The Loss of Heterozygosity (LOH) Algorithm in Genotyping Console 2.0

Introduction

Loss of heterozygosity (LOH) represents the loss of allelic differences. The SNP markers on the SNP Array 6.0 can be used to detect LOH. Specifically, the genotypes for each SNP marker are used to find regions with large numbers of homozygous genotype calls.

There are two distinct copy number/LOH algorithms in Genotyping Console 2.0:

- Copy number/LOH algorithm using a predefined reference model file. This utilizes a single-sample workflow that does genotyping on the fly using an algorithm similar to the BRLMM-P-plus genotyping algorithm in Affymetrix Power Tools, which uses a no-call confidence threshold of 0.05.

- Copy number/LOH algorithm in which both test and reference samples are specified. This utilizes a batch workflow that calls the Birdseed genotyping algorithm with a no-call confidence threshold of 0.1 to find the genotypes.

This white paper outlines the proof of concept work performed during the development of the LOH algorithms in the context for the first workflow above.

Results and Discussion

The performance of the LOH algorithm was evaluated using genotypes that were computed in the following manner. First, a reference quantile normalization distribution and a set of plier feature effects were determined based upon 270 HapMap sample CEL files using Affymetrix Power Tools (APT).

Then, each sample CEL file was individually processed using the APT BLRMM-P-plus implementation with the default parameters (this includes a no-call confidence threshold of 0.05) and specifying the pre-computed normalization distribution and feature effects.

This procedure is analogous to the GTC 2.0 single-sample workflow. In practice, no-call rates and call accuracy are driven by sample quality rather than differences between the algorithms; hence, we expect the conclusions in this white paper to be independent of the algorithm.

The algorithm frames the LOH problem in terms of a statistical hypothesis test. Given a specific region containing N SNP markers with n_{het} heterozygous and n_{hom} homozygous, genotype calls decide between the following two hypotheses:

1. Null: Region is LOH
2. Alternative: Region is non-LOH

Treat the SNP markers as independent binomial variables that can be in one of two states, a heterozygous (AB) or homozygous (AA or BB) genotype call. No-call SNP markers are ignored. Let the probability of making a heterozygous call at any position along the genome be p_{het} and the heterozygous error rate in a homozygous region be p_{error}. Assume a significance level of α and a power of $1-\beta$. Two important values are chosen based on these quantities. The first is *N*, the number of markers in a region, and the second is n_{crit}, the smallest number of heterozygous calls that can be observed before we must conclude that a region is not LOH. An iterative procedure is used to estimate these quantities. Specifically, first n_{crit} is estimated by choosing the smallest *n* such that

$$P(X < n | N, p_{\text{error}}) = 1 - \alpha$$

And then *N* is estimated via the solution of

$$\left(n_{\text{crit}} - Np_{\text{het}}\right) \Big/ \sqrt{Np_{\text{het}}\left(1 - p_{\text{het}}\right)} = Z^{-1}(\beta)$$

where $Z^{-1}(\beta)$ is the inverse of the standard normal function. These are iterated using a fixed number of iterations or until *N* does not change, whichever occurs first. To simplify things, if the final *N* is not odd we increase it by 1 to ensure that it is odd.

To decide between the two hypotheses, the number of heterozygous call n_{het} is compared with the critical value n_{crit}. In the case that $n_{\text{het}} >= n_{\text{crit}}$, decide for alternative hypothesis that there is no LOH. If there are not a sufficient number of heterozygous calls, the decision is made in favor of LOH. Thus, the algorithm is very simple, consisting merely of counting how many heterozygous calls are in a sequence of *N* consecutive genotype calls and comparing with an appropriate cut-off value. Figure 1 visually demonstrates two different regions on different sections of the genome. One where there are few heterozygous SNP genotype calls is called LOH. The other where there are many heterozygous SNP genotype calls is not called LOH.

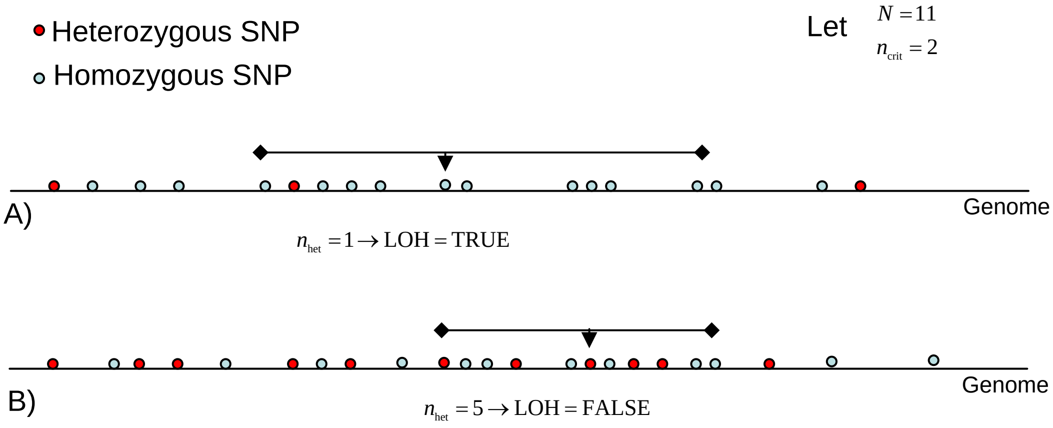

● Heterozygous SNP
○ Homozygous SNP
Let $N = 11$
$n_{\text{crit}} = 2$
A)
Genome
$n_{\text{het}} = 1 \rightarrow \text{LOH} = \text{TRUE}$
B)
Genome
$n_{\text{het}} = 5 \rightarrow \text{LOH} = \text{FALSE}$

Figure 1: Demonstrating how LOH calls are made based on counting the number of heterozygous SNP calls in a given window for two sections of a genome. Hypothetical values are chosen for window size *N* and n_{crit} . A) Region is called LOH because there are fewer heterozygous SNP calls (1) than the cut-off (2). B) Regions are not called LOH because the number of heterozygous SNP calls (5) exceeds the critical value (2).

Figure 2 demonstrates how the procedure works for the entire set of SNP markers. The algorithm is applied to the genome by sliding a moving window of size *N* SNPs along each chromosome. The window is centered at each SNP position along the genome, and in each case the number of heterozygous genotype calls in the window is quantified and a decision is made as to whether we are in the LOH state or the non-LOH state at that SNP position.

The transition between the states represents a special situation. When transitioning from the non-LOH state to the LOH state, a number of previously evaluated SNPs are marked as also being LOH. Similarly, when transitioning from the LOH state to the non-LOH state, SNPs immediately adjacent along the genome can also be expected to be LOH and are marked as thus.

A special set of rules is used for these back and forward filling operations. In particular, starting at the central position, move toward the end of the window, stopping either when the end of the window is reached or at the last homozygous call before the second heterozygous call is reached. This second rule helps to prevent situations where a single het error divides two long stretches of homozygous calls. Additionally, it helps to more accurately estimate the boundaries of the LOH region.

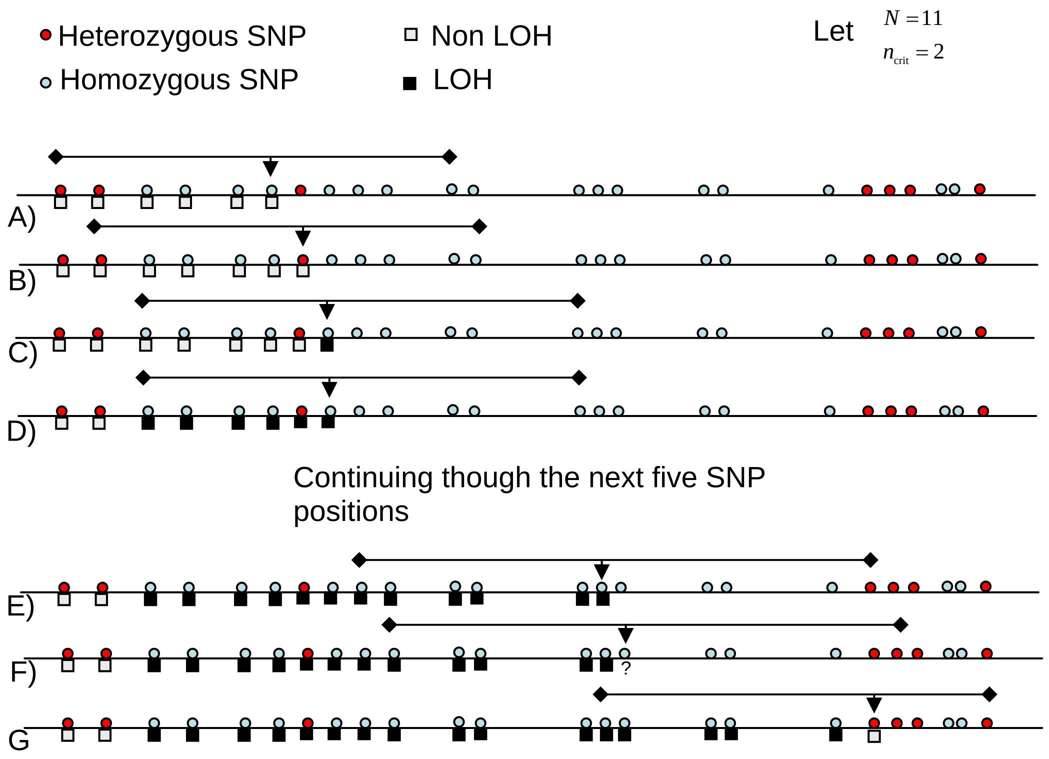


Figure 2: An illustration of how the LOH algorithm proceeds along the genome and how the transitions between LOH and non-LOH regions are handled. A) Window contains three heterozygous SNPs, so call non-LOH at that position. B) Moving to the next SNP position the window now contains two heterozygous SNPs, so call non-LOH at that position. C) At the next SNP position there is only a single heterozygous SNP, so the position is called LOH. D) Because transitioned to the LOH state, you need to back fill the window to also mark it as being LOH. E) Moving farther along the genome, the next five SNP positions are all called LOH. F) Now there are two heterozygous markers in the window, so the call for this position would be non-LOH, but isn't because a transition is occurring. G) Instead, forward fill the window until the last homozygous marker before the second heterozygous marker is reached, then move the window center to the next uncalled position along the genome.

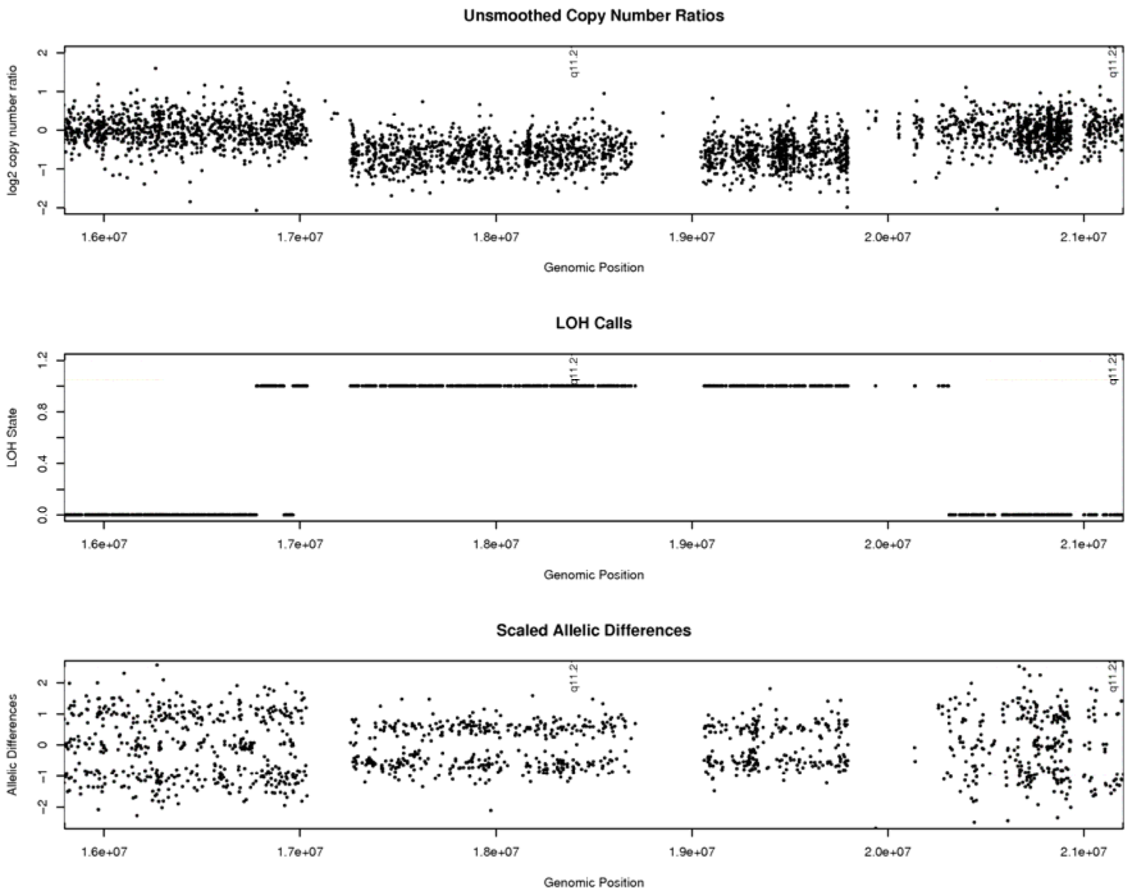
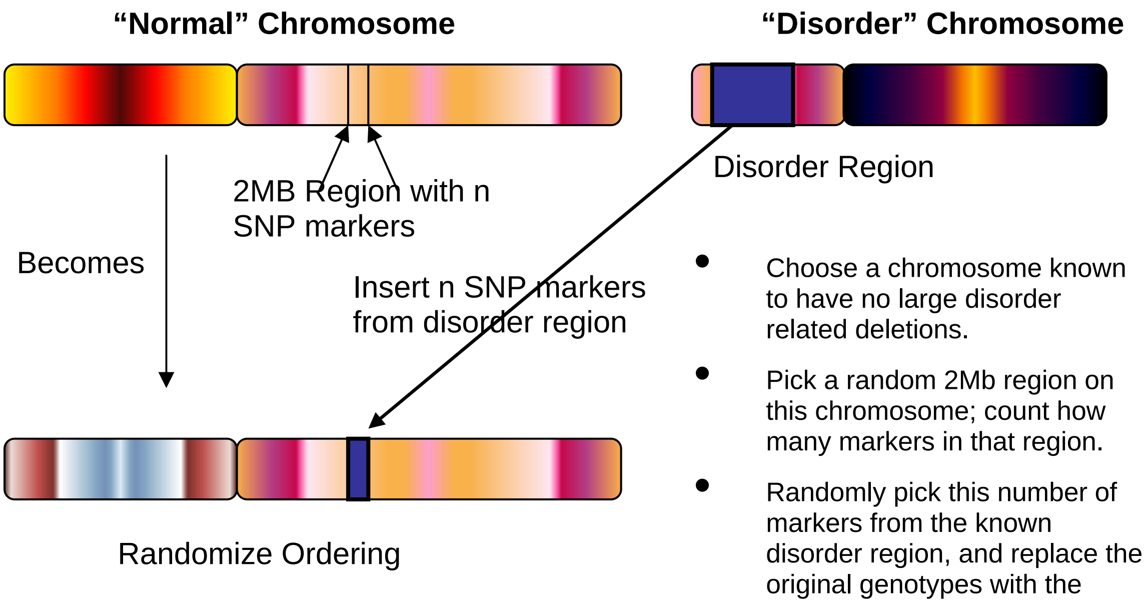


Figure 3: Unsmoothed $\log_2$ copy number ratio (top), LOH (middle) and scaled allelic difference (lower) estimates in the region of a 1 copy deletion.

Figure 3 demonstrates the results of running the algorithm on a particular section of genome where a one-copy deletion is present. The LOH state is either 0, representing no LOH, or 1, representing LOH. Examining the top and middle panels, we see that the LOH region corresponds directly with the region with lower unsmoothed copy. The lower panel contains scaled allelic differences, helping to confirm that LOH is present. The left-most parts of LOH do not seem to correspond directly with the deletion. Instead, close examination of the scaled allelic differences suggests that this is perhaps a copy-neutral LOH region.



- Choose a chromosome known to have no large disorder related deletions.
- Pick a random 2Mb region on this chromosome; count how many markers in that region.
- Randomly pick this number of markers from the known disorder region, and replace the original genotypes with the disorder-related genotypes.
- Randomize the ordering of the remaining genotypes on that chromosome.

Figure 4: A simulation framework for assessing LOH calling algorithms. A 2 Mb region is selected from a "normal" chromosome not known to have any large deletion-related LOH. Genotypes from a region having a known 1 copy deletion are substituted into this 2 Mb region. To ensure that no real copy neutral LOH is detected, randomize the ordering of the remaining markers on the chromosome.

One method of assessing the performance of the LOH algorithm is via simulation. With a viable method of generating known sections of LOH and known sections of non-LOH, the sensitivity and specificity of the algorithm can be examined.

Figure 4 shows the simulation framework used here. Specifically, samples with known, validated copy number 1 deletions greater than 2 Mb in size were chosen. For each sample, the following procedure was used: Randomly select a 2 Mb region on a chromosome not known to have any large copy number deletions. Call this chromosome the "normal" chromosome and count how many SNP markers are in the selected 2 Mb region; call this *n*.

From the chromosome having the known deletion, call this the "disorder" chromosome, select *n* consecutive SNPs and their genotype calls. If there are less than *n* markers in the disorder region and a consecutive set can not be found, select the *n* markers from the disorder region randomly with replacement. Replace the genotype calls of the markers in the selected region of the "normal" chromosome with those from the "disorder" chromosome. To ensure that any other pre-existing LOH on the "normal" chromosome is removed, the genotypes for the SNP markers outside the selected 2 Mb region are randomly ordered.

Note that this has the downside that it removes the normal linkage structure. The LOH algorithm is then applied to this data to examine its performance.

Assessment is conducted by repeating the simulation multiple times for each sample. In each case, a marker in the 2 Mb region that is correctly called LOH is called a true positive. A marker outside the LOH region which is called LOH is a false positive. The ideal is to have a high true positive rate and a low false positive rate. Each of the algorithm parameters discussed above will have an impact on the results.

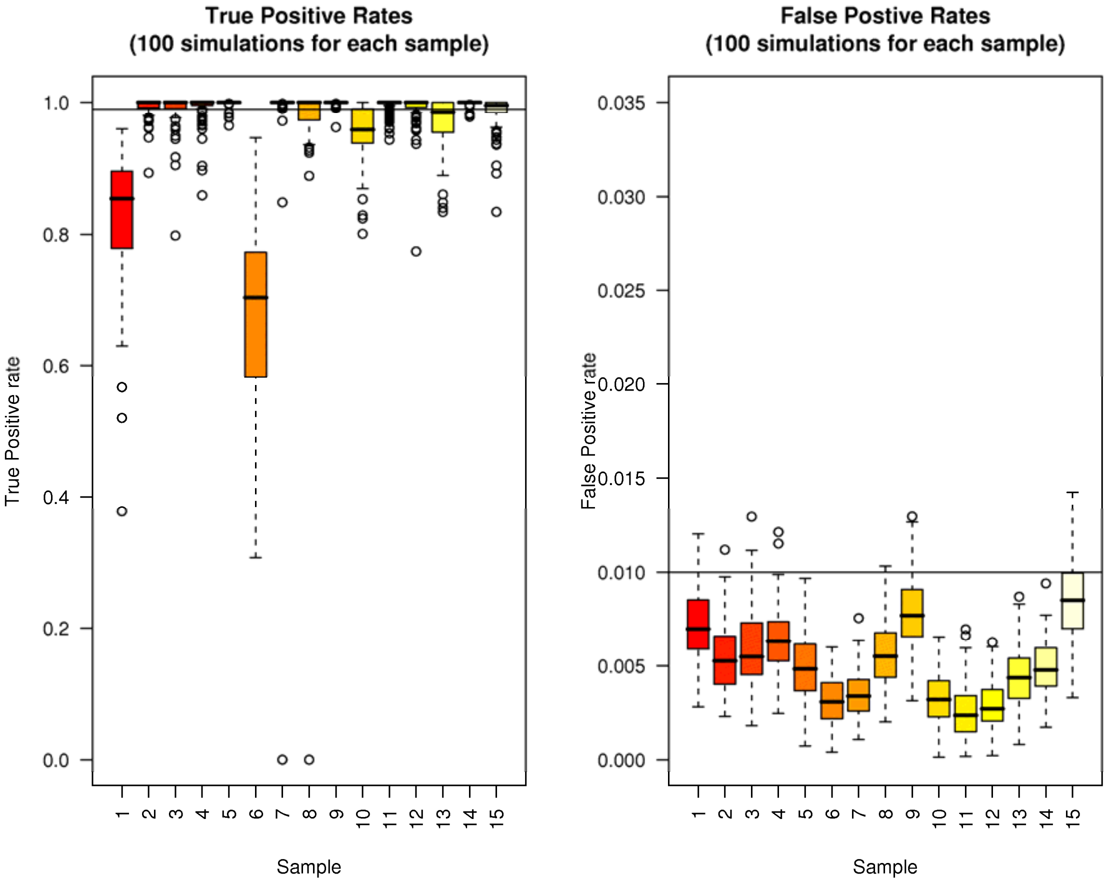


Figure 5: True and false positive rates for LOH detection from 100 simulations for each of 15 different samples. Using p_{error} =0.05, α =0.001 and β =0.005. Horizontal lines indicate 99 percent true positives and 1 percent false positives.

Figure 5 shows the results of running the simulation 100 times for each of 15 different samples. Each sample has a different large known deletion. For this simulation, the same set of parameters was used for each sample p_{error} =0.05, α =0.001 and β =0.005. Note that p_{het} is calculated separately for each sample based on the entirety of its respective genotype calls. Two samples, 1 and 6, performed particularly poorly across all 100 simulations. Note that another two samples, 7 and 8, had simulation results with true positive rates of 0. Closer examination of these two cases showed the selected 2 Mb region having few SNP markers, 49 and 26 markers, respectively. Because the LOH algorithm described here is implicitly a counting algorithm, when there is a region with very few markers it is going to be difficult to correctly detect. Across all 1,500 simulations the median number of markers in the selected 2 Mb region was 632. The smallest region that had a non-zero true positive rate had 47 SNP markers.

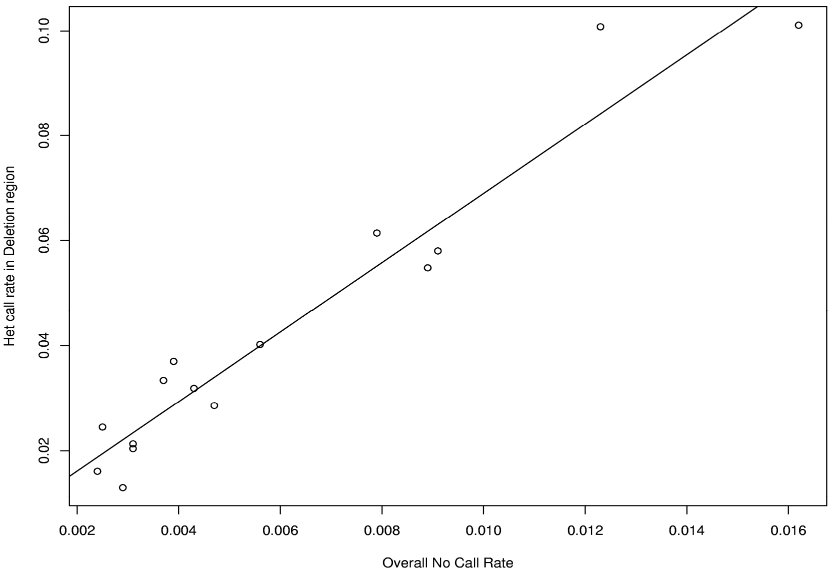


Figure 6: Comparing the overall no-call rate with the heterozygous call rate in the deletion regions. A strong relationship exists.

Making the correct genotype calls in a deletion region could be expected to be more difficult than along a portion of the genome having normal copy number. In particular, when in a 1 copy region, the desirable result would be to have only homozygous calls. But because only a single copy exists in such a region, rather than having “AA” and “BB” for a homozygous SNP, only “A” and “B” are really present.

The lower signal for each allele, in this situation, makes it more difficult to discriminate between the correct homozygous state and the heterozygous state. This difficulty in making the appropriate genotype call also affects the ability to make the correct decision on LOH state for each marker. The p_{error} parameter is used to account for this difficulty. If this parameter is too low, then the true positive results will be lower. If this parameter is too high, then the false positive results will be higher.

Additionally, this parameter might differ between samples. In general, deletion regions are not known *a priori*, so this parameter can not be directly estimated. Instead, another method is used. Figure 6 shows the relationship between the overall no-call rate for each of the 15 samples and the heterozygous call rate in the deletion region. There is a strong relationship between these two values. This suggests that p_{error} can be estimated using the overall no-call rate. In particular, a linear regression fit to the above could be used. The two points with heterozygous error rates of approximately 0.1 correspond to samples 1 and 6.

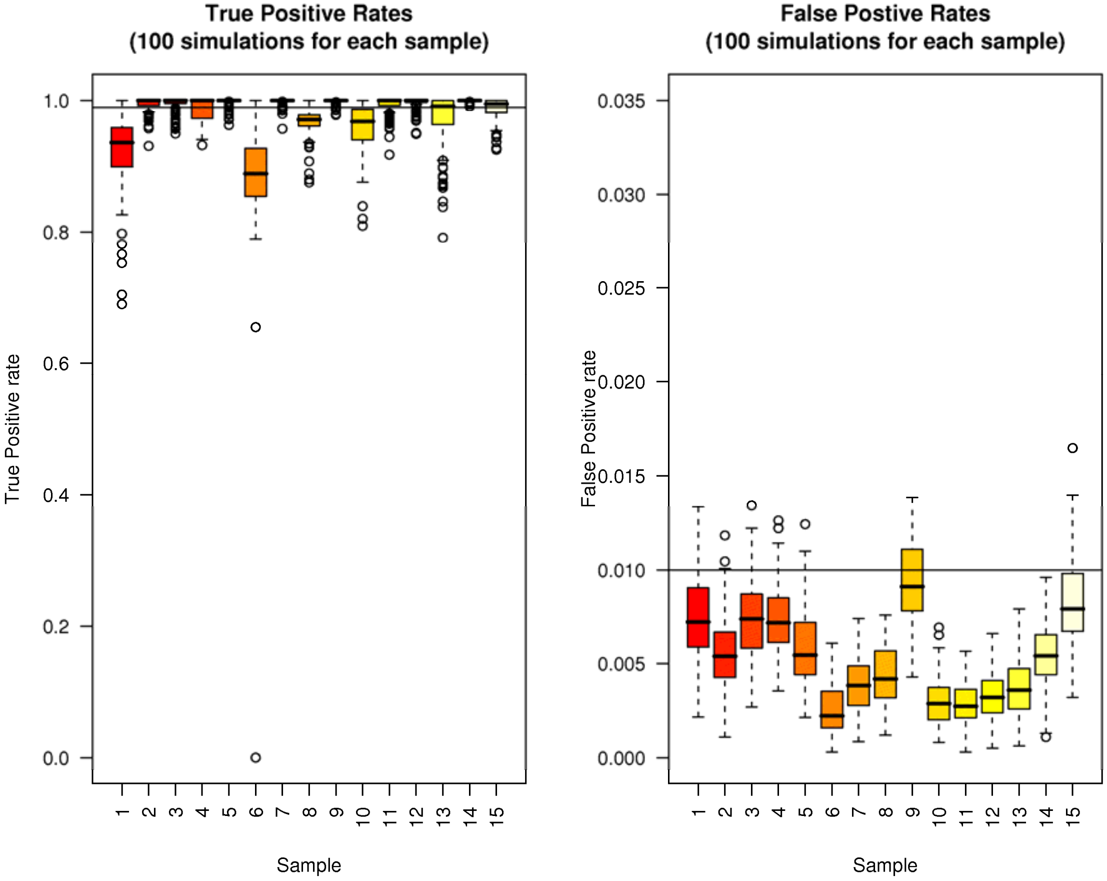


Figure 7: True and False positive rates for LOH detection from 100 simulations for each of 15 different samples. Using α =0.001 and β =0.001 and dynamically selected p_{error} . Horizontal lines indicate 99 percent true positives and 1 percent false positives.

One procedure for using the overall no-call rate to dynamically select p_{error} on a sample by sample basis is as follows: First, use the linear regression proposed above to make an initial estimate p_{error} . If this estimate is less than a given minimum value (say, 0.04) then increase to the minimum threshold. To accommodate error in estimating to p_{error} this way, its potential values are restricted to fall in equally sized steps. This is accomplished by rounding p_{error} up in increments of 0.01 when necessary. Figure 7 shows the results of the simulation when using this procedure for dynamically selecting p_{error} . The results for sample 1 and sample 6 have improved considerably when compared to the previous results, indicating that allowing p_{error} to be selected provided improved results.