Notes on CNV Methods

Mehrtash Babadi,* David Benjamin,† and Samuel K. Lee‡
Broad Institute, 75 Ames Street, Cambridge, MA 02142
(Dated: July 21, 2016)

Some notes on current and proposed methods used in the GATK CNV and ACNV workflows.

I. INTRODUCTION

The GATK uses two types of information from sequencing data to detect copy number variations (CNVs). First, targets (usually exons, but in principle any genomic locus) with abnormally high or low coverage suggest amplifications or deletions, respectively. Second, sites that are heterozygous in a normal sample and have allele ratios significantly different from 1:1 in the matched tumor sample imply a CNV event involving one or both alleles. The workflow is correspondingly split into two major portions:

1. GATK CNV: Using coverage data that has been normalized against a panel of normals to remove sequencing noise, targets are partitioned into continuous segments that represent the same copy-number event. The segmentation is performed by a circular-binary-segmentation (CBS) algorithm described by Olshen et al. 2004 that was originally developed to segment noisy array copy-number data.[1] Amplifications, deletions, and copy-neutral regions are called from the segmentation.

2. GATK ACNV: Heterozygous sites are identified in the normal case sample and segmented, again using CBS, according to their ref:alt allele ratios in the tumor sample. These allele-fraction segments are combined with the copy-ratio segments found by GATK CNV to form a common set of segments. Modeling of both the copy ratio and minor allele fraction of each segment is alternated with the merging adjacent segments that are sufficiently similar according to this model, until convergence.

II. STEPS IN THE GATK CNV AND ACNV WORKFLOWS

A. Coverage collection

[SL: Details of coverage collection go here.] This is implemented by the GATK command-line tool `CalculateTargetCoverage`.

B. Creation of a panel of normals

We cannot simply divide the coverage of each target by the average sequencing depth to obtain an estimate of its copy ratio. This is because the coverage of different targets is heavily-biased by factors including the efficiency of their baits, GC content, and mappability. In order to detect CNVs, we must determine these systematic effects on the coverage of each target in the absence of CNVs, which requires a panel of normal samples (PoN) that are representative of the sequencing conditions of the case sample. PoN samples must also be created using the same baits as the case sample.

The steps for creating a panel of normals are:

1. Obtain the coverage (total number of overlapping reads) of every target and sample.

2. Calculate the median coverage of each target over all samples.

---

*Electronic address: `mbabadi@broadinstitute.org`
†Electronic address: `davidben@broadinstitute.org`
‡Electronic address: `slee@broadinstitute.org`
[1] Specifically, the CBS implementation provided by the `R` package `DNACopy` is used.

3. Filter out targets whose median coverage is below a given percentile (by default 25%) of target medians.
4. Divide all coverages by their corresponding target medians.
5. Filter out samples with too great a proportion of zero-coverage targets (by default 5%).
6. Filter out targets with zero coverage in too great a proportion of samples (by default 2%).
7. Filter out samples whose median coverage is above or below certain percentiles (by default 2.5% and 97.5%) of sample medians.
8. Replace all remaining zero coverages with their corresponding target median.
9. Calculate the range of coverage from percentile p% to $(100-p)$% for each target and truncate coverages at each target to lie within these ranges. By default $p = 0.1$.
10. Divide each coverage by its sample median.
11. Take the $\log_2$ of each coverage.
12. Calculate the median of each sample and take the median of these over all targets. Subtract this median of medians from each coverage.
13. Perform a singular value decomposition (SVD) of the resulting matrix and calculate its pseudo-inverse truncated to the space spanned by the k right eigenvectors with largest singular values. Choose k using Jollife's heuristic of retaining singular values greater than 0.7 times the mean singular value.

This procedure is implemented by the GATK tool `CreatePanelOfNormals`. The output is: a $N \times k$ matrix P, the columns of which are the the retained right eigenvectors (eigensamples), and its pseudoinverse P^+; and the target medians (before any transformations). Here N denotes the number of targets.

C. Segmentation by tangent-normalized coverage

We first divide the integer coverage of the case sample at each target by the corresponding target median from the PoN and take the $\log_2$ transformation to obtain an $N \times 1$ column matrix $\mathbf{x}$. We then calculate the "tangent-normalized" coverage: $\mathbf{x} - PP^+\mathbf{x}$. The meaning of this is as follows: PP^+ is an operator that projects onto the column space of P. That is, it projects onto the space spanned by the k most significant eigensamples representing the (non-CNV-related) variability of the coverage. Subtracting the projection $PP^+\mathbf{x}$ therefore isolates the CNV signal and removes noise due to fluctuations in sequencing bias. This is implemented by the GATK tool `NormalizeSomaticReadCounts`.

Finally, the tangent-normalized coverage vector is passed to CBS to obtain coverage segments. This is implemented by the GATK tool `PerformSegmentation`.

D. Calling of events from coverage segments

[SL: Description of caller goes here.] This is performed by the GATK tool `CallSegments`, which is the final step in the GATK CNV portion of the case-sample workflow.

E. Collection of allele counts at het sites

The first step in the GATK ACNV portion of the case-sample workflow is to gather the necessary allele-count data. This procedure is implemented by the GATK tool `GetHetCoverage`.

Given a large database of common SNPs, we search the normal control sample for heterozygous sites. To determine whether a site with r ref reads and a alt reads is heterozygous, we calculate the two-sided p-value under the null hypothesis that the number of alt reads follows a binomial distribution: $a \sim \text{Binom}(a + r, 1/2)$. If the p-value is not too small we consider the site heterozygous. Ref and alt counts are then obtained at these sites in the tumor case sample.

Alternatively, the GATK tool `GetBayesianHetCoverage`, which instead performs Bayesian hypothesis testing to identify het sites, can be used to perform this step; see Sec. II K for details.

The results of this step are combined with the coverage segments generated by GATK CNV and passed to the GATK tool `AllelicCNV`, which performs the rest of the steps in the GATK ACNV workflow.

F. Segmentation by minor allele fraction

To obtain initial minor-allele-fraction segments, we estimate the minor allele fraction for each het site under the allele-fraction model discussed in Sec. III B. Specifically, we begin by taking the maximum-likelihood estimates given by Equation 16 with reference bias ignored (i.e., $\lambda_j = 1$) and segmenting them with CBS. Using this initial segmentation, the maximum-likelihood values for the reference-bias hyperparameters α and β (see Sec. III B) are found. The minor allele fraction for each het site is then re-estimated as before, but now assuming $\lambda_j = \alpha/\beta$ (i.e., the reference bias at each site is taken to be the mean reference bias across all sites[2]), and again segmented upon. This procedure is iterated until the segmentation converges or the number of iterations reaches a specified limit; if a cycle occurs, the first repeated segmentation is taken.

G. Union of copy-ratio and minor-allele-fraction segments

At this point, per-target estimates of copy ratio (i.e., tangent-normalized coverage) and per-het estimates of minor allele fraction have been segmented separately by CBS. We now perform a segment-union step to combine both segmentations into a single one, the idea being that there may be breakpoints present in the copy-ratio segmentation that were missed in the minor-allele-fraction segmentation and vice versa.[3]

Such a segment union could be created naively by simply taking all segments created by the union of both sets of breakpoints. However, this typically results in oversegmentation, for two reasons: 1) the copy-ratio segmentation itself is typically oversegmented, due to residual systematic noise that is not removed normalization against the panel of normals[4], and 2) the genomic locations of the copy-ratio and minor-allele-fraction breakpoints corresponding to a single underlying event do not exactly overlap (instead, the breakpoints are given by the minimum spanning intervals containing the targets and hets in that event, respectively, and these intervals are not necessarily identical). We thus perform some heuristic steps to reduce the resulting number of segments.

First, if the copy-ratio segments have been called amplified, deleted, or copy neutral (e.g., by the caller described in Sec. II D), we can use the calls to merge adjacent copy-neutral segments, which partially addresses the first issue. Note, however, that oversegmentation can still remain in adjacent amplified or deleted segments.

Second, we can improve upon a naive union of breakpoints by merging segments that are spuriously created due to the inexact overlap of copy-ratio and minor-allele-fraction breakpoints. Not allowing the formation of segments by the addition of minor-allele-fraction breakpoints near the starts and ends of segments that originate from the copy-ratio segmentation partially addresses this issue. Likewise, merging those segments formed by the addition of minor-allele-fraction breakpoints to the middle of segments that originate from the copy-ratio segmentation also reduces oversegmentation.[5] Here, whether such a segment is merged with the adjacent segment to the left or with the adjacent segment to the right is decided by nonparametric tests of similarity between the tangent-normalized coverages and the allele counts in the segments and the genomic distances between the segments. These tests are also used for small-segment merging, which we describe next.

---

[2] An issue is filed to relax this assumption, which is only made so that maximum-likelihood estimates of the minor allele fraction can be easily cached as a function of the ref and alt counts. In future releases, we may also incorporate the distribution across samples of reference bias at each site, which can be learned from an allelic panel of normals (discussed in Sec. V A), into the het-segmentation step.

[3] Note that there are several issues filed concerning this step and that it will most likely be modified in the near future.

[4] This can be somewhat alleviated by using the `undo.splits` parameter of CBS, with varying results. With `undo.splits="sdundo"`, CBS will merge segments within a certain number of standard deviations of each other (analogous to the heuristic procedure we use to perform similar-segment merging, which is discussed in Sec. II J). With `undo.splits="prune"`, CBS instead merges segments that result in a proportional increase of the squared error that is less than a specified threshold; however, note that enabling this option often causes CBS to hang indefinitely on samples with a large number ($\gtrsim$200-300) of segments. Issues are filed to investigate alternatives to segmentation algorithms that could offer the possibility of controlling the amount of segmentation in a more principled way than those offered by CBS.

[5] Both of these procedures treat the copy-ratio segments as the primary segmentation, to which the minor-allele-fraction breakpoints are added. This is done for legacy reasons, even though, as previously mentioned, the copy-ratio segmentation often contains residual segmentation from systematic noise. Furthermore, both corrections are only necessary because a naive union of breakpoints only approximates the true procedure we would like to perform here—that is, using changepoints in one series of data to identify changepoints in a second series of data, in the case where the locations of the data points from both series do not strictly overlap. Perhaps a better procedure would be: 1) perform multiple changepoint detection (e.g., using CBS) on both series, 2) use single changepoint detection to look for additional changepoints in the first series that fall in between those from the second series, and vice versa.

H. Small-segment merging

Using CBS to segment the targets in GATK CNV results in segments that contain at least a specified minimum number of targets n_t (by default, $n_t = 3$). However, after taking the union of target and SNP segments, small segments with less than n_t targets may be introduced. To be consistent with CBS and CNV, ACNV treats these small segments as spurious, and removes them by merging them with adjacent segments.

For each small segment, the question we want to answer is: which of the two adjacent segments (i.e., those to the left or right of the small segment in the center) is most similar to the small segment, and hence, should be merged with it? We would like to determine this based on the tangent-normalized $\log_2$ coverages, allele counts, and genomic locations of each of the three segments. Note that each segment may be missing either coverages or allele counts, but not both. However, this question is not statistically well defined, so we will use the following heuristic procedure:

We first examine the allele counts (as opposed to the coverages—this is because the center segment has a small number of targets, by construction, and so the ability to determine which adjacent segment is more similar to it using the coverages is inherently limited). In particular, we examine the alternate-allele fractions in each of the segments; these should have a bimodal (unimodal) distribution for unbalanced (balanced) segemnts. The Kolmogorov-Smirnov test statistic, which measures the similarity of two data sets, is used to construct two distances between the alternate-allele fractions in the left and center segments and those in the right and center segments, respectively. Including corrections that account for the sizes of the data sets, the distances are used to construct a pair of scores for merging with the left and right segments, respectively. The adjacent segment with the higher score has alternate-allele fractions that are more similar, and hence should be merged with the center small segment.

However, if the Kolmogorov-Smirnov distances are not sufficiently dissimilar, if they are both close to unity (i.e., if the alternate-allele fractions in neither the left nor the right segment overlap significantly with those in the center segment), or if there are not enough hets (>2 data points in each data set) to calculate the Kolmogorov-Smirnov distances, we instead use the inverse minor-allele fractions in each of the three segments. These, ideally, have a distribution that is roughly unimodal in each segment. Two distances between the two pairs of data sets are constructed using the Hodges-Lehmann estimator, which gives a measure of the difference in the location parameters of two data sets, and these distances are used to construct a pair of scores as above.

If the scores generated from the allele counts are too similar or if any of the segments is missing hets, then we attempt to use the coverages instead. Here, we simply use the Hodges-Lehmann estimator to construct two distances and a corresponding pair of scores as above.

If the scores generated from the coverages are too similar or if any of the segments is missing targets, we then simply use genomic distance (defined between adjacent breakpoints) to decide which adjacent segment is closer. For consistency, we also convert the two genomic distances into a pair of scores that sum to unity.

In the unlikely event that all of the above scores are equal, we randomly choose one of the adjacent segments and merge it with the center small segment.

Although this procedure is admittedly quite ad-hoc, it performs reasonably well on simulated data. Furthermore, it is unlikely that incorrect merging of small segments has a significant adverse effect on the subsequent model-fitting step, which we discuss next.

I. Model fitting

At this point, a common segmentation of the genome has been derived from both the tangent-normalized coverage and the het allele counts. Each segment contains at least n_t coverages and may or may not also contain hets.

We now proceed to separately fit a copy-ratio model to the coverages and a minor-allele-fraction model to the allele counts. Both models contain local, segment-level parameters which represent the $\log_2$ copy ratio and minor allele fraction, respectively, as well as global parameters that attempt to model systematic noise and biases arising from sequencing. These models are discussed in detail in Sec. III.

We use Markov Chain Monte Carlo (MCMC) to generate a specified number of samples from the posteriors of each of the parameters. These samples are used to generate posterior summary statistics; the posterior mode, the 95% highest posterior density credible interval, and deciles are reported for the segment-level parameters.

J. Similar-segment merging

We next perform a smoothing step on the fitted copy-ratio and minor-allele-fraction models to further reduce the segmentation.[6] Proceeding across the genome from left to right, we examine the posterior summaries for adjacent pairs of segments. If both the credible intervals for the local copy-ratio and minor-allele-fraction parameters overlap by a specified amount, the segments are merged. The posterior summaries for the newly created segment are determined from those of the original two segments by approximating all posteriors as Gaussian (i.e., using inverse-variance weighting). The new segment is then repeatedly checked for similarity against the adjacent segment to the right and merged until it is no longer similar, at which point we proceed to the next pair of adjacent segments. After one complete traversal of the genome, we optionally refit both models using MCMC.

This procedure is iterated until the segmentation converges or the number of iterations reaches a specified limit. Both models are then refit using MCMC if necessary, resulting in the final output of GATK ACNV: posterior summaries for both $\log_2$ copy ratio and minor allele fraction in each segment.

K. Detection of het sites using a Bayesian model

Here, we describe a procedure for calling heterozygous (Het) sites that (1) takes into account the base read alignment and sequencing qualities, and (2) works for both normal and tumor data. This procedure is implemented by the GATK command-line tool `GetBayesianHetCoverage`.

Provisioning situations that only the tumor data is available to us ("tumor-only"), in addition to the presently considered situation of paired normal-tumor data ("paired normal-tumor"), we need to modify our criterion for calling a Het site. Conceptually, since reads from tumor samples are not pure (contaminated with subclones, normals, etc), a statistical test that rejects the Het hypothesis based on the premise of having equal probability of Ref and Alt reads is bound to reject Het cases when applied to tumor reads. Here, we propose a more sensible Bayesian model.

Notation: Let us first focus on a single site j, with $R_{kj} \in \{A, C, T, G\}$ denoting the mapped base at site j from read k, and ε^B_{kj} and ε^M_k denoting the err probability of base calling and mapping. Also, let Ref_j and Alt_j denote the Ref and Alt alleles at this site.

Definition of error: In case of a base error event, the base could be read as any other three bases with equal probability. In case of a mapping error event, we assume equal probability for all four bases.

Rareness of somatic SNP events: In order to proceed with the model, we assume that somatic SNPs are rare events such that Hom/Het sites retain their germline identity.

Likelihood of Hom_j: Assuming that site j is homozygous (Hom), we find the likelihood of the reads by conditioning over the allele and error events. We easily find:

$$P(R_{kj}|\mathsf{Hom}_j) = P(\mathsf{Ref}_j|\mathsf{Hom}_j) \prod_{k=1}^{N_j} \left[\frac{\varepsilon^B_{kj}}{3} + \frac{\varepsilon^M_k}{4} + \left(1 - \frac{4}{3}\varepsilon^B_{kj} - \varepsilon^M_k\right) \delta_{R_{kj},\mathsf{Ref}_j} \right] + P(\mathsf{Alt}_j|\mathsf{Hom}_j) \prod_{k=1}^{N_j} \left[\frac{\varepsilon^B_{kj}}{3} + \frac{\varepsilon^M_k}{4} + \left(1 - \frac{4}{3}\varepsilon^B_{kj} - \varepsilon^M_k\right) \delta_{R_{kj},\mathsf{Alt}_j} \right]. \quad (1)$$

We need to know the two priors $P(\mathsf{Ref}_j|\mathsf{Hom}_j)$ and $P(\mathsf{Alt}_j|\mathsf{Hom}_j)$, both of which can be estimation from the statistics of the population to which the sample belongs. If this data is not available, we may use the flat prior $P(\mathsf{Ref}_j|\mathsf{Hom}_j) = P(\mathsf{Alt}_j|\mathsf{Hom}_j) = 1/2$ with little harm.

Likelihood of Het_j: Assuming that site j is Het, and that the the probability of the Ref allele in the sample is $f_{j,R}$,

---

[6] This step is analogous to the aforementioned `undo.splits="sdundo"` procedure optionally performed by CBS.

we have:

$$p_{kj,R} \equiv P(R_{kj} = \mathsf{Ref}_j|\mathsf{Het}_j, f_{j,R}) = (1-\varepsilon^B_{kj}-\varepsilon^M_k)\, f_{j,R} + \frac{\varepsilon^B_{kj}}{3}(1-f_{j,R}) + \varepsilon^M_k/4,$$
$$p_{kj,A} \equiv P(R_{kj} = \mathsf{Alt}_j|\mathsf{Het}_j, f_{j,R}) = (1-\varepsilon^B_{kj}-\varepsilon^M_k)\,(1-f_{j,R}) + \frac{\varepsilon^B_{kj}}{3} f_{j,R} + \varepsilon^M_k/4,$$
$$p_{kj,\circ} \equiv P(R_{kj} \neq \mathsf{Ref}_j, \mathsf{Alt}_j|\mathsf{Het}_j, f_{j,R}) = \frac{\varepsilon^B_{kj}}{3} + \frac{\varepsilon^M_k}{4}. \tag{2}$$

Therefore, the likelihood reads:

$$P(\{R_{kj}\}|\mathsf{Het}_j) = \int_0^1 \mathrm{d}f_{j,R}\, P(f_{j,R}|\mathsf{Het}_j) \prod_{k=1}^{N_j} p_{kj,R}^{I(R_{kj}=\mathsf{Ref}_j)}\, p_{kj,A}^{I(R_{kj}=\mathsf{Alt}_j)}\, p_{kj,\circ}^{I(R_{kj}\neq\mathsf{Ref}_j,\mathsf{Alt}_j)}. \tag{3}$$

The integration over $f_{R,j}$ is not as trivial as before since (1) the error probabilities differs from site to site, and (2) the prior is not necessary conjugate to Het likelihood. For the uniformity of notation, we define:

$$R_{kj} = \mathsf{Ref}_j \quad \Rightarrow \quad \alpha_{kj} \equiv \frac{\varepsilon^B_{kj}}{3} + \frac{\varepsilon^M_k}{4}, \quad \beta_{kj} = 1 - \frac{4\,\varepsilon^B_{kj}}{3} - \frac{\varepsilon^M_k}{4},$$
$$R_{kj} = \mathsf{Alt}_j \quad \Rightarrow \quad \alpha_{kj} \equiv 1 - \varepsilon^B_{kj} - \frac{3\,\varepsilon^M_k}{4}, \quad \beta_{kj} = -1 + \frac{4\,\varepsilon^B_{kj}}{3} + \varepsilon^M_k. \tag{4}$$

such that:

$$P(\{R_{kj}\}|\mathsf{Het}_j) = \left[\prod_{k\in\mathcal{I}_\circ} \frac{\varepsilon_{kj}}{3}\right] \left[\int_0^1 \mathrm{d}f\, P(f|\mathsf{Het}) \prod_{k\in\mathcal{I}_{RA}} (\alpha_{kj} + \beta_{kj} f)\right], \tag{5}$$

where $\mathcal{I}_\circ$ are the indices of reads that are neither Ref or Alt at site j, and $\mathcal{I}_{RA}$ are indices of reads that either Ref or Alt. Furthermore, $P(f|\mathsf{Het})$ is the common prior for Ref allele fraction. For a given prior, we calculate the f-integral numerically with a fixed-order quadrature. Since the integrand is polynomial of f, a Gaussian quadrature is well-suited to approximate the integral provided that the prior is also smooth.

Caveats: (1) sensitivity to error underestimation: if the read/alignment qualities are overestimated, even a single deviation from the Hom_j hypothesis can dramatically reduce the likelihood. (2) Loss of heterozygosity can manifest itself has homozoygosity; in practice, it should not be an issue since the samples are not pure and germline heterozygosity should yield sufficient evidence to reject the Hom hypothesis. (3) Heterozygosity in a sizable subclone resulting from a somatic SNP may manifest itself as germline heterozygosity. This is also expected not to be a major issue since somatic SNPs are rare.

A model prior for allele fraction at Het sites: In this section, we construct a simple prior for the Ref allele fraction at Het sites. To this end, we assume (1) a minimum (maximum) fraction $\rho_{\min}$ ($\rho_{\max}$) of the cells in the sample may have events that change the allele fraction with respect to germline (large copy number events, CNLOH, etc). Furthermore, we assume that the maximum copy number is bounded from above by N_c. Otherwise, we assume flat priors over both the copy number and non-germline fraction. Under these assumptions, the distribution of the Ref allele fraction is given by:

$$P(f|\mathsf{Het}) = \frac{1}{(N_c+1)^2} \sum_{n,m=0}^{N_c} \int_{\rho_{\min}}^{\rho_{\max}} \frac{\mathrm{d}\rho}{\rho_{\max}-\rho_{\min}}\, \delta\left(f - \frac{(1-\rho)+\rho m}{2(1-\rho)+\rho(m+n)}\right). \tag{6}$$

Since the prior will be symmetric under the transformation $f \to 1-f$, we will assume $f < 1/2$ hereafter. The ρ integration is trivially performed and we find:

$$P(f|\mathsf{Het}) = \frac{1}{(N_c+1)^2} \sum_{n,m=0}^{N_c} \frac{1}{\rho_{\max}-\rho_{\min}} \frac{|n-m|}{[1-m+f(n+m-2)]^2}\, \theta\left(f - \frac{(1-\rho_{\max})+\rho_{\max} m}{2(1-\rho_{\max})+\rho_{\max}(m+n)}\right)$$
$$\times\, \theta\left(\frac{(1-\rho_{\min})+\rho_{\min} m}{2(1-\rho_{\min})+\rho_{\min}(m+n)} - f\right). \tag{7}$$

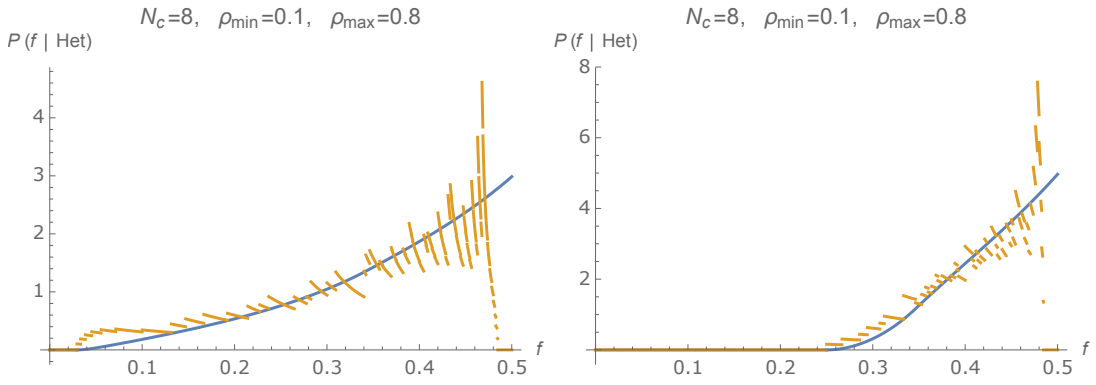

FIG. 1: Two examples of the Ref allele fraction prior $P(f|\mathsf{Het})$ at Het sites based on minimum/maximum non-germline cells and maximum copy number. The blue lines denote the continuous approximation given in Eq. (8), The discontinuous organge lines denote the result with discrete copy number summation given in Eq. (the delta function peak at $f = 1/2$ is not shown).

The summand is ambiguous for $n = m = 1$ since f evalues to $1/2$ independent of ρ. The correct prescription is to replace it with $\delta(f - 1/2)/(\rho_{\max} - \rho_{\min})$.

The discrete summation over the copy numbers (n, m) result in a discontinuous prior. It is convenient to approximate the discrete summations with integrals over n and m. This approximation preserves the main features of the prior while converges to the discrete result for large N_c. The double integral over (n, m) must be performed with diligence since the Heaviside functions restrict the integration region depending on the value of f. We leave out the details and just quote the final result:

$$P(f|\mathsf{Het}) = \begin{cases} P_<(f) & f_{\rm th} \le f \le f^*, \\ P_>(f) & f^* < f \le \frac{1}{2}, \end{cases} \tag{8}$$

where:

$$f_{\rm th} = \frac{1-\rho_{\max}}{N_c\,\rho_{\max} + 2(1-\rho_{\max})},$$
$$f^* = \frac{1-\rho_{\min}}{N_c\,\rho_{\min} + 2(1-\rho_{\min})},$$
$$P_<(f) = \frac{(\rho_{\max}(fN_c-1)-1)(f((N_c-2)\rho_{\max}+2)+\rho_{\max}-1) + 2\rho_{\max}(f(fN_c-2)+1)\log\left(\frac{\rho_{\max}(f(N_c-2)+1)}{1-2f}\right)}{2\,(f-1)^2\,f^2\,N_c^2\,\rho_{\max}\,(\rho_{\max}-\rho_{\min})},$$
$$P_>(f) = \frac{(\rho_{\max}-\rho_{\min})(\rho_{\max}\rho_{\min}(f(N_c-2)+1)(fN_c-1)+2f-1) + 2\rho_{\max}\rho_{\min}(f(fN_c-2)+1)\log\left(\frac{\rho_{\max}}{\rho_{\min}}\right)}{2\,(f-1)^2\,f^2\,N_c^2\,\rho_{\max}\,\rho_{\min}\,(\rho_{\max}-\rho_{\min})}. \tag{9}$$

Fig. 1 shows two examples of this prior along with the version with discrete copy number summations.

The Bayesian decision rule: Using the Bayes' theorem, the log odds of heterozygosity is found as:

$$\log\text{odds}(\mathsf{Het}_j) = \log\,P(\{R_{kj}\}|\mathsf{Het}_j) + \log\,P(\mathsf{Het}_j) - \log\,P(\{R_{kj}\}|\mathsf{Hom}_j) - \log\,P(\mathsf{Hom}_j). \tag{10}$$

In order to evaluate the right hand side, we need to have knowledge of the prior $P(\mathsf{Het}_j)$. This can be worked out from population statistics. Otherwise, we may use the flat prior $P(\mathsf{Het}_j) = 1/2$.

Having the log odds, the decision rule is simple: we call a Het site if its odds exceeds a given threshold:

$$\text{Call } \mathsf{Het}_j \quad \Leftrightarrow \quad \log\text{odds}(\mathsf{Het}_j) > \log\frac{1-10^{-s_{\mathsf{Het}}}}{10^{-s_{\mathsf{Het}}}} = \log(10^{s_{\mathsf{Het}}}-1), \tag{11}$$

where we have defined the Het *calling stringency parameter* s_{Het} as a convenient parametrization of the decision boundary. Finally, we note that the log likelihoods scale linearly with the read depth N_j (each read results in an additional multiplicative term). Therefore, the statistic $\log\text{odds}(\mathsf{Het}_j)$ linearly deviates from the decision threshold

$\log(10^{s_{\mathsf{Het}}} - 1) \propto s_{\mathsf{Het}}$ as the read depth increases.

Increasing power using haplotype information: Todo. The basic idea is to utilize SNP correlations to test multiple correlated sites simultaneously for heterozygosity (1) to increase power, and (2) to improve the prior on $\mathsf{Ref}_j/\mathsf{Alt}_j$. A good starting point to run HaplotypeCaller on a few normal/tumor reads and check the strength/range/size of correlations between SNP constellations.

III. GATK CNV/ACNV MODELS

A. Copy-ratio model

We fit a simple, heuristic copy-ratio model to the segmented, tangent-normalized $\log_2$ coverages. Our primary goal is to estimate accurate mean $\log_2$ copy ratios within each segment for use in downstream tools, so a simple model suffices assuming that the tangent-normalization step has removed most of the systematic noise.

The model assumes that the tangent-normalized $\log_2$ coverages in each segment are distributed as a mixture of: 1) a Gaussian distribution, with mean given by a segment-level $\log_2$ copy-ratio parameter and a variance that is common to all the segments[7], and 2) a uniform distribution, which is meant to model outlier $\log_2$ coverages and is truncated at the minimum and maximum observed values.

This model can be expressed in terms of local target-level outlier indicators and segment-level parameters for the mean $\log_2$ copy ratio, as well as global parameters for the variance and outlier-distribution mixture fraction. We generate posterior samples by Gibbs sampling the conditional distributions for each parameter in succession; slice sampling is used for continuous parameters.

B. Allelic model

We want a generative model for allelic fractions that infers its parameters from the data. We observe alt and ref read counts for each het site and wish to infer the minor allelic fraction of every segment. Let's consider what other hidden variables belong in the model. Read counts obey an overdispersed binomial distribution in which the probability of an alt read is a site-dependent random variable. Letting θ_j be the probability that a mapped read at het j is an alt we have

$$P(a_j, r_j|\theta_j) = \binom{a_j + r_j}{a_j}\theta_j^{a_j}(1-\theta_j)^{r_j} = \binom{n_j}{a_j}\theta_j^{a_j}(1-\theta_j)^{r_j}, \tag{12}$$

where a_j and r_j are alt and ref read counts and $n_j = a_j + r_j$ is the total read count at site j. Now we consider θ_j. Suppose site j belongs to a segment with minor allelic fraction f and is alt minor, such that $P(\text{alt}) = f$ and $P(\text{ref}) = 1 - f$ are the probabilities that a random DNA fragment will contain the alt and ref alleles. Let $x_j^{\text{alt(ref)}} = P(\text{mapped}|\text{alt(ref)})$ be the probabilities that an alt (ref) DNA fragment at site j eventually gets sequenced and mapped. Then θ_j is the conditional probability that a mapped read comes from an alt fragment:

$$\theta_j = P(\text{alt}|\text{mapped}) = \frac{P(\text{alt})P(\text{mapped}|\text{alt})}{P(\text{alt})P(\text{mapped}|\text{alt}) + P(\text{ref})P(\text{mapped}|\text{ref})} \tag{13}$$

$$= \frac{f x_j^{\text{alt}}}{f x^{\text{alt}} + (1-f)x_j^{\text{ref}}} = \frac{f}{f + (1-f)\lambda_j}, \tag{14}$$

where $\lambda_j = x_j^{\text{ref}}/x_j^{\text{alt}}$ is the "bias ratio" of ref to alt sequenceability and mappability at site j. A similar result for ref minor sites follows from substituting $f \leftrightarrow 1 - f$. In addition to the bias ratio λ_j we need an indicator variables

---

[7] Note that this differs from `AllelicCapSeg`, which assumed that non-outlier, tangent-normalized non-$\log_2$ coverages were Gaussian distributed with variance proportional to the mean in each segment. However, examination of many samples shows that this proportionality is only weakly exhibited, on average, in the non-$\log_2$ coverages—and even less so in the $\log_2$ coverages, which exhibit a variance than is, on average, constant with mean $\log_2$ copy ratio. It is true that allowing the variance in each segment to be drawn from a global distribution, rather than fixed to a single number, would provide a better fit to the data; however, the estimates of the mean $\log_2$ copy ratio are not likely to be strongly biased under the simple model. Furthermore, this copy-ratio model will soon be superseded by the generative coverage model proposed in Sec. V B.

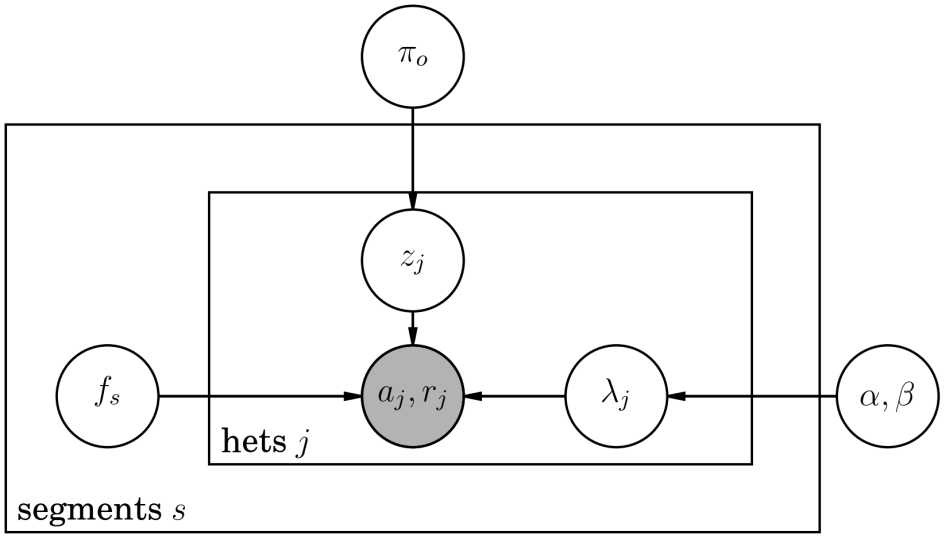

FIG. 2: Graphical model for ACNV allelic model

z_j with three states, alt minor, ref minor, and an outlier state that gives robustness to anomalous events. For this outlier state we average the binomial likelihood over all θ to get:

$$P(a_j, r_j|\text{outlier}) = \binom{n_j}{a_j} \int_0^1 \theta_j^{a_j} (1-\theta_j)^{r_j} \, d\theta_j = \binom{n_j}{a_j} \frac{a_j! r_j!}{(n_j+1)!} \tag{15}$$

For notational convenience we give z_j a one-of-K encoding $z_j = (z_{ja}, z_{jr}, z_{jo})$ in which one component equals 1 and the rest 0.

The contribution of site j to the likelihood is

$$P(a_j, r_j|f_j, \lambda_j, z_j) = \binom{n_j}{a_j} \left[\frac{f_j^{a_j}(1-f_j)^{r_j}\lambda_j^{r_j}}{(f_j+(1-f_j)\lambda_j)^{n_j}} \right]^{z_{ja}} \left[\frac{(1-f_j)^{a_j} f_j^{r_j}\lambda_j^{r_j}}{(1-f_j+f_j\lambda_j)^{n_j}} \right]^{z_{jr}} \left[\frac{a_j! r_j!}{(n_j+1)!} \right]^{z_{jo}} \tag{16}$$

where f_s is the minor allele fraction of the segment containing site j. We will consider f to be drawn from a uniform distribution on $[0, 1/2]$ – that is, we give it a flat prior – but in the future we can obtain some sort of clustering behavior, representing the fact that events in the same subclone that exhibit the same integer copy numbers will have the same minor allelic fractions, by drawing f_s from a Dirichlet process.

We assume that the bias ratios come from a common Gamma distribution with parameters α, β:

$$P(\lambda_j|\alpha, \beta) = \frac{\beta^\alpha}{\Gamma(\alpha)} \lambda_j^{\alpha-1} e^{-\beta\lambda_j} \tag{17}$$

Note that bias ratios tend to be near 1.0 and so the choice of distribution is not too important as long as it has adjustable mean and standard deviation. We choose the Gamma distribution because it is the simplest such distribution on $\mathbb{R}^+$. We will give the parameters α and β a flat prior $P(\alpha, \beta) \propto 1$.

Finally, the indicator z_j is a multinomial random variable distributed according to parameter vector π:

$$P(z_{ja(r,o)} = 1|\pi) = \pi_{a(r,o)} \tag{18}$$

We set the alt and ref minor probabilities equal so that the only free parameter is $\pi = \pi_o$, with $\pi_{a(r)} = (1-\pi)/2$. The Bayesian network corresponding to this model is shown in Figure III B.

As with the other parameters, we put a flat prior on π. Putting all the pieces together the likelihood is

$$\mathbb{L} = \prod_j \frac{\beta^\alpha}{\Gamma(\alpha)} \lambda_j^{\alpha-1} e^{-\beta\lambda_j} \left[\frac{(1-\pi) f_j^{a_j}(1-f_j)^{r_j}\lambda_j^{r_j}}{(f_j+(1-f_j)\lambda_j)^{n_j}} \right]^{z_{ja}} \left[\frac{(1-\pi)(1-f_j)^{a_j} f_j^{r_j}\lambda_j^{r_j}}{(1-f_j+f_j\lambda_j)^{n_j}} \right]^{z_{jr}} \left[\frac{2\pi a_j! r_j!}{(n_j+1)!} \right]^{z_{jo}} . \tag{19}$$

The dependence on λ for alt minor sites is

$$g(\lambda_j, \alpha, \beta, f_j, a_j, r_j) = \frac{\beta^\alpha}{\Gamma(\alpha)} \frac{f_j^{a_j}(1-f_j)^{r_j}\lambda_j^{\alpha+r_j-1}e^{-\beta\lambda_j}}{(f_j+(1-f_j)\lambda_j)^{n_j}}. \tag{20}$$

For ref minor sites the dependence is the same but with $f \leftrightarrow 1-f$. We show in show in Appendix A that this function can be integrated analytically, and thus we can marginalize λ out of the model to obtain the likelihood

$$\prod_j \left[\frac{1-\pi}{2}\phi(\alpha,\beta,f_j,a_j,r_j)\right]^{z_{ja}} \left[\frac{1-\pi}{2}\phi(\alpha,\beta,1-f_j,a_j,r_j)\right]^{z_{jr}} \left[\frac{\pi a_j! r_j!}{(n_j+1)!}\right]^{z_{jo}}, \tag{21}$$

where $\phi(\alpha,\beta,f_j,a_j,r_j) = \int_0^\infty g(\lambda,\alpha,\beta,f,a,r)\,d\lambda_j$. Pseudocode for computing ϕ is presented in Appendix A. Furthermore, marginalizing out z is trivial – simply sum each term over its three possible states. We then have a collapsed likelihood

$$p(f,\alpha,\beta,\pi) \propto \prod_j \left[\frac{1-\pi}{2}\phi(\alpha,\beta,f_j,a_j,r_j) + \frac{1-\pi}{2}\phi(\alpha,\beta,1-f_j,a_j,r_j) + \frac{\pi a_j! r_j!}{(n_j+1)!}\right] \tag{22}$$

Integrating out the latent variables removes the strongest correlations from the model – intuitively, f should not be too sensitive to α and β, for example – and significantly improves mixing. The exception is α and β, since adjusting one with the other fixed changes the mean of the prior on λ. Thus we reparameterize in terms of μ and σ^2, the mean and variance of the common gamma distribution of biases, where $\alpha = \mu^2/\sigma^2$ and $\beta = \mu/\sigma^2$. Due to the weak correlations our MCMC method does not need to be very sophisticated. We choose to sample each variable with one-dimensional adaptive Metropolis, tuning the proposal step size to achieve some reasonable acceptance rate like 0.4 or so. Thus we have completely specified an MCMC scheme for this model, given by Algorithm 1:

Algorithm 1 MCMC algorithm for ACNV allelic model

1: Initialize all parameters to a maximum likelihood initial guess (see below).
2: **repeat**
3: Sample each f_s with adaptive Metropolis
4: Sample π with adaptive Metropolis
5: Sample μ with adaptive Metropolis
6: Sample β with adaptive Metropolis
7: **until** convergence

We initialize the model by finding the mode of likelihood. This significantly reduces burn-in time of our MCMC sampling. Also, it allows us to give the adaptive Metropolis samplers better initial guesses for their step sizes. Since in practice there is a single global maximum of the likelihood it is easy to find. After initializing the initialization with rough guesses for the parameters, we successively find one-dimensional maxima adjusting one parameter at a time until the likelihood converges. One could use multidimensional optimization to obtain faster convergence, but after marginalizing out latent parameters the remaining correlations are weak and thus this simple approach performs quite well. Since we may delegate one-dimensional maximization to mathematical libraries, the only thing left to describe is our initial coarse guess.

In the initial guess we set the outlier probability $\pi_o = 0.01$, $\mu = 1.0$, and $\sigma^2 = 0.1$. With the exception of σ^2 these are all reasonable guesses. We choose σ^2 to be larger than what we actually believe because μ converges more slowly from a bad initial guess if σ^2 is too small. The only non-trivial part of the initial guess is the minor allele fractions. For each segment, we wish to set the minor allele fraction to the number of reads from minor alleles divided by to total number of reads – this is an unbiased estimator if allelic bias is absent. The problem is that we have counts of alt and ref reads, not minor and major reads. Our solution is to weight the alt and ref read counts on each het by probabilities that the het is alt and ref minor, respectively. That is, we set

$$f_S \approx \frac{\sum_{j\in S} a_j P(z_{ja}=1) + r_j P(z_{jr}=1)}{\sum_{j\in S}(a_j+r_j)(P(z_{ja}=1)+P(z_{jr}=1))} \tag{23}$$

For this coarse guess we ignore the possibility of outliers, so that $P(z_{ja}=1)+P(z_{jr}=1)=1$. Ignoring bias and outliers the alt minor likelihood of het j is proportional to $f_j^{a_j}(1-f_j)^{r_j}$. Since we don't know f yet, we integrate this (including the normalization) from $f=0$ to $f=1/2$ in order to get $P(z_{ja}=1)$. This quantity is called the incomplete regularized beta function I. Thus we have

$$P(z_{ja}=1) \approx I(1/2, a_j+1, r_j+1), \quad P(z_{jr}=1) = 1 - P(z_{ja}=1). \tag{24}$$

C. Calling segments after allelic CNV workflow

After running the allelic fraction and copy ratio model, we have a list of segments s, each with its own posterior pdfs f_s^{CR} and f_s^{MAF} of the copy ratio and minor allele fraction[8]. That is, $f_s^{MAF}(x)$ is the posterior probability density from ACNV that segment s has minor allele fraction x. We assume that for each segment some fraction ρ of sequenced cells carry m and n copies of the original homologs, while the remaining $1-\rho$ cells are diploid. This assumption is compatible with both normal contamination and tumor heterogeneity but not with distinct subclones containing different CNVs at overlapping segments. It *can* express distinct subclones that inherit a CNV from a common ancestor, as well as a single subclone that incurs overlapping CNVs as long as both are fixed (in the population genetics sense) in that subclone.

Each distinct value of ρ therefore corresponds to a node in the tumor's phylogenetic tree, its value being the proportion of sequenced cells belonging to subclones descended from that node. We therefore expect its values to be drawn from a discrete multinomial distribution, on which we place a symmetric and sparse Dirichlet prior. That is, let ρ take on values $\rho_1, \rho_2 \dots \rho_K$ and let z_s be a binary-valued indicator vector such that $z_{sk} = 1$ if the CNV on segment s occurs in fraction ρ_k of sequenced cells. Then

$$P(\pi|\alpha) = \frac{\Gamma(\alpha)}{\Gamma(\alpha/K)^K} \prod_k \pi_k^{\alpha/K-1} \tag{25}$$

$$P(z_s|\pi) = \prod_k \pi_k^{z_{sk}} \tag{26}$$

Here α is the concentration parameter such that the smallness of α/K enforces sparseness[9], i.e. most cluster components will not be used. The $K \to \infty$ limit is a Dirichlet process and for finite K to work well, K must be larger than the number of components needed; in practice making K twice as large as the number of components works well. The expected number of clusters found in data of size N (here, the number of segments) is roughly $\alpha \ln N$, so we place a vague prior on α that corresponds to roughly a single- or double-digit number of clusters. For example, a broad gamma prior with mean 1:

$$P(\alpha) = \mathrm{Gamma}(\alpha|1,1) \tag{27}$$

We have little prior knowledge on tumor's phylogeny, so we put a uniform prior on the values of ρ: $P(\rho_k) = 1$.

Next we relate copy ratio and minor allele fraction to (m, n, ρ). The total copy number is a weighted sum of $(1-\rho)$ diploid cells and ρ cells with copy number $m+n$.

$$\mathrm{cr}(m, p, \rho) \equiv (2(1-\rho) + \rho(m+n))/2. \tag{28}$$

Similarly, the minor allele fraction is a weighted sum of $1-\rho$ diploid cells with a single copy of the minor allele and ρ cells with $\min(m,n)$ copies, divided by the total:

$$\mathrm{maf}(m, n, \rho) \equiv \frac{(1-\rho) + \rho \min(m,n)}{2(1-\rho) + \rho(m+n)} \tag{29}$$

It is convenient to represent the latent state (m, n) via binary indicator variables v and w with e.g. $v_{sm} = 1, w_{sn} = 1$ if segment s has m and n copies of the original homologs.

Finally, we place a simple factorized multinomial prior on (m, n): $P(m, n) = P(m)P(n) = \phi_m \phi_n$, which we can do if we set of maximum copy number of, say, $m, n < 4$. The factorization assumption realistic regarding the origin of CNVs but not necessarily regarding their *viability*. For example, a homozygous deletion could be lethal when a heterozygous deletion is not. However, we expect this effect to be less dramatic for small segments, which have less phenotypic impact. Large segments ought to have sufficient statistical power that the prior is less important. Taking into account the copy ratio and minor allele fraction posteriors from ACNV as well as the multinomial prior, the model likelihood is

$$P(z_s, v_s, w_s, \pi, \phi, \rho, \alpha) = P(\alpha)\frac{\Gamma(\alpha)}{\Gamma(\alpha/K)^K} \prod_k \pi_k^{\alpha/K-1} \prod_{s,k,n,m} \left[\pi_k \phi_m \phi_n f_s^{\mathrm{CR}}(\mathrm{cr}(m,n,\rho_k) f_s^{\mathrm{MAF}}(\mathrm{maf}(m,n,\rho_k)\right]^{z_{sk} v_{sm} w_{sn}} \tag{30}$$

---

[8] ACNV obtains MCMC samples from these posteriors; we assume that a reasonable distribution has been fit to these posterior samples.
[9] If $\alpha < K$ the prior is singular as $\pi_k \to 0$ for any k.

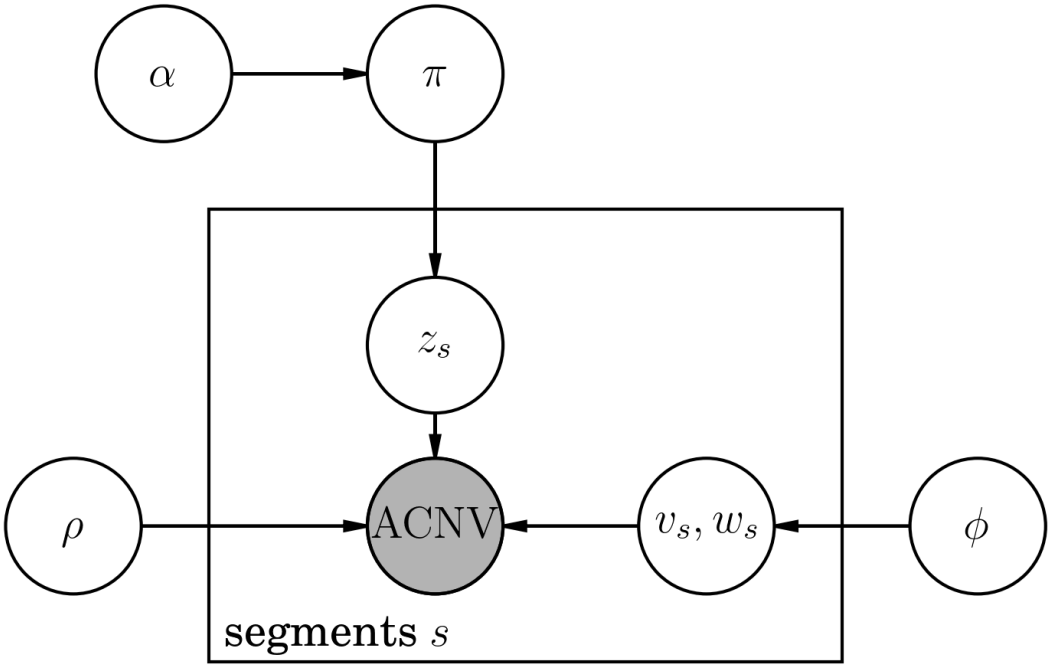

FIG. 3: Graphical model for ACNV caller. "ACNV" represents posterior probability output of ACNV; v, w are indicators of homolog integer copy numbers; ρ is the set of values of purity × cancer cell fraction; z is the corresponding indicator; ϕ is the multinomial prior on homolog counts; π is the multinomial prior on z; α is a Dirichlet concentration parameter encouraging a sparse set of ρ values.

Note that we have simply multiplied of contributions of copy number and minor allele fraction. This is justified because we inferred the former only from total read counts, while the inference for the latter was *conditioned* on the total read depth of each het. Thus there is no double-counting of evidence. This argument is somewhat heuristic because ACNV infers copy number from *target* read counts and minor allele fraction from *SNP* allele counts, but is valid to the extent that total depth at het sites is correlated with depth and the targets they belong to. For off-target SNPs it is not heuristic at all.

The graphical model is shown in Figure III C.

We will obtain maximum likelihood estimates of ρ and ϕ and give the remaining variables the variational factorized distribution $p(\alpha, \pi, z, v, w) \to q(\alpha)q(\pi)q(z, v, w)$. We now proceed to carry out the standard recipe of the EM and variational Bayes algorithms. Denoting one variable or group of variables by X, all other variables by Z, and the joint probability by $p(X, Z)$, the mean-field posterior on X is

$$\ln q(X) = E_{q(Z)}[\ln p(X, Z)] + \text{const} \tag{31}$$

For those variables X for which we seek a point estimate and not a full posterior we employ a similar formula

$$X = \arg\max \left[E_{q(Z)}[\ln p(X, Z)]\right] \tag{32}$$

We will henceforth drop the subscript $q(Z)$ from the expectation $E_{q(Z)}$ – all expectations are with respect to the factorized distribution. Following this prescription, we find that the posterior on α is

$$q(\alpha) \propto \frac{P(\alpha)\Gamma(\alpha)}{\Gamma(\alpha/K)^K} \exp\left(\frac{\alpha}{K}\sum_k E\left[\ln \pi_k\right]\right) \tag{33}$$

The posteriors on π and ϕ are

$$q(\pi) \propto \prod_k \pi_k^{E[\alpha]/K - 1 + \sum_s E[z_{sk}]},\ q(\phi) \propto \prod_j \phi_j^{\sum_s E[v_{sj} + w_{sj}]} \tag{34}$$

The maximization objective for ρ is

$$\rho_k = \arg\max \sum_{s,k,n,m} E[z_{sk} v_{sm} w_{sn}] \left[\ln f_s^{\mathrm{CR}}(\mathrm{cr}(m,n,\rho_k) + \ln f_s^{\mathrm{MAF}}(\mathrm{maf}(m,n,\rho_k)\right] \tag{35}$$

Lastly, $q(z,v,w)$ is a categorical distribution which we evaluate by plugging in values:

$$E\left[z_{sk} v_{sm} w_{sn}\right] = \frac{\phi_m \phi_n e^{E[\ln \pi_k]} f_s^{\mathrm{CR}}(\mathrm{cr}(m,n,\rho_k) f_s^{\mathrm{MAF}}(\mathrm{maf}(m,n,\rho_k)}{\sum_{k,m,n} \text{“ ”}} \tag{36}$$

Equations 33 – 36 require the expectations $E[\alpha]$, $E[\ln \pi]$, $E\left[z_{sk}\right]$, $E[v_{sj}]$, $E[w_{sj}]$, and $E[z_{sk} v_{sm} w_{sn}]$. The last of these is the E step Equation 36. Three more follow directly from marginalization:

$$E\left[z_{sk}\right] = \sum_{m,n} E[z_{sk} v_{sm} w_{sn}],\; E[v_{sj}] = E[w_{sj}] = \sum_{k,n} E[z_{sk} v_{sj} w_{sn}] \tag{37}$$

By inspection, the Dirichlet posterior $q(\pi)$ of Equation 34 yields the following logarithmic moments:

$$E[\ln \pi_k] = \psi\left(E[\alpha]/K + \sum_s E\left[z_{sk}\right]\right) - \psi\left(E[\alpha] + \sum_{s,k} E\left[z_{sk}\right]\right), \tag{38}$$

where ψ is the digamma function. Likewise, $q(\phi)$ is Dirichlet and is maximized with

$$\phi_j = \frac{\sum_s E\left[v_{sj} + w_{sj}\right]}{\sum_{s,i} E\left[v_{si} + w_{si}\right]} \tag{39}$$

$E[\alpha]$ is not analytic but requires only a single numerical integral per iteration:

$$E[\alpha] = \frac{\int \alpha\; q(\alpha)\; d\alpha}{\int q(\alpha)\; d\alpha} \tag{40}$$

We therefore have a self-contained iteration scheme in terms of expectations only, Algorithm 2.

Algorithm 2 calling allele counts of ACNV segments

1: Initialize $E[\alpha] = 1$
2: Initialize $(\rho_1, \rho_2, \ldots \rho_K) = (1/K, 2/K, \ldots 1)$
3: Initialize $E[\ln \pi_j] = \ln(1/K)$ for all j.
4: Initialize ϕ in some reasonable way, i.e. $\phi_1 > \phi_2 > \phi_0 > \phi_3$.
5: **repeat**
6: Update $E[z_{sk} v_{sm} w_{sn}]$ via Equation 36.
7: Update $E\left[z_{sk}\right]$, $E[v_{sj}]$, $E[w_{sj}]$ via Equation 37.
8: Update $E[\ln \pi_k]$ via Equation 38
9: Update ϕ via Equation 39
10: Update $E[\alpha]$ via Equation 40
11: Update ρ via Equation 35
12: **until** convergence

Once this converges, the main objects of interest are the posterior probabilities of different allele counts, $P(v_{sm} = 1, w_{sn} = 1) = \sum_k E\left[z_{sk} v_{sm} w_{sp}\right]$. For the purposes of guessing phylogeny the fractions ρ_k are also interesting.

IV. GERMLINE EXOME CNVS

The GATK treats germline CNVs differently from somatic CNVs. This is partly due to fundamental differences, such as the absence of subclones in the germline setting. However, many arbitrary inconsistencies are historic in nature, arising from the germline algorithm's origins in the XHMM method. It is important to keep this in mind when reading these notes. The two most significant differences between the GATK's germline and somatic workflows is are the neglect of allelic information (i.e. alt and ref read counts at het sites) in the germline workflow and the use of an HMM for simultaneous segmentation and calling in the germline workflow.

We will treat the HMM as a black box. Although the GATK has its own implementation, the functionality is standard. Thus we will only describe how we define its states, initial probabilities, transition probabilities, and emission distributions. Besides that, it suffices to describe what is done to raw coverage data before it is fed into the HMM.

A. Normalization of raw germline data

The germline model does not separate the creation of a panel of normals from a case workflow. Rather, it calls CNVs simultaneously for all samples in a cohort. Its starting point is an $S \times T$ matrix of raw coverage, where S is the number of samples and T is the number of targets. We then normalize by each sample's average coverage to get an $S \times T$ *proportional coverage* matrix P:

$$P_{st} = \frac{(\text{raw coverage})_{st}}{\text{average depth of sample } s} \tag{41}$$

Next, as in the somatic workflow, we perform principal components analysis (PCA) on the proportional coverage in order to remove noise due to laboratory conditions etc. from the coverage, leaving (we hope) only a CNV signal and a small amount of residual noise. For purely historical reasons PCA is expressed here in slightly different terms from the somatic case. PCA yields a length-T mean proportional coverage vector μ and set of M principal vectors $\mathbf{v}_k$, also of length T, such that the proportional coverage of each sample is approximated by the mean coverage μ plus a linear combination of the principal components:

$$P_s \approx \mu + \sum_{k=1}^{M} \beta_{sk} \mathbf{v}_k \tag{42}$$

Because the principal components capture the shared variation among all samples, we expect them *not* to capture individual variation due to CNVs. There is necessarily some contamination because the samples we call are the same samples used to decide the principal components – there is no separate PoN. Nonetheless, this effect should be small if there are enough samples. Therefore, the next step is to produce the tangent-normalized coverage X, which is again an $S \times T$ matrix:

$$X_s = P_s - \mu - \sum_{k=1}^{M} \beta_{sk} \mathbf{v}_k. \tag{43}$$

(Here a single subscript for a matrix denotes an entire row).

Finally, the tangent-normalized coverage is converted to a Z-score coverage in which each target is mean-centered (tangent-normalization should yield a mean of zero for each target over all samples, so this part is trivial) and divided by the standard deviation of tangent-normalized coverage of that target over all samples:

$$Z_{st} = X_{st} / \text{std}(X_{\cdot t}) \tag{44}$$

The codebase also allows for filtering at each stage of coverage based on target GC and repeat fraction and various coverage descriptive statistics such as mean, standard deviation and interquartile range of targets across samples and vice versa. However, we do not yet have a sense of best practices for these. Furthermore, what constitutes best practices will change as we improve the model.

B. Germline HMM

Each sample's Z-score coverage is segmented and called separately via the Viterbi algorithm, which finds the maximum-likelihood solution of an HMM. The hidden states are neutral, deletion, and duplication – the XHMM model does not take into account homozygous deletions or multiple duplications.

The HMM's transition matrix is guided by the principle (an approximation, of course) that there is some underlying biological HMM on a *per-base* level and that *per-target* transitions are simply the realization of this underlying HMM on a coarser scale. The per-base transition matrix is defined by two parameters. The first is the probability p to to make a transition from a neutral state to a CNV state. Equivalently, $1/p$ is, roughly, the average separation between CNVs. The second is the probability $1/D$ that a CNV state ends. Equivalently, D is the average CNV length in base pairs. The probability for a CNV to terminate between two consecutive targets a distance d apart is $1 - e^{-d/D}$.

Letting $f = e^{-1/D}$ the transition matrix T between two adjacent bases is

$$T = \begin{array}{c} \text{from\to} \\ - \\ 0 \\ + \end{array} \begin{array}{c} \begin{array}{ccc} - & 0 & + \end{array} \\ \begin{pmatrix} f & 1-f & 0 \\ p & 1-2p & p \\ 0 & 1-f & f \end{pmatrix} \end{array} \tag{45}$$

We neglect transitions between different types of CNVs at consecutive bases, which are extremely rare. Note that this in no way precludes CNVs of different types occurring at adjacent targets. The transition matrix for two targets separated by d bases is T^d. We can compute this very cheaply by first diagonalizing T as $T = U\Sigma V^T$, where Λ is a diagonal matrix. Then $T^d = U^T\Lambda^d U$. For numerical stability one usually works with log transition probabilities, so we have:

$$\log\left(T^d\right)_{ij} = \log\sum_k U_i^T k\Lambda_{kk}^d U_{kj} \tag{46}$$

$$= \log\sum_k \Lambda_{kk}^d U_{kj} U_{ki} \tag{47}$$

$$= \log\sum_k \exp\left(d\log\Lambda_{kk} + \log U_{kj} + \log U_{ki}\right) \tag{48}$$

In this form we can work entirely in log space and exploit the log-sum-exp trick for stability.

The emission model is as follows. Each hidden state emits a normally-distributed Z-score. The means are $-M$, 0, and $+M$ for deletion, neutral, and duplication states, respectively, where M is s user-specified parameter whose default is 3. Each emission distribution is given unit variance. This model is quite wrong. Consider a duplication. The tangent-normalized coverage ought be be roughly 0.5 times the proportional coverage – the raw coverage is 3/2 that of a diploid target, leaving 1/2 remaining after (ideal) tangent-normalization. Then division by the target standard deviation to get a Z-score yields who-knows-what. Since different targets have different average proportional coverage, the global parameter M is misguided. Basically, the current model is not a model at all, but a heuristic.

V. PROPOSED METHODS

A. Using Panel of Normals for Allelic Fraction Model

The GATK ACNV allelic model learns a global distribution on allelic biases and uses it as a shared prior for the allelic biases of SNPs. While better than nothing, it would be much more powerful to use prior knowledge of the allelic bias at each SNP individually. We can learn these per-SNP biases from a panel of normals using the allelic model, but with two simplifications. First, minor allele fractions are always 1/2 since normal samples are diploid and do not exhibit subclonality. Second, we do not account for outliers; that is, we set the outlier probability $\pi = 0$. The reason for this is that the panel of normals reflects typical distributions of allelic biases and censoring data via an outlier classification could render these distributions artificially tight. If the allelic bias at some SNP site varies a lot we want to know about it. The overall likelihood is

$$\prod_j \frac{\beta^\alpha}{\Gamma(\alpha)}\lambda_j^{\alpha-1}e^{-\beta\lambda_j}\prod_{s\in\mathcal{H}_j}\frac{\lambda_j^{r_{sj}}}{(1+\lambda_j)^{n_{sj}}} \tag{49}$$

$$= \prod_j \frac{\beta^\alpha}{\Gamma(\alpha)}e^{-\beta\lambda_j}\frac{\lambda_j^{\alpha+r_{\cdot j}-1}}{(1+\lambda_j)^{n_{\cdot j}}} \tag{50}$$

where λ_j is the allelic bias ratio of SNP j (for samples sequenced and mapped using the same technology as the panel of normals), $\mathcal{H}_j$ is the set of samples in the panel of normals that are heterozygous at SNP j, $r_{\cdot j} = \sum_{s\in\mathcal{H}_j} r_{sj}$, and $n_{\cdot j} = \sum_{s\in\mathcal{H}_j} n_{sj}$. As before, the biases are assumed to come from a common distribution Gamma(α, β), but due to the large number of samples in the panel of normals the data will yield a posterior distribution on each λ_j that may be quite different from the global prior. It is these posteriors that we will use as input to ACNV. Although they are the object of interest, however, we will first marginalize them out of the likelihood in order to obtain maximum likelihood estimates of α and β. We have in fact already performed this marginalization – Equation 50 is the special case $f = 1/2$, $\pi = 0$ of the allelic-model likelihood, Equation 16, and thus its marginalization over latent variables is obtained by substituting $f = 1/2$, $\pi = 0$ into Equation 22, which yields

$$p(\alpha, \beta) = \prod_j \phi(\alpha, \beta, f = 1/2, n_{\cdot j} - r_{\cdot j}, r_{\cdot j}). \tag{51}$$

This likelihood is easily maximized numerically to obtain MLE values of α and β. Having done this, we can then approximate the posterior on each λ_j as a gamma distribution using the method of Appendix A. As shown there, the posterior on λ_j is Gamma(ρ_j, τ_j) where ρ_j and τ_j are computed in Algorithm 3, with $a \to n_{\cdot j} - r_{\cdot j}$ and $r \to r_{\cdot j}$.

Once we have the posteriors on each λ_j from the panel of normals, they are used as priors for λ_j in the ACNV allelic model. This obviates the hyperparameters α and β, and Equation 22 becomes

$$p(f,\pi) \propto \prod_j \left[\frac{1-\pi}{2}\phi(\rho_j,\tau_j,f_j,a_j,r_j) + \frac{1-\pi}{2}\phi(\rho_j,\tau_j,1-f_j,a_j,r_j) + \frac{\pi a_j! r_j!}{(n_j+1)!} \right] \tag{52}$$

where f and π may once again be sampled via adaptive Metropolis.

B. Generative Model for Read Counts

We wish to address several goals in this section:

1. Connect copy ratio (or copy number) and raw read counts in a single probabilistic model without transforming the data.
2. Take into account the Poisson nature of coverage depth, thereby giving less weight to low-coverage targets and separating the inherent variance due to Poisson statistics from experimental noise. We want to use the panel of normals to subtract only the latter.
3. Choose the number of principal components to use in an automatic and principled manner.
4. Use an algorithm that does not waste time calculating all principal components when we only want the few most significant ones.
5. Make a universal panel of normals for both sexes by taking into account targets on sex chromosomes on par with autosomal targets.
6. Make a reliable method for detecting and excluding outlier samples from the panel of normals, in particular, those with abnormal ploidy.
7. Correct for CNV events that occur in the panel of normals.

1. The model

Suppose we have vectors of read counts over a set of T targets for S samples, $\mathbf{n}_s$, $s = 1 \dots S$ where n_{st} is the coverage of sample s at target t. In order to include both sexes on an equal footing, we further define a "germline ploidy matrix" $\mathcal{P}_{st}$ such that $\mathcal{P}_{st}$ is the germline ploidy[10] of target t of sample s. We imagine that laboratory conditions for a particular sample yielding an underlying bias vector $\mathbf{b}_s$, where $e^{b_{st}}$ is the propensity of target t to be captured, sequenced, and mapped in the preparation of sample s. Suppose also that sample s has an average depth d_s and a vector of copy numbers $\mathbf{c}_s$, where the latent variable c_{st} is the copy number of sample s at target t. Our model for coverage is[11]:

$$n_{st} \sim \text{Poisson}(d_s \mathcal{P}_{st} c_{st} e^{b_{st}}) \tag{53}$$

We can achieve many of the goals listed above by performing probabilistic PCA not on directly $\mathbf{n}$, but rather on $\mathbf{b}$. One one hand, the Poisson parameters must be positive and therefore, $\exp(\mathbf{b})$ is a well-defined parametrization of the multiplicative bias. On the other hand, a Gaussian model for $\mathbf{b}$ implies a log-normal distribution for $\exp(\mathbf{b})$ which is indeed the expected distribution when the multiplicative bias arises from several independent sources according to the central limit theorem[12]. We model $\mathbf{b}$ as:

$$\begin{aligned} \mathbf{z} &\sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \\ \mathbf{b} &\sim \mathcal{N}(\mathbf{W}\mathbf{z} + \mathbf{m}, \mathbf{\Psi}), \end{aligned} \tag{54}$$

---

[10] For human autosomal targets, $\mathcal{P}_{st} = 2$ for both sexes. In female samples, $P_{st} = 2$ for X chromosome targets and $P_{st} = 0$ for Y chromosome targets. Finally, $P_{st} = 1$ for X and Y chromosomes in male samples

[11] In Equation 53 the quantity d_s is a hypothetical quantity representing average coverage in the absence of bias. Since this is impossible to know we use average coverage instead. This doesn't matter since any constant factor will be absorbed into e^b via the parameter $\mathbf{m}$.

[12] Let $B = \prod_{j=1}^{N_B} B_j$ be the total multiplicative bias where $B_j \in (0, \infty)$ are independent components of the bias. For $N_B \gg 1$, $\ln(B) \sim \mathcal{N}$ and therefore, B has a log-normal distribution.

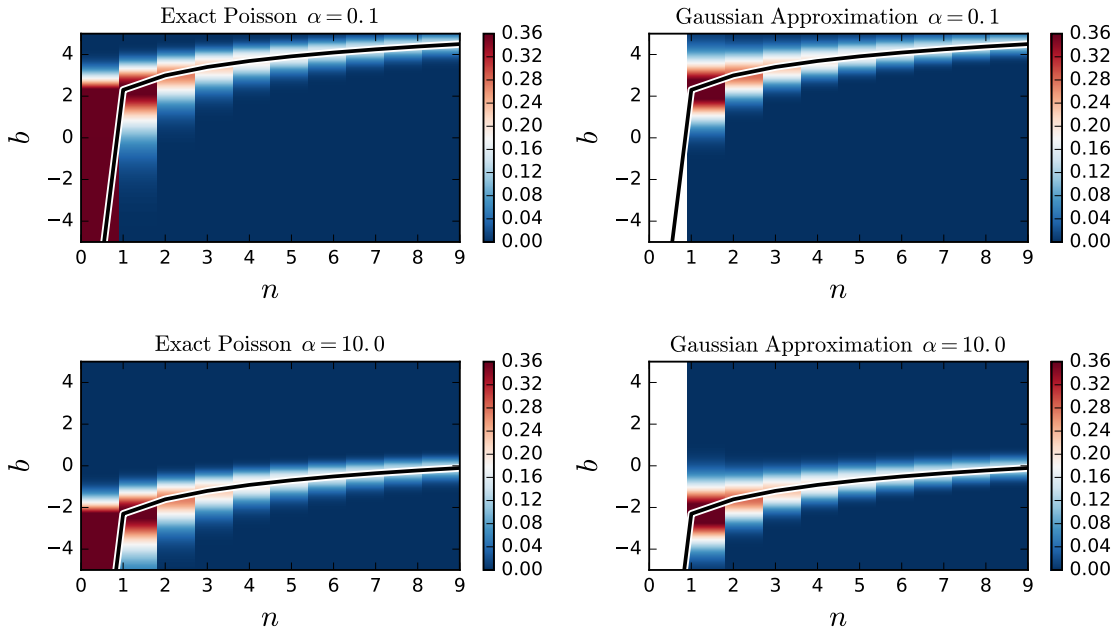

FIG. 4: Gaussian approximation to the Poisson likelihood (see Eq. 55). The left and right panels show $\text{Poisson}(n|\alpha\, e^b)$ and $n^{-1}\mathcal{N}(b|\ln(n/\alpha), n^{-1})$, respectively for $\alpha = 0.1$ (top) and $\alpha = 10.0$ (bottom). The black lines show $b = \ln(n/\alpha)$ the maximum likelihood bias estimate. The Gaussian approximation breaks down at $n = 0$ (no coverage). It also slightly overestimates the variance at small n. Otherwise, it is an excellent approximation.

where $\mathbf{z} \in \mathbb{R}^D$ is a low-dimensional latent vector of laboratory conditions, $\mathbf{W} \in \mathbb{R}^{T\times D}$ is a linear map from latent space to target space, $\mathbf{m} \in \mathbb{R}^T$ is the vector of mean biases, and $\mathbf{\Psi} \in \mathbb{R}^{T\times T}$ is a diagonal matrix of residual variance not explained by the latent features. We approximate the Poisson as a Gaussian and expand the argument of the Gaussian exponential about the mode of b_{st} to quadratic order to obtain:

$$\text{Poisson}(n_{st}|d_s\mathcal{P}_{st}c_{st}e^{b_{st}}) \simeq \Sigma_{st}\,\mathcal{N}(b_{st}|m_{st},\Sigma_{st}), \tag{55}$$

where:

$$\begin{aligned} m_{st} &= \ln(n_{st}/(\mathcal{P}_{st}c_{st}d_s)), \\ \Sigma_{st} &= 1/n_{st}. \end{aligned} \tag{56}$$

Note that Σ_{st} can be thought of as the width of the distribution of b_{st} about its maximum likelihood estimate such that in the limit $n_{st}, d_s \to \infty$, $\text{Poisson}(n_{st}|d_s c_{st} e^{b_{st}}) \to \delta(b_{st} - b^*_{st})$ where $b^*_{st} = \lim_{n,d\to\infty} m_{st}$ is the true bias. The above approximation, while being excellent for well-covered targets (see Fig. 4), inevitably breaks down for targets that are uncovered *ex ante* in some samples, such as Y chromosome targets in female samples. To this end, we define a "sample-target mask matrix" M_{st} such that $M_{st} = 0$ if $\mathcal{P}_{st} = 0$, and $M_{st} = 1$ if $\mathcal{P}_{st} \neq 0$, and for each sample-target pair (s,t), we only consider targets where the $M_{st} \neq 0$ in the joint likelihood function. The latter is thus written as:

$$P(\mathbf{n},\mathbf{b},\mathbf{z}|\mathbf{c},\mathbf{W},\mathbf{m},\mathbf{\Psi}) \propto \prod_s \mathcal{N}(\mathbf{z}_s|\mathbf{0},\mathbf{I}) \prod_{t|M_{st}\neq 0} \mathcal{N}(b_{st}|(\mathbf{W}\mathbf{z}_s)_t + m_t, \Psi_t)\,\mathcal{N}(b_{st}|m_{st},\Sigma_{st}) \tag{57}$$

We can integrate out $\mathbf{b}$ via the identity $\int_{-\infty}^{+\infty} \mathcal{N}(x|\mu_1,\sigma_1^2)\,\mathcal{N}(x|\mu_2,\sigma_2^2)\,\mathrm{d}x = \mathcal{N}(\mu_1|\mu_2,\sigma_1^2+\sigma_2^2)$ to obtain:

$$P(\mathbf{n},\mathbf{z}|\mathbf{c},\mathbf{W},\mathbf{m},\mathbf{\Psi}) \propto \prod_s \mathcal{N}(\mathbf{z}_s|\mathbf{0},\mathbf{I}) \prod_{t|M_{st}\neq 0} \mathcal{N}(m_{st}|(\mathbf{W}\mathbf{z}_s)_t + m_t, \Psi_t + \Sigma_{st}) \tag{58}$$

Save for the presence of a sample-dependent bias uncertainty Σ_{st}, our learning objective is essentially the factor analysis problem. The MLE for parameters $(\mathbf{\Psi},\mathbf{W},\mathbf{m})$ can be obtained via the EM algorithm. The presence of Σ_{st}, in particular its dependence on s, introduces undesirable complexities which we wish to avoid. To gain more insight

about the role of Σ_{st}, let us marginalizing $\mathbf{z}_s$ from Eq. (58). The final result can be put in a simple form using the Woodbury identity and properties of projection matrices:

$$P(\mathbf{n}|\mathbf{c},\mathbf{W},\mathbf{m},\mathbf{\Psi}) \propto \exp\left(-\frac{1}{2}\sum_s (\mathbf{m}-\mathbf{m}_s)^T\mathbf{M}_s(\mathbf{\Psi}+\mathbf{\Sigma}_s+\mathbf{M}_s\mathbf{W}\mathbf{W}^T\mathbf{M}_s)^{-1}\mathbf{M}_s(\mathbf{m}-\mathbf{m}_s)\right) \tag{59}$$

To simplify the discussion, let us assume that all samples share the same targets, i.e. $\mathbf{M}_s = \mathbb{I}$. First, note that $\mathbf{\Psi}+\mathbf{W}\mathbf{W}^T$ is the covariance of log-coverage due to experimental conditions, both those included explicitly in $\mathbf{W}$ and the residual ("unexplained") covariance $\mathbf{\Psi}$. The quantity $\mathbf{\Psi}+\mathbf{\Sigma}_s+\mathbf{W}\mathbf{W}^T$ is essentially a weighting factor that decreases the role of lower-coverage samples (i.e. those with larger $\mathbf{\Sigma}$) in the likelihood. If variations in $\mathbf{\Sigma}$ tend to be small compared to $\mathbf{\Psi}+\mathbf{W}\mathbf{W}^T$, then these weights will be very similar between samples and we may use the simple formula that assigns each sample the same weight. *It is an empirical fact that the noise before normalization is much greater than noise after normalization, which implies that $\mathbf{\Sigma}$ is small compared to $\mathbf{\Psi}+\mathbf{W}\mathbf{W}^T$. Variations in $\mathbf{\Sigma}$ are, of course, even smaller, especially in the typical case that all PoN samples have similar depth.* Furthermore, if the depths of PoN samples are not correlated with their positions in latent space (as is reasonable) our approximation is an unbiased estimator, because for any given position in latent space our failure to use the exact weights will on average wash out. Finally, suppose, contrary to empirical fact, that $\mathbf{\Sigma}$ tended to be large compared to $\mathbf{\Psi}+\mathbf{W}\mathbf{W}^T$. In this case, either the shared source of bias is small and we don't need this fancy model in the first place, or the depth of coverage is very poor, in which case accurate normalization is hopeless regardless. In summary, replacing Σ_{st} with a typical sample-independent value $\bar{\mathbf{\Sigma}}$ is a very reasonable approximation.

A natural choice for $\bar{\Sigma}_t$ will be made apparent soon. In the meantime, we notice that $\mathbf{\Sigma}_s$ always appears in the combination $\mathbf{\Sigma}_s+\mathbf{\Psi}$ such that the specific choice of $\bar{\Sigma}_t$ may seem to be immaterial: if $\mathbf{\Psi}^*$ is the MLE estimate obtained using the choice $\bar{\mathbf{\Sigma}}^*$, we may infer that $\mathbf{\Psi}^*+\bar{\mathbf{\Sigma}}^*-\bar{\mathbf{\Sigma}}'$ is the MLE estimate had we used $\bar{\mathbf{\Sigma}}'$ instead. However, the constraint $\Psi_t > 0$ implies that $\Psi_t^* + \bar{\Sigma}_t > 0$, such that our choice of $\bar{\Sigma}_t$ effectively sets a lower bound on the unexplained variance.

2. Estimating the mean read depth d_s

Let us define $\lambda_{st} \equiv \mathcal{P}_{st}c_{st}e^{b_{st}}$ as the total multiplicative bias per sample per target. Assuming λ_{st} is sharply peaked about its mode, the MLE for d_s can be easily found by approximating the Poisson distributions with a Gaussian distribution as discussed earlier:

$$\begin{aligned} P(\mathbf{n}_s|d_s\boldsymbol{\lambda}_s) &= \prod_{t|M_{st}\neq 0} \text{Poisson}(n_{st}|d_s\lambda_{st}) \simeq \prod_{t|M_{st}\neq 0} \mathcal{N}(n_{st}|d_s\lambda_{st}, d_s\lambda_{st}) \\ &\propto \prod_t \left[\frac{1}{\sqrt{d_s\lambda_{st}}}\exp\left(-\frac{(n_{st}-d_s\lambda_{st})^2}{2d_s\lambda_{st}}\right)\right]^{M_{st}}. \end{aligned} \tag{60}$$

Replacing λ_{st} with its mode, the log likelihood of read depth is given as:

$$\log P(\mathbf{n}_s|d_s) = -\frac{1}{2}\sum_t M_{st}\left[\log d_s + \frac{1}{d_s\lambda_{st}}(n_{st}-d_s\lambda_{st})^2\right] + \text{const.} \tag{61}$$

Maximizing yields a quadratic equation with only one acceptable root:

$$d_s = \frac{\sqrt{4\langle\!\langle \mathbf{n}^2\rangle\!\rangle_s\langle\!\langle \boldsymbol{\lambda}^2\rangle\!\rangle_s + \langle\!\langle \boldsymbol{\lambda}\rangle\!\rangle_s^2} - \langle\!\langle \boldsymbol{\lambda}\rangle\!\rangle_s}{2\langle\!\langle \boldsymbol{\lambda}^2\rangle\!\rangle_s}, \tag{62}$$

where we have defined the masked target average $\langle\!\langle\cdot\rangle\!\rangle_s$ as:

$$\langle\!\langle \mathbf{v}\rangle\!\rangle_s \equiv \frac{\sum_t M_{st}v_{st}}{\sum_t M_{st}}. \tag{63}$$

If $n_{st} \gg 1$, we may approximate the result as $d_s \simeq \left(\langle\!\langle \mathbf{n}^2\rangle\!\rangle_s / \langle\!\langle \boldsymbol{\lambda}^2\rangle\!\rangle_s\right)^{\frac{1}{2}}$. Before learning the parameters of the coverage model, we may set $\lambda_{st} \to \mathcal{P}_{st}$ to obtain a first estimate of d_s. This estimate can be updated along the way as better estimates for c_{st} and b_{st} is found.

3. EM algorithm for obtaining MLE of $(\mathbf{W}, \mathbf{\Psi}, \mathbf{m})$:

We may obtain maximum likelihood estimates of the parameters via an iterative EM approach in which we alternate between computing the Gaussian posteriors of each $\mathbf{z}_s$ (E step) and maximizing the log-likelihood with respect to $\mathbf{W}$, $\mathbf{\Sigma}$, $\mathbf{m}$, and $\mathbf{c}$ (M step). The EM algorithm for $\mathbf{W}$, $\mathbf{\Sigma}$, and $\mathbf{m}$ is similar to the factor analysis model discussed in Chapter 12 of Bishop. We note calling $\mathbf{c}$ while learning $(\mathbf{W}, \mathbf{\Psi}, \mathbf{m})$ allows us to correct for CNV events that may be present in the panel of normals[13].

The E step follows from substituting $\mathbf{\Psi} \to \mathbf{\Psi}_s \equiv \mathbf{\Psi} + \mathbf{\Sigma}_s$ in Bishop's Eqs. 12.66-67 and including the sample-target mask matrix $\mathbf{M}$. The result is:

$$\begin{aligned} \mathbf{G}_s &= \left(\mathbf{I} + \mathbf{W}^T \mathbf{M}_s \mathbf{\Psi}_s^{-1} \mathbf{W}\right)^{-1}, \\ \mathbb{E}\left[\mathbf{z}_s\right] &= \mathbf{G}_s \mathbf{W}^T \mathbf{M}_s \mathbf{\Psi}_s^{-1} (\mathbf{m}_s - \mathbf{m}), \\ \mathbb{E}\left[\mathbf{z}_s \mathbf{z}_s^T\right] &= \mathbf{G}_s + \mathbb{E}[\mathbf{z}_s]\mathbb{E}[\mathbf{z}_s]^T. \end{aligned} \tag{64}$$

In the M step, we calculate the expectation value of the complete-data log likelihood with respect to the posterior estimate of $\mathbf{z}_s$. Save for terms independent of the model parameters, the result is:

$$\mathcal{L} = -\frac{1}{2} \sum_{st} \left\{ M_{st} \ln \Psi_{st} + M_{st} \Psi_{st}^{-1} \left[\left(\mathbf{W} \mathbb{E}[\mathbf{z}_s \mathbf{z}_s^T] \mathbf{W}^T \right)_{tt} + 2(m_t - m_{st}) \left(\mathbf{W} \mathbb{E}[\mathbf{z}_s] \right)_t + (m_t - m_{st})^2 \right] \right\}, \tag{65}$$

The stationarity condition for $\mathcal{L}$ with respect to $\mathbf{m}$ gives:

$$m_t = \left(\sum_s \mathbf{M}_s \mathbf{\Psi}_s^{-1} \right)^{-1} \sum_s \left[\mathbf{M}_s \mathbf{\Psi}_s^{-1} (\mathbf{m}_s - \mathbf{W} \mathbb{E}[\mathbf{z}_s]) \right]. \tag{66}$$

The stationarity condition with respect to Ψ_t gives:

$$\sum_s M_{st} \left[\frac{1}{\Psi_t + \Sigma_{st}} - \frac{B_{st}}{(\Psi_t + \Sigma_{st})^2} \right] = 0, \tag{67}$$

where:

$$B_{st} = \left(\mathbf{W} \mathbb{E}[\mathbf{z}_s \mathbf{z}_s^T] \mathbf{W}^T \right)_{tt} + 2(m_t - m_{st}) \left(\mathbf{W} \mathbb{E}[\mathbf{z}_s] \right)_t + (m_t - m_{st})^2 \tag{68}$$

The above nonlinear equation must be solved for each target, which is a computationally demanding task for a large number of targets. If this is to be avoided, we offer two approximation schemes:

(Scheme 1) Assuming small sample to sample variations in Σ_{st}: Had Σ_{st} been constant, then Eq. (67) would have the following simple solution:

$$\Psi_t^{\text{approx}} = \langle\langle \mathbf{B} \rangle\rangle_t - \bar{\Sigma}_t, \tag{69}$$

where $\bar{\Sigma}_t$ is the sample-independent value of Σ_{st}, and we have defined the double angle bracket average as:

$$\langle\langle \mathbf{B} \rangle\rangle_t \equiv \frac{\sum_s M_{st} B_{st}}{\sum_s M_{st}}. \tag{70}$$

It is tempting to replace $\bar{\Sigma}_t$ with its M-averaged value. However, a more principled approach is to choose $\bar{\Sigma}_t$ such that the approximation solution given in (70) is as close to the exact solution as possible. To this end, we assume $\Sigma_{st} = \bar{\Sigma}_t + (\Sigma_{st} - \bar{\Sigma}_t)$ such that $|\Sigma_{st} - \bar{\Sigma}_t| \ll \bar{\Sigma}_t$, expand Eq. (67) in $\Sigma_{st} - \bar{\Sigma}_t$ to linear order, and require that Ψ_t^{approx} represents the exact solution. This procedure yields:

$$\bar{\Sigma}_t = \frac{2\langle\langle \mathbf{B}\mathbf{\Sigma} \rangle\rangle_t - \langle\langle \mathbf{\Sigma} \rangle\rangle_t \langle\langle \mathbf{B} \rangle\rangle_t}{\langle\langle \mathbf{B} \rangle\rangle_t}. \tag{71}$$

---

[13] If our copy-ratio calling algorithm is a maximum-likelihood method, for example calling via the Viterbi algorithm for an HMM, then provided that our likelihood is averaged with respect to the E step for $\mathbf{z}$ the entire process including optimization of $(\mathbf{c}, \mathbf{W}, \mathbf{\Psi}, \mathbf{m})$ is an EM algorithm and thus will increase likelihood at each iteration and converge.

Plugging this result back in Eq. (70), we find:

$$\Psi_t^{\text{approx}} = \langle\!\langle \mathbf{B} \rangle\!\rangle_t + \langle\!\langle \mathbf{\Sigma} \rangle\!\rangle_t - 2\,\frac{\langle\!\langle \mathbf{B\Sigma} \rangle\!\rangle_t}{\langle\!\langle \mathbf{B} \rangle\!\rangle_t}. \tag{72}$$

This solution is accurate to linear order in deviations of Σ_{st} about $\bar{\Sigma}_t$.

(Scheme 2) Newton iterations: The complexity of solving Eq. (67) numerically is not too high given that the Hessian matrix is diagonal. Expanding $\mathcal{L}$ about $\mathbf{\Psi}_0$, we find:

$$\mathcal{L}(\mathbf{\Psi}) = \mathcal{L}(\mathbf{\Psi}_0) + \alpha_t\,(\Psi_t - \Psi_{0,t}) + \frac{1}{2}\,\beta_t\,(\Psi_t - \Psi_{t,0})^2 + \ldots \tag{73}$$

where:

$$\begin{aligned}
\alpha_t &= \frac{\partial \mathcal{L}}{\partial \Psi_t} = -\frac{1}{2}\sum_s M_{st}\left[\frac{1}{\Psi_t + \Sigma_{st}} - \frac{B_{st}}{(\Psi_t + \Sigma_{st})^2}\right], \\
\beta_t &= \frac{\partial^2 \mathcal{L}}{\partial \Psi_t^2} = +\frac{1}{2}\sum_s M_{st}\left[\frac{1}{(\Psi_t + \Sigma_{st})^2} - \frac{2B_{st}}{(\Psi_t + \Sigma_{st})^3}\right].
\end{aligned} \tag{74}$$

The Newton's approximate solution is therefore:

$$\Psi_{t,1} = \Psi_{t,0} - \frac{\alpha_t(\mathbf{\Psi}_0)}{\beta_t(\mathbf{\Psi}_0)}. \tag{75}$$

One may start iterations using the result of Scheme 1 as the initial guess and continue until convergence.

Remark: In practice, when sample-to-sample variance of Σ was large (e.g. read depths varying from 50 to 1000 randomly), we noticed that the best approach was to use Brent root finding for each target. On average, 10 function calls yields the solution within a 10^{-6} tolerance. Newton's method required approximately 20 evaluations of α_t and β_t to converge within the same tolerance. Also, we found that the most robust scheme was to start from $\Psi_t = 0$ rather than using Eq. (72).

In the M step equation for $\mathbf{W}$, we may incorporate an automatic relevance determination (ARD) prior:

$$P(\mathbf{W}) = \prod_\mu \left(\frac{\alpha_\mu}{2\pi}\right)^{T/2} \exp\left(-\frac{1}{2}\alpha_\mu \sum_t W_{t\mu}^2\right). \tag{76}$$

If $\alpha_\mu \to \infty$, the latent feature μ is effectively turned off. Thus we can initially choose a liberal estimate of D and the model will automatically become more parsimonious. The M step log likelihood times the ARD prior depend on the t-th row of $\mathbf{W}$ as:

$$-\frac{1}{2}\left(-\sum_\mu \ln \alpha_\mu + \sum_{\mu\nu} W_{t\mu}\,(A_{\mu\nu} + Q_{t\mu\nu})\,W_{t\nu} - 2\sum_\mu W_{t\mu} v_{t\mu}\right), \tag{77}$$

where $\mathbf{A} \equiv \text{diag}(\alpha_1, \alpha_2 \ldots \alpha_D)$ and we have defined:

$$Q_{t\mu\nu} = \sum_s M_{st}\Psi_{st}^{-1}\,\mathbb{E}\left[z_{s\mu} z_{s\nu}\right], \qquad v_{t\mu} = \sum_s M_{st}\Psi_{st}^{-1}(m_{st} - m_t)\,\mathbb{E}[v_{s\mu}]. \tag{78}$$

The maximum a posteriori result for $W_{t\mu}$ is:

$$W_{t\mu} = \sum_\nu \left(\mathbf{A} + \mathbf{Q}_t\right)^{-1}_{\mu\nu}\, v_{t\nu}. \tag{79}$$

In the approximation $\mathbf{\Sigma}_s \to \bar{\mathbf{\Sigma}}$, this formula is unchanged but $\mathbf{Q}_t$ is S times as fast to calculate. The other M steps and the E step are not affected by the ARD prior. To determine α_μ, we re-exponentiate expression (77)[14] and integrate

[14] This is the distribution on $\mathbf{W}$ that we would obtain from a mean-field variational factorization $q(\mathbf{z})q(\mathbf{W})$.

out $\mathbf{W}$ to obtain the evidence for $\mathbf{A}$:

$$P(\mathbf{n}|\mathbf{A}) \propto \prod_k \alpha_k^{T/2} \prod_t \int q(\mathbf{W}|t) \prod_\mu \mathrm{d}W_{t\mu}, \tag{80}$$

where:

$$q(\mathbf{W}|t) \equiv \exp\left(-\frac{1}{2}\sum_{\mu\nu} W_{t\mu}(A_{\mu\nu} + Q_{t\mu\nu})W_{t\nu} - \sum_\mu W_{t\mu} v_{t\mu}\right). \tag{81}$$

The ARD coefficients α are determined by maximizing the log evidence. That is, we set

$$\frac{\partial}{\partial \alpha_k} \ln P(\mathbf{n}|\mathbf{A}) = 0 \Rightarrow \frac{1}{2}\left(\frac{T}{\alpha_\mu} - \sum_t \left\langle W_{t\mu}^2 \right\rangle\right) = 0 \Rightarrow \alpha_\mu = \frac{T}{\sum_t \left\langle W_{t\mu}^2 \right\rangle}, \tag{82}$$

where the average $\left\langle W_{t\mu}^2 \right\rangle$ is taken with respect to the density $q(\mathbf{W}|t)$. Completing the square, we find that $q(\mathbf{W}_{t\cdot})$ is Gaussian with covariance $(\mathbf{A} + \mathbf{Q}_t)^{-1}$ and mean $(\mathbf{A} + \mathbf{Q}_t)^{-1}\mathbf{v}_t$. Note that this mean is precisely the M step value for $q(\mathbf{W}|t)$, as we would hope! Thus we get:

$$\left\langle W_{t\mu}^2 \right\rangle = W_{t\mu}^2 + (\mathbf{A} + \mathbf{Q}_t)^{-1}_{\mu\mu} \tag{83}$$

Let us now summarize these steps:

- E step: $\mathbf{G}_s = \left(\mathbf{I} + \mathbf{W}^T \mathbf{M}_s \mathbf{\Psi}_s^{-1} \mathbf{W}\right)^{-1}$, $\mathbb{E}\left[\mathbf{z}_s\right] = \mathbf{G}_s \mathbf{W}^T \mathbf{\Psi}_s^{-1}(\mathbf{m}_s - \mathbf{m})$, $\mathbb{E}\left[\mathbf{z}_s \mathbf{z}_s^T\right] = \mathbf{G}_s + \mathbb{E}[\mathbf{z}_s]\mathbb{E}[\mathbf{z}_s]^T$. $\mathbf{G}_s$ and all the $\mathbb{E}[\mathbf{z}_s]$ are each $O(D^2TS)$. $\mathbb{E}[\mathbf{z}_s \mathbf{z}_s^T]$ is $O(D^2S)$. The E step overall is $O(D^2TS)$.
- $\mathbf{m} = \left(\sum_s \mathbf{M}_s \mathbf{\Psi}_s^{-1}\right)^{-1} \sum_s \left[\mathbf{M}_s \mathbf{\Psi}_s^{-1}(\mathbf{m}_s - \mathbf{W}E[\mathbf{z}_s])\right]$ is $O(DTS)$.
- $B_{st} = \left(\mathbf{W}\mathbb{E}[\mathbf{z}_s \mathbf{z}_s^T]\mathbf{W}^T\right)_{tt} + 2(m_t - m_{st})\left(\mathbf{W}\mathbb{E}[\mathbf{z}_s]\right)_t + (m_t - m_{st})^2$ is $O(D^2TS)$
- Solving Eq. (67) for each t is $O(TS)$ with a prefactor equal to the number of evaluations required to find a root (approximately $10 \sim 20$). As long as this number is less than D^2 this step is subleading.
- $\mathbf{Q}_t = \sum_s M_{st} \Psi_{st}^{-1} \mathbb{E}[\mathbf{z}_s \mathbf{z}_s^T]$ is $O(D^2TS)$.
- $\mathbf{v}_t = \sum_s M_{st} \Psi_{st}^{-1}(m_{st} - m_t) E[\mathbf{z}_s]$ is $O(DTS)$.
- $\mathbf{W}_{t\cdot} = W_{t\mu} = \sum_\nu (\mathbf{A} + \mathbf{Q}_t)^{-1}_{\mu\nu}\, v_{t\nu}$ is $O(D^3T)$.
- $\left\langle W_{t\mu}^2 \right\rangle = W_{t\mu}^2 + (\mathbf{A} + \mathbf{Q}_t)^{-1}_{\mu\mu}$ is $O(D^3T)$.
- $\alpha_\mu = T / \sum_t \left\langle W_{t\mu}^2 \right\rangle$ is $O(DT)$.

The leading cost is a few terms of $O(D^2TS)$ flops, each with small prefactors, per iteration. Assuming a total prefactor of 10 and $T = 2 \times 10^5$, $D = 10$, $S = 500$ a full EM iteration costs 10^{11} flops in exact mode. On a single 1 GHz core (10^9 flops per second) this comes out to roughly 100 or 10 seconds.

We can apply the parameters learned from the panel of normals to single-sample calling, which requires the likelihood as a function of the copy numbers $\mathbf{c}$. Applying the same E step as above, the likelihood is

$$P(\mathbf{n}|\mathbf{c}, \mathbf{W}, \mathbf{m}, \mathbf{\Psi}) \propto \prod_t \exp\left[-\frac{1}{2} M_{st} \Psi_{st}^{-1}\left(\ln\left(n_{st}/(d_s \mathcal{P}_{st})\right) - \ln(c_{st}) - m_t - \left(\mathbf{W}\mathbb{E}[\mathbf{z}_s]\right)_t\right)^2\right], \tag{84}$$

We have only kept factors that depends on $\mathbf{c}$ in the above likelihood. Note that this is factorized into independent likelihood terms for each target and is thus suitable for the emission model of an HMM. This likelihood is not Gaussian in $\mathbf{c}$, but it does not need to be for the Viterbi and forward-backward algorithms. Also, note that when $c_{st} = 0$ is the most likely solution, this must be incorporated in the mask matrix M_{st} in order to avoid ambiguous expressions due to the breakdown of the Laplace approximation used to replace the Poisson with a Gaussian.

4. GC bias correction

We can easily integrate sample-specific GC bias into this model. Let $f_s(\mathrm{GC})$ be the GC bias of GC content GC for sample s. Then this enters into the model as an additional multiplier to the Poisson parameter. That is, we replace $d_s c_{st} \to d_s c_{st} f_s(\mathrm{GC}_t)$, which affects the model learning and inference only via the definition of m_{st}. We can iteratively re-estimate the GC bias function f_s by regressing the bias not explained by the latent factors. That is, for each target the Poisson parameter is, ignoring GC effects,

$$c_{st} d_s \exp\left(\mathbf{W}_t \mathbb{E}[\mathbf{z}_s] + m_t\right) \tag{85}$$

and thus the ratio $n_{st} / \left[c_{st} d_s \exp\left(\mathbf{W}_t \mathbb{E}[\mathbf{z}_s] + m_t\right)\right]$ (with error bars of size $1/\sqrt{n}$ if we want to do a weighted regression) is an estimate of $f_s(\mathrm{GC}_t)$ that we can feed into our favorite regression model. This is more sophisticated than the standard approach of simply regressing n versus GC in that it seeks to explain with GC only the bias that cannot be explained by linear latent features.

5. PCA and the curse of small samples

PCA-like denoising approaches minimize the *total variance* by learning and subtracting the contribution of the underlying latent features from the data. In practice, this objective is achieved using either the maximum variance principle (usual PCA) or the maximum likelihood principle on a linear-Gaussian model as explained earlier. In either method, when the number of samples largely exceeds the dimension of the latent space, sample-specific variations become immaterial and the true underlying latent features can be learned from the data. On the other hand, when the number of samples is comparable to the number of latent features, the statistical power for separating sample-specific variations from mutual variations is significantly reduced.

Let us assume that we have an oracle for the first few major latent features, and that we have already subtracted the contribution arising from these features. Let σ_ℓ^2 be the variance associated with the next leading latent feature. Subtracting this latent feature reduces the total variance by $S\sigma_\ell^2$, where S is the number of samples. Now, if one of the samples has an individual leftover variance of magnitude σ_s^2 such that $\sigma_s^2 \gtrsim S\sigma_\ell^2$, then the maximum variance principle implies choosing the next principal component along the direction of that specific sample. In other words, the procedure erroneously learns a sample-specific signal as a source of noise. Note that this artifact occurs only if $S \lesssim \sigma_s^2/\sigma_\ell^2$.

Why does it matter? — We have no theoretical guarantee that the MLE problem for $(\mathbf{W}, \mathbf{m}, \mathbf{\Psi}, \mathbf{c})$ is convex. In all likelihood, if one of the samples has a large germline CNV event, it may be picked up as a principal component and be interpreted as experimental noise such that the MLE for $\mathbf{c}$ fails to call that CNV event. It is possible that the likelihood function has numerous such local maxima. Therefore, we wish to ensure that sample-specific *nuisances* are not picked up as Gaussian noise, no matter how strong they are. We discuss a number of such approaches in what follows.

(Idea 1) Blind source separation — One remedy is to use a blind source separation approach, such as independent component analysis (ICA), to separate the signal from the noise as a first step, followed by learning the latent features of the noise using PCA. In ICA-like methods, one decomposes the signal into additive subcomponents and minimizes the mutual information between them (or maximizes the non-Gaussianity by taking into account higher moments such as the kurtosis). Even though this method is quite appealing, we follow a more context-specific heuristic approach here.

(Idea 2) Imposing context-specific constraints on the structure of variations — Fortunately, we have some idea about the spatial structure of the CNV events: they are amplification or attenuation of the read count over several consecutive targets. In the absence of noise, we expect the frequency spectrum of the CNV signal (as obtained by taking a Fourier transform of m_{st} in t) to be significantly enhanced at spatial frequencies corresponding to the inverse length scale of the size of the CNV event. Similarly, the variation subtracted from sample s, i.e. $\mathbf{W}\mathbf{z}_s$, will exhibit an enhanced spectral power if a CNV event is erroneously picked up. Let $\tilde{f}(k)$ be the Fourier transform of a linear filter that approximately represents a range of CNV events. For example, we may use a midpass filter such as:

$$\tilde{f}(k) = \begin{cases} 1 & k_l \le k^* \le k_h, \\ 0 & k^* > k_h \text{ or } k^* < k_l, \end{cases} \tag{86}$$

Here, $k^* = \min(k, T-1-k)$, $k_l \sim \lfloor T/\ell_{\max} \rfloor$ and $k_l \sim \lfloor T/\ell_{\min} \rfloor$, where $\ell_{\min}$ and $\ell_{\max}$ denote roughly the minimum and maximum length of the CNV events in the units of targets. The filtered spectral power of the noise in sample s

is given as:

$$\kappa_s \equiv \sum_{k=0}^{T-1} \tilde{f}(k) \left|\mathrm{FFT}\left[\mathbf{W}\mathbf{z}_s\right]_k\right|^2 = \frac{1}{T}\sum_{t,t'=0}^{T-1} F_{tt'}\, W_{t\mu}\, W_{t'\nu}\, z_{s\mu}\, z_{s\nu}, \tag{87}$$

where $F_{tt'} = f(t-t') \equiv T^{-1}\sum_{k=0}^{T-1} e^{2\pi i k(t-t')/T}\tilde{f}(k)$ is the inverse DFT of $\tilde{f}(k)$. Now, in order to avoid picking up event-like variations as noise, we simply penalize variations with large κ. To this end, we regularize the coverage likelihood function Eq. (58) with the following multiplicative factor:

$$R_f \equiv \exp\left(-\frac{\lambda}{2}\sum_{s=1}^{S}\kappa_s\right) = \exp\left(-\frac{\lambda}{T}\sum_{s=1}^{S}\sum_{t,t'=1}^{T-1} f(t-t')\, W_{t\mu}\, W_{t'\nu}\, z_{s\mu}\, z_{s\nu}\right). \tag{88}$$

We will discuss a proper choice of λ later. Since this regularizer is quadratic in z, the Gaussian structure of the likelihood is preserved and the E step remains as simple as before. The only difference is the presence of an additional term in $\mathbf{G}_s^{-1}$:

$$\mathbf{G}_s^{-1} \to \mathbf{I} + \mathbf{W}^T\left[\mathbf{M}_s\mathbf{\Psi}_s^{-1} + \lambda\mathbf{F}\right]\mathbf{W}. \tag{89}$$

Since $\mathbf{F}$ is not diagonal in the target space, a naive matrix multiplication implies a multiplication complexity of $\mathcal{O}(D^2T^2)$ for the new term. However, this complexity can be reduced to a manageable $\mathcal{O}(D^2T\log T)$ using FFT:

$$(\mathbf{W}^T\mathbf{F}\mathbf{W})_{\mu\nu} = \sum_{t=0}^{T-1} W_{\mu t}\,\mathrm{FFT}_t^{-1}\left[\sum_{k=0}^{T-1}\tilde{f}(k)\,\mathrm{FFT}_k[W_{t\nu}]\right]. \tag{90}$$

The M step equations for $\mathbf{\Psi}$ and $\mathbf{M}$ remain the same. For $\mathbf{W}$, we find:

$$\sum_{\nu} Q_{t\mu\nu}W_{t\nu} + \lambda\sum_{\nu,t'} Z_{\mu\nu}F_{tt'}W_{t'\nu} = v_{t\mu}, \tag{91}$$

where $\mathbf{Q}$ and $\mathbf{v}$ are defined as before and:

$$\mathbf{Z} = \sum_{s=1}^{S}\mathbb{E}[\mathbf{z}_s\mathbf{z}_s^T]. \tag{92}$$

As one would expect, the regularizer mixes different targets such that t is no longer a mere label in the stationarity condition for $\mathbf{W}$. The direct solution to Eq. (91) is impractical since it involves inverting a matrix of size $DT \times DT$.

Fortunately, the linear operator in question, $\mathbf{Q} + \lambda\mathbf{Z}\otimes\mathbf{F}$, is the sum of two sparse operators: $\mathbf{Q}$ is diagonal in the target space, and $\mathbf{A} \equiv \mathbf{Z}\otimes\mathbf{F}$ is diagonal in Fourier space ($\mathbf{Z}$ acts on the latent space, $\mathbf{F}$ acts on the target space). Both $\mathbf{Q}$ and $\mathbf{Z}\otimes\mathbf{F}$ are dense in the latent space, but this space has a low dimensionality and is not prohibitive in numerics. Eq. (91) can be solved very efficiently using preconditioned iterative Krylov space solvers such as conjugate gradients or GMRES. A decent preconditioner for $\mathbf{A}$ can be constructed by taking a target average of $Q_{t\mu\nu}$:

$$\tilde{\mathbf{A}} \equiv \tilde{\mathbf{Q}} + \lambda\mathbf{Z}\otimes\mathbf{F}, \qquad \tilde{\mathbf{Q}} = \frac{1}{T}\sum_t \mathbf{Q}_t. \tag{93}$$

Note that $\tilde{\mathbf{A}}$ is now easily invertible in the Fourier space. In iterative methods, we only need to be able to calculate $\tilde{\mathbf{A}}^{-1}\mathbf{W}$ for arbitrary $\mathbf{W}$. The complexity for this is $\mathcal{O}(D^3T\log T)$ using FFT:

$$\left(\tilde{\mathbf{A}}^{-1}\mathbf{W}\right)_{t\mu} = \mathrm{FFT}_t^{-1}\left[\left(\tilde{\mathbf{Q}} + \lambda\tilde{f}(k)\mathbf{Z}\right)^{-1}\mathrm{FFT}_k[\mathbf{W}_t]\right]. \tag{94}$$

Note that if target-to-target variance $\mathbf{Q}_t$ is small (which is the case if the targets have a comparable degree of unexplained variance), $\tilde{\mathbf{A}}^{-1}\mathbf{v}$ is an excellent approximate solution to Eq. (91) and can be used as a starting point. In practice, we found preconditioned CG iterations to converge within an error tolerance of 10^{-6} within less than 10 steps. The complexity of each CG step is also $\mathcal{O}(D^3T\log T)$.

Choice of λ — The regularizer "kicks in" when $\lambda \sim \Psi^{-1}$, as it can be inferred directly from Eq. (91). One may initially choose $\lambda \sim 1000\,\Psi^{-1}$ and progressively relax it as CNV calls stabilize.

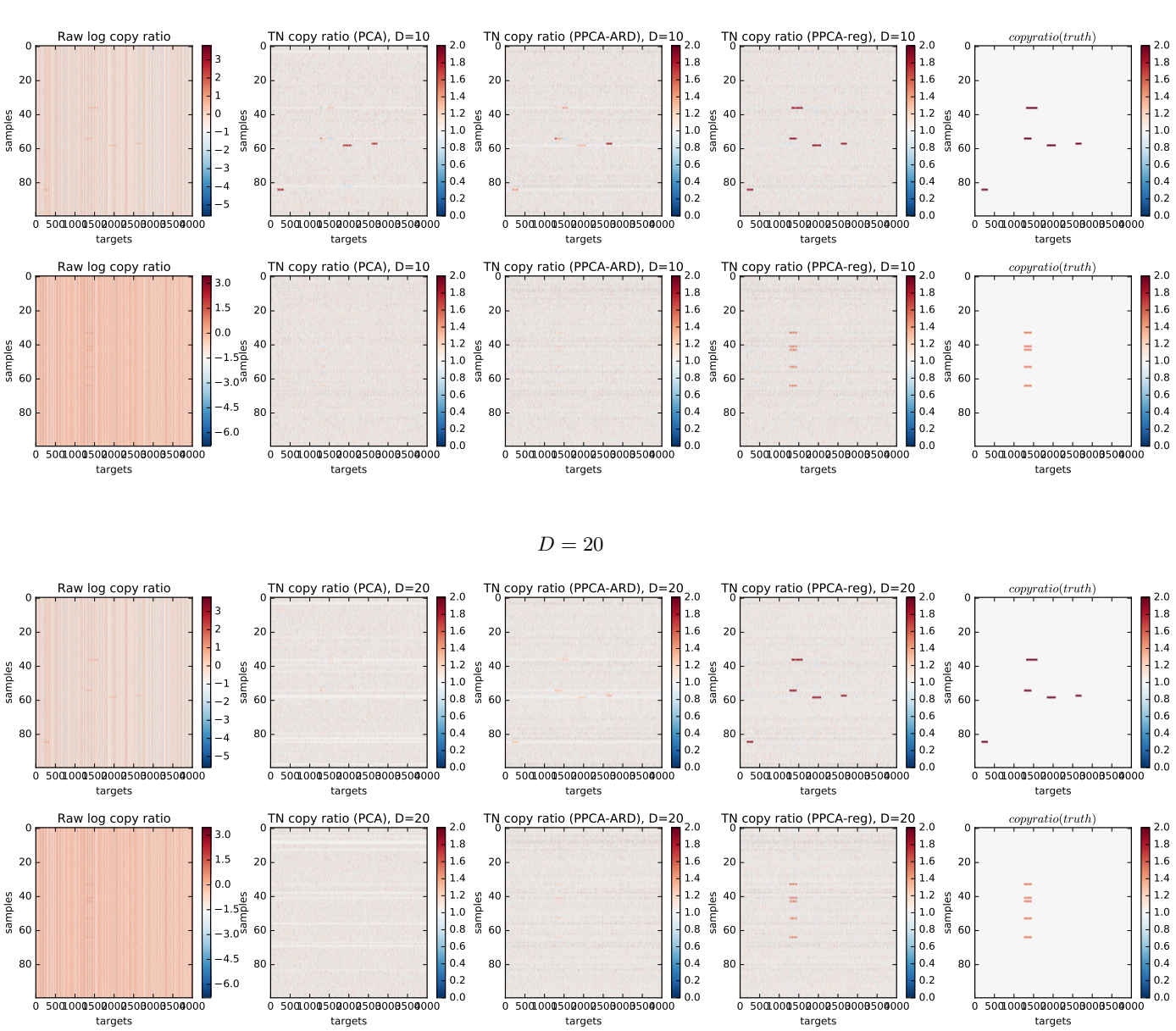

FIG. 5: Comparison of PCA with the probabilistic coverage model in different modes. Top two rows: $D = 10$; random events, correlated events. Bottom two rows: $D = 20$; random events, correlated events.

6. *Results*

In this section, we present the result of the algorithm on synthetic coverage data where the ground truth is known (this section must be eventually supplemented with real data). We synthesize the data according to Eq. (53) along with random duplication events of varying lengths. We choose $T = 4000$ targets, $D = 10$ true latent variables, $S = 100$ samples, mean read depth d uniformly sampled from $[50, 1000]$, mean bias $m_t \sim \mathcal{N}(0, 1)$, eigenvalues of the covariance matrix $\mathbf{W}\mathbf{W}^T$ uniformly sampled from $[0, 10]$, and residual variance Ψ_t uniformly sampled from $[0.01, 0.05]$. Finally, the length of CNV events are randomly sampled from $[50, 500]$ targets.

Figs. **??** and **??** compares PCA denoising against our probabilistic model with different features turned on/off (ARD, CNV event regularization) for random and correlated events, respectively. It is clearly observed that the regularized model retains all of the events even when the number of latent features chosen is greater than the true number.

C. Inferring sex genotype from raw read counts

The probabilistic target coverage model requires the germline copy number of the targets as an input for dealing with cohorts composed of samples with different sexes. If the phenotypic sex of the human subject associated to a sample is known, the nominal germline ploidy of each target is readily known. Unfortunately, the gender annotation is not available for most samples. In this section, we propose a simple statistical test for inferring the sex associated to a sample from human subjects from raw read counts on X and Y chromosomes.

Let T_X and T_Y be a set of targets on X and Y chromosomes. Let $\{n_t\}$ be the raw read counts on sex chromosome targets. The likelihood of reads for either of XX or XY genotypes can be calculated from Eq. (53). Since we have no prior knowledge of the germline CNV events[15], we proceed by assuming $c_{st} = 1$. We have no prior knowledge about the multiplicative bias e^{b_t} either. However, it is reasonable to assume that a major source of the mean target bias is the target length l_t. In particular, if the reads are corrected for the GC bias, we expect a strong $e^{b_t} \propto l_t$ proportionality. Finally, the average read depth, d, can be reliably estimated from the read counts on autosomal targets (see Sec. V B 2). Put together, for the XX genotype we have:

$$\text{(XX genotype)} \qquad n_t \sim \begin{cases} \text{Poisson}(2\,\rho\,l_t) & t \in T_X, \\ \text{Poisson}(\varepsilon_M\,l_t) & t \in T_Y, \end{cases} \tag{95}$$

where ρ is *average read count per base per strand*, l_t is the target length (number of overlapping bases), and $\varepsilon_M \sim 10^{-5}$ is the (small) probability of a mapping error that may result in a (small) number of reads to be mapped to the Y chromosome for an XX genotype sample. Similarly, for the YY genotype we have:

$$\text{(XX genotype)} \qquad n_t \sim \text{Poisson}(\rho\,l_t) \qquad t \in T_X \cup T_Y. \tag{96}$$

At this point, we may either proceed with a likelihood model for individual reads, or for compound reads. We discuss both cases below.

Individual reads likelihood: using the previous two equations, and assuming independent reads on different targets, we have:

$$\begin{aligned} P(\mathbf{n}|\text{XX}) &= \prod_{t\in T_X} \text{Poisson}(n_t|2\,\rho\,l_t) \prod_{t\in T_X} \text{Poisson}(n_t|\varepsilon_M\,\rho\,l_t\,e^{b_t}), \\ P(\mathbf{n}|\text{XY}) &= \prod_{t\in T_X\cup T_Y} \text{Poisson}(n_t|\rho\,l_t). \end{aligned} \tag{97}$$

Compound reads likelihood: we define compounds reads on X and Y chromosomes as $N_X n_t$ and $N_Y n_t$, respectively. Both of these compound random variables have a Poisson distribution with the sum of the Poisson parameter of the underlying reads. The compounds likelihoods are given as:

$$\begin{aligned} P(N_X, N_Y|\text{XX}) &= \text{Poisson}(N_X|2\rho L_X) \times \text{Poisson}(N_Y|\varepsilon_M \rho N_Y), \\ P(N_X, N_Y|\text{XY}) &= \text{Poisson}(N_X|\rho L_X) \times \text{Poisson}(N_Y|\rho L_Y). \end{aligned} \tag{98}$$

We have defined $L_{X(Y)} \equiv \sum_{t\in T_{X(Y)}} l_t$ as the total length of targets in X and Y chromosomes.

Having a likelihood function, the logarithmic odds of XX to XY genotype is readily found from the Bayes theorem:

$$\log \text{odds}(\text{XX}/\text{XY}) = \log P(\mathbf{n}|\text{XX}) + \log P(\text{XX}) - \log P(\mathbf{n}|\text{XY}) - \log P(\text{XY}). \tag{99}$$

If the compounds reads likelihood model is used, the compound likelihood functions must be accordingly:

$$\log \text{odds}(\text{XX}/\text{XY}) = \log P(N_X, N_Y|\text{XX}) + \log P(\text{XX}) - \log P(N_X, N_Y|\text{XY}) - \log P(\text{XY}). \tag{100}$$

---

[15] The germline CNV events will be known only after training the target coverage model. The latter, however, requires the sex genotype as an input.

Appendix A: Marginalizing out latent variables of the allelic model

We wish to evaluate

$$\phi(\alpha, \beta, f, a, r) = \int_0^\infty g(\lambda, \alpha, \beta, f, a, r)\, d\lambda \tag{A1}$$

where

$$g(\lambda, \alpha, \beta, f, a, r) = \frac{\beta^\alpha}{\Gamma(\alpha)} \frac{f_j^a (1-f)^r \lambda^{\alpha+r-1} e^{-\beta\lambda}}{(f + (1-f)\lambda)^{a+r}} \tag{A2}$$

An extremely good approximation for all values of f, α, β, and a, r is

$$g(\lambda, \alpha, \beta, f, a, r) \approx c\lambda^{\rho-1} e^{-\tau\lambda}. \tag{A3}$$

where ρ and τ are chosen to reproduce the mode of $g(\lambda, \alpha, \beta, f, a, r)$ and the curvature at its mode. Having approximated our integrand as a gamma distribution's pdf on λ, we integrate it analytically

$$\phi(\alpha, \beta, f, a, r) = c \int_0^\infty \lambda^{\rho-1} e^{-\tau\lambda}\, d\lambda = c\frac{\Gamma(\rho)}{\tau^\rho} \tag{A4}$$

The mode λ_0 is found by setting logarithmic derivatives to zero:

$$\frac{d}{d\lambda} \left[(\alpha + r - 1)\ln\lambda - \beta\lambda - n\ln\left(f + (1-f)\lambda\right)\right]_{\lambda_0} = 0 \tag{A5}$$

$$\frac{\alpha + r - 1}{\lambda_0} - \beta - \frac{n(1-f)}{f_j + (1-f_j)\lambda_0} = 0 \tag{A6}$$

Multiplying out the denominators yields a quadratic equation. Taking the positive root gives

$$\lambda_0 = \frac{\sqrt{w^2 + 4\beta f(1-f)(r+\alpha-1)} - w}{2\beta(1-f)}, \quad w = (1-f)(a-\alpha+1) + \beta f. \tag{A7}$$

The second derivative of $\ln f$ at λ_0 is

$$\kappa = -\frac{r + \alpha - 1}{\lambda_0^2} + \frac{n(1-f)^2}{\left(f + (1-f)\lambda_0\right)^2} \tag{A8}$$

The mode of the approximating gamma distribution is $(\rho-1)/\tau$ and the log second derivative is $-(\rho-1)/\lambda_0^2$. Equating these, we obtain

$$\rho = 1 - \kappa\lambda_0^2, \quad \tau = -\kappa\lambda_0 \tag{A9}$$

Finally, we choose c so that the values of $\ln f$ and the approximation match at λ_0:

$$\ln c = \alpha\ln\beta - \ln\Gamma(\alpha) + a\ln f + r\ln(1-f) + (r + \alpha - \rho)\ln\lambda_0 + (\tau - \beta)\lambda_0 - n\ln\left(f + (1-f)\lambda_0\right) \tag{A10}$$

Algorithm 3 shows the entire computation.

Algorithm 3 Calculating $\phi(\alpha, \beta, f, a, r)$

$n = a + r$
$w = (1 - f)(a - \alpha + 1) + \beta f$
$\lambda_0 = \left(\sqrt{w^2 + 4\beta f(1-f)(r + \alpha - 1}} - w\right) / \left(2\beta(1-f)\right)$
$\kappa = \left(n(1-f)^2\right) / \left(f + (1-f)\lambda_0\right)^2 - (r + \alpha - 1) / \lambda_0^2$
$\rho = 1 - \kappa\lambda_0^2$
$\tau = -\kappa\lambda_0$
$\ln c = \alpha \ln \beta - \ln \Gamma(\alpha) + a \ln f + r \ln(1-f) + (r + \alpha - \rho) \ln \lambda_0 + (\tau - \beta)\lambda_0 - n \ln\left(f + (1-f)\lambda_0\right)$
return $c\Gamma(\rho)/\tau^\rho$