A unified haplotype-based method for accurate and comprehensive variant calling

Paper Presentation by Chuanyi Zhang

Daniel P Cooke$^{1,2}$, David C Wedge$^{2}$, and Gerton Lunter$^{1}$

$^1$Wellcome Centre for Human Genetics, University of Oxford, Oxford, UK
$^2$Big Data Institute, University of Oxford, Oxford, UK

March 7, 2019
# Table of Contents

1 Motivation & Background
   Variant Calling
   Somatic Variant Calling

2 Method
   Cancer Prior
   Subclone genotype model
   Cancer calling model

3 Result
Motivation & Background

Variant Calling

- Ideal scenario: enough read depth
- (1) read processing, (2) mapping, (3) calling
- haplotype analysis: HaplotypeCaller in Genome Analysis Toolkit (GATK)
Somatic Variant Calling

Differences from germline calling

- Allele frequency assumption: purity, multiple subclones, CNA
- Low VAF vs. Artifacts
- Somatic vs. Germline: matched tumor-normal sample
Table of Contents

1 Motivation & Background
   Variant Calling
   Somatic Variant Calling

2 Method
   Cancer Prior
   Subclone genotype model
   Cancer calling model

3 Result
To assess germline calling accuracy, we called variants in three well-characterized Genome in a Bottle (GIAB) samples: HG001 (NA12878), HG002 (NA24385), and HG005 (NA24631), in addition to the synthetic-diploid (Syndip) sample CHM1-CHM13 (NA24385), in addition to barcoding libraries prepared using the 10X Genomics Chromium protocol. We tested several HG001 and HG002 replicates, including two libraries, using an approach orthogonal to that used for the GIAB genetics guidelines - degraded performance, so we only used QUAL and GQ for filtering GATK4 calls. All calls were evaluated with RTG3 metrics guidelines - degraded performance, so we only used QUAL and GQ. Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similarly, we found that using GATK4-like quality (QUAL and GQ). Similar...
Haplotype generating

- build from candidate alleles
- haplotype tree
- prune, stages: (1) pre: haplotype likelihood, (2) post: haplotype posterior
Genotype Prior Models

1. Uniform
2. Hardy-Weinberg-Equilibrium (HWE)
3. Coalescent-HWE
4. Trio
5. ★ Cancer

For ploidy $m$, genotypes: $g = (h_1, \ldots, h_m)$;
for $n$ populations inside a tumor, joint genotypes: $g = (g_1, \ldots, g_n)$. 
Cancer

\[ g_{\text{cancer}} = (g_{\text{germ}}, g_{\text{som}}) \]

\[ p(g_{\text{cancer}} \mid M_{\text{cancer}}) = p(g_{\text{germ}} \mid M_{\text{germ}}) p(g_{\text{som}} \mid g_{\text{germ}}, M_{\text{som}}), \]

\[ M_{\text{germ}} \] can be Coalescent-HWE prior model, if there’s only 1 somatic haplotype

\[ p(g_{\text{som}} \mid g_{\text{germ}}, M_{\text{som}}) = \frac{1}{|g_{\text{germ}}|} \sum_{i=1}^{|g_{\text{germ}}|} p(g_{\text{som}} \mid g_{\text{germ}, i}, M_{\text{som}}) \]

if multi somatic haplotypes: (assume all haplotypes originate from germline, independently)

\[ p(g_{\text{som}} \mid g_{\text{germ}}, M_{\text{som}}) = \prod_{j=1}^{|g_{\text{som}}|} p(g_{\text{som}, j} \mid g_{\text{germ}}) = \prod_{j=1}^{|g_{\text{som}}|} \frac{1}{|g_{\text{germ}, j}|} \sum_{i=1}^{|g_{\text{germ}, j}|} p(g_{\text{som}} \mid g_{\text{germ}, j, i}, M_{\text{som}}) \]
We want to know joint posterior distribution

\[
p(g, \pi \mid \mathcal{R}, \alpha, M_g) = \frac{p(\pi, g, \alpha, M_g, \mathcal{R})}{p(\alpha, M_g, \mathcal{R})}
\]

\[
= \frac{p(\mathcal{R} \mid \pi, g)p(\pi \mid \alpha)p(g \mid M_g)}{p(\alpha, M_g, \mathcal{R})}
\]

\[
\propto p(g \mid M_g) \prod_{s=1}^{S} p(\mathcal{R}_s \mid \pi_s, g)p(\pi_s \mid \alpha_s)
\]

\[
= p(g \mid M_g) \prod_{s=1}^{S} \int p(\mathcal{R}_s \mid \phi_s, g)p(\phi_s \mid \alpha_s) \, d\phi_s
\]

\[
= p(g \mid M_g) \prod_{s=1}^{S} \int \prod_{i=1}^{\mid g \mid} \phi_{si} p(r \mid h_i)p(\phi_s \mid \alpha_s) \, d\phi_s
\]
Problem of computing

- This posterior is intractable. Since \( \phi_s \sim \text{Dir}(\alpha_s) \) So the integration over \( \phi_s \) is intractable. \( \phi \) is latent variables.
- Using **Variational Bayes** (VB)

  Approximate \( p^*(x) \triangleq p(x \mid D) \) (intractable posterior) with \( q(x) \). Maximize \( L(q) \triangleq -D_{KL}(q \parallel \tilde{p}) \) (not \( D_{KL}(p^* \parallel p) \)), where \( \tilde{p} = p(x, D) = p^*(x)p(D) \)

\[
L(q) = -\mathbb{E}_q \left[ \log \frac{q}{\tilde{p}} \right] = - \int q(x) \log \frac{q(x)}{p^*(x)p(D)} \, d\mu(x) \\
= - \int q(x) \log \frac{q(x)}{p^*(x)} - q(x) \log p(D) \, d\mu(x) \\
= -D_{KL}(q \parallel p^*) + \log p(D) \\
\leq \log p(D)
\]
\( L(q) \) is evidence lower bound (ELBO). Maximizer is \( q = p^* \).
Bayes:

\[ p(x \mid \mathcal{D}) = \frac{p(\mathcal{D} \mid x)p(x)}{p(\mathcal{D})} \]

\[ \text{post} = \frac{\text{likelihood} \cdot \text{prior}}{\text{evidence}} \]

And by assuming this factorization

\[ q(g, Z, \phi) = q(g) \prod_{s=1}^{S} q(Z_s)q(\phi_s) \]

where we introduce the latent binary matrix \( Z_s \), \( q(Z_{snk}) \) are so-called *responsibilities* of assuming haplotype \( k \) for read \( n \) in sample \( s \). By this factorization (mean field) we can optimize on them alternately. Moreover, if we assume these priors are Dirichlet, then prior and posterior are *conjugated*. Categorical (likelihood) and Dirichlet are conjugate distributions.
Assume 3 possible cases:

1. No somatic mutations, clean germline, the individual model with any germline prior (merge) $M_{ind}$

2. Copy number changes, but no somatic, the *subclone model* with germline prior (e.g. Coalescent-HWE) $M_{ind}$

3. Somatic occurs, possible CNA, the *subclone model* with cancer genotype prior.
Calling Model

Germline genotype posterior

\[ p(g \mid \mathcal{R}) = \sum_x p(g, \mathcal{M}_x \mid \mathcal{R}) \]

\[ = \sum_x p(g \mid \mathcal{M}_x, \mathcal{R}) p(\mathcal{M}_x \mid \mathcal{R}) \]

\[ = p(g, \mid \mathcal{M}_{ind}) p(\mathcal{M}_{ind} \mid \mathcal{R}) \]

\[ + p(g, \mid \mathcal{M}_{CNV}) p(\mathcal{M}_{CNV} \mid \mathcal{R}) \]

\[ + p(g, \mid \mathcal{M}_{somatic}) p(\mathcal{M}_{somatic} \mid \mathcal{R}) \]

where \( p(\mathcal{M}_x \mid \mathcal{R}) = p(\mathcal{M}_x)p(\mathcal{R} \mid \mathcal{M}_x) \), and \( p(\mathcal{R} \mid \mathcal{M}_x) \) is the “evidence”; and \( p(g \mid \mathcal{M}_{somatic}) = \sum_{\tilde{g} : \tilde{g} \in g} p(\tilde{g} \mid \mathcal{M}_{somatic}) \), \( \tilde{g} = (g_{\text{germ}}, g_{\text{som}}) \), from cancer prior

Germline allele posterior \( p(a \mid \mathcal{R}) = \sum_{g : a \in g} p(g \mid \mathcal{R}) \).
Credible somatic mass

\[ p_{somatic}(a \mid R) \leftarrow \sum ? p(\tilde{g} \mid M_{somatic}, \text{credible}) \]

There are \( K \) somatic haplotypes, then the \textit{credible somatic frequencies} satisfy

\[
p(\phi_{sk} > \tau \mid M_{somatic}) = \int_\tau^1 \text{Beta} \left( \theta ; \alpha_{P+1}, \sum_{i=1}^{P} \alpha_i \right) d\theta
\]

where \( \phi_{sk} \sim \text{Beta}(\alpha_k, \sum \alpha - \alpha_k) \) since \( \phi_s \sim \text{Dir}(\alpha_s) \) i.e. \( p(\phi_s) = \frac{1}{B(\alpha)} \prod_{k=1}^{K} \phi_{sk}^{\alpha_k-1} \).

The \textit{credible somatic mass} is

\[
\lambda_s = 1 - \prod_k 1 - p(\phi_{sk} > \tau \mid M_{somatic})
\]

means the probability mass of \( \exists \) at least 1 credible in \( K \) somatic haplotypes.
Calling allele

Then $\lambda = 1 - \prod_s \lambda_s$. 

$$\prod \exists_1 \land \cdots \land \exists_S = \#_1 \lor \cdots \lor \#_S$$

$$p_{somatic}(a \mid \mathcal{R}) = \lambda \left( 1 - \prod_s \sum_a 1_{\{a \notin \text{germ} \land a \in \text{som}\}} p(\tilde{g} \mid \mathcal{R}, M_{somatic}) \right)$$

Might be a typo?
# Table of Contents

1. Motivation & Background
   - Variant Calling
   - Somatic Variant Calling

2. Method
   - Cancer Prior
   - Subclone genotype model
   - Cancer calling model

3. Result
Synthetic Tumors

Evaluation of somatic mutation calling is challenging

- Real tumor with manual inspected mutations
- Mix reads from unrelated individuals
- ★ Spike mutations directly into raw sequencing reads from healthy tissue
Synthetic Tumors

1. Select sample with known germline

2. Assign reads to germline haplotypes

3. Realign reads to germline haplotypes

4. Sample PCAWG tumour-specific calls

5. Spike PCAWG mutations onto reads

6. Remap spiked reads

- Reads from NA12878 ~ 300× → \{30×, 35×, 60×, 65×\}
- Assign and realign to make sure spiked mutations fall on same haplotype, and position consistent
- Sample mutations from pan-cancer analysis of whole genomes (PCAWG) uniformly

Produce raw unmapped reads (FASTQ) files.
Somatic Mutations Calling Accuracy

1. Precision-Recall curve: top-right is optimal
2. Recalls for each VAF, using PASS variants
3. Most differences in recall is due to low frequencies

- Octopus
- Mutect2
- Strelka2
- Lancet
- LoFreq
- VarDict
- Platypus

**Figure 4**

- **a** Precision-recall curves. Scoring metrics used to generate curves were RFQUAL (Octopus), TLOD (Mutect2), SomaticEVS (Strelka2), QUAL (Lancet), QUAL (LoFreq), SSF (VarDict), and QUAL (Platypus). Only PASS calls are used. VarDict is not visible as it is outside the axis limits due to low precision. Precisions on the two tests are substantially different as the skin set has almost 50 times as many true mutations as the breast set. Dots on the Octopus curve are placed at RFQUAL 7 (3 is used for the entire curve).

- **b** Recalls for each Variant Allele Frequency (VAF) using PASS variants. Points show true spike-in VAFs. The approximate depth for the synthetic skin and breast tumours were 60x and 65x, and 30x and 35x for their normal pairs, respectively. All comparisons to the synthetic tumour truth sets were performed using RTG Tools vcfeval. Spectral differences in recall, particularly between the best performing tools (Octopus, Strelka2, Mutect2, Lancet), were due to low frequency mutations (Fig. 4b and Supplementary Fig. 5). Sensitivity for mutations below 2.5% is poor for all callers (0.01 for Octopus and 0.002 for Mutect2). At 60x sequencing depth, a 2.5% VAF corresponds to an expectation of less than two observations. However, Octopus had considerably better sensitivity for mutations with VAFs between 4.5-10% (3-5 expected observations at 60x) and had only slightly worse recall than VarDict, which represents an approximate upper bound on sensitivity. Mutect2 had marginally better sensitivity at some moderate VAFs between 12.5% and 20%.

Finally, we re-ran all methods after downsampling the synthetic tumour and normal samples. We observed an even greater performance differential between Octopus and the other callers in all downsampled tests (Supplementary Table 2 and Figure 6).
Supplementary Figure 4

Somatic mutation calling accuracy for synthetic skin and breast tumours with a paired normal sample with downsampling applied.

(a) Precision-recall curves. Scoring metrics used to generate curves were RFQUAL (Octopus), TLOD (Mutect2), SomaticEVS (Strelka2), QUAL (Lancet), QUAL (LoFreq), SSF (VarDict), and QUAL (Platypus). Only PASS calls are used. VarDict is not visible as it is outside the axis limits due to low precision. Precisions on the two tests are substantially different as the skin set has almost 50 times as many true mutations as the breast set. Dots on the Octopus curve are placed at RFQUAL 7 (3 is used for the entire curve).

(b) Recalls for each Variant Allele Frequency (VAF) using PASS variants. Points show true spike-in VAFs. All comparisons to the synthetic tumour truth sets were performed using RTG Tools vcfeval (version 3.9.1).

Chuanyi Zhang (UIUC)
Somatic Mutations Calling Accuracy without paired normal

Octopus is able to discover mutations even without paired normal sample