HMMploidy: inference of ploidy levels from short-read sequencing data

Samuele Soraggi[1,2][0000-0002-1159-5535], Johanna Rhodes[3][0000-0002-1338-7860], Isin Altinkaya[2,4,5][0000-0002-6364-3332], Oliver Tarrant[2], François Balloux[6][0000-0003-1978-7715], Matthew C. Fisher[3][0000-0002-1862-6402], and Matteo Fumagalli[2][0000-0002-4084-2953]

[1] Bioinformatics Research Center (BiRC), University of Aarhus, 8000 Aarhus, Denmark, samuele@birc.au.dk

[2] Department of Life Sciences Silwood Park, Imperial College London, Ascot, SL5 7PY, UK, m.fumagalli@imperial.ac.uk

[3] MRC Centre for Global Infectious Disease Analysis, Department of Infectious Disease Epidemiology, Imperial College London, London, W2 1PG, UK

[4] Department of Biology, Hacettepe University, 06800 Beytepe Campus, Ankara, Turkey

[5] GLOBE, Section for Geogenetics, Øster Voldgade 5-7, 1350, Copenhagen, Denmark

[6] UCL Genetics Institute, University College London, London, WC1E 6BT, UK

Abstract. The inference of ploidy levels from genomic data is important to understand molecular mechanisms underpinning genome evolution. However, current methods based on allele frequency and sequencing depth variation do not have power to infer ploidy levels at low- and mid-depth sequencing data, as they do not account for data uncertainty. Here we introduce `HMMploidy`, a novel tool that leverages the information from multiple samples and combines the information from sequencing depth and genotype likelihoods. We demonstrate that `HMMploidy` outperforms existing methods in most tested scenarios, especially at low-depth with large sample size. `HMMploidy` further allows for local inferences of ploidy change to detect within-chromosome variations. We apply `HMMploidy` to sequencing data from the pathogenic fungus *Cryptococcus neoformans* and retrieve pervasive patterns of polyploidy and aneuploidy, even when artificially downsampling the sequencing data. We envisage that `HMMploidy` will have wide applicability to low-depth sequencing data from polyploid and aneuploid species.

Keywords: high-throughput DNA sequencing · ploidy · poliploidy · aneuploidy · hidden Markov model · genotype likelihood

1 Introduction

Despite the significant methodological advances for processing high-throughput DNA sequencing data we are experiencing nowadays, most of these efforts have been focused towards model species with predefined characteristics. Specifically, there has been a lack of research into modelling sequencing data from non-diploid species or organisms with unknown ploidy. Polyploidy (i.e. ploidy greater than two) is often the consequence of hybridisation or whole genome duplication, and has a significant role in the evolution and speciation of plants (20). Moreover, 34.5% of vascular plants (including leading commercial crop species) are shown to be polyploid (26). Aneuploidy (i.e. a chromosomal aberration where the number of chromosomes is abnormal) is commonly detected in cancer cells and associated with a plastic response to stress (e.g. drug-induced or due to environmental factors) in several pathogenic fungi and monocellular parasites (21; 16; 27). For these reasons, inferring the ploidy of a sample from genomic data is essential to shed light onto the evolution and adaptation across the domains of life.

Available computational methods to infer ploidy levels from genomic data are based either on modelling the distribution of observed allele frequencies (`nQuire` (25)), comparing frequencies and coverage to a reference data set (`ploidyNGS` (1)), or using inferred genotypes and information on GC content - although this is an approach specific for detecting aberrations in cancer genomes (e.g. `AbsCN-seq` (2), `sequenza` (8)). A popular approach is based on the simple eyeballing method, that is, on the visual inspection of variation of sequencing depth (compared to another ground-truth data set sequenced with the same setup) and allele frequencies (1). However, methods based only on sequencing depth, allele frequencies and genotypes limit the inference on the multiplicity factor of different ploidy levels only (if present), often need a reference data at known ploidy set to be compared to, and lack power for low- or mid-depth sequencing data applications, which are typically affected by large data uncertainty. As low-coverage whole genome sequencing is a popular and cost-effective strategy in the population genetics of both model and non-model species (22), a tool that incorporate data uncertainty is in dire need.

To overcome these issues, we introduce a new method called `HMMploidy` to infer ploidy levels from low- and mid-depth sequencing data. `HMMploidy` comprises a Hidden Markov Model (HMM) (18) where the emissions are both sequencing depth levels and observed reads. The latter are translated into genotype likelihoods (17) and population frequencies to leverage the genotype uncertainty. The hidden states of the HMM represent the ploidy levels which are inferred in windows of polymorphisms, allowing local changes in ploidy to be detected. Notably, `HMMploidy` determines automatically its number of latent states through a heuristic procedure and reduction of the transition matrix. Moreover, our method can leverage the information from multiple samples in the same population by estimate of population frequencies, making it effective at very low depth. `HMMploidy` is written in `R/C++` and `python`. Source code is freely available at https://github.com/SamueleSoraggi/HMMploidy, integrated into `ngsTools` (10), and FAIR data sharing is available at the OSF repository https://osf.io/5f7ar/.

We will first introduce the mathematical and inferential model underlying `HMMploidy`, then show its performance to detect ploidy levels compared to existing tools, and finally illustrate an application to sequencing data from the pathogenic fungus *Cryptococcus neoformans*.

2 Material and methods

This section describes the methods used in the implementation of the `HMMploidy` software. In what follows, data is assumed to be diallelic, without loss of generality. Allowing for more than two alleles would add a summation over all possible pairs of alleles in all calculations. In our notation, indices are lower case and vary within an interval ranging from 1 to the index's upper case letter, e.g. $m = 1, \dots, M$.

2.1 Probability of Sequenced Data

Let $O = (O_1, \dots, O_M)$ be the observed Next Generation Sequencing (NGS) data for M sequenced genomes at N polymorphic sites. Consider a fixed m-th genome and n-th locus. For such genome and locus define $Y_{m,n}$, $G_{m,n}$ and $O_{m,n}$ as the ploidy, genotype and sequencing data, respectively. Given $Y_{m,n}$, the genotype $G_{m,n}$ assumes values in $\{0, 1, ..., Y_{m,n}\}$, i.e. the number of alternate (or derived) alleles of the genotype. The likelihood of the sequenced data, conditionally on the ploidy $Y_{m,n}$ and the population frequency F_n at locus n, is expressed by

$$p(O_{m,n}|Y_{m,n}, F_n) = \sum_{G_{m,n} \in \{0,\dots,Y_{m,n}\}} p(O_{m,n}|G_{m,n}, Y_{m,n}) p(G_{m,n}|Y_{m,n}, F_n), \quad (1)$$

where the left-hand side of the equation has been marginalised over the genotypes, and the resulting probabilities have been rewritten as product of two terms using the tower property of the probability. The first factor of the product is the genotype likelihood (17); the second factor is the probability of the genotype given the population frequency and the ploidy number. The marginalisation over all possible genotypes has therefore introduced a factor that takes into account the genotype uncertainty. Throughout the analyses carried out in this paper, we assume Hardy-Weinberg equilibrium (HWE) and thus model the genotype probability with a negative binomial distribution (11; 24). Other methods considering departure from HWE (DHW), can be considered and implemented by *ad hoc* substitutions of the formula coded in the software. Such functions can be useful in specific situations, such as pathology-, admixture- and selection-induced DHW scenarios (6; 12; 13). However, we will leave the treatment of DHW for the inference of ploidy variation in future studies.

2.2 Genotype likelihood for arbitrary ploidy number

The genotype likelihood is the probability of the observed data (here, sequence counts) given the model (genotype). The base quality of each read is treated as the probability of the incorrect sequenced base, assuming independence of the bases across the reads (15).

Consider the sequencing data $O_{m,n}$ for a diallelic locus n and a genome m, and the coverage $C_{m,n}$ at such locus. Consider $O_{m,n}$ represented as a vector of length $C_{m,n}$ of observed nucleotides $[O_{m,n,1}, O_{m,n,2}, \dots]$. Let $q_{m,n,r}$ be the

Phred base quality (7) for each observed nucleotide $O_{m,n,r}$ at such locus and genome, for $r = 1, \dots, C_{m,n}$. It is straightforward to extend the diploid model to calculate the likelihood of a genotype $G_{m,n}$ at ploidy $Y_{m,n}$ as it follows:

$$\ln p(O_{m,n}|G_{m,n}, Y_{m,n}) = \sum_{r=1}^{C_{m,n}} \ln\Big(\sum_{i=1}^{Y_{m,n}} \frac{1}{Y_{m,n}} p(O_{m,n,r}|G_{m,n}, q_{m,n,r}, Y_{m,n})\Big), \quad (2)$$

$$\textit{where} \quad p(O_{m,n,r}|G_{m,n}, q_{m,n,r}, Y_{m,n}) = \begin{cases} 1 - \epsilon_{m,n,r}, & \textit{if } O_{m,n,r} \text{ in } G_{m,n} \\ \frac{\epsilon_{m,n,r}}{3} & \textit{otherwise} \end{cases}$$

and $\epsilon_{m,n,r}$ is the Phred probability related to the score $q_{m,n,r}$. The probabilities of observing incorrect nucleotides are considered homogeneous over all possible nucleotides.

2.3 Estimation of population frequencies

Population allele frequencies are calculated prior to the HMM optimisation to decrease the computational time. Specifically, the population frequency F_n at the n-th locus is estimated under the assumption of ploidy level being arbitrarily very high to let frequencies represent any possible genotype. Let $\hat{F}_{m,n}$ be the observed minor allele frequency for sample m at locus n, and assume it to be a proxy for the population frequency calculated in each sample. The population frequency estimator for F_n, say $\hat{F}_n$, is defined by pooling the sample estimators into the weighted sum

$$\hat{F}_n = \sum_{m=1}^{M} \frac{C_{m,n}}{C_n} \hat{F}_{m,n}, \quad (3)$$

where $C_n = \sum_{m=1}^{M} C_{m,n}$. The choice of the weights is motivated by the fact that each sequenced read is sampled without replacement from the true genotype, thus the amount of information contained in the estimator $\hat{F}_{m,n}$ w.r.t. the other individuals is proportional to $C_{m,n}$. Therefore, samples with higher coverage are given more weight in estimating the population frequency.

2.4 Hidden Markov Model for Ploidy Inference

Here, the HMM is defined, and the inferential process of ploidy levels from the HMM is illustrated. Further mathematical details, proofs and algorithm analysis are available in the supplementary material.

Let $O = (O_1, \dots, O_M)$ be the observed sequencing data for M sequenced genomes at N polymorphic sites. Consider the N sites arranged in K adjacent and non-overlapping windows, where the ploidy is assumed to be constant. For each individual m, `HMMploidy` defines a HMM with a Markov chain of length K of latent states $Y_m^{(1)}, \dots, Y_m^{(K)}$, as shown for a sequence of two ploidies (Fig. 1A) in the graphical model of dependencies of Fig. 1B. Each k-th latent state represents the ploidy level at a specific window of loci, and each window's ploidy

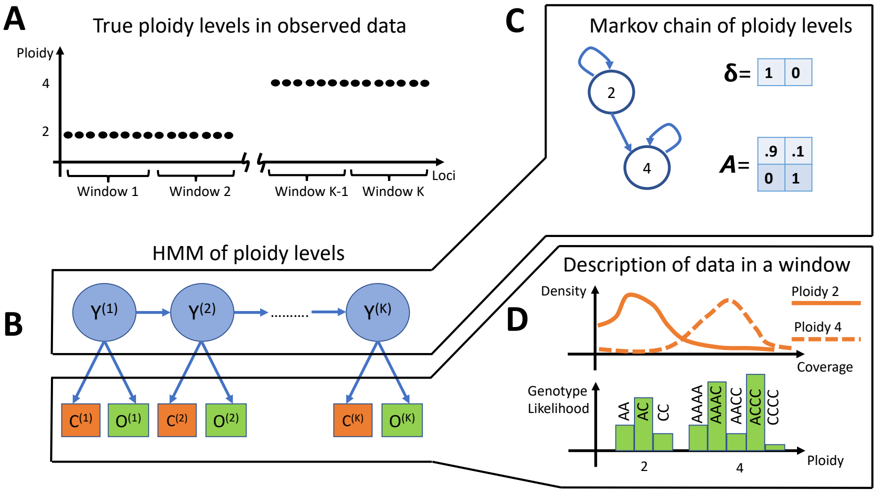

Fig. 1: **HMM for two ploidy levels.** (A) Consider a NGS dataset consisting of a sequence of two ploidy levels. (B) The HMM describing the data has a sequence of hidden states $Y^{(1)}, \ldots, Y^{(K)}$ - one for each window of loci - that can assume one of two values of the ploidies. Observations $C^{(1)}, \ldots, C^{(K)}$ and $O^{(1)}, \ldots, O^{(K)}$ describe respectively the coverage and observed reads in each window. The index related to the sample is omitted to simplify the notation. (C) The sequence of ploidies is described by a Markov chain with two states, governed by a starting vector $\boldsymbol{\delta}$ and a Markov matrix $\boldsymbol{A}$. (D) At each window, the observations are described by the coverage distribution of the loci. There are two distributions, each one dependant on the ploidy level. Similarly, genotype likelihoods describe the observed reads by modelling the genotypes at two distinct ploidy levels.

level depends only on the previous one. Therefore, the sequence of states is described by a transition matrix $\boldsymbol{A}$ of size $\mathcal{Y} \times \mathcal{Y}$ and a $\mathcal{Y}$-long vector of starting probabilities $\boldsymbol{\delta}$, where $\mathcal{Y}$ is the number of ploidies. (Fig. 1C).

In the HMM structure, each k-th ploidy emits two observations. Those contain a dependency on which ploidy is assigned to that window. The observations consist of the sequenced reads $O_{m,n}^{(k)}$ and the average sequencing depth $C_{m,n}^{(k)}$ in the k-th window (Fig. 1B). The former is modelled by the probability in Equation 4; the latter by a Poisson-Gamma distribution (3; 5) (Fig. 1D). The Poisson-Gamma distribution consists of a Poisson distribution whose mean parameter is described by a Gamma random variable. This generates a so-called super-Poissonian distribution, for which the mean is lower than the variance. This allows modelling overdispersed counts, a common issue in NGS datasets.

For the m-th HMM, the Poisson-Gamma distribution in window k is modelled by the ploidy-dependent parameters $\alpha_{Y_m^{(k)}}, \beta_{Y_m^{(k)}} \in \mathbb{R}$, describing mean and dispersion, where $Y_m^{(k)}$ is the ploidy in the considered window. In each window, the population frequencies estimated through Equation (6) serve as a proxy for the probability of sequenced reads.

Heuristic Expectation Conditional Maximization (HECM) Here we propose a heuristic optimisation algorithm to automatically find the number of latent states of the HMM, and to assign them to the correct ploidy through the genotype likelihoods. Our implementation is a heuristic version of the well-known Expectation Conditional Maximisation (ECM) algorithm (4).

The ECM algorithm is used to infer the parameters $\boldsymbol{A}$, $\boldsymbol{\delta}$ modelling the sequence of ploidies. This is done in two iterative steps by exploiting the ploidy-dependent distributions of the observed data (sequenced reads and coverage in each window). The first step is the well-known forward-backward algorithm (18; 4), that computes in each window the probability of a ploidy given all the observed data. This is done in an efficient way through dynamic programming and exploitation of Markov properties in a time bounded by $\mathcal{Y}\mathcal{K}$ by implementing two calculation sweeps, starting respectively at the end and at the beginning of the observation sequence.

The forward-backward algorithm thus creates the mathematical link between ploidies and observed data, and allows to update the parameters governing the Markov chain of ploidies with the Expectation-maximization (EM) algorithm in a subsequent step. The EM algorithm maximizes a value (called intermediate quantity) strictly related to the likelihood of the model, where the free variables of the maximization are the matrix $\boldsymbol{A}$ and the vector $\boldsymbol{\delta}$. This procedure continues iteratively by recalculating the forward-backward posteriors and the update parameters with the EM, until the intermediate quantity cannot be further improved.

Heuristic step and ploidy inference. The ECM algorithm is repeated as an iterative sequence of forward-backward and EM routines, until the intermediate quantity satisfies a convergence criteria. When convergence is achieved, `HMMploidy` performs the heuristic step, by running few iterations of the ECM over the HMM, where the set of ploidy levels is reduced by one, and the parameters for initialization are the final ones from the ECM. We assume that, if the HMM has an overfitting set of ploidy levels, observation parameters are overlapping (14) for two or more ploidy levels. Therefore, removing one unnecessary ploidy requires only few extra iterations for the EM to converge again. The Bayesian Information Criterion (BIC) (3; 4) is used to compare the HMM with the reduced HMMs. If there is a reduced HMM with a better BIC score, then the ECM runs again on such HMM, otherwise it stops. Such method is an adaptation of the suggestion in (14). After the HMM is reduced through the BIC comparison, we reduce the transition matrix between ploidies, i.e. we remove ploidies for which there is almost zero probability of lasting a reasonable number of adjacent windows. In other words, we remove ploidies that will last for the length of one or few more windows of loci. Once the HMM parameters are determined through the heuristic sweep, the standard Viterbi algorithm (23) is applied to infer the most likely sequence of ploidies from the parameters of the HMM. The Viterbi algorithm is another example of dynamic programming, that avoiding calculating all possible $\mathcal{Y}^{\mathcal{K}}$ sequences of ploidies to determine which one is the best.

3 Simulated data

The assessment of memory, runtime and ploidy detection power of `HMMploidy` compared to the other methods is performed on a wide range of simulated scenarios. We simulated sequencing data under a wide range of scenarios using a previously proposed approach (9). Specifically, each locus is treated as an independent observation, without modelling the effect of linkage disequilibrium. The number of reads is modeled with a Poisson distribution with parameter given by the input coverage multiplied by the ploidy level. At each locus, individual genotypes are randomly drawn according to a probability distribution defined by set of population parameters (e.g. shape of the site frequency spectrum). Once genotypes are assigned, sequencing reads (i.e. nucleotidic bases) are sampled with replacement with a certain probability given by the base quality scores.

For comparing the performance of detecting ploidy between `HMMploidy` and existing tools, 100 simulations of M genomes are performed for every combination of ploidy (from 1 to 5, constant along each genome), sample size (1, 2, 5, 10, 20), and sequencing depth (0.5X, 1X, 2X, 5X, 10X, 20X). The sequencing depth is defined as the number of sequenced bases averaged by the polyploid genome size. Each simulated genome has a length of 5Kb with all loci being polymorphic in the population.

Simulated data for the analysis of of runtimes and memory usage consist of 100 diploid genomes of length 10kb, 100kb, 1Mb, 10Mb. Each simulated genome has of the expected density of polymorphic sites equal to 1%. The simulation scripts and pipelines are included in the Github and OSF repositories. Performance analysis was performed on a cluster node with four reserved cores of an Intel Xeon Gold 6130 @1.00GHz with 24GB of RAM and the Ubuntu 18.04.3 OS.

4 Results and discussion

We assess the power of `HMMploidy` to infer ploidy levels on simulated genomes ranging from haploid to pentaploid. Samples sizes varied from 1 to 20 individuals haplotypes, and sequencing depths from 0.5X to 20X. `HMMploidy` is compared to the two state-of-the-art methods `ploidyNGS` (1) and `nQuire` (including a version with denoising option, `nQuire.Den`) (25). The former performs a Kolmogorov-Smirnov test between the minor allele frequencies of the observed data and of simulated data sets at different ploidy levels (simulated at 50X). The latter models the minor allele frequencies with a Gaussian mixture model. We exclude depth-based methods because they are hardly applicable to low sequencing depth (Fig. S2,S3) and work as empirical visual checks rather than algorithmic procedures. While nQuire and ploidyNGS sweep the whole simulated genomes, HMMploidy analyses windows of 250bp, so the detection rate is calculated as the windows' average, making the comparison deliberately more unfair to our method.

HMMploidy reaches maximum power at depth 0.5X with 20 individuals for all scenarios excluding the tetraploid case (Fig. 2)). This might be because it is difficult to distinguish diploid and tetraploid genotypes at such low depth. In the haploid and diploid case ploidyNGS has a remarkable 100% success at very low depths (Fig. 2). This is likely because having only few reads make it easier to compare the data to a simulated genome with low ploidy number and a simpler distribution of observed alleles. However, this erratic behaviour disappears at higher ploidy levels, and ploidyNGS is generally outperformed by nQuire.Den and/or HMMploidy. HMMploidy is outperformed at low depth in the tetraploid scenario by both versions of nQuire. This might indicate that genotype likelihoods are not successful in modelling tetraploid genotypes as well as allele frequencies in this specific scenario.

Note also that none of the methods perform well with a single haploid sample. This happens because many loci show only one possible genotype, and even with the genotype likelihoods it is impossible to determine the multiplicity of the ploidy. With more samples it is possible to exploit loci with at least another allele to inform on the most likely genotype.

In all tested scenarios, HMMploidy greatly improves its accuracy with increasing sample size, with unique good performances at low depth (Fig. 2) not observed with other methods. Additionally, HMMploidy infers ploidy numbers in windows (as in Fig. 3), and is able to detect changes in ploidy levels across the data, while other tools return a whole-genome value. Moreover, HMMploidy does not require a reference genome at a known ploidy, unlike ploidyNGS. HMMploidy can identify haploid genomes, unlike nQuire. Note that either deeper sequencing coverage or larger sample size is likely to be required for HMMploidy to detect higher levels of ploidy, as the power of the method lowers with increasing ploidy (Fig. S4).

The benchmark of HMMploidy shows rather constant CPU time across genome lengths by keeping the number of windows fixed at $K = 100$ (Fig. S5A). The shortest simulations are an exception, due to a very fast processing of the data to be used in the HMM. Occasionally, runtimes are elevated corresponding to cases where the inference algorithm is converging with difficulty. Fig. S5B shows the effect of increasing the number of windows on 10MB genomes. The growth of execution time follows linearly the increase of K, plus a probable extra overhead for preprocessing the data in many windows, showing that the forward-backward cost of $O(Y^2K)$ dominates the algorithm. In both the length- and windows-varying scenarios, memory usage was kept at an almost constant value of $350MB$. This is possible thanks to the implementation of file reading and frequency estimation in C++. Both nQuire and ploidyNGS are obviously extremely fast and run in less than one second because they only need to calculate and compare observed allele frequencies, with a cost approximately comparable to the number of loci in the data. Therefore, their performance is not reported in the benchmark figures. Analogous trends on execution times would follow for genomes longer than 10MB and we expect `HMMploidy` to run without issues on larger genomes.

Note that `HMMploidy` trains a separate HMM on each genome even for larger sample sizes. As shown above, each HMM might require considerable CPU time if many windows are used, or if the HECM algorithms has a slow convergence. However, training a separate HMM on each genome allows to overcome two main issues: sequencing data at different depths, and variations in ploidy among sampled genomes. On the former point, rescaling sequencing depth across genomes is not possible since `HMMploidy` models a distribution of read counts. On the latter point, it would not be possible to detect sample-specific variation in ploidy levels when training the HMM on pooled genomic data. Therefore, training a separate HMM on each genome is an important feature in `HMMploidy`. However, a simple extension of `HMMploidy` would allow to estimate an HMM on the pooled data from multiple genomes, and to initiate HMM parameters and number of latent states to reduce the model estimation tuntimes. Such options might be implemented in future versions of the software.

To illustrate the use of `HMMploidy`, we apply it to sequencing data from 23 isolates of the pathogenic fungus *Cryptococcus neoformans* recovered from HIV infected patients (19). Changes in ploidy in *C. neoformans* have been associated with drug resistance (21). In accordance with the original study (19), we retrieve patterns of polyploidy and aneuploidy within each isolate. Most of the analysed samples are haploid (Fig. 3 and Fig. S6-S28). Interestingly, samples CCTP27 and CCTP27 at day 121 (CCTP27-d121) are inferred to have same ploidy, even though CCTP27-d121 triplicates its sequencing depth on chromosome 12 (Fig. 3). This finding suggests the presence of a recent copy number variation. In fact, as no sufficient genetic variation has built up on the recently duplicated triploid chromosome yet, the data is modeled as a single chromosome by the genotype likelihoods. Sample CCTP50 has on average a higher depth at day 409, but chromosome 1 changes from diploid (day 1) to haploid (day 409). Chromosome 12 is triploid at day 409 although the high variability of sequencing depth is not informative on the ploidy. Notably, we were able to retrieve the same patterns of predicted ploidy when artificially down-sampling the sequencing data to 20% of the original data set (Fig. S6-S28). Interestingly, `ploidyNGS`, `nQuire` and `nQuire.Den` infer the highest tested ploidy in almost all windows of the 23 samples (Supplementary Table 1). This is likely because these methods fit the distribution of widely varying allele frequencies in each sample with the most complex ploidy model, as they do not consider the information of genotype likelihoods. All results and data are available in the OSF repository.

5 Conclusions

Here we introduce `HMMploidy`, a method to infer ploidy levels suitable for low- and mid-depth sequencing data, as it jointly uses information from sequencing

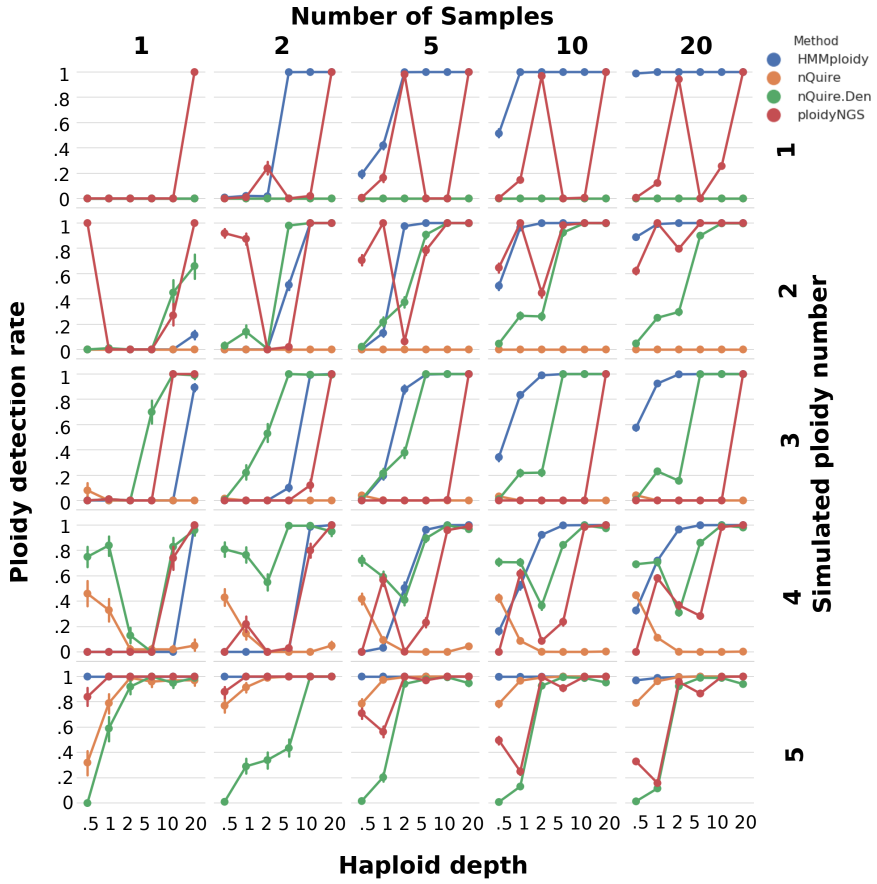

Fig. 2: **Comparison of ploidy detection rates for different methods at various experimental scenarios.** The rate of detecting the correct ploidy (y-axis) is shown against the haploid sequencing depth (x-axis) for different sample sizes (on columns) and simulated ploidy levels (on rows). For every simulated ploidy level, at each value of the sequencing depth we generate 100 times M genomes, where M is the number of simulated samples. The ploidy detection rate is the proportion of correctly detected ploidies in the windows of loci with the HMM method, and the proportion of correctly detected ploidies along each whole genome with the other genome-wide methods.

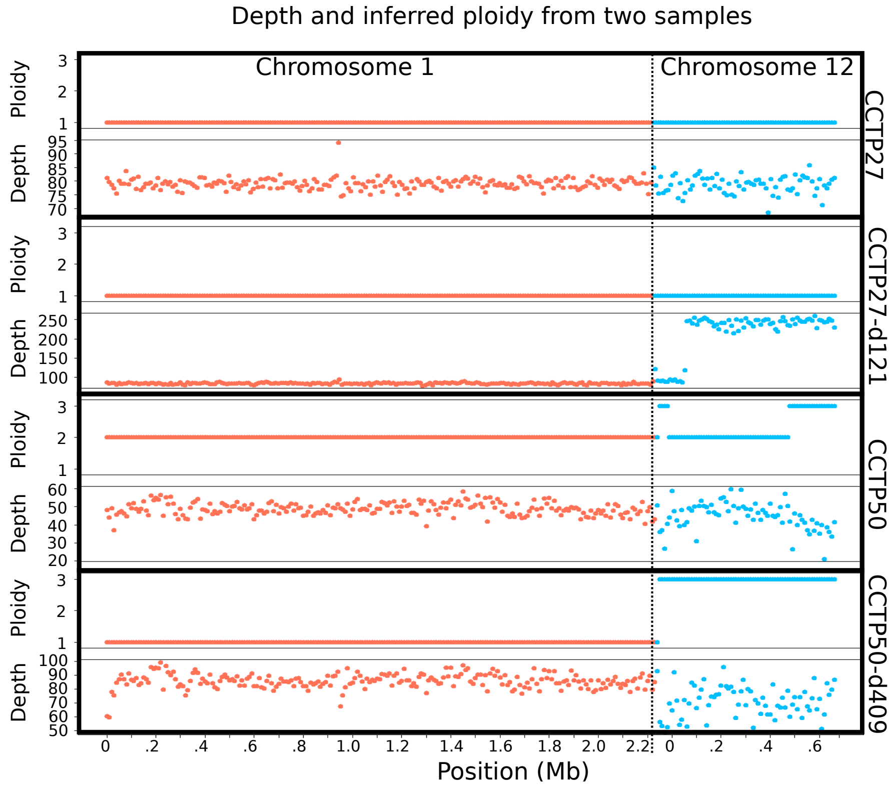

Fig. 3: **Inference of ploidy levels on two samples of *Cryptococcus neoformans* at different time points using `HMMploidy`.** Inferred ploidy and corresponding sequencing depth are shown in genomic windows for two samples at day 1 (CCTP27 and CCTP50) and day 121 (CCTP27-d121) and 409 (CCTP50-d409) on chromosomes 1 and 12.

depth and genotype likelihoods. HMMploidy outperforms traditional methods based on observed allele frequencies, especially when combining multiple samples. We predict that HMMploidy will have a broad applicability in studies of genome evolution.

Bibliography

[1] Augusto Corrêa dos Santos, R., Goldman, G.H., Riaño-Pachón, D.M.: ploidyNGS: visually exploring ploidy with Next Generation Sequencing data. Bioinformatics **33**(16), 2575–2576 (aug 2017). https://doi.org/10.1093/bioinformatics/btx204, http://www.ncbi.nlm.nih.gov/pubmed/28383704

[2] Bao, L., Pu, M., Messer, K.: AbsCN-seq: a statistical method to estimate tumor purity, ploidy and absolute copy numbers from next-generation sequencing data. Bioinformatics **30**(8), 1056–1063 (apr 2014). https://doi.org/10.1093/bioinformatics/btt759, https://academic.oup.com/bioinformatics/ARTICLE-lookup/doi/10.1093/bioinformatics/btt759

[3] Bishop, C.M.: Pattern recognition and machine learning. Springer (2006)

[4] Cappe, O., Moulines, E., Ryden, T.: Inference in Hidden Markov Models. Springer Science+Business Media, Inc (2005)

[5] Casella, G., Berger, R.L.: Statistical inference. Thomson Learning (2002)

[6] Chen, B., Cole, J.W., Grond-Ginsbach, C.: Departure from Hardy Weinberg Equilibrium and Genotyping Error. Front Genet. (8) (2017)

[7] Ewing, B., Hillier, L., Wendl, M.C., Green, P.: Base-calling of automated sequencer traces using phred. I. Accuracy assessment. Genome research **8**(3), 175–85 (mar 1998), http://www.ncbi.nlm.nih.gov/pubmed/9521921

[8] Favero, F., Joshi, T., Marquard, A.M., Birkbak, N.J., Krzystanek, M., Li, Q., Szallasi, Z., Eklund, A.C.: Sequenza: allele-specific copy number and mutation profiles from tumor sequencing data. Annals of Oncology **26**(1), 64–70 (jan 2015). https://doi.org/10.1093/annonc/mdu479, http://www.ncbi.nlm.nih.gov/pubmed/25319062

[9] Fumagalli, M., Vieira, F.G., Korneliussen, T.S., Linderoth, T., Huerta-Sánchez, E., Albrechtsen, A., Nielsen, R.: Quantifying population genetic differentiation from next-generation sequencing data. Genetics **195**(3), 979–992 (2013). https://doi.org/10.1534/genetics.113.154740, https://www.genetics.org/content/195/3/979

[10] Fumagalli, M., Vieira, F.G., Linderoth, T., Nielsen, R.: ngsTools: methods for population genetics analyses from next-generation sequencing data. Bioinformatics **30**(10), 1486–1487 (01 2014). https://doi.org/10.1093/bioinformatics/btu041, https://doi.org/10.1093/bioinformatics/btu041

[11] Hardy, G.H.: Mendelian Proportions in a Mixed Population. Science, New Series **28**(706), 49–50 (1908)

[12] Jacqueline, K.W., Anna, P., Nancy, J.C.:

[13] Lachance, J.: Detecting selection-induced departures from hardy-weinberg proportions. Genetics Selection Evolution (1), 15 (2009)

[14] Li, C., Biswas, G.: Temporal Pattern Generation Using Hidden Markov Model Based Unsupervised Classification. In: IDA 1999: Advances in In-

telligent Data Analysis, pp. 245–256. Springer, Berlin, Heidelberg (1999). https://doi.org/10.1007/3-540-48412-4$_2$1

[15] McKenna, A., Hanna, M., Banks, E., Sivachenko, A., Cibulskis, K., Kernytsky, A., Garimella, K., Altshuler, D., Gabriel, S., Daly, M., DePristo, M.A.: The Genome Analysis Toolkit: a MapReduce framework for analyzing next-generation DNA sequencing data. Genome research **20**(9), 1297–303 (sep 2010). https://doi.org/10.1101/gr.107524.110, http://www.ncbi.nlm.nih.gov/pubmed/20644199

[16] Morrow, C.A., Fraser, J.A.: Ploidy variation as an adaptive mechanism in human pathogenic fungi. Seminars in Cell and Developmental Biology **24**(4), 339–346 (apr 2013)

[17] Nielsen, R., Paul, J., Albrechtsen, A., Song, Y.: Genotype and snp calling from next-generation sequencing data. Nature Reviews. Genetics **12**(6), 443–451 (2011). https://doi.org/10.1038/nrg2986

[18] Rabiner, L.: A tutorial on hidden Markov models and selected applications in speech recognition. Proceedings of the IEEE **77**(2), 257–286 (1989). https://doi.org/10.1109/5.18626, http://ieeexplore.ieee.org/document/18626/

[19] Rhodes, J., Beale, M.A., Vanhove, M., Jarvis, J.N., Kannambath, S., Simpson, J.A., Ryan, A., Meintjes, G., Harrison, T.S., Fisher, M.C., Bicanic, T.: A Population Genomics Approach to Assessing the Genetic Basis of Within-Host Microevolution Underlying Recurrent Cryptococcal Meningitis Infection. G3 Genes—Genomes—Genetics (2017). https://doi.org/10.1534/g3.116.037499, https://doi.org/10.1534/g3.116.037499

[20] Sattler, M.C., Carvalho, C.R., Clarindo, W.R.: The polyploidy and its key role in plant breeding. Planta (243), 281–296 (2016)

[21] Stone, N.R., Rhodes, J., Fisher, M.C., Mfinanga, S., Kivuyo, S., Rugemalila, J., Segal, E.S., Needleman, L., Molloy, S.F., Kwon-Chung, J., Harrison, T.S., Hope, W., Berman, J., Bicanic, T.: Dynamic ploidy changes drive fluconazole resistance in human cryptococcal meningitis. Journal of Clinical Investigation **129**(3), 999–1014 (mar 2019)

[22] Therkildsen, N.O., Palumbi, S.R.: Practical low-coverage genomewide sequencing of hundreds of individually barcoded samples for population and evolutionary genomics in nonmodel species. Molecular Ecology Resources **17**(2), 194–208 (2017). https://doi.org/https://doi.org/10.1111/1755-0998.12593, https://onlinelibrary.wiley.com/doi/abs/10.1111/1755-0998.12593

[23] Viterbi, A., A.: Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. IEEE Transactions on Information Theory **13**(2), 260–269 (apr 1967). https://doi.org/10.1109/TIT.1967.1054010, http://ieeexplore.ieee.org/document/1054010/

[24] Weinberg, W.: Über den Nachweis der Vererbung beim Menschen. Jahresh. Ver. Vaterl. Naturkd. Württemb. **64**, 369–382 (1908)

[25] Weiß, C.L., Pais, M., Cano, L.M., Kamoun, S., Burbano, H.A.: nQuire: a statistical framework for ploidy estimation using next generation sequenc-

ing (2018). https://doi.org/10.1186/s12859-018-2128-z, https://doi.org/10.1186/s12859-018-2128-z

[26] Wood, T.E., Takebayashi, N., Barker, M.S., Mayrose, I., Greenspoon, P.B., H, R.L.: The frequency of polyploid speciation in vascular plants. Proc Natl Acad Sci USA (106), 13875–13879 (2009)

[27] Zhu, J., Tsai, H.J., Gordon, M.R., Li, R.: Cellular Stress Associated with Aneuploidy. Developmental cell **44**(4), 420–431 (feb 2018). https://doi.org/10.1016/j.devcel.2018.02.002, http://www.ncbi.nlm.nih.gov/pubmed/29486194

6 Supplementary Material

6.1 Supplementary Methods

This section describes supplementary information on methods used for the implementation of HMMploidy. The first three chapters are identical to the ones from the main manuscript, but we report them here as well to make the reading of the supplementary manuscript complete. The rest of the material contains all mathematical details, proof and detailed explanations not included in the main manuscript.

6.2 Probability of Sequenced Data

Let $O = (O_1, \dots, O_m)$ be the observed Next Generation Sequencing (NGS) data for M sequenced genomes at N polymorphic sites. Consider a fixed m-th genome and n-th locus. For such genome and locus define $Y_{m,n}$, $G_{m,n}$ and $O_{m,n}$ as the ploidy, genotype and sequencing data, respectively. Given $Y_{m,n}$, the genotype $G_{m,n}$ assumes values in $\{0, 1, ..., Y_{m,n}\}$, i.e. the number of alternate (or derived) alleles of the genotype. The likelihood of the sequenced data, conditionally on the ploidy $Y_{m,n}$ and the population frequency F_n at locus n, is expressed by

$$p(O_{m,n}|Y_{m,n}, F_n) = \sum_{G_{m,n} \in \{0,\dots,Y_{m,n}\}} p(O_{m,n}|G_{m,n}, Y_{m,n}) p(G_{m,n}|Y_{m,n}, F_n), \quad (4)$$

where the left-hand side of the equation has been marginalised over the genotypes, and the resulting probabilities have been rewritten as product of two terms using the tower property of the probability. The first factor of the product is the genotype likelihood (17); the second factor is the probability of the genotype given the population frequency and the ploidy number. The marginalisation over all possible genotypes has therefore introduced a factor that takes into account the genotype uncertainty. Throughout the analyses carried out in this paper, we assume Hardy-Weinberg equilibrium (HWE) and thus model the genotype probability with a negative binomial distribution (11; 24). Other methods considering departure from HWE (DHW), can be considered and implemented by *ad hoc* substitution of the formula coded in the software. Such functions can be useful in specific situations, such as pathology-, admixture- and selection-induced DHW scenarios (6; 12; 13). However, we will leave the treatment of DHW for the inference of ploidy variation in future studies.

6.3 Genotype likelihood for arbitrary ploidy number

The genotype likelihood is the probability of observing a specific genotype given the observed sequencing data. The base quality of each read is treated as the probability of the incorrect sequenced base, assuming independence of the bases across the reads (15).

Consider the sequencing data $O_{m,n}$ for a locus n and a genome m, and the coverage $C_{m,n}$ at such locus. Let q_r be the Phred base quality (7) for each observed nucleotide r at such locus and genome, for $r = 1, \dots, C_{m,n}$. It is straightforward to extend the diploid model calculate the likelihood of a genotype $G_{m,n}$ at ploidy $Y_{m,n}$ as it follows:

$$\ln p(O_{m,n}|G_{m,n}, Y_{m,n}) = \sum_{r=1}^{C_{m,n}} \ln\Bigg(\sum_{i=1}^{Y_{m,n}} \frac{1}{Y_{m,n}} p(r|G_{m,n}, q_r, Y_{m,n}) \Bigg), \tag{5}$$

$$where \quad p(r|G_{m,n}, q_r, Y_{m,n}) = \begin{cases} 1 - \epsilon_r, & if\ r = G_{m,n} \\ \frac{\epsilon_r}{3} & otherwise \end{cases}$$

and ϵ_r is the Phred probability related to the score q_r. The probabilities of observing incorrect nucleotides are considered homogeneous over all possible nucleotides.

6.4 Estimation of population frequencies

Population allele frequencies are calculated prior to the HMM optimisation to decrease the computational time. Specifically, the population frequency F_n at the n-th locus is estimated under the assumption of ploidy level being arbitrarily very high to let frequencies represent any possible genotype. Let $\hat{F}_{m,n}$ be the observed minor allele frequency for sample m at locus n, and assume it to be a proxy for the population frequency calculated in each sample. The population frequency estimator for F_n, say $\hat{F}_n$, is defined by pooling the sample estimators into the weighted sum

$$\hat{F}_n = \sum_{m=1}^{M} \frac{C_{m,n}}{C_n} \hat{F}_{m,n}, \tag{6}$$

where $C_n = \sum_{m=1}^{M} C_{m,n}$. The choice of the weights is motivated by the fact that each sequenced read is sampled without replacement from the true genotype, thus the amount of information contained in the estimator $\hat{F}_{m,n}$ w.r.t. the other individuals is proportional to $C_{m,n}$. In this way samples with higher coverage are given more weight in estimating the population frequency.

6.5 Hidden Markov Model for Ploidy Inference

Here, the HMM is defined, and the inferential process of ploidy levels as latent variables of the HMM is described in a more illustrative fashion. Further mathematical details an proofs are found in the supplementary material.

Let $O = (O_1, \dots, O_m)$ be the observed sequencing data for M sequenced genomes at N polymorphic sites. Consider the N sites arranged in K adjacent and non-overlapping windows, where the ploidy is assumed to be constant. For each individual m, `HMMploidy` defines a HMM with a Markov chain of length K of latent states $Y_m^{(1)}, \dots, Y_m^{(K)}$. Each latent state represents the ploidy level at a specific window of loci.

The k-th state (ploidy) of the Markov chain emits two observations, that is, the sequenced reads $O_{m,n}^{(k)}$ and the average sequencing depth $C_{m,n}^{(k)}$ in the k-th window (Fig. S1). The former is modelled by the probability in Equation 4; the latter by a Poisson-Gamma distribution (3; 5). The Poisson-Gamma distribution consists in a Poisson distribution whose mean parameter is described by a Gamma random variable. This allows to obtain a so-called superpoissonian distribution, for which the mean and the variance are no longer the same, but the variance is larger than the mean. This allows us to infer a model for overdispersed counts, a common issue in NGS datasets.

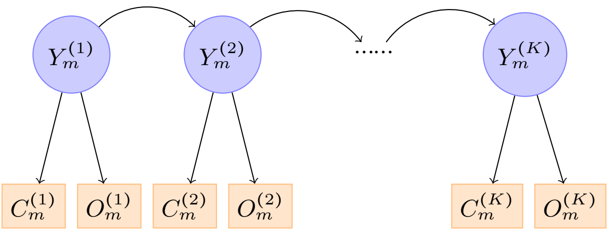

Figure S1: **Hidden Markov Model for ploidy inference.** Graphical representation of the HMM to infer the ploidies of the m-th genome. $Y_m^{(k)}$ is the ploidy level of the k-th window of genome m. The ploidy-dependent emissions consist of the average sequencing depth $C_m^{(k)}$ and the sequenced data $O_m^{(k)}$, whose distributions are respectively described by a Poisson-gamma distribution and by Equation (4).

The Markov chain of ploidies aforementioned is characterised by a $|\mathcal{Y}| \times |\mathcal{Y}|$ transition matrix $\boldsymbol{A}$, and a $|\mathcal{Y}|$-long vector $\boldsymbol{\delta}$ of starting probabilities for the first latent state. Here, $\mathcal{Y}$ is the set of ploidies included in the model, and $|\mathcal{Y}|$ is its cardinality. The average depth for genome m in window k is characterised by the ploidy-dependent parameters $\alpha_{Y_m^{(k)}}, \beta_{Y_m^{(k)}} \in \mathbb{R}$, describing mean and dispersion of the data, for each $Y_m^{(k)} \in \mathcal{Y}$. For brevity we write the parameters of the depth distribution in vector form, i.e. $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$. The allele frequencies calculated through Equation (6) in the k-th window of loci serve as a proxy for the probability of sequenced reads.

Heuristic Expectation Conditional Maximization (HECM) Here we propose a heuristic optimisation algorithm to automatically find the number of latent states, and to assign them to the correct ploidy through the genotype likelihoods. Our implementation is a heuristic version of the well-known Expectation Conditional Maximisation (ECM) algorithm (4). The objective of the HECM algorithm applied to our HMM is to find the optimal (according to the ECM formulation) values of the parameters $\boldsymbol{A}$, $\boldsymbol{\delta}$ of the Markov chain, the ploidy-dependent coverage distribution parameters $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$, and to apply a heuristic procedure to find a set $\mathcal{Y}$ that satisfies a specific criterion.

We start by illustrating the steps of the ECM algorithm, and subsequently add the heuristic procedure. For ease of notation, denote by λ the triplet of parameters $(\boldsymbol{A}, \boldsymbol{\delta}, \boldsymbol{\beta}, \boldsymbol{\alpha}) \in \Lambda$, and consider it written as two separate triplets as it follows: $\lambda = (\lambda_1, \lambda_2) = ((\boldsymbol{A}, \boldsymbol{\delta}, \boldsymbol{\beta}), (\boldsymbol{\alpha})) \in \Lambda_1 \times \Lambda_2$. Given the parameters $\lambda^{\ell-1}$ calculated at the $(\ell-1)$-th step of the ECM, the ℓ-th iteration to calculate λ^{ℓ} follows essentially the steps below:

1. calculate the intermediate quantity
$$Q(\lambda_1^{\ell}|\lambda^{\ell-1}) = \mathbb{E}\Big[\ln p\Big(O_m^{(1:K)},\, Y_m^{(1:K)}|(\lambda_1^{\ell}, \lambda_2^{\ell-1})\Big)\,\Big|\, O_m^{(1:K)}, \lambda^{\ell-1}\Big];$$
2. calculate $\lambda_1^{\ell} = \arg_{\lambda_1^{\ell} \in \Lambda_1} \max Q(\lambda_1^{\ell}|\lambda^{\ell-1}))$;
3. calculate the intermediate quantity $Q(\lambda_2^{\ell}|(\lambda_1^{\ell}, \lambda_2^{\ell-1}))$ analogously to step 1;
4. calculate $\lambda_2^{\ell} = \arg_{\lambda_2^{\ell} \in \Lambda_2} \max Q((\lambda_2^{\ell})|(\lambda_1^{\ell}, \lambda_2^{\ell}))$.

Here, we used $O_m^{(1:K)}$, $Y_m^{(1:K)}$ to denote $O_m^1, \ldots, O_m^K$ and $Y_m^1, \ldots, Y_m^K$, respectively. The first step and the calculation of $\boldsymbol{A}, \boldsymbol{\delta}$ at iteration ℓ are solved by using the classical forward-backward algorithm (18; 4), therefore we will only briefly mention the necessary elements of it.

The intermediate quantity at step 1 can be explicitly written as the sum of three terms involving separately the matrix $\boldsymbol{A}$, the vector $\boldsymbol{\delta}$ and the vectors $\boldsymbol{\alpha}, \boldsymbol{\beta}$:

$$Q(\lambda_1^{\ell}|\lambda^{\ell-1}) = \sum_{Y_m^{(1:K)} \in \mathcal{Y}} ln(\delta^{\ell}_{Y_m^{(1)}})p(Y_m^{(1:K)}|O_m^{(1:K)}, C_m^{(1:k)}, \lambda^{\ell-1}) \tag{7}$$
$$+ \sum_{Y_m^{(1:K)} \in \mathcal{Y}} \textstyle\sum_{k=2}^{K} ln(\boldsymbol{A}^{\ell}_{Y_m^{(k-1)} Y_m^{(k)}})p(Y_m^{(1:K)}|O_m^{(1:K)}, C_m^{(1:k)}, \lambda^{\ell-1}) \tag{8}$$
$$+ \sum_{Y_m^{(1:K)} \in \mathcal{Y}} \sum_{k=1}^{K} \Bigg(ln\Big(p(O_m^{(k)}|Y_m^{(k)}, F^{(k)})\Big) + ln\Big(p(C_m^{(k)}|Y_m^{(k)}, \alpha^{\ell-1}_{Y_m^{(k)}}, \beta^{\ell}_{Y_m^{(k)}}))\Bigg)$$
$$p(Y_m^{(1:K)}|O_m^{(1:K)}, C_m^{(1:k)}, \lambda^{\ell-1}) \tag{9}$$

Consider the (m,k)-th forward variable defined by
$$f(y_m^{(k)}) = p(O_m^{(1:k)}, C_m^{(1:k)}, Y_m^{(k)} = y_m^{(k)}|\lambda),$$
that is, the probability of the first k observations and k-th ploidy $y_m^{(k)}$ given the parameters λ. Define the (m,k)-th backward variable as
$$b(y_m^{(k)}) = p(O_m^{(k+1:K)}, C_m^{(k+1:K)}|Y_m^{(k)} = y_m^{(k)}, \lambda),$$
that is, the probability of the latest $(K-k)$ observations, given the k-th ploidy $y_m^{(k)}$ and the parameters λ. The forward and backward variables can be computed with an iterative procedure (18, eq. 19,20,24,25) and allow to calculate efficiently the likelihood of the data as
$$p(O_m^{(1:K)}, C_m^{(1:k)}|\lambda) = \sum_{y_m^{(k)} \in \mathcal{Y}} f(y_m^{(k)})b(y_m^{(k)}) \quad \text{for any } k = 1, \ldots, K.$$

The two terms in Equations (7) and (8) include only the parameters $\boldsymbol{\delta}$ and $\boldsymbol{A}$, respectively. This simplifies finding an optimisation formula by considering separately each term of Equations (7) and (8). Optimisation equations for $\boldsymbol{\delta}$ and $\boldsymbol{A}$ are easily derived through Lagrange multipliers (18, eq. 40a,40b). This does not solve the second step of the ECM algorithm, because the optimum for $\boldsymbol{\beta}$ is still not calculated.

It is easy to see that both $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ concur in defining Equation (9). This is what originates the conditional nature of the ECM algorithm, i.e. α and β cannot be optimised independently. Therefore we first optimise $\boldsymbol{\beta}$ considering the values of $\boldsymbol{\alpha}$ calculated at the $(\ell-1)$-th iteration of the ECM algorithm. Using the forward and backward variables, and excluding terms independent from the Poisson-Gamma parameters, Equation (9) can be written as follows:

$$\begin{aligned}
&\sum_{Y_m^{(1:K)}\in\mathcal{Y}} \sum_{k=1}^{K} u(m,k) ln\Big(p(C_m^{(k)}|Y_m^{(k)}, \alpha^{\ell-1}_{Y_m^{(k)}}, \beta^{\ell}_{Y_m^{(k)}})\Big)\\
&= \sum_{Y_m^{(1:K)}\in\mathcal{Y}} \sum_{k=1}^{K} u(m,k) \ln\left(\frac{\Gamma(\alpha^{\ell-1}_{Y_m^{(k)}} + C_m^{(k)})}{\Gamma(C_m^{(k)}+1)\Gamma(\alpha^{\ell-1}_{Y_m^{(k)}})}\right)\\
&+ \sum_{Y_m^{(1:K)}\in\mathcal{Y}} \sum_{k=1}^{K} u(m,k)\left(C_m^{(k)} \ln\left(\frac{1}{\beta^{\ell}_{Y_m^{(k)}}+1}\right) + \alpha^{\ell-1}_{Y_m^{(k)}} \ln\left(\frac{\beta^{\ell}_{Y_m^{(k)}}}{\beta^{\ell}_{Y_m^{(k)}}+1}\right)\right)
\end{aligned}$$

where $u^{\ell}(m,k) = f(y_m^{(k)})b(y_m^{(k)})/p(O_m^{(1:K)}, C_m^{(1:k)}|\lambda^{\ell})$. Let us equal the partial derivative of $Q(\lambda_1^{\ell}|\lambda^{\ell-1})$ w.r.t. a certain $\beta^{\ell}_{y_m^{(k)}}$, $y_m^{(k)} \in \mathcal{Y}$, to zero, to calculate the optimum for the parameter of interest:

$$\frac{\partial Q(\lambda_1^{\ell}|\lambda^{\ell-1})}{\partial \beta^{\ell}_{y_m^{(k)}}} = \sum_{k=1}^{K} -u(m,k)\frac{C_m^{(k)}}{\beta^{\ell}_{y_m^{(k)}}+1} + \sum_{k=1}^{K} u(m,k)\frac{\alpha^{\ell-1}_{Y_m^{(k)}}}{\beta^{\ell}_{y_m^{(k)}}\big(\beta^{\ell}_{y_m^{(k)}}+1\big)} = 0.$$

Solving for $\beta^{\ell}_{y_m^{(k)}}$ leads to

$$\beta^{\ell}_{y_m^{(k)}} = \frac{\alpha^{\ell-1}_{Y_m^{(k)}} \sum_{k=1}^{K} u(m,k)}{\sum_{k=1}^{K} u(m,k) C_m^{(k)}}.$$

This completes the step 2 of the ECM. In our implementation of `HMMploidy`, we want to leverage the information contained in the genotype likelihoods, whose partial derivative goes to zero and in principle are not integrated in the optimisation. In `HMMploidy`, we add the genotype likelihoods to the depth distribution prior to optimisation, so that forward and backward variables contain information on both depth and genotypes, and allow the identification of different states with distinct ploidies.

The value of $Q(\lambda_2^{\ell}|(\lambda_1^{\ell}, \lambda_2^{\ell-1}))$ can be easily calculated as in step 1, and by setting the partial derivative of $Q(\lambda_2^{\ell}|(\lambda_1^{\ell}, \lambda_2^{\ell-1}))$ w.r.t. $\alpha^{\ell}_{y_m^{(k)}}$, $y_m^{(k)} \in \mathcal{Y}$, to zero, we obtain:

$$\sum_{k=1}^{K} u(m,k)\Bigg(\ln\Big(\frac{\beta^{\ell}_{y_m^{(k)}}}{\beta^{\ell}_{y_m^{(k)}}+1}\Big) + \psi_0\big(\alpha^{\ell}_{y_m^{(k)}} + C_m^{(k)}\big) - \psi_0\big(\alpha^{\ell}_{y_m^{(k)}}\big)\Bigg) = 0.$$

Solving for $\alpha^{\ell}_{y_m^{(k)}}$ is done through the Newton-Rapson method (3), completing step 4 of the ECM.

Reduction of the transition matrix An important element of a HMM is the transition matrix between states and the meaning of each state. Thanks to the heuristic ECM, `HMMploidy` is able to assign a ploidy to each state of the Markov chain in an unsupervised mode without overfitting the data. However, one needs to check whether transitions between states follow a biological meaning. For example, it is unlikely that a ploidy occurs only in a small window of loci, and then shifts again to the previous value, i.e. such event is likely due to noise or other biological artefact altering the quality and behaviour of the data (e.g. the presence of a centromere).

Once the HECM algorithm has converged to a set of parameters $\lambda \in \Lambda$, it is possible to perform an optional filtering on the transition matrix $\boldsymbol{A}$ of the HMM. Given the matrix $\boldsymbol{A}$ of size $|\mathcal{Y}| \times |\mathcal{Y}|$, the time of permanence in a state $y \in \mathcal{Y}$ has geometric distribution with parameter $\boldsymbol{A}_{y,y}$ (5). If the user expects that a ploidy level has to remain uninterrupted for at least a certain number of windows N, then a corresponding minimum value for the parameter of the geometric distribution can be estimated. In fact, the probability of permanence in ploidy y for at least $N > 0$ windows is given by the CDF of the geometric distributions, that is, $1-(1-y)^N$.

Given N, `HMMploidy` calculates the minimum value of y that has to be on the diagonal of $\boldsymbol{A}$. Rows and columns corresponding to diagonal entries lower than y are cancelled and $\boldsymbol{A}$ is rendered stochastic again. Corresponding values of δ, α, β are also removed. Afterwards, the HMM optimization is performed again on the new subset of ploidies for adjustment of the remaining parameters.

Application in presence of sparse polymorphic sites Given an individual m, consider each k-th window of its genome. In presence of very few polymorphic sites in each window, the genotype likelihoods might not be enough to determine the ploidy, especially when the data is at low-depth and in presence of error, as it is often the case with high-throughput data.

To consider this case, the option `useGeno` is added to the software. When `useGeno='yes'`, the HMM infers the ploidy numbers as explained in the main text. If otherwise, at first only the sequencing depth data is used to infer the hidden states of a negative-binomial HMM. This allows to have as large windows of loci as possible. Each latent state is then assigned a ploidy by maximising Equation (4) over all the windows with same hidden state.

6.6 Supplementary Figures

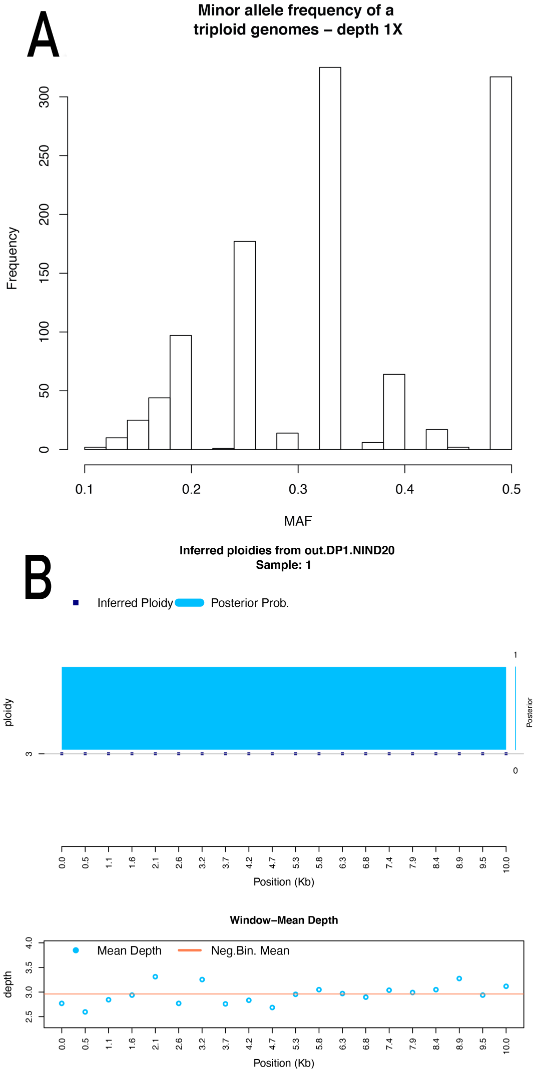

Figure S 2**Histograms of minor allele frequencies and inferred ploidy with `HMMploidy` at low depth.** (A) Distribution of the minor allele frequencies of one simulated triploid genomes (out of a sample of 20 individuals) of 10kbp at depth $1X$ for the haploid state. It is not trivial to determine the ploidy by visual inspection of this graph. (B) Inferred ploidy with `HMMploidy` from the same individual on windows of 500 bases. Using the information contained in all the other individuals, it is possible to infer the correct ploidy.

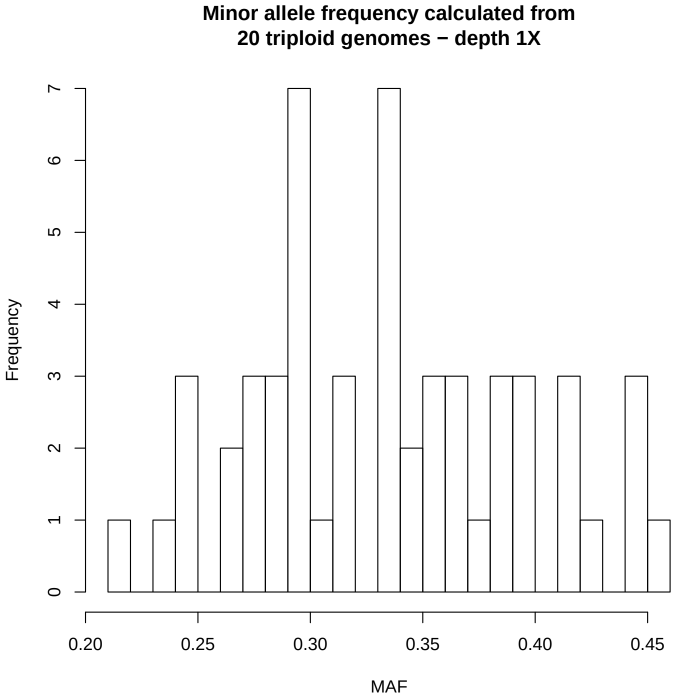

Figure S 3**Histograms of minor allele frequencies for many samples at low depth.** Histogram of the estimated minor allele frequency for 20 simulated triploid individuals in a window of 500 bases. The distribution is closer to the one expected for a triploid individual, but it is still not possible to infer the ploidy by a simple visual inspection of the graph. The use of genotype likelihoods in `HMMploidy` supplies additional information to infer the correct ploidy.

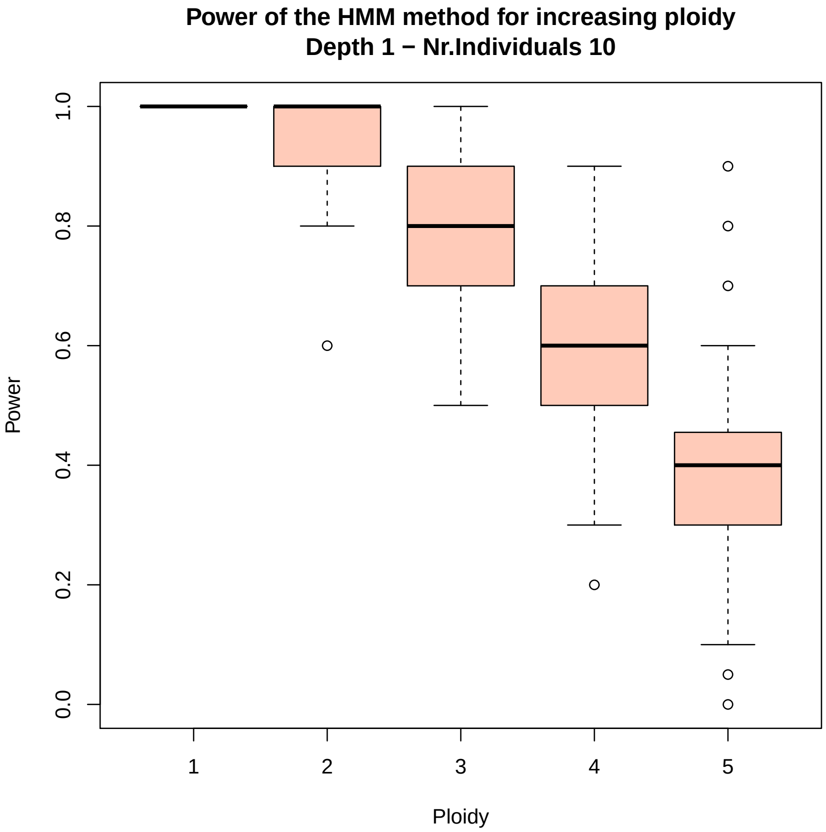

Figure S 4**Relationship between ploidy levels and detection rate.** Power of `HMMploidy` to detect the correct ploidy level (on y-axis) on simulated genomes with increasing ploidy (on x-axis) from one to six at depth $1X$. The power decreases with higher ploidy numbers because genotype likelihoods lack information to characterise correct genotypes.

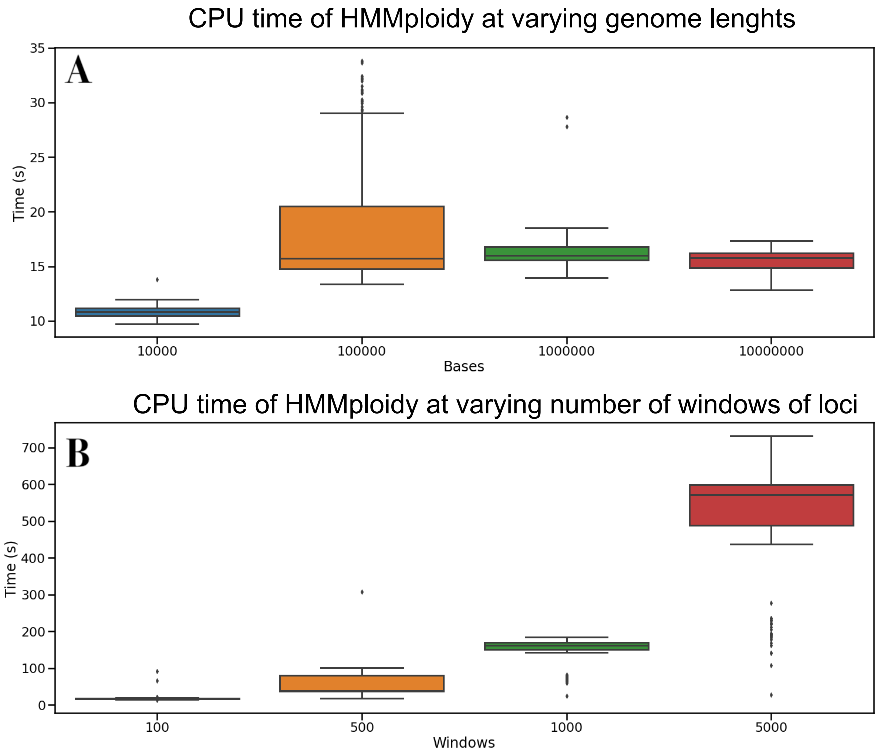

Figure S 5**CPU running time for `HMMploidy`.** (A) CPU running time of `HMMploidy` by simulating genomes of various lengths and keeping the windows number to 100. The time is quite constant, meaning that the loading and processing of the data is very fast, and most of the time is taken by the HMM inference. (B) CPU running time of `HMMploidy` by increasing the number of windows on a $10MB$ genome. The time grows accordingly with K in an almost linear fashion (due to a probable overhead for preprocessing the data in many windows), as predicted by the computational cost of the forward-backward algorithm.

6.7 Results from the analysis of *Cryptococcus Neoformans*

Here we present all the inferred ploidy levels from the 23 isolates of *Cryptococcus Neoformans* from the original study (19). Each figure contains:

- In the first line, inferred ploidy levels from chromosome 1 and 12 using the full data,
- In the second line, inferred ploidy levels from chromosome 1 and 12 using 20% of the original sequencing data.

Most of the results from the downsampled data coincide with the inference from the whole data. Higher ploidy levels can be hard to detect in some cases, and are occasionally detected as a constant lower ploidy , or as a highly varying sequence of adjacent ploidy levels . However, downsampling seems to recover a constant haploid chromosome 12 in sample cctp50 (Fig. S12B-D) according to what the sequencing depth indicates. This means that downsampling might reduce the effect of noisy data points that could alter the detected ploidy. In fact, the triploid sections of chromosome 12 are at the extremities of the chromosome, where the data is more affected by noise and in general by a lower sequencing quality.

All the other samples recover successfully the original ploidy levels in downsampled data. However, note that there are few changes in ploidy probably due to noise or to the presence of reads close to the centromere (Fig. S21, S16, S15, S13, S14).

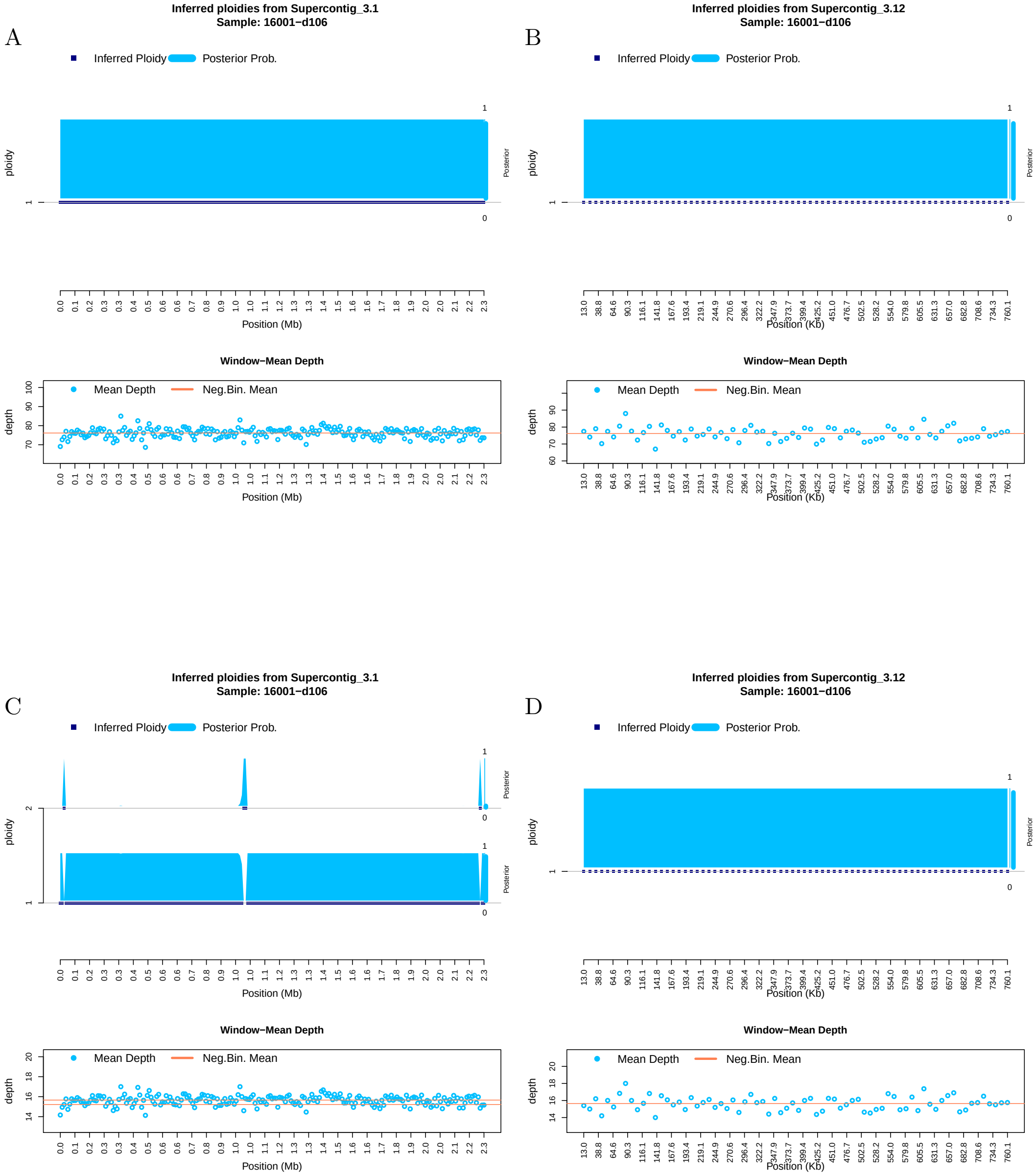

Figure S 6 **Ploidy inference on full and downsampled sequencing data.** Inferred ploidy levels from HMMploidy for chromosome 1 and 12 of isolate 16001-d106. **(A-B)** Results using the whole data on chromosomes 1 (A) and 12 (B). **(C-D)** Results using the data downsampled to 20% of its original depth on chromosomes 1 (C) and 12 (D).

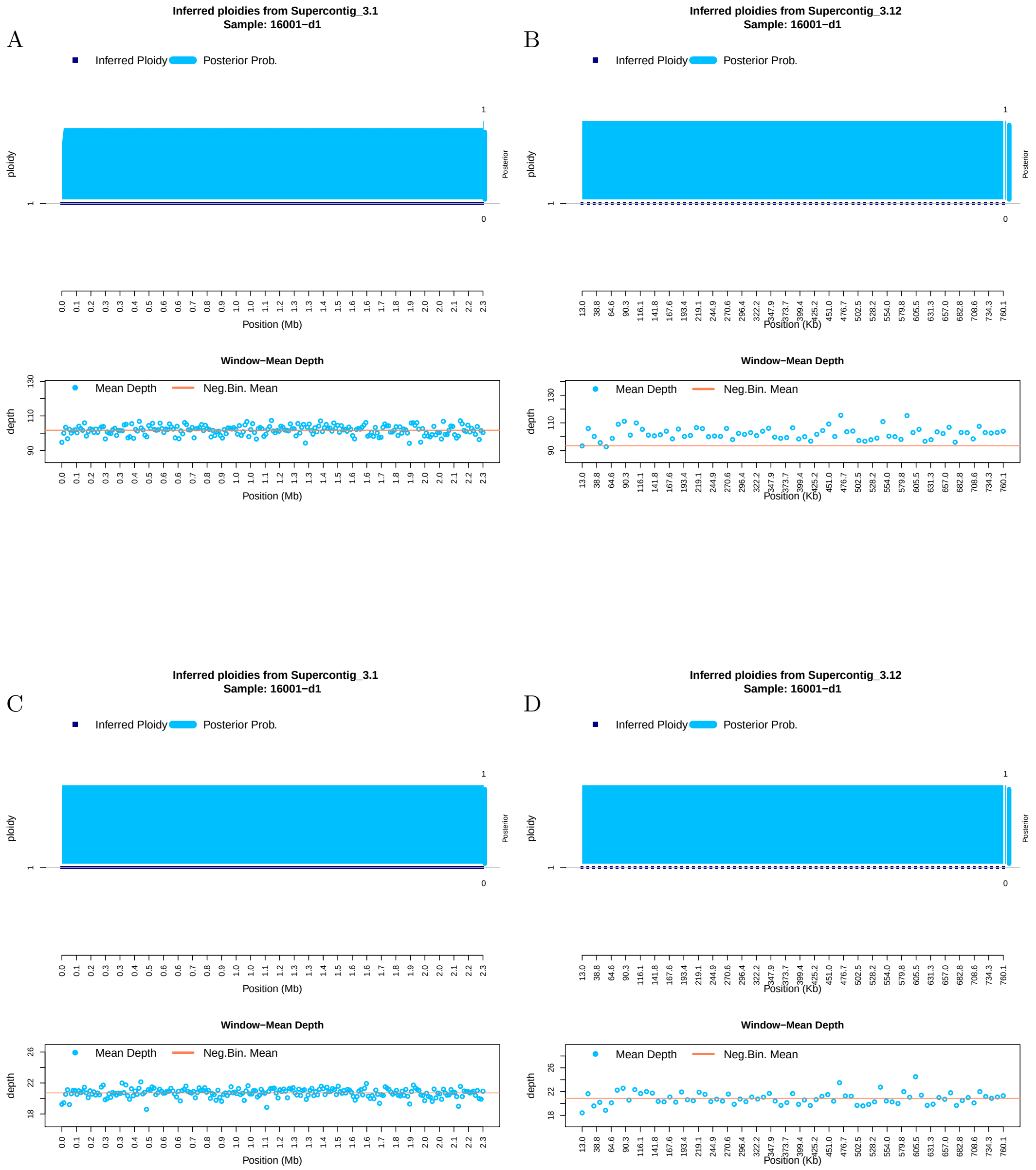

Figure S 7**Ploidy inference on full and downsampled sequencing data.** Inferred ploidy levels from `HMMploidy` for chromosome 1 and 12 of isolate 16001-d1. **(A-B)** Results using the whole data on chromosomes 1 (A) and 12 (B). **(C-D)** Results using the data downsampled to 20% of its original depth on chromosomes 1 (C) and 12 (D).

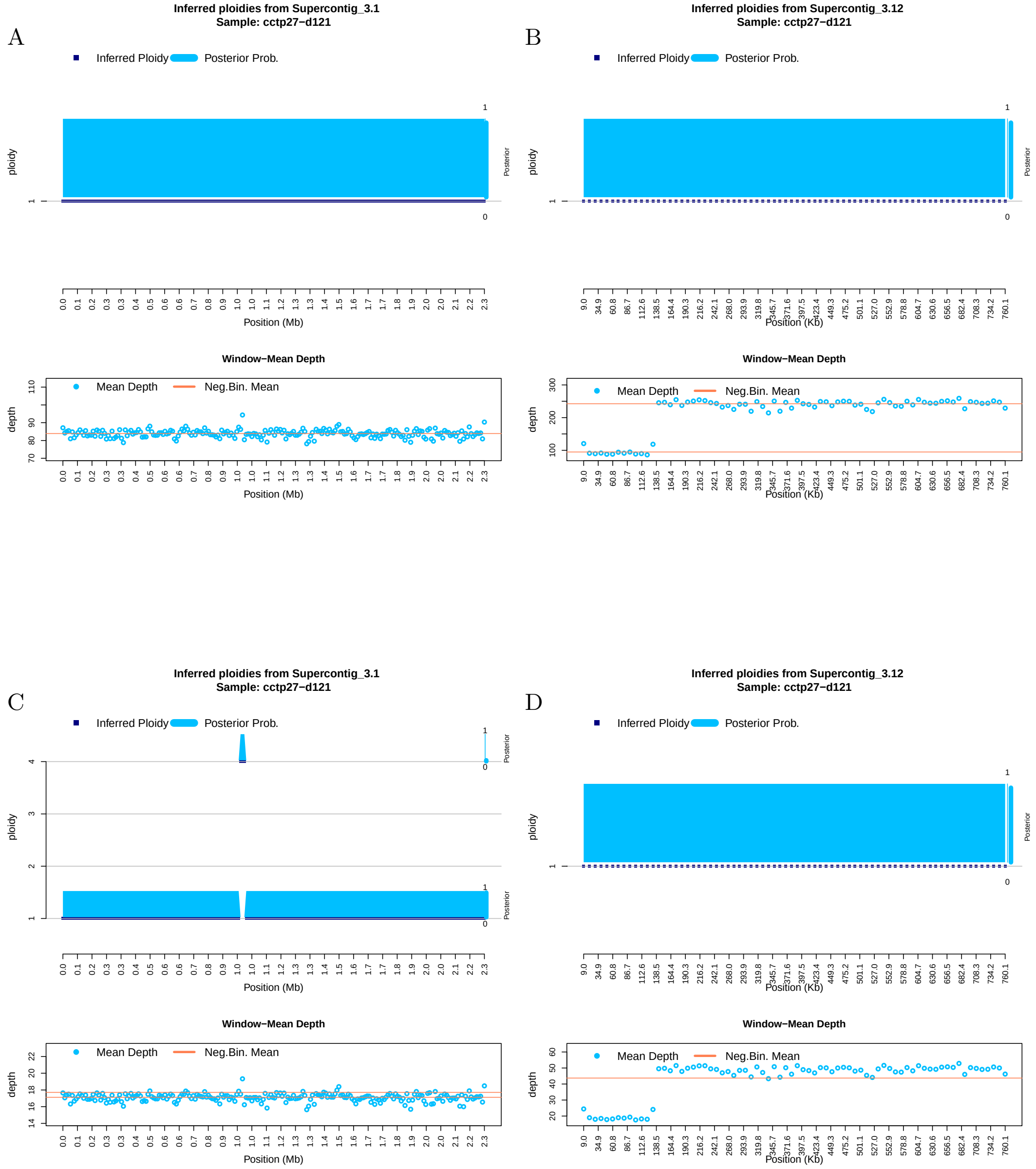

Figure S 8**Ploidy inference on full and downsampled sequencing data.** Inferred ploidy levels from `HMMploidy` for chromosome 1 and 12 of isolate cctp27-d121. **(A-B)** Results using the whole data on chromosomes 1 (A) and 12 (B). **(C-D)** Results using the data downsampled to 20% of its original depth on chromosomes 1 (C) and 12 (D).

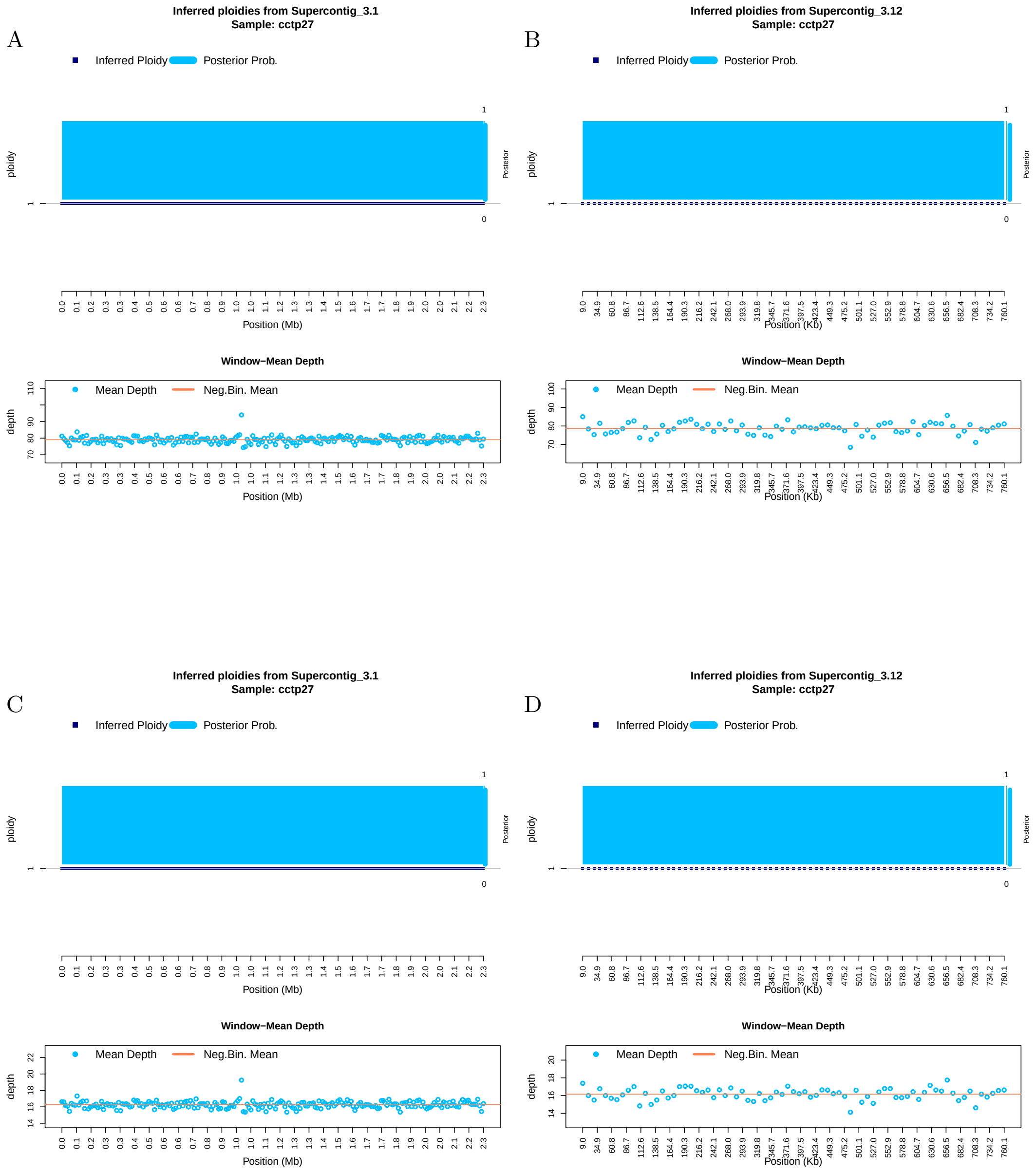

Figure S 9**Ploidy inference on full and downsampled sequencing data.** Inferred ploidy levels from HMMploidy for chromosome 1 and 12 of isolate cctp27. **(A-B)** Results using the whole data on chromosomes 1 (A) and 12 (B). **(C-D)** Results using the data downsampled to 20% of its original depth on chromosomes 1 (C) and 12 (D).

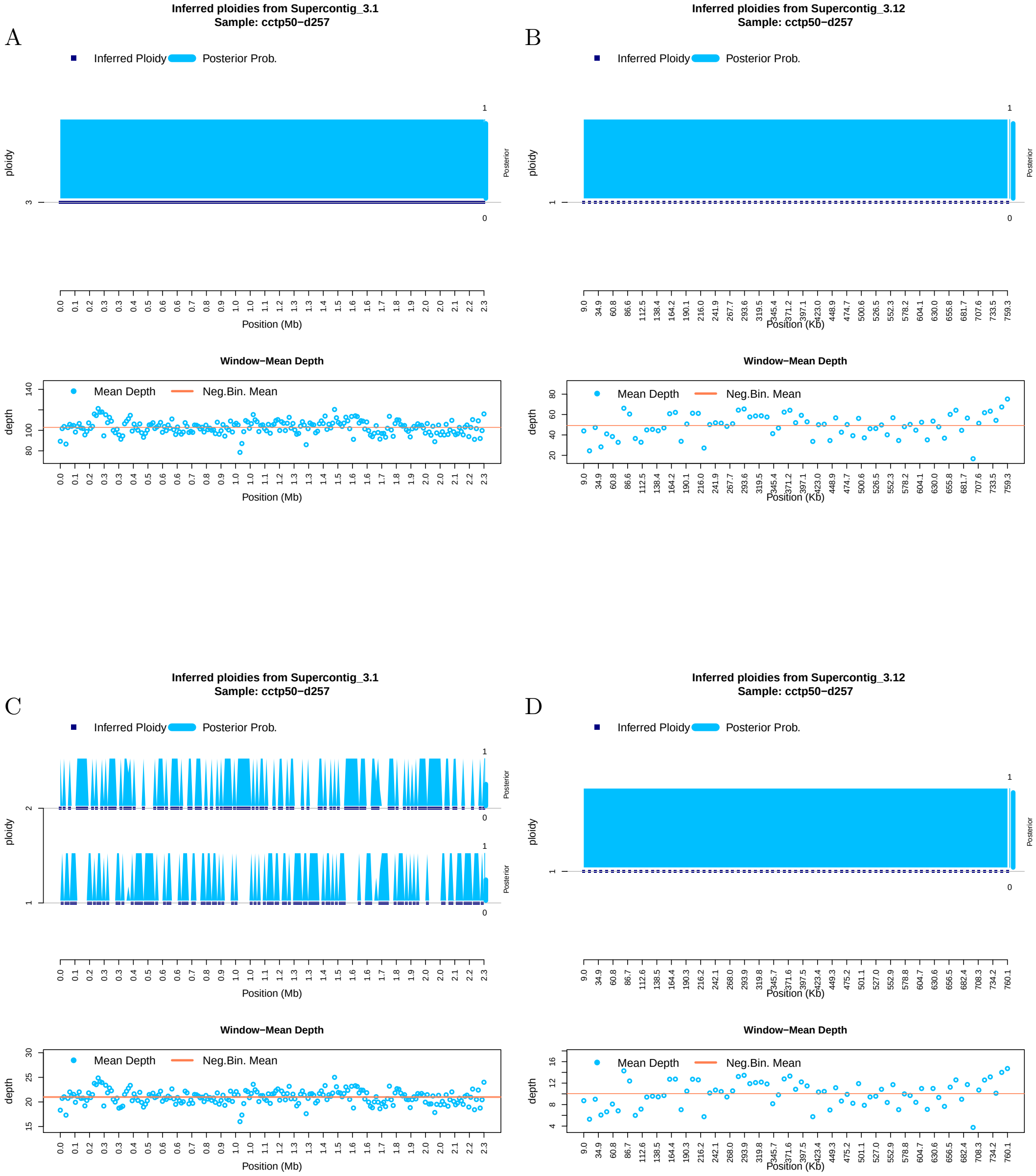

Figure S 10**Ploidy inference on full and downsampled sequencing data.** Inferred ploidy levels from `HMMploidy` for chromosome 1 and 12 of isolate cctp50-d257. **(A-B)** Results using the whole data on chromosomes 1 (A) and 12 (B). **(C-D)** Results using the data downsampled to 20% of its original depth on chromosomes 1 (C) and 12 (D).

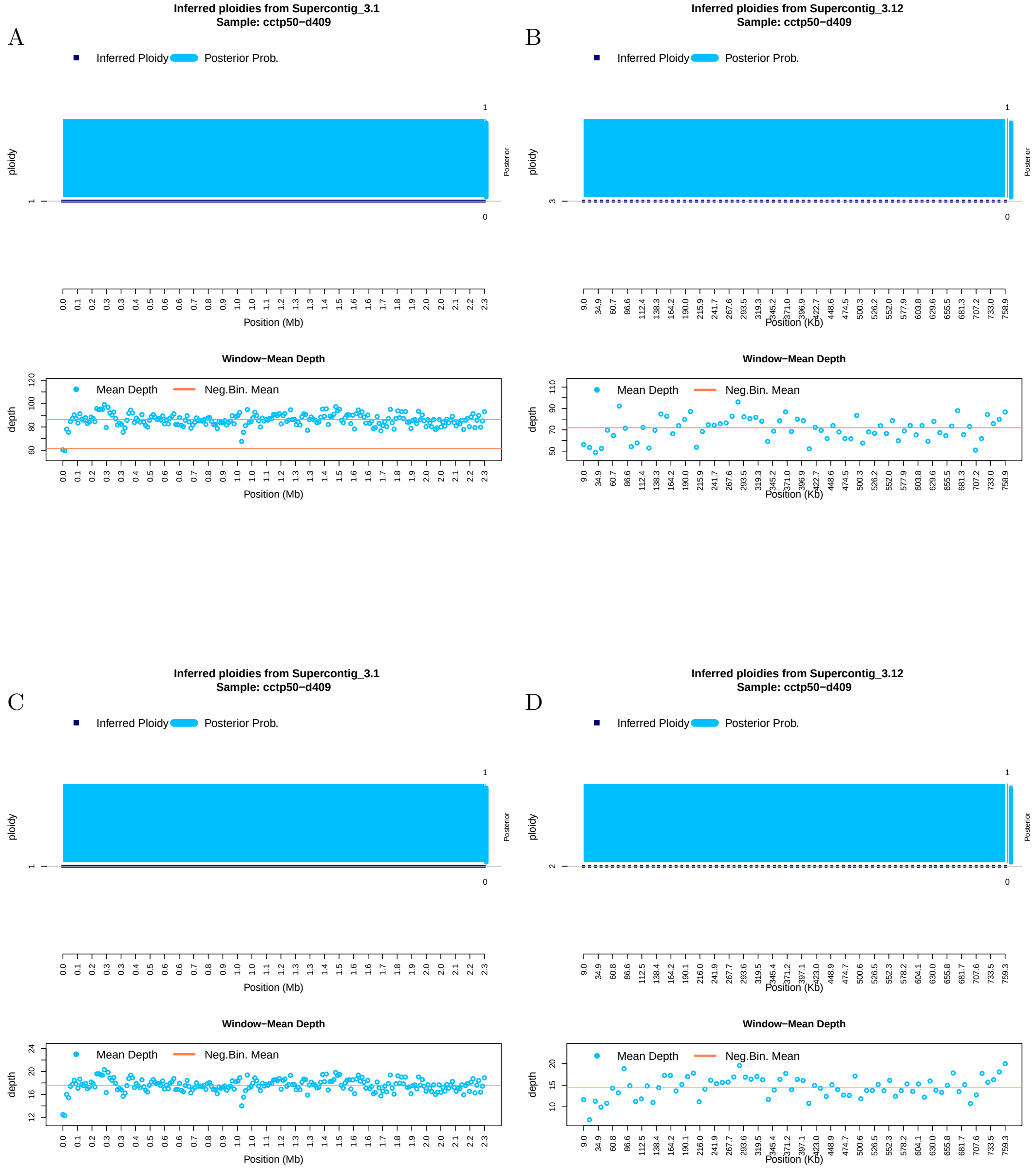

Figure S 11**Ploidy inference on full and downsampled sequencing data.** Inferred ploidy levels from `HMMploidy` for chromosome 1 and 12 of isolate cctp50-d409. **(A-B)** Results using the whole data on chromosomes 1 (A) and 12 (B). **(C-D)** Results using the data downsampled to 20% of its original depth on chromosomes 1 (C) and 12 (D).

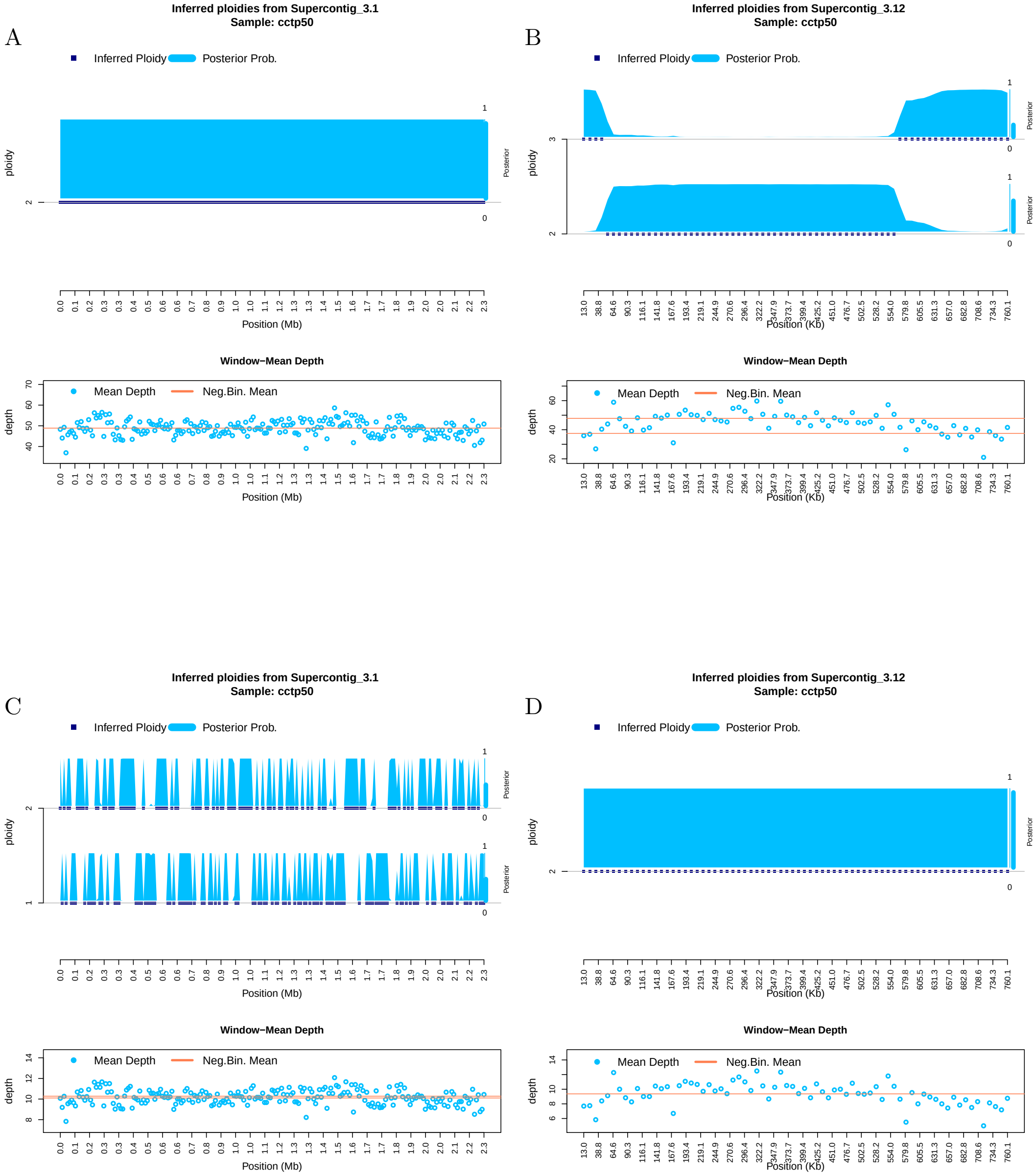

Figure S 12**Ploidy inference on full and downsampled sequencing data.** Inferred ploidy levels from `HMMploidy` for chromosome 1 and 12 of isolate cctp50. **(A-B)** Results using the whole data on chromosomes 1 (A) and 12 (B). **(C-D)** Results using the data downsampled to 20% of its original depth on chromosomes 1 (C) and 12 (D).

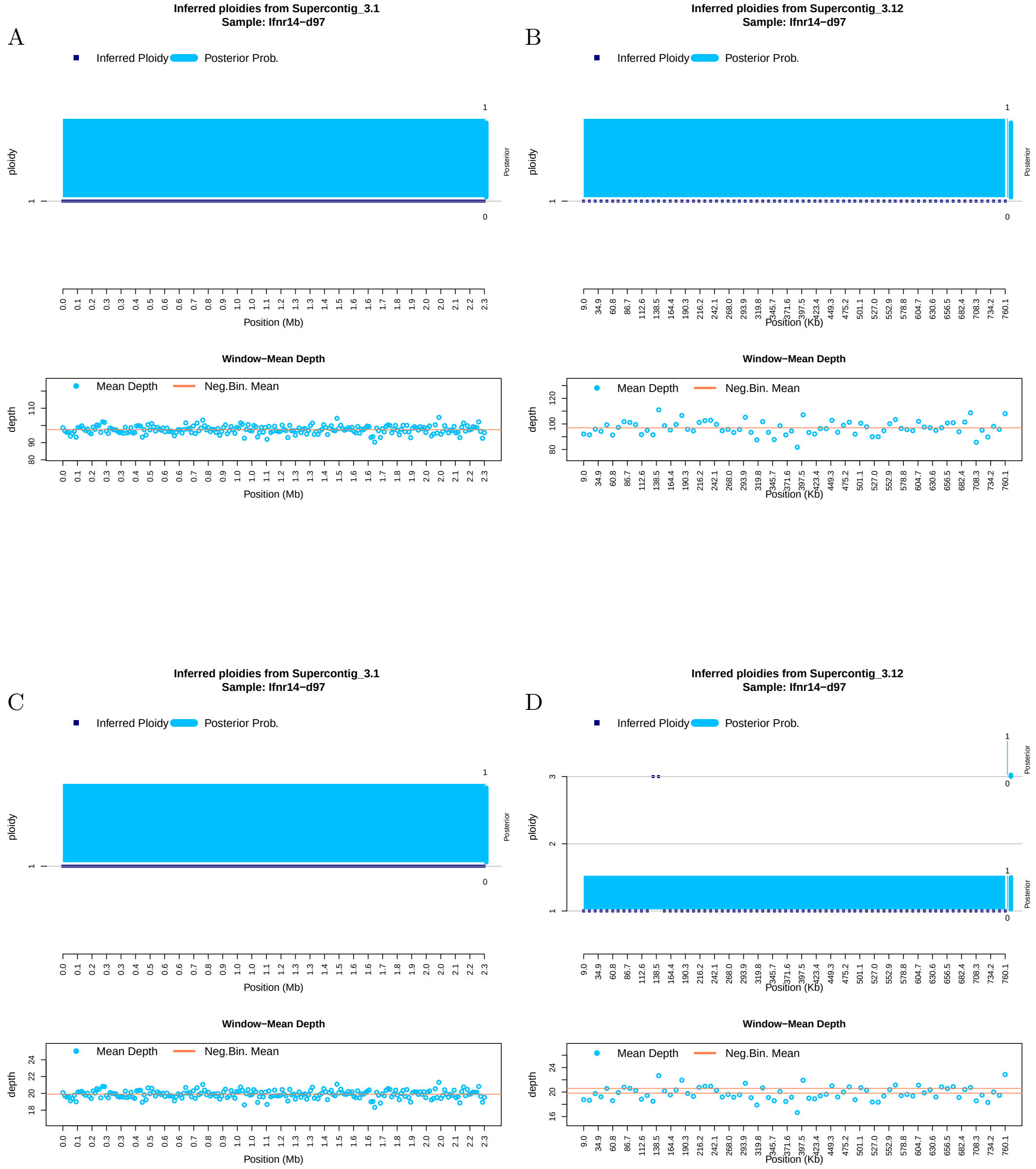

Figure S 13**Ploidy inference on full and downsampled sequencing data.** Inferred ploidy levels from `HMMploidy` for chromosome 1 and 12 of isolate ifnr14-d97. **(A-B)** Results using the whole data on chromosomes 1 (A) and 12 (B). **(C-D)** Results using the data downsampled to 20% of its original depth on chromosomes 1 (C) and 12 (D).

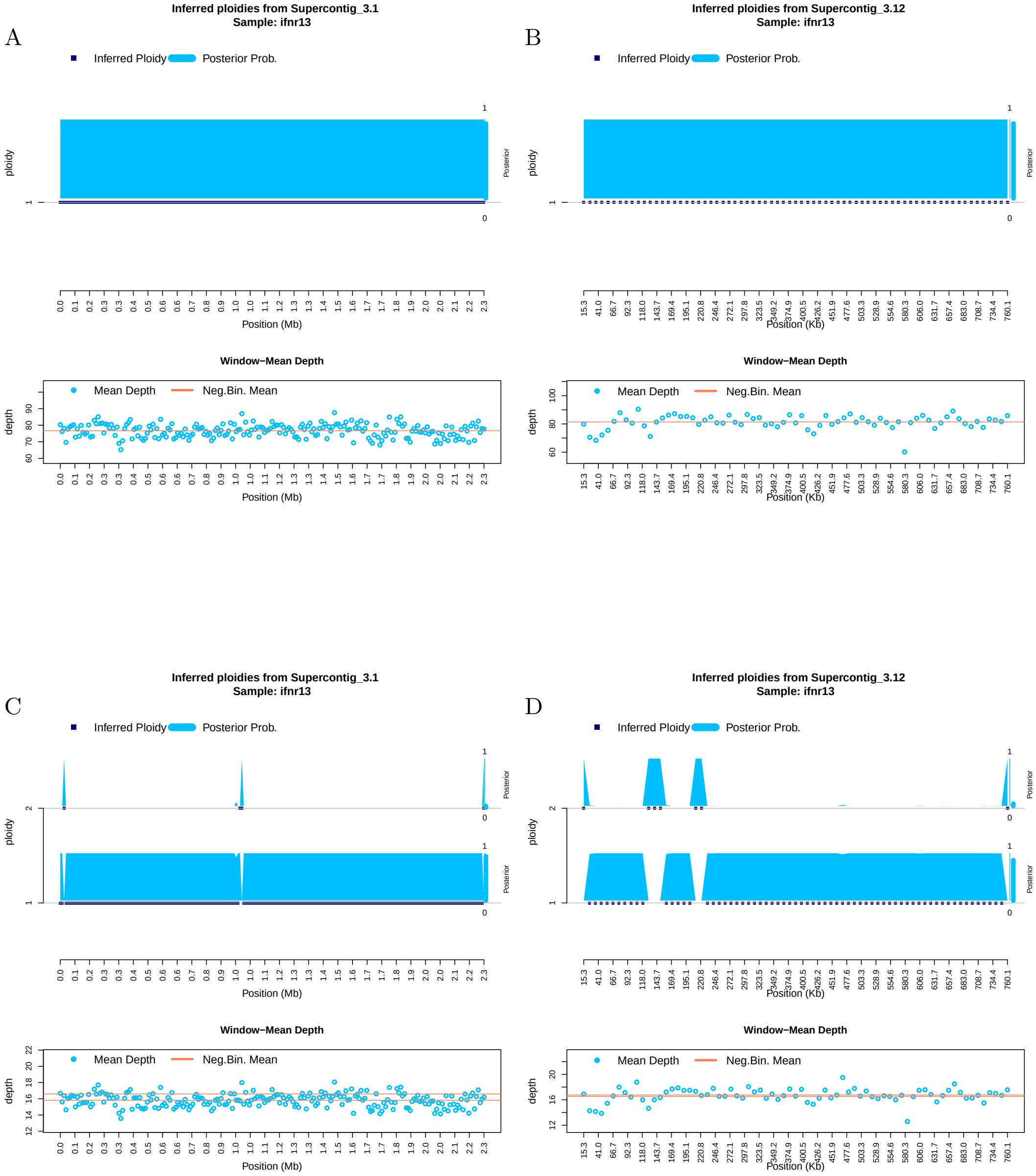

Figure S 14**Ploidy inference on full and downsampled sequencing data.** Inferred ploidy levels from `HMMploidy` for chromosome 1 and 12 of isolate ifnr13. **(A-B)** Results using the whole data on chromosomes 1 (A) and 12 (B). **(C-D)** Results using the data downsampled to 20% of its original depth on chromosomes 1 (C) and 12 (D).

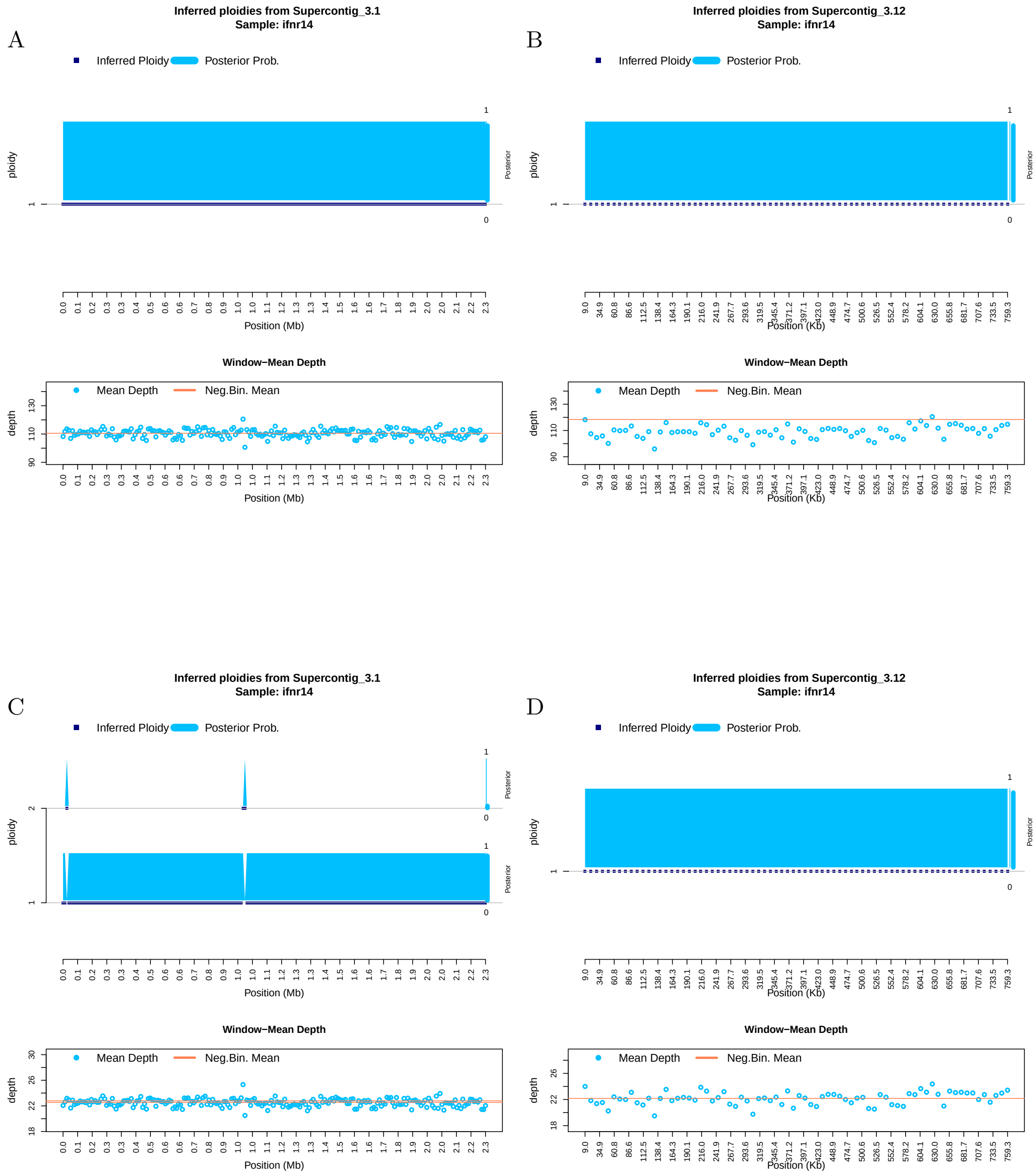

Figure S 15**Ploidy inference on full and downsampled sequencing data.** Inferred ploidy levels from `HMMploidy` for chromosome 1 and 12 of isolate ifnr14. **(A-B)** Results using the whole data on chromosomes 1 (A) and 12 (B). **(C-D)** Results using the data downsampled to 20% of its original depth on chromosomes 1 (C) and 12 (D).

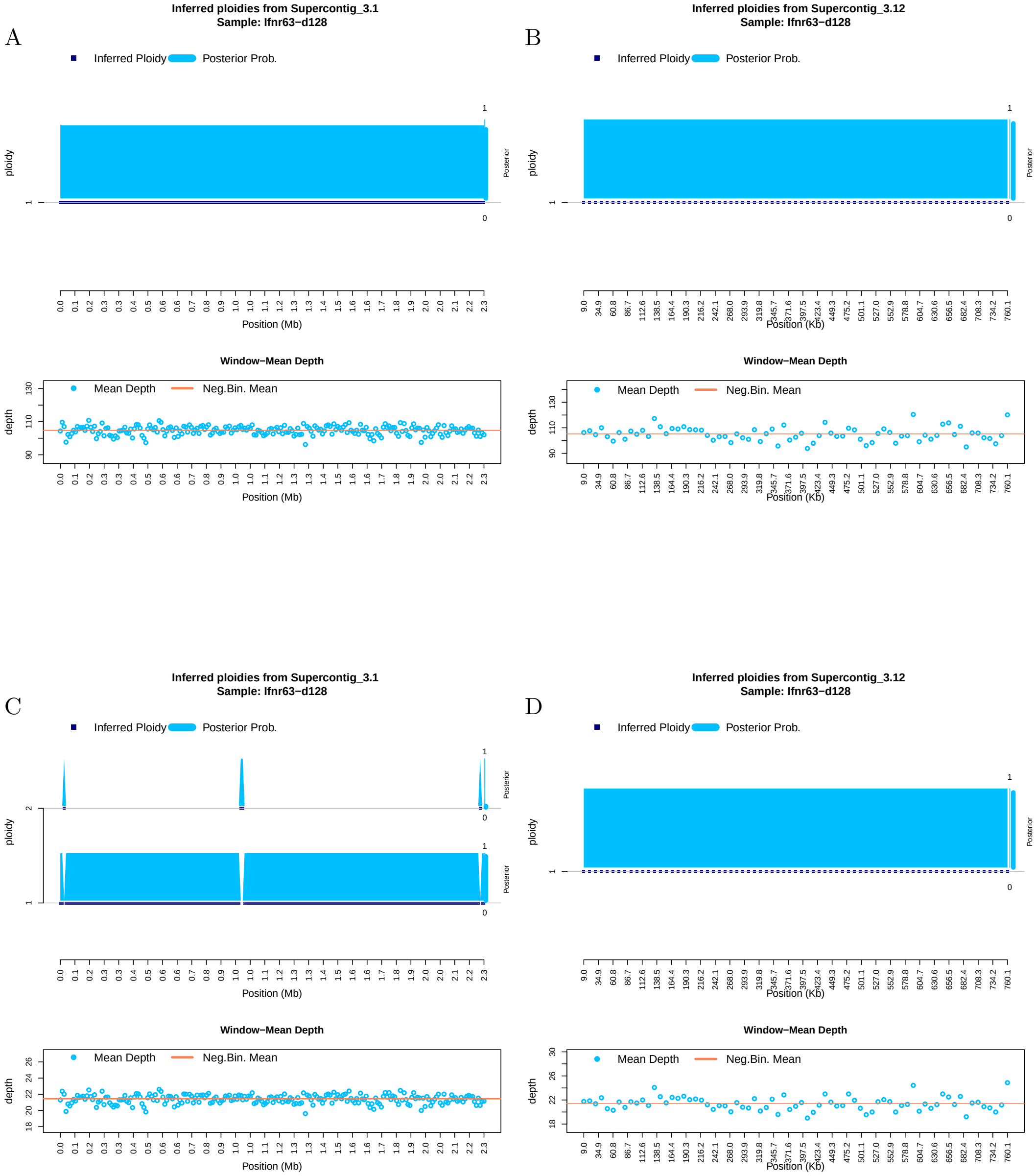

Figure S 16**Ploidy inference on full and downsampled sequencing data.** Inferred ploidy levels from `HMMploidy` for chromosome 1 and 12 of isolate ifnr63-d128. **(A-B)** Results using the whole data on chromosomes 1 (A) and 12 (B). **(C-D)** Results using the data downsampled to 20% of its original depth on chromosomes 1 (C) and 12 (D).

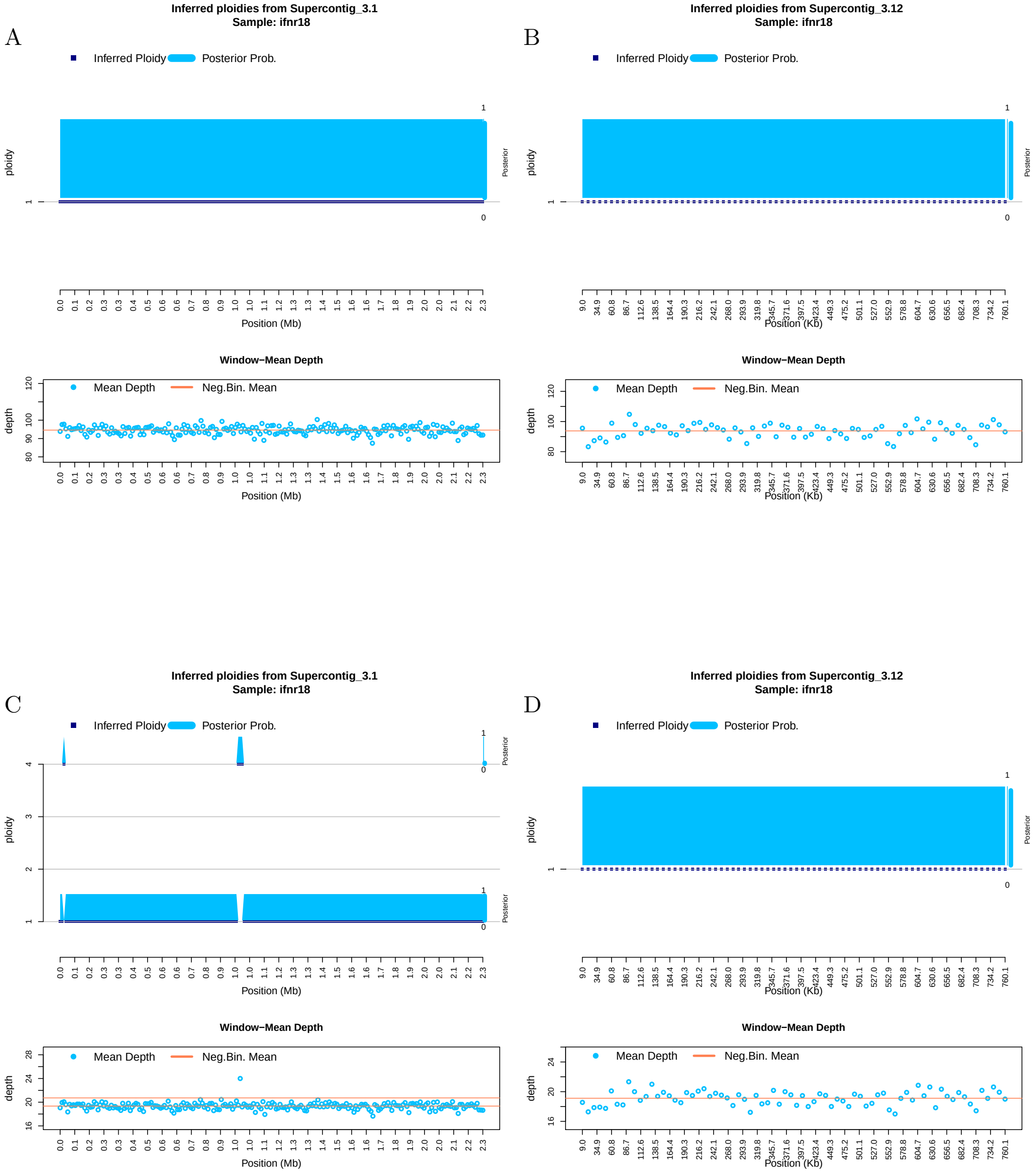

Figure S 17**Ploidy inference on full and downsampled sequencing data.** Inferred ploidy levels from `HMMploidy` for chromosome 1 and 12 of isolate ifnr18. **(A-B)** Results using the whole data on chromosomes 1 (A) and 12 (B). **(C-D)** Results using the data downsampled to 20% of its original depth on chromosomes 1 (C) and 12 (D).

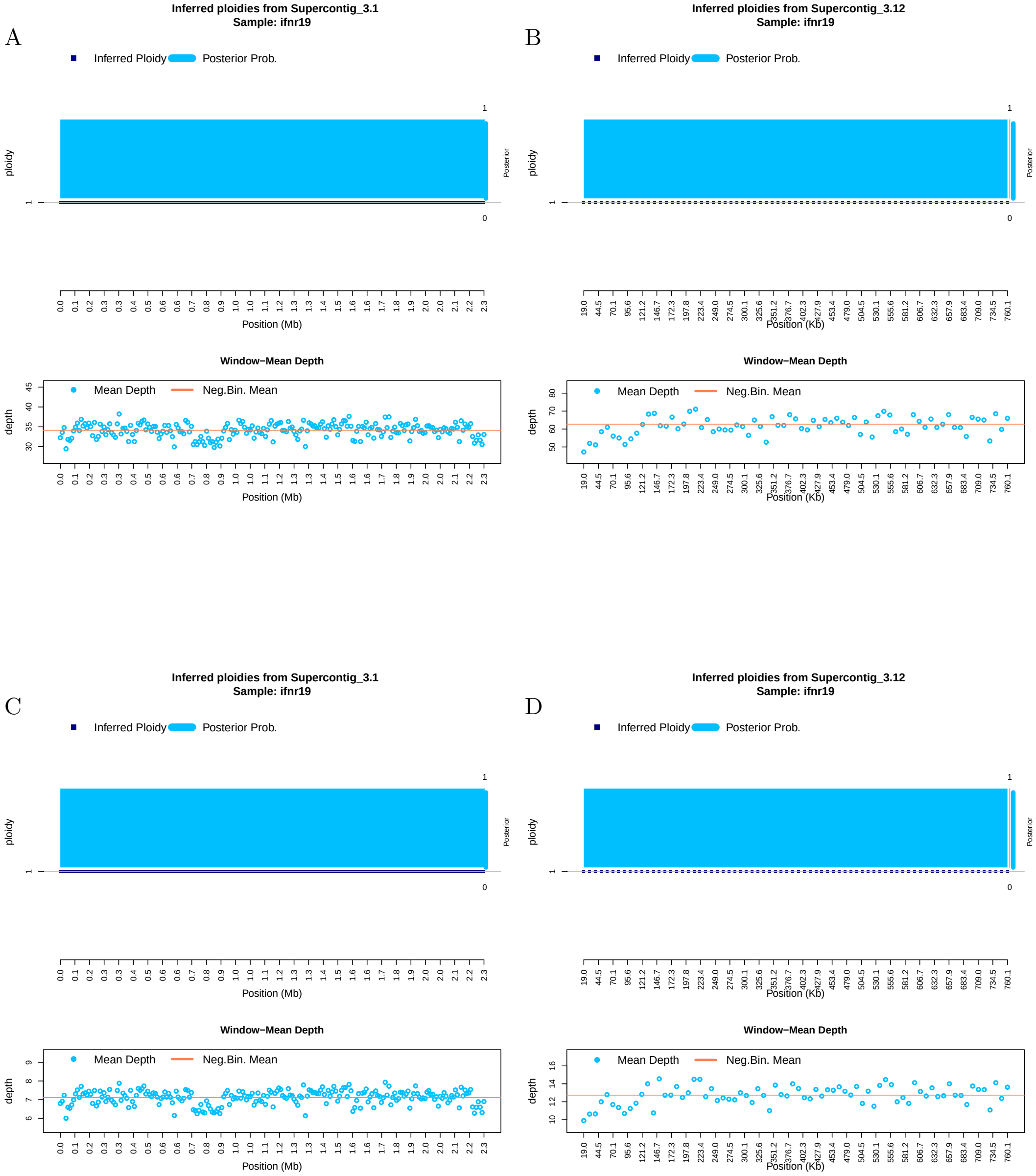

Figure S 18**Ploidy inference on full and downsampled sequencing data.** Inferred ploidy levels from HMMploidy for chromosome 1 and 12 of isolate ifnr19. **(A-B)** Results using the whole data on chromosomes 1 (A) and 12 (B). **(C-D)** Results using the data downsampled to 20% of its original depth on chromosomes 1 (C) and 12 (D).

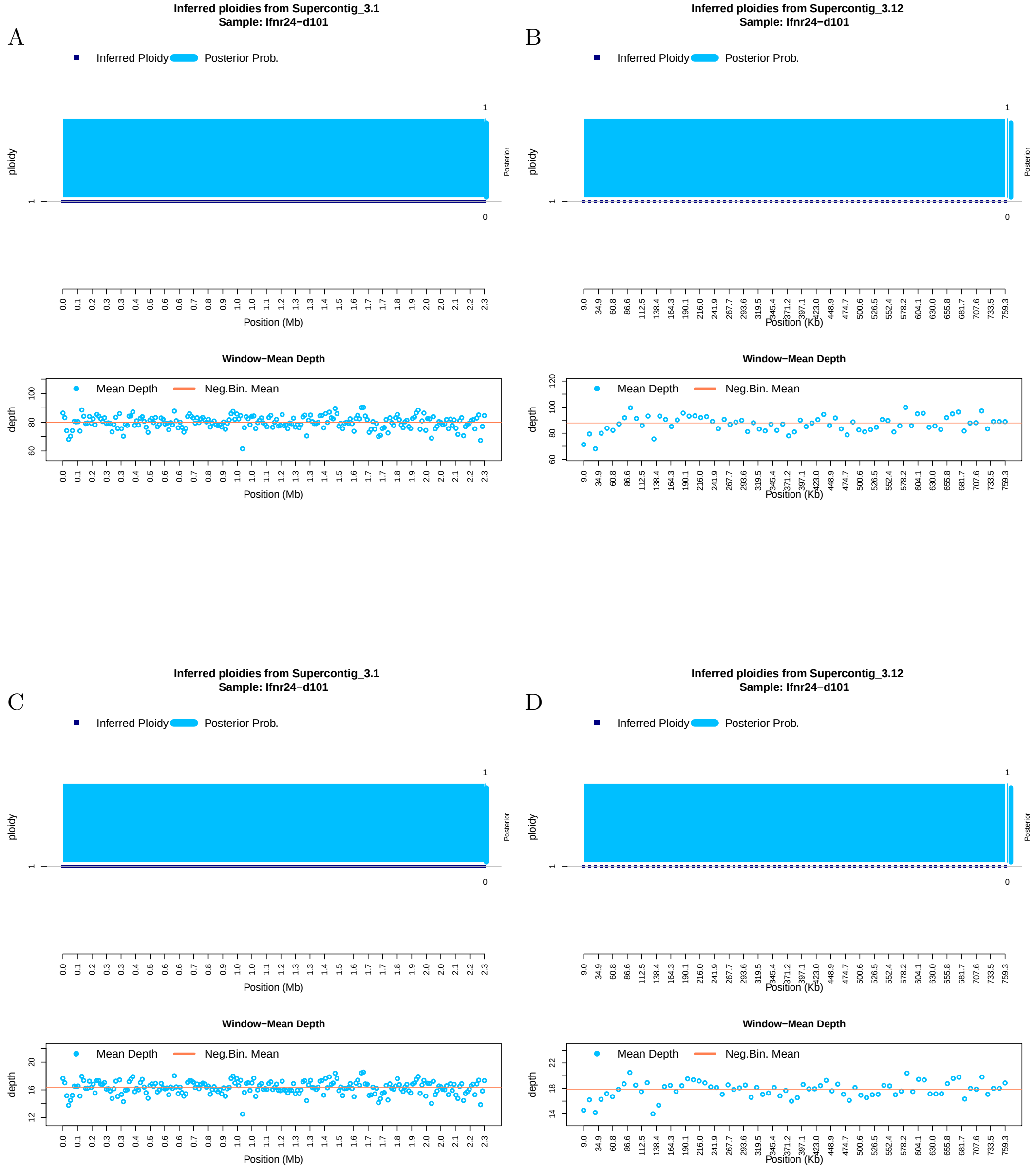

Figure S 19 **Ploidy inference on full and downsampled sequencing data.** Inferred ploidy levels from `HMMploidy` for chromosome 1 and 12 of isolate ifnr24-d101. **(A-B)** Results using the whole data on chromosomes 1 (A) and 12 (B). **(C-D)** Results using the data downsampled to 20% of its original depth on chromosomes 1 (C) and 12 (D).

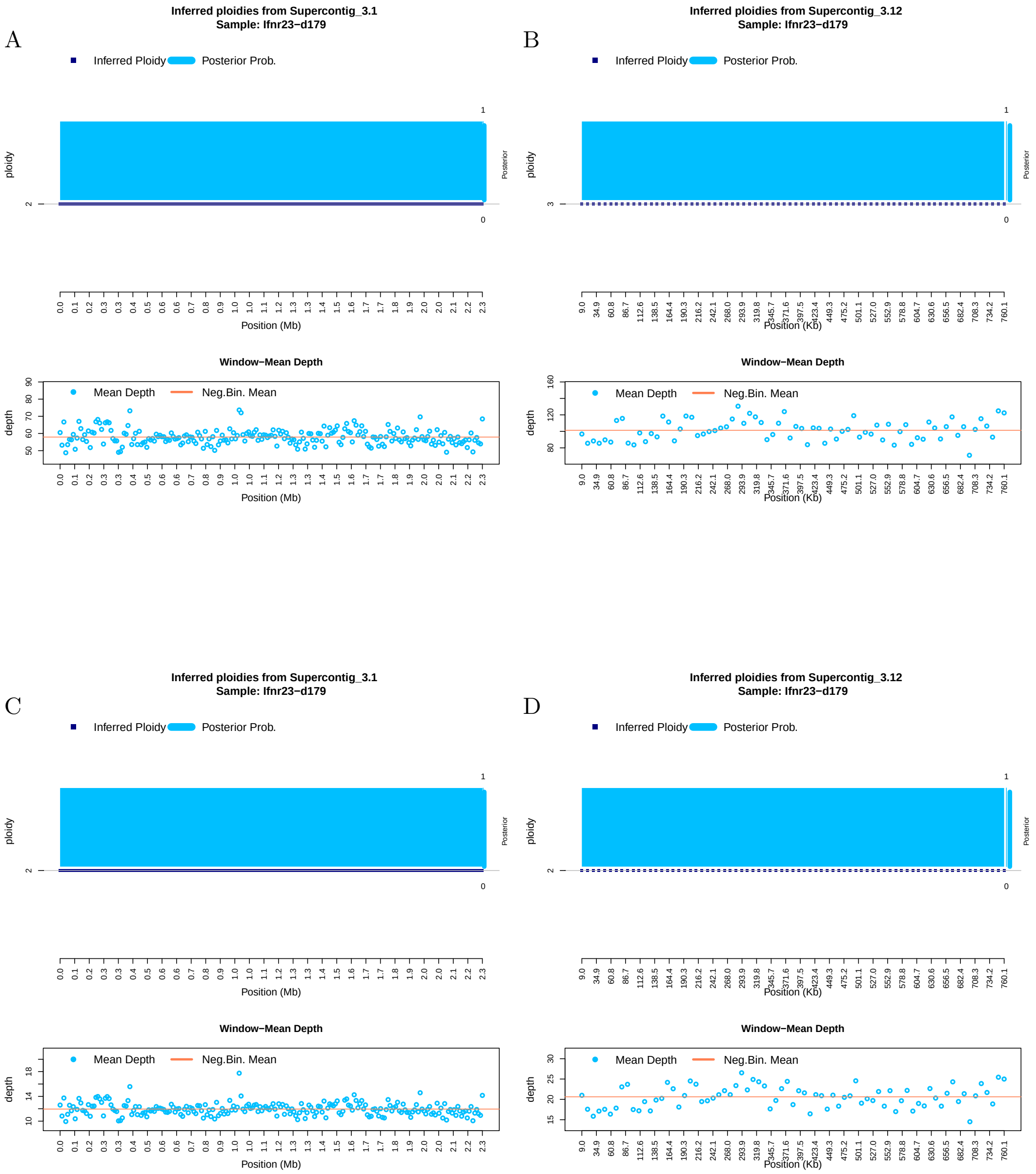

Figure S 20**Ploidy inference on full and downsampled sequencing data.** Inferred ploidy levels from `HMMploidy` for chromosome 1 and 12 of isolate ifnr23-d179. **(A-B)** Results using the whole data on chromosomes 1 (A) and 12 (B). **(C-D)** Results using the data downsampled to 20% of its original depth on chromosomes 1 (C) and 12 (D).

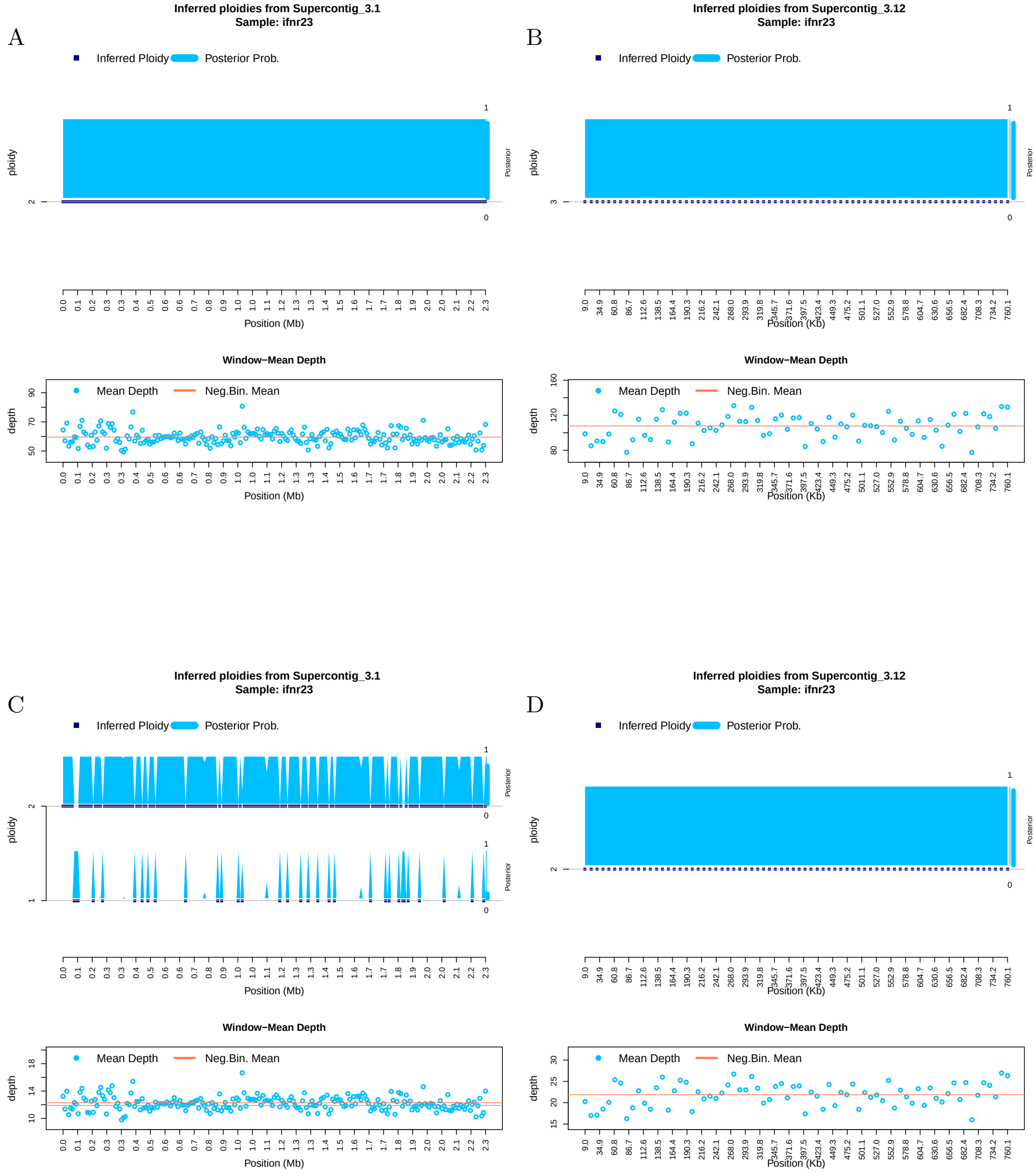

Figure S 21**Ploidy inference on full and downsampled sequencing data.** Inferred ploidy levels from `HMMploidy` for chromosome 1 and 12 of isolate ifnr23. **(A-B)** Results using the whole data on chromosomes 1 (A) and 12 (B). **(C-D)** Results using the data downsampled to 20% of its original depth on chromosomes 1 (C) and 12 (D).

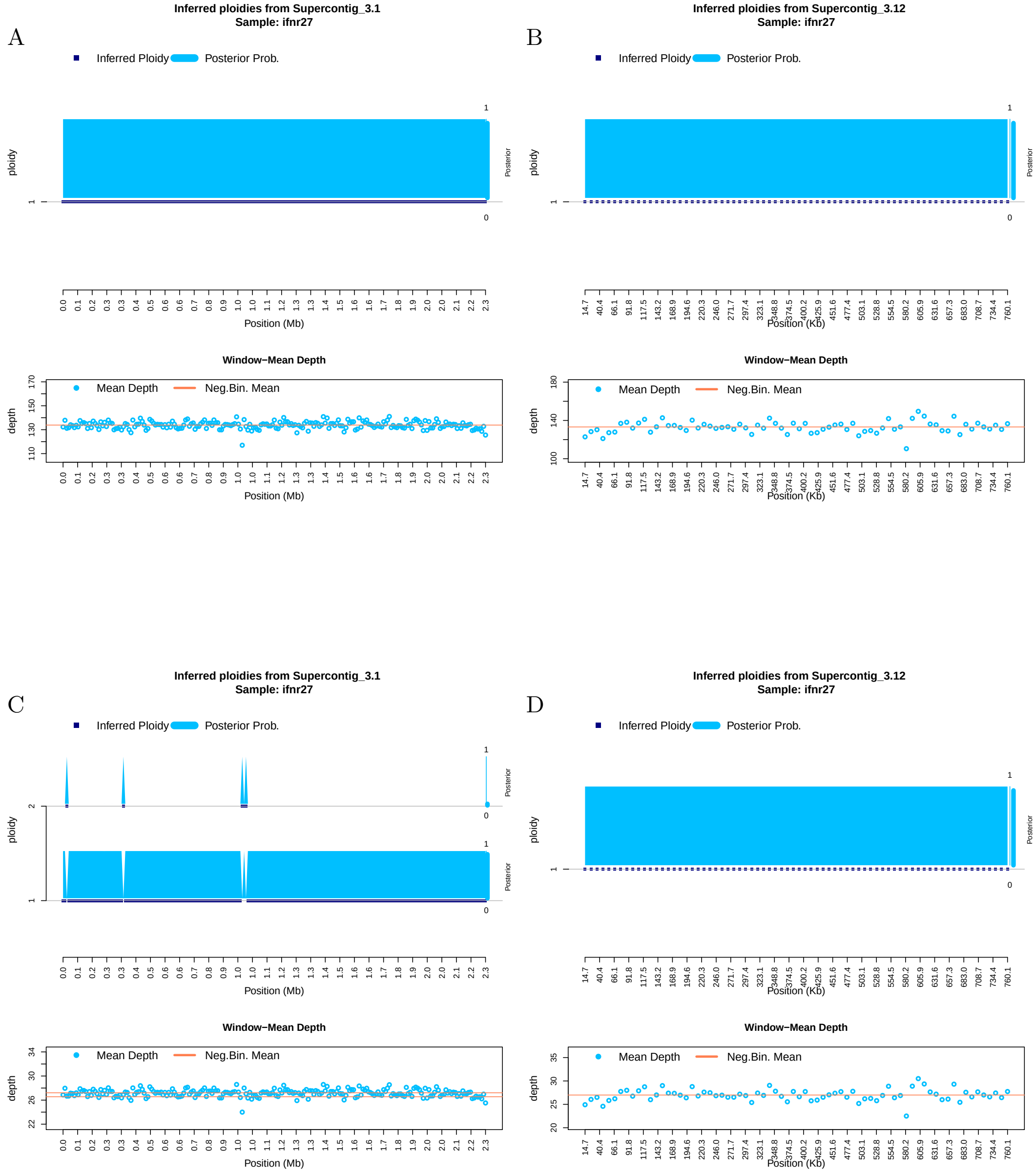

Figure S 22**Ploidy inference on full and downsampled sequencing data.** Inferred ploidy levels from `HMMploidy` for chromosome 1 and 12 of isolate ifnr27. **(A-B)** Results using the whole data on chromosomes 1 (A) and 12 (B). **(C-D)** Results using the data downsampled to 20% of its original depth on chromosomes 1 (C) and 12 (D).

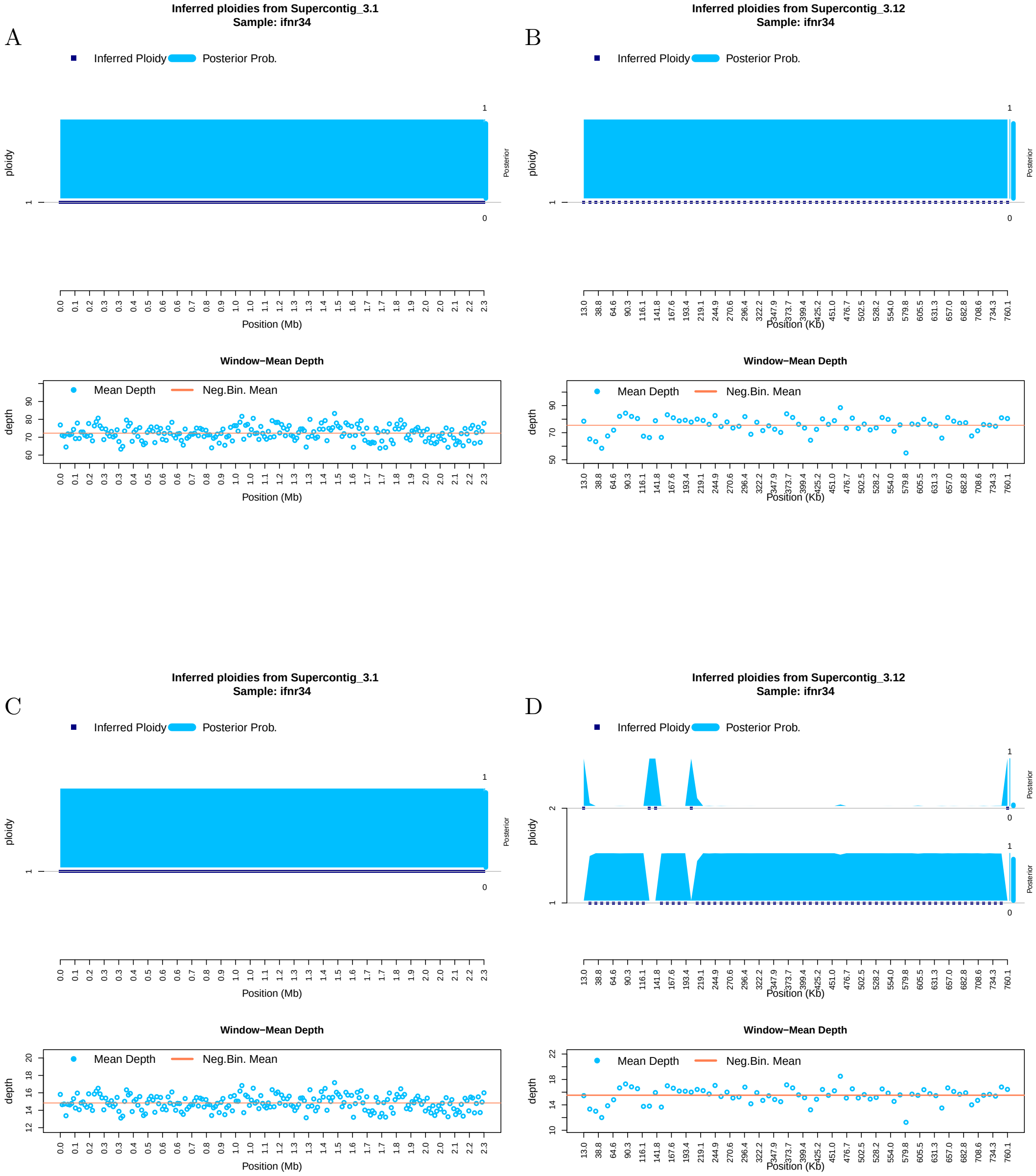

Figure S 23**Ploidy inference on full and downsampled sequencing data.** Inferred ploidy levels from HMMploidy for chromosome 1 and 12 of isolate ifnr34. **(A-B)** Results using the whole data on chromosomes 1 (A) and 12 (B). **(C-D)** Results using the data downsampled to 20% of its original depth on chromosomes 1 (C) and 12 (D).

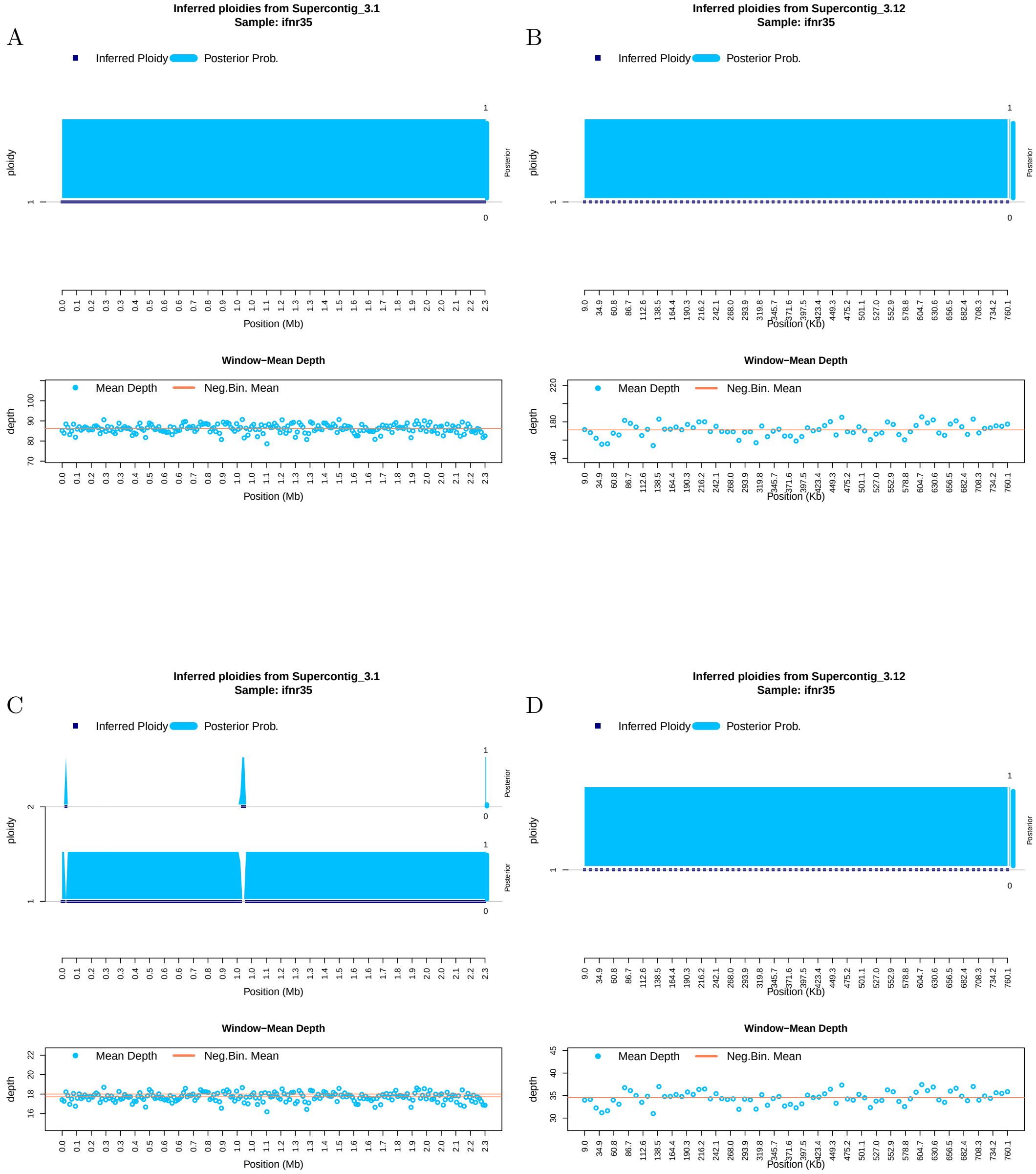

Figure S 24**Ploidy inference on full and downsampled sequencing data.** Inferred ploidy levels from `HMMploidy` for chromosome 1 and 12 of isolate ifnr35. **(A-B)** Results using the whole data on chromosomes 1 (A) and 12 (B). **(C-D)** Results using the data downsampled to 20% of its original depth on chromosomes 1 (C) and 12 (D).

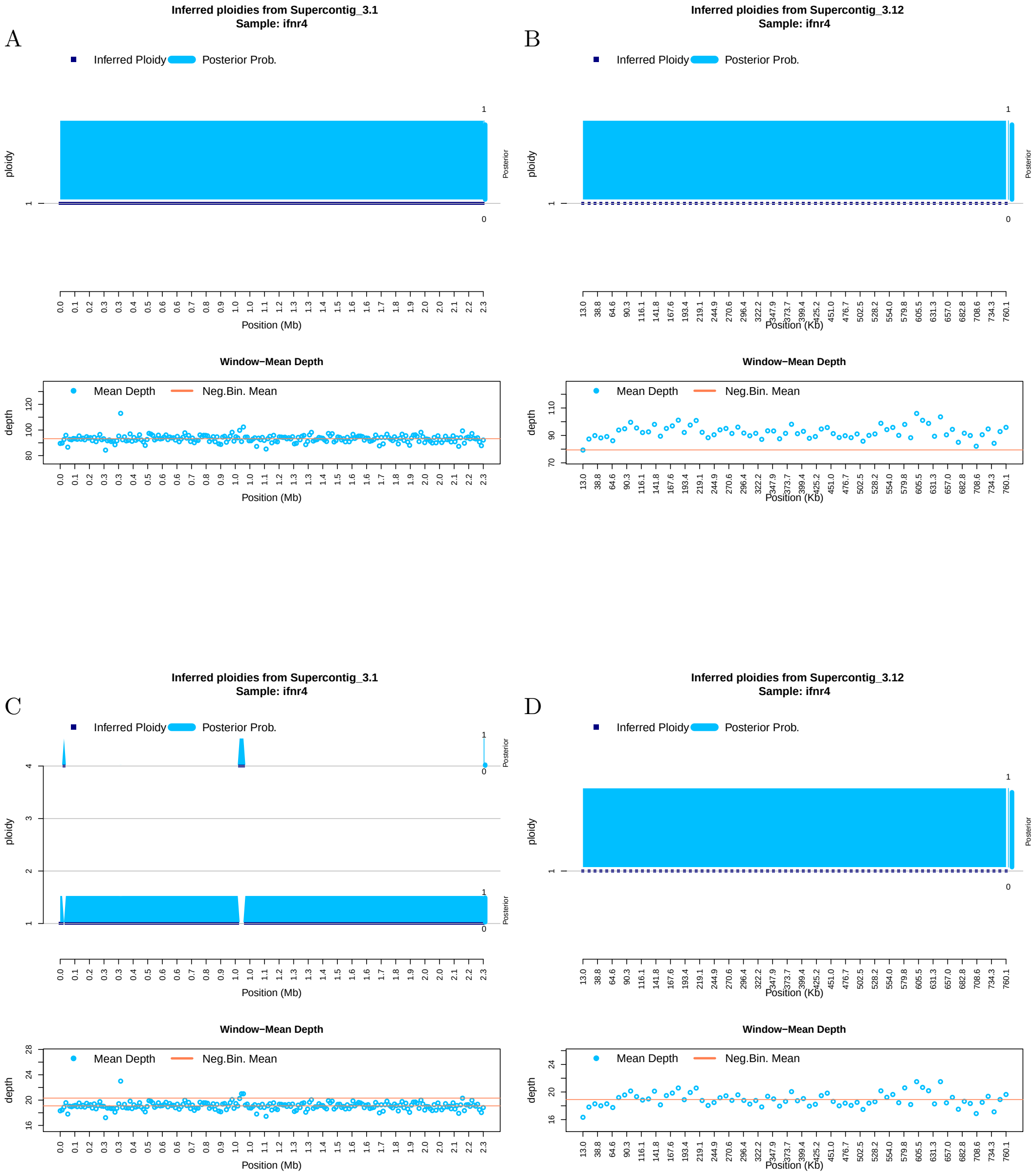

Figure S 25**Ploidy inference on full and downsampled sequencing data.** Inferred ploidy levels from `HMMploidy` for chromosome 1 and 12 of isolate ifnr4. **(A-B)** Results using the whole data on chromosomes 1 (A) and 12 (B). **(C-D)** Results using the data downsampled to 20% of its original depth on chromosomes 1 (C) and 12 (D).

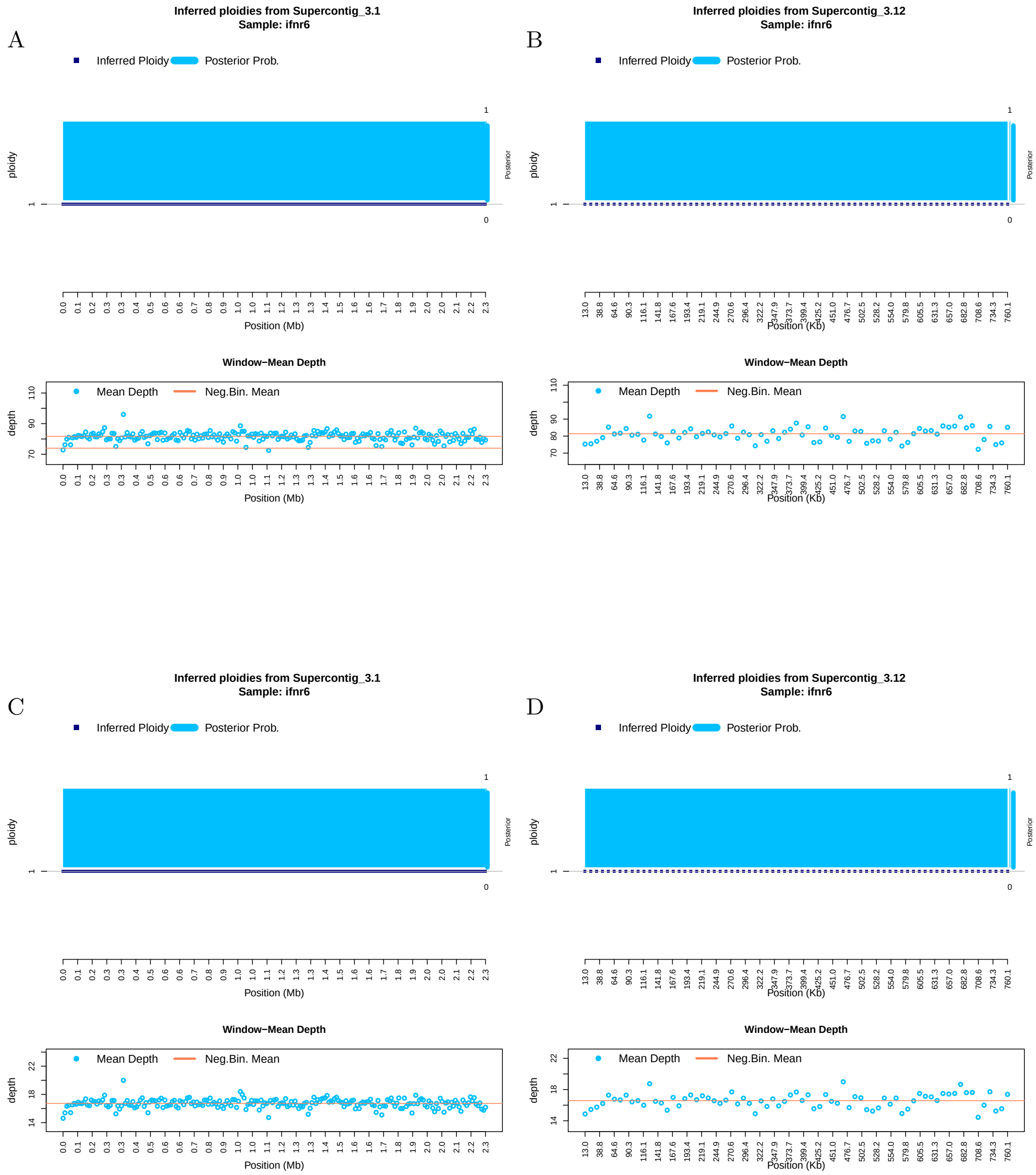

Figure S 26 **Ploidy inference on full and downsampled sequencing data.** Inferred ploidy levels from `HMMploidy` for chromosome 1 and 12 of isolate ifnr6. **(A-B)** Results using the whole data on chromosomes 1 (A) and 12 (B). **(C-D)** Results using the data downsampled to 20% of its original depth on chromosomes 1 (C) and 12 (D).

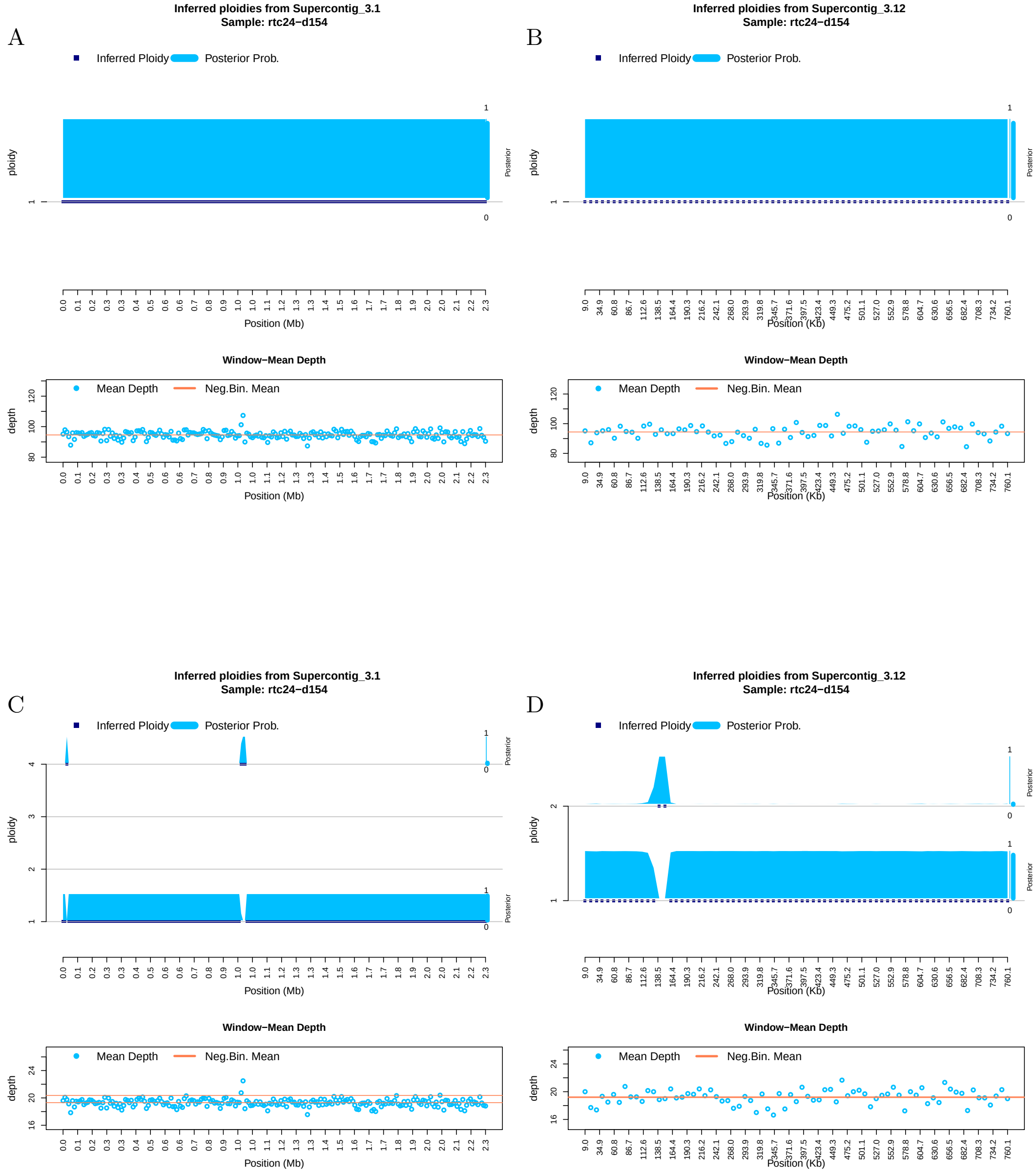

Figure S 27**Ploidy inference on full and downsampled sequencing data.** Inferred ploidy levels from `HMMploidy` for chromosome 1 and 12 of isolate rtc24-d154. **(A-B)** Results using the whole data on chromosomes 1 (A) and 12 (B). **(C-D)** Results using the data downsampled to 20% of its original depth on chromosomes 1 (C) and 12 (D).

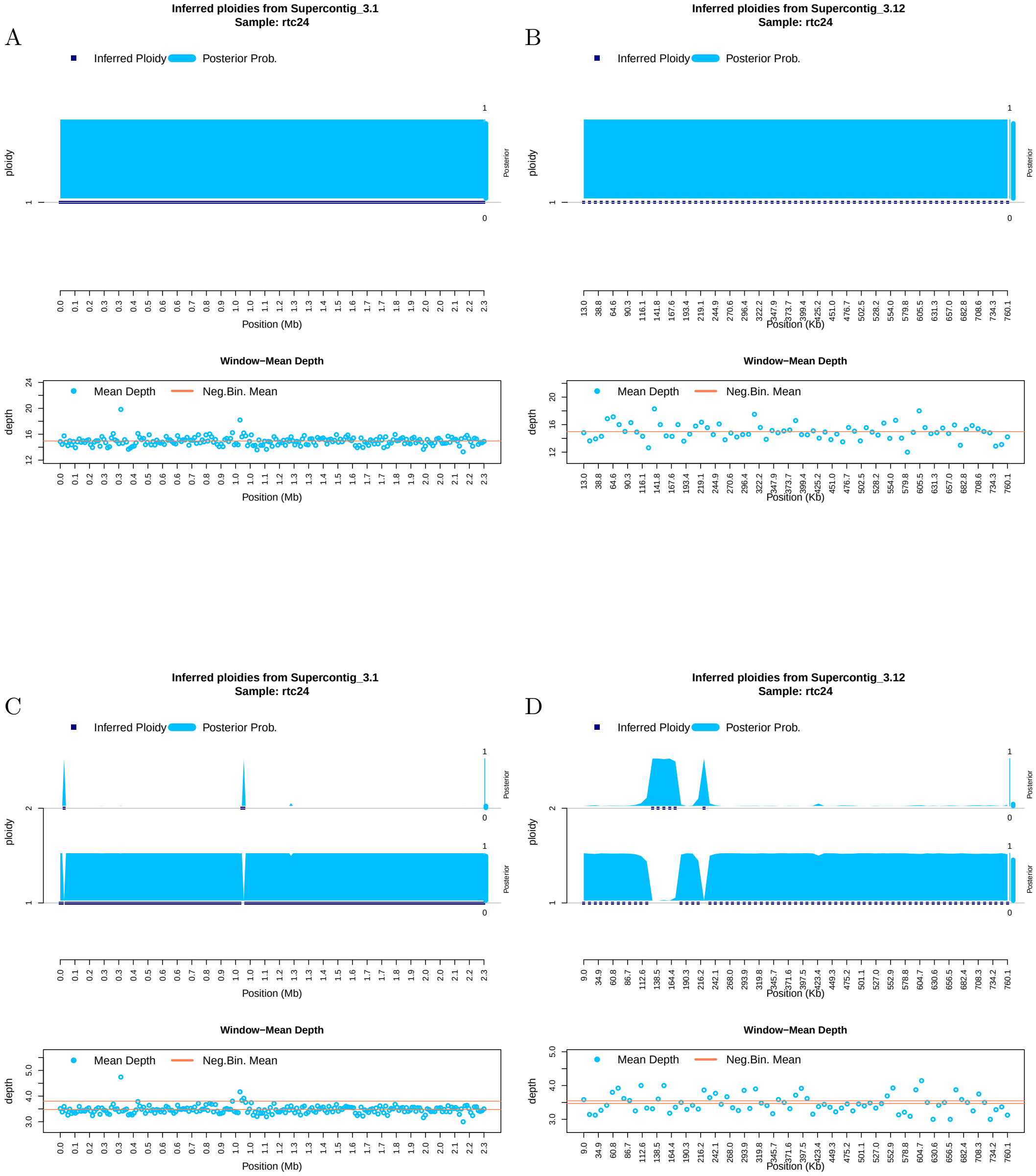

Figure S 28**Ploidy inference on full and downsampled sequencing data.** Inferred ploidy levels from `HMMploidy` for chromosome 1 and 12 of isolate rtc24. **(A-B)** Results using the whole data on chromosomes 1 (A) and 12 (B). **(C-D)** Results using the data downsampled to 20% of its original depth on chromosomes 1 (C) and 12 (D).

Bibliography

[1] Augusto Corrêa dos Santos, R., Goldman, G.H., Riaño-Pachón, D.M.: ploidyNGS: visually exploring ploidy with Next Generation Sequencing data. Bioinformatics **33**(16), 2575–2576 (aug 2017). https://doi.org/10.1093/bioinformatics/btx204, http://www.ncbi.nlm.nih.gov/pubmed/28383704

[2] Bao, L., Pu, M., Messer, K.: AbsCN-seq: a statistical method to estimate tumor purity, ploidy and absolute copy numbers from next-generation sequencing data. Bioinformatics **30**(8), 1056–1063 (apr 2014). https://doi.org/10.1093/bioinformatics/btt759, https://academic.oup.com/bioinformatics/ARTICLE-lookup/doi/10.1093/bioinformatics/btt759

[3] Bishop, C.M.: Pattern recognition and machine learning. Springer (2006)

[4] Cappe, O., Moulines, E., Ryden, T.: Inference in Hidden Markov Models. Springer Science+Business Media, Inc (2005)

[5] Casella, G., Berger, R.L.: Statistical inference. Thomson Learning (2002)

[6] Chen, B., Cole, J.W., Grond-Ginsbach, C.: Departure from Hardy Weinberg Equilibrium and Genotyping Error. Front Genet. (8) (2017)

[7] Ewing, B., Hillier, L., Wendl, M.C., Green, P.: Base-calling of automated sequencer traces using phred. I. Accuracy assessment. Genome research **8**(3), 175–85 (mar 1998), http://www.ncbi.nlm.nih.gov/pubmed/9521921

[8] Favero, F., Joshi, T., Marquard, A.M., Birkbak, N.J., Krzystanek, M., Li, Q., Szallasi, Z., Eklund, A.C.: Sequenza: allele-specific copy number and mutation profiles from tumor sequencing data. Annals of Oncology **26**(1), 64–70 (jan 2015). https://doi.org/10.1093/annonc/mdu479, http://www.ncbi.nlm.nih.gov/pubmed/25319062

[9] Fumagalli, M., Vieira, F.G., Korneliussen, T.S., Linderoth, T., Huerta-Sánchez, E., Albrechtsen, A., Nielsen, R.: Quantifying population genetic differentiation from next-generation sequencing data. Genetics **195**(3), 979–992 (2013). https://doi.org/10.1534/genetics.113.154740, https://www.genetics.org/content/195/3/979

[10] Fumagalli, M., Vieira, F.G., Linderoth, T., Nielsen, R.: ngsTools: methods for population genetics analyses from next-generation sequencing data. Bioinformatics **30**(10), 1486–1487 (01 2014). https://doi.org/10.1093/bioinformatics/btu041, https://doi.org/10.1093/bioinformatics/btu041

[11] Hardy, G.H.: Mendelian Proportions in a Mixed Population. Science, New Series **28**(706), 49–50 (1908)

[12] Jacqueline, K.W., Anna, P., Nancy, J.C.:

[13] Lachance, J.: Detecting selection-induced departures from hardy-weinberg proportions. Genetics Selection Evolution (1), 15 (2009)

[14] Li, C., Biswas, G.: Temporal Pattern Generation Using Hidden Markov Model Based Unsupervised Classification. In: IDA 1999: Advances in In-

telligent Data Analysis, pp. 245–256. Springer, Berlin, Heidelberg (1999). https://doi.org/10.1007/3-540-48412-4$_2$1

[15] McKenna, A., Hanna, M., Banks, E., Sivachenko, A., Cibulskis, K., Kernytsky, A., Garimella, K., Altshuler, D., Gabriel, S., Daly, M., DePristo, M.A.: The Genome Analysis Toolkit: a MapReduce framework for analyzing next-generation DNA sequencing data. Genome research **20**(9), 1297–303 (sep 2010). https://doi.org/10.1101/gr.107524.110, http://www.ncbi.nlm.nih.gov/pubmed/20644199

[16] Morrow, C.A., Fraser, J.A.: Ploidy variation as an adaptive mechanism in human pathogenic fungi. Seminars in Cell and Developmental Biology **24**(4), 339–346 (apr 2013)

[17] Nielsen, R., Paul, J., Albrechtsen, A., Song, Y.: Genotype and snp calling from next-generation sequencing data. Nature Reviews. Genetics **12**(6), 443–451 (2011). https://doi.org/10.1038/nrg2986

[18] Rabiner, L.: A tutorial on hidden Markov models and selected applications in speech recognition. Proceedings of the IEEE **77**(2), 257–286 (1989). https://doi.org/10.1109/5.18626, http://ieeexplore.ieee.org/document/18626/

[19] Rhodes, J., Beale, M.A., Vanhove, M., Jarvis, J.N., Kannambath, S., Simpson, J.A., Ryan, A., Meintjes, G., Harrison, T.S., Fisher, M.C., Bicanic, T.: A Population Genomics Approach to Assessing the Genetic Basis of Within-Host Microevolution Underlying Recurrent Cryptococcal Meningitis Infection. G3 Genes—Genomes—Genetics (2017). https://doi.org/10.1534/g3.116.037499, https://doi.org/10.1534/g3.116.037499

[20] Sattler, M.C., Carvalho, C.R., Clarindo, W.R.: The polyploidy and its key role in plant breeding. Planta (243), 281–296 (2016)

[21] Stone, N.R., Rhodes, J., Fisher, M.C., Mfinanga, S., Kivuyo, S., Rugemalila, J., Segal, E.S., Needleman, L., Molloy, S.F., Kwon-Chung, J., Harrison, T.S., Hope, W., Berman, J., Bicanic, T.: Dynamic ploidy changes drive fluconazole resistance in human cryptococcal meningitis. Journal of Clinical Investigation **129**(3), 999–1014 (mar 2019)

[22] Therkildsen, N.O., Palumbi, S.R.: Practical low-coverage genomewide sequencing of hundreds of individually barcoded samples for population and evolutionary genomics in nonmodel species. Molecular Ecology Resources **17**(2), 194–208 (2017). https://doi.org/https://doi.org/10.1111/1755-0998.12593, https://onlinelibrary.wiley.com/doi/abs/10.1111/1755-0998.12593

[23] Viterbi, A., A.: Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. IEEE Transactions on Information Theory **13**(2), 260–269 (apr 1967). https://doi.org/10.1109/TIT.1967.1054010, http://ieeexplore.ieee.org/document/1054010/

[24] Weinberg, W.: Über den Nachweis der Vererbung beim Menschen. Jahresh. Ver. Vaterl. Naturkd. Württemb. **64**, 369–382 (1908)

[25] Weiß, C.L., Pais, M., Cano, L.M., Kamoun, S., Burbano, H.A.: nQuire: a statistical framework for ploidy estimation using next generation sequenc-

ing (2018). https://doi.org/10.1186/s12859-018-2128-z, https://doi.org/10.1186/s12859-018-2128-z

[26] Wood, T.E., Takebayashi, N., Barker, M.S., Mayrose, I., Greenspoon, P.B., H, R.L.: The frequency of polyploid speciation in vascular plants. Proc Natl Acad Sci USA (106), 13875–13879 (2009)

[27] Zhu, J., Tsai, H.J., Gordon, M.R., Li, R.: Cellular Stress Associated with Aneuploidy. Developmental cell **44**(4), 420–431 (feb 2018). https://doi.org/10.1016/j.devcel.2018.02.002, http://www.ncbi.nlm.nih.gov/pubmed/29486194