozygous germ line variants nearby, using exome tumor and matched normal sequences of 10 ccRCC patients (Sato *et al.*, 2013; Shiraishi *et al.*, 2013). First, we focused on orthogonally validated somatic mutations with a heterozygous germ line single nucleotide polymorphism (SNP) nearby. The somatic mutations were detected based on tumor and matched normal exome sequences, and then validated by using RNAseq and whole genome sequence data. For each true somatic mutation, we classified the reads, which overlap both the somatic mutation position and the heterozygous SNP position, into four types: (i) RR, reads that do not support either the somatic mutation or the heterozygous SNP; (ii) VR, reads that support the somatic mutation without the heterozygous SNP; (iii) RV, reads that support the heterozygous SNP without the somatic mutation; and (iv) VV, reads that support both the somatic mutation and the heterozygous SNP. As a heterozygous germ line SNP was located nearby, we observed both RR and RV reads for each somatic mutation. However, we mostly observed either VR or VV reads (Fig. 2B). This experiment indicated that somatic mutations occur on one of normal haplotypes. Second, we investigated the behaviors of sequence errors. We collected mismatches in error loci (Shiraishi *et al.*, 2013) using the same exome data as above, and excluded potential mapping errors by using human chained self-alignments (Chiaromonte *et al.*, 2002; Kent *et al.*, 2003; Schwartz *et al.*, 2003), simple tandem repeats (Benson, 1999), dbSNP 138 (Sherry *et al.*, 2001), BLAT aligner (Kent, 2002) with human decoy sequences (37d5) (Genovese *et al.*, 2013) and indels located nearby (see Supplementary Method S2). For each probable sequence error position, we classified the reads, which overlap the sequence error position and the heterozygous SNP position, into four types (RR, VR, RV and VV). We observed both VR and VV reads simultaneously in most cases (Fig. 2B), in contrast to the observations made for the somatic mutations. This observation indicates that sequence errors occur randomly on normal haplotypes.

3.2 Simulation Study

We tested HapMuC using synthetic data. In this simulation, we used several read lengths (100, 150, 300 and 1000 bp) and compared the results between these as well as single-end reads versus paired-end reads.

We generated positive and negative examples as follows:

(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, according to an empirical distribution (Supplementary Fig. S1), as well as two haplotypes (h3 and h4)

(4) Setting parameters were as follows: tumor purity was set at 20% for positive examples and 0% for negative examples. Note that a tumor purity of 20% leads to an average allele frequency of 10%.

(5) For 50 repetitions:

- Select a source haplotype randomly, according to the multinomial distribution (haplotype frequencies), given the parameters
- Select a start point of a read randomly such that the read overlaps the candidate position
- Select a strand direction randomly, and generate a paired-read with 1% sequence errors and an insertion size given by a normal distribution ($\mu = 3 \times l$, $\sigma = l/2$), where l is the read length, by referring to an empirical distribution (Supplementary Fig. S2).

Note that we used only those examples that passed the minimum criteria for allele frequencies, read coverage, the number of variant-supporting reads and strand bias (Supplementary Method S3). Note also that the read depth is 50×. As a counterpart method, we prepared a simple Fisher's exact test-based method, which uses a 2 × 2 base counts table.

We calculated the area under curve (AUC) values (Bradley, 1997) for each setting. We summarize these results in Table 1. The AUC values of the Fisher method and BFs without SNPs are comparable; however, BFs that used heterozygous germ line variants outperformed these. In addition, the number of positions, for which heterozygous germ line variants could be used, increases as the read length increases.

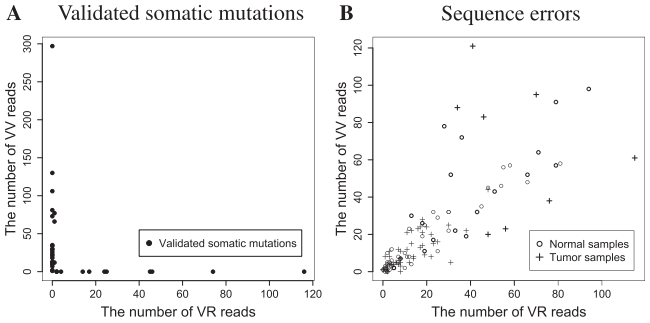

Fig. 2. Relationships between candidate mutations and nearby heterozygous germ line variants from 10 ccRCC patients' sequence data. For each candidate mutation, we counted the reads that cover the candidate mutation and the heterozygous germ line variant (VV), and the reads that cover the candidate mutation without the heterozygous variant (VR). (**A**) For the orthogonally validated somatic mutations, we mostly observed the reads of a single haplotype (VV or VR). (**B**) However, for sequence errors, we observed the reads of both haplotypes (VV and VR) randomly

3.3 Real data

We evaluated our new algorithm, HapMuC, using the whole-genome sequence data of breast cancer cell lines (HCC1954 and HCC1143), which are publicly available as a part of the TCGA Mutation Calling Benchmark 4 datasets (https://cghub.ucsc.edu/datasets/benchmark_download.html). We used the tumor samples (computationally mixed with normal) and the matched pure normal samples as evaluation datasets, whose average read coverage was ∼30× (Supplementary Table S1). We first detected the candidate positions that passed minimum criteria in the evaluation dataset. Then, we assumed that candidate positions were true positives if, and only if, the positions were also called on the corresponding pure dataset with an average coverage of 60×. The criteria are shown in later sections. The workflow of HapMuC is given in Supplementary Figure S3.

3.3.1 Superior performance using nearby heterozygous germ line SNPs We used the 60% tumor dataset of HCC1954 for evaluating the effect of using heterozygous germ line variants located near candidate mutations. The criteria are shown in Supplementary Method S4 and S5 for the evaluation dataset and the pure dataset, respectively. We first restricted the candidate positions so that heterozygous SNPs were located within a 100 bp distance. Second, we excluded potential mapping errors using human chained self-alignments, simple tandem repeats, dbSNP 138, BLAT aligner with human decoy sequences (37d5), and indels located nearby, in a manner similar to that described in Section 3.1 (Supplementary Method S6). We then confirmed the advantage of using the information of heterozygous germ line variants by comparing the results of HapMuC and a version of HapMuC that does not use the information of heterozygous germ line variants (Fig. 3A). By drawing ROC curves (Fawcett, 2006; Robin *et al.*, 2011), the AUC value of HapMuC was 0.937, and that of the counterpart was 0.898 (Fig. 3B). The difference in AUCs was statistically significant (P-value = 0.00124). Furthermore, the version of HapMuC

Table 1. Results from the simulation study

Read length	Distance	Number of candidate SNVs (Paired)	Number of candidate SNVs (Single)	HapMuC (Paired)	HapMuC (Single)	HapMuC without SNP	Fisher
100	0–100	635	635	0.8077	0.8032	0.5892	0.578
	100–	542	0	0.7614	0.6706	0.6706	0.6177
	None	5920	6462	0.6255	0.6246	0.6246	0.6262
150	0–150	939	939	0.8254	0.8147	0.5985	0.587
	150–	693	0	0.6976	0.6082	0.6082	0.6151
	None	5407	6100	0.6402	0.6393	0.6393	0.6052
300	0–300	1714	1714	0.8501	0.8457	0.6414	0.6222
	300–	778	0	0.6779	0.6046	0.6046	0.5846
	None	4916	5964	0.6285	0.6275	0.6275	0.6246
1000	0–1000	3461	3461	0.8628	0.8609	0.6302	0.6106
	1000–	762	0	0.6881	0.6092	0.6092	0.6048
	None	3280	4042	0.6432	0.6423	0.6423	0.6198

Notes: The AUC values of HapMuC of paired-end reads and single-reads, HapMuC, which does not use heterozygous germ line variants, and the Fisher method, were compared. We also tested the methods on several read lengths (100, 150, 300 and 1000 bp). In each read length, we grouped the candidate positions by the distances between candidate mutations and closest heterozygous germ line variants, which HapMuC used ('Distance' column). 'None' means that HapMuC could not use any heterozygous germ line variants.

that does not use heterozygous variants demonstrated superior performance compared with the simple Fisher method, suggesting that HapMuC can use information other than neighboring heterozygous germ line variants.

3.3.2 Alternative methods To compare the performance of our approach in terms of mutation detection, we also ran VarScan 2, SomaticSniper, Strelka and MuTect on the same datasets. VarScan 2 is a Fisher's exact test-based method with an indel filter and a triallelic site filter, and SomaticSniper, Strelka and MuTect are Bayesian modeling-based methods. We summarized the parameters that we used for these alternative methods in Supplementary Method S7.

3.3.3 Performance with whole-genome data We used the 60% tumor datasets and the 40% tumor datasets of HCC1954 and HCC1143 to benchmark the performance of mutation detection in whole genome. For this analysis, we used the criteria in Supplementary Method S8 and S9 for the evaluation datasets and the pure datasets, respectively. We summarize the results in Table 2. Overall, the AUC values of our HapMuC algorithm were the highest under all conditions, reflecting superior sensitivity and specificity. The AUC values of the original HapMuC were slightly higher than that of a version of HapMuC that does not use the information of heterozygous germ line variants. In this analysis, HapMuC detected heterozygous germ line variants nearby in ~15% of all the candidate positions. In addition, we show the ROC curves on HCC1954/n40t60 (Fig. 4). The runtime of HapMuC for HCC1954/n40t60 was <4 h, using 48 computing nodes. The maximum memory usage of HapMuC was 0.3 GB. We report the runtime and memory usage of the other mutation-calling programs and HapMuC in Supplementary Table S2.

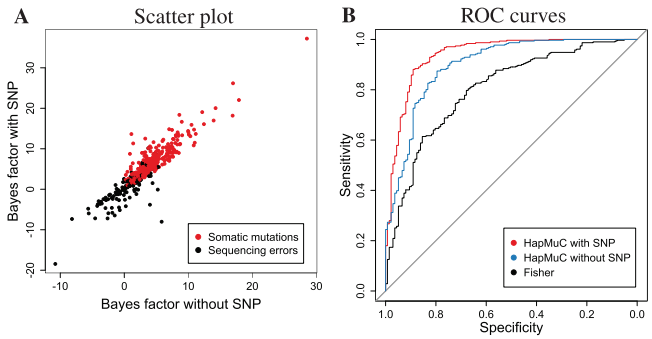

Fig. 3. Results obtained with the HCC1954/n40t60 dataset (TCGA Mutation Calling Benchmark 4) in cases where heterozygous germ line SNPs are present near candidate mutations. In (**A**), a scatter plot and (**B**) ROC curves, the result obtained with HapMuC is compared with that of a version of HapMuC, which does not use the information of neighboring heterozygous germ line SNPs

3.3.4 Most mutations called only by the Fisher method were false positives We focused on single nucleotide variants (SNVs) of the HCC1954/n60t40 dataset to investigate the reasons behind HapMuC's superiority other than using heterozygous germ line variants. We excluded positions that were within a 10 000 bp distance from gap locations, in addition to the criteria in Supplementary Method S4 and S5. When we set the *P*-value threshold of 0.05 for the Fisher method, the specificity was 0.744 and the sensitivity was 0.719. The BF threshold of 5 for HapMuC resulted in a similar specificity (0.746) and a high sensitivity (0.974). Based on these thresholds, most of the positions called only by the simple Fisher method were found to be false positives (Table 3).

3.3.5 Sequence errors caused by homopolymers We first focused on homopolymers near candidate mutations. We measured the length of homopolymers that cover candidate mutations; the stacked histograms (Supplementary Fig. S4) showed that the false positives generated by the Fisher method involved longer homopolymers when compared with HapMuC. The average homopolymer length of the positions that were called only by Fisher was 8.41 and that only by HapMuC was 4.79. This difference was statistically significant (*t*-test $P < 2.2e\text{-}16$). This is because the profile HMM realignment algorithm in HapMuC takes into account the fact that NGS mechanisms tend to increase error rates in homopolymers (Albers *et al.*, 2011). Thus, mismatches in homopolymers have smaller weights in our haplotype inference and BF. In Supplementary Fig. S5, we show an

Table 2. Results obtained with whole-genome data of the TCGA Mutation Calling Benchmark 4 datasets

Type	Sample	Number of candidates	Number of with heterozygous germ line variants	HapMuC	HapMuC without SNP	VarScan2	Strelka	MuTect	SomaticSniper
SNVs	HCC1954/n40t60	6153	936	0.8145	0.8088	0.7192	0.7878	0.7638	0.713
	HCC1954/n60t40	4442	693	0.9333	0.9327	0.8286	0.9236	0.9203	0.7291
	HCC1143/n40t60	7080	995	0.864	0.8578	0.7764	0.8397	0.8444	0.7108
	HCC1143/n60t40	4451	735	0.9601	0.9594	0.8566	0.9406	0.946	0.7013
Indels	HCC1954/n40t60	762	100	0.7433	0.7412	0.6145	0.7261	–	–
	HCC1954/n60t40	437	67	0.729	0.7303	0.6184	0.72	–	–
	HCC1143/n40t60	349	45	0.72	0.7135	0.6181	0.7132	–	–
	HCC1143/n60t40	243	39	0.7524	0.7457	0.6833	0.7381	–	–

Notes: The number of candidate positions (Number of candidates), the number of candidate positions where HapMuC used heterozygous germ line variants (Number of candidates with heterozygous germ line variants) and the AUC values of HapMuC, HapMuC that does not use heterozygous germ line variants, VarScan 2, Strelka, MuTect and SomaticSniper. The candidate positions are positions that passed the minimum criteria (Supplementary Method S8).

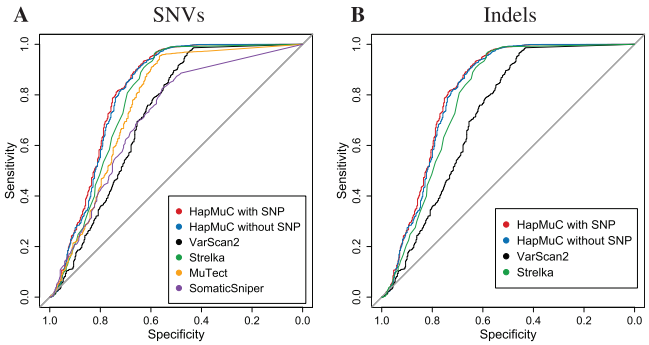

Fig. 4. Benchmarking mutation-detection methods for the whole genome using the HCC1954/n60t40 dataset (TCGA Mutation Calling Benchmark 4). The ROC curves of HapMuC (red), HapMuC, which does not use heterozygous germ line variants (blue), VarScan 2 (black), Strelka (orange), MuTect (Purple) and SomaticSniper (green) in (**A**) SNVs and (**B**) Indels

Table 3. The number of somatic SNVs called by HapMuC and the simple Fisher method in the whole genome from the HCC1954/n40t60 dataset (TCGA Mutation Calling Benchmark 4)

Type	Both	Only by HapMuC	Only by Fisher
True positive	2461	897	15
False positive	760	945	959

Notes: The threshold for HapMuC BF was 5, and that for the simple Fisher method was 0.05. Note that we excluded positions within a 10 000 bp distance from gap locations.

example of false positive calls of the Fisher method in a long homopolymer.

3.3.6 Dealing with reads with several mismatches We also investigated the number of mismatches within a 100 bp distance from candidate mutations. The stacked histograms (Supplementary Fig. S6) showed a trend that candidate mutations accompanying several mismatches might be false positives. The average number of mismatches near the candidate mutations that were called only by the Fisher method was 4.82, and that only by HapMuC was 1.85 (*t*-test $P < 2.2e\text{-}16$). This is because the reads, which contain several mismatches with relatively high base qualities, are judged not to be mapped to any of the prepared haplotypes, but are mapped to different locations in the genome during the calculation of alignment probabilities. Thus, those reads are ignored in our haplotype inference and have no impact on the BF. In Supplementary Figure S7, we show a representative false positive call by the Fisher method.

3.4 Estimation of sensitivity using real data

We used the COLO-829, malignant melanoma cell line, and COLO-829BL, a lymphoblastoid line, to evaluate the sensitivity of our method. The data were downloaded from the European Genome-Phenome Archive (accession number EGAS00000000052), and we used the list of validated somatic SNVs ($n = 497$) and validated somatic indels ($n = 66$) (Pleasance *et al.*, 2010). These variants were converted to hg19 from their positions on the hg18 reference using the University of California Santa Cruz (UCSC) Genome Browser liftover utility. We called 494 SNVs, with a sensitivity of 99.4%, and 32 indels, with a sensitivity of 48.5%. In both cases, the sensitivity was comparable with or higher than that of the counterpart methods (Supplementary Method S10).

4 DISCUSSION

In this article, we addressed the problem of accurately detecting somatic mutations in cancer by using tumor and matched normal NGS sequences. Based on 10 ccRCC patients' data, we observed that true somatic mutations mostly occurred on one of two normal haplotypes; however, sequencing errors occurred randomly on normal haplotypes. Based on this observation, we developed a novel Bayesian statistical method, HapMuC, which can use the information of heterozygous germ line variants located near candidate mutations.

In our simulation study, we demonstrated that our algorithm has superior performance compared with a simple Fisher method when a candidate mutation occurs near a heterozygous germ line variant. We also tested our algorithm on several read lengths (100, 150, 300 and 1000 bp) as well as single-end reads and paired-end reads; HapMuC achieved higher AUC values with longer reads and paired reads by using additional information. For calculating the likelihood of read pairs, we approximated it by multiplying the alignment likelihoods of the first and second read, which assumes that read pairs are generated independently given the haplotype. However, we should modify our modeling for read pairs, considering their insertion size distribution, as a read-pair usually comes from a fragment.

We also confirmed the advantages of using heterozygous germ line variants based on breast cancer cell line samples, which are publicly available as a part of the TCGA Mutation Calling Benchmark 4 datasets. In this experiment, we excluded potential alignment artifacts by using some genomic annotations, a BLAT filter with human decoy sequence (37d5) and an indel filter because mapping errors can confuse the haplotype inference of our algorithm. However, detecting and eliminating alignment errors appropriately remains a problem.

Additionally, we evaluated our algorithm in the context of a whole genome using the TCGA Mutation Calling Benchmark 4 datasets. We compared the results of HapMuC with those of the simple Fisher method, Bayesian methods, which includes Strelka, MuTect and SomaticSniper, and VarScan 2, which is based on Fisher's exact test but which uses some filters. For both SNVs and indels, our method shows higher AUC values than the alternative methods. During subsequent manual inspection using Integrative Genomics Viewer (IGV) (Thorvaldsdóttir *et al.*, 2013), we confirmed that HapMuC succeeded in using contextual information, such as mismatches and homopolymers, near candidate mutations. Although several filtering approaches have been proposed, to date, no somatic mutation caller that integrates the contextual information into its score has been available. Furthermore, the analysis using the COLO-829 dataset with the list of validated somatic mutations (Pleasance *et al.*, 2010)

showed that our algorithm has sensitivity better than or comparable with that of the existing methods.

For future studies, we intend to incorporate further information into our algorithm. First, sequence strand bias, which is suggested as a false-positive filter by SomaticSniper, VarScan 2, MuTect and so on, could be considered. We can integrate the information of strand bias into HapMuC's score by extending our generative model. Second, we may consider the distance to the effective 3′end, which is suggested as an additional filter by SomaticSniper. We can achieve this by modifying the profile HMM alignment algorithm to consider the distances from the edges of reads. In addition, when sequencing data of other samples are available, we can infer background error rates and use these as prior information. A recent statistical approach, called EBCall (Shiraishi *et al.*, 2013), has succeeded in achieving outstanding performance using pooled normal samples. MuTect also uses a simple false-positive filter, which checks whether candidate mutations exist in other normal samples. In terms of statistical modeling, if we could modify the mutation model and the error model as a nested model-pair, we can use a variable selection approach, such as lasso, for detecting somatic mutations.

ACKNOWLEDGEMENTS

The super-computing resource was provided by Human Genome Center, the Institute of Medical Science, the University of Tokyo.

Funding: This work was partly supported by Grant-in-Aid for Scientific Research on Innovative Areas (22134004).

Conflict of interest: none declared.

REFERENCES

Albers,C.A. *et al.* (2011) Dindel: accurate indel calls from short-read data. *Genome Res.*, **21**, 961–973.

Beal,M.J. (2003) Variational Algorithms for Approximate Bayesian Inference. PhD thesis, The Gatsby Computational Neuroscience Unit, University College London, London, UK.

Benson,G. (1999) Tandem repeats finder: a program to analyze DNA sequences. *Nucleic Acids Res.*, **27**, 573.

Blei,D.M. *et al.* (2003) Latent dirichlet allocation. *J. Mach. Learn. Res.*, **3**, 993–1022.

Bradley,A.P. (1997) The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern Recognit.*, **30**, 1145–1159.

Chiaromonte,F. *et al.* (2002) Scoring pairwise genomic sequence alignments. In: *Pacific Symposium on Biocomputing*. Vol. 7, pp. 115–126.

Cibulskis,K. *et al.* (2013) Sensitive detection of somatic point mutations in impure and heterogeneous cancer samples. *Nat. Biotechnol.*, **31**, 213–219.

Cilibrasi,R. *et al.* (2005) On the complexity of several haplotyping problems. In: *Algorithms in Bioinformatics*. Springer, pp. 128–139.

Ding,L. *et al.* (2012) Clonal evolution in relapsed acute myeloid leukaemia revealed by whole-genome sequencing. *Nature*, **481**, 506–510.

Eddy,S.R. (1998) Profile hidden Markov models. *Bioinformatics*, **14**, 755–763.

Fawcett,T. (2006) An introduction to ROC analysis. *Pattern Recognit. Lett.*, **27**, 861–874.

Forbes,S.A. *et al.* (2011) COSMIC: mining complete cancer genomes in the Catalogue of Somatic Mutations in Cancer. *Nucleic Acids Res.*, **39**, D945–D950.

Genovese,G. *et al.* (2013) Mapping the human reference genomes missing sequence by three-way admixture in latino genomes. *Am. J. Hum. Genet.*, **93**, 411–421.

Genovese,L.M. *et al.* (2008) Speedhap: an accurate heuristic for the single individual SNP haplotyping problem with many gaps, high reading error rate and low coverage. *IEEE/ACM Trans. Comput. Biol. Bioinform.*, **5**, 492–502.

Goya,R. *et al.* (2010) SNVMix: predicting single nucleotide variants from next-generation sequencing of tumors. *Bioinformatics*, **26**, 730–736.

He,D. *et al.* (2010) Optimal algorithms for haplotype assembly from whole-genome sequence data. *Bioinformatics*, **26**, i183–i190.

Kent,W.J. (2002) BLAT-the BLAST-like alignment tool. *Genome Res.*, **12**, 656–664.

Kent,W.J. *et al.* (2003) Evolution's cauldron: duplication, deletion, and rearrangement in the mouse and human genomes. *Proc. Natl Acad. Sci. USA*, **100**, 11484–11489.

Koboldt,D.C. *et al.* (2012) VarScan 2: somatic mutation and copy number alteration discovery in cancer by exome sequencing. *Genome Res.*, **22**, 568–576.

Larson,D.E. *et al.* (2012) SomaticSniper: identification of somatic point mutations in whole genome sequencing data. *Bioinformatics*, **28**, 311–317.

Ley,T.J. *et al.* (2008) DNA sequencing of a cytogenetically normal acute myeloid leukaemia genome. *Nature*, **456**, 66–72.

Meyerson,M. *et al.* (2010) Advances in understanding cancer genomes through second-generation sequencing. *Nat. Rev. Genet.*, **11**, 685–696.

Nik-Zainal,S. *et al.* (2012) The life history of 21 breast cancers. *Cell*, **149**, 994–1007.

Pleasance,E.D. *et al.* (2010) A comprehensive catalogue of somatic mutations from a human cancer genome. *Nature*, **463**, 191–196.

Robin,X. *et al.* (2011) pROC: an open-source package for R and S+ to analyze and compare ROC curves. *BMC Bioinformatics*, **12**, 77.

Roth,A. *et al.* (2012) JointSNVMix: a probabilistic model for accurate detection of somatic mutations in normal/tumour paired next-generation sequencing data. *Bioinformatics*, **28**, 907–913.

Sato,Y. *et al.* (2013) Integrated molecular analysis of clear-cell renal cell carcinoma. *Nat. Genet.*, **45**, 860–867.

Saunders,C.T. *et al.* (2012) Strelka: accurate somatic small-variant calling from sequenced tumor-normal sample pairs. *Bioinformatics*, **28**, 1811–1817.

Schwartz,S. *et al.* (2003) Human-mouse alignments with BLASTZ. *Genome Res.*, **13**, 103–107.

Shah,S.P. *et al.* (2009) Mutational evolution in a lobular breast tumour profiled at single nucleotide resolution. *Nature*, **461**, 809–813.

Sherry,S.T. *et al.* (2001) dbSNP: the NCBI database of genetic variation. *Nucleic Acids Res.*, **29**, 308–311.

Shiraishi,Y. *et al.* (2013) An empirical Bayesian framework for somatic mutation detection from cancer genome sequencing data. *Nucleic Acids Res.*, **41**, e89–e89.

Thorvaldsdóttir,H. *et al.* (2013) Integrative Genomics Viewer (IGV): high-performance genomics data visualization and exploration. *Brief. Bioinform.*, **14**, 178–192.

Yoshida,K. *et al.* (2011) Frequent pathway mutations of splicing machinery in myelodysplasia. *Nature*, **478**, 64–69.