Instructions for curating issues discovered from long read alignments to separate haplotypes, April 17, 2023

This document is a guide to curating anomalies in HG002 diploid assemblies (currently the v0.7 verkko/rukki assembly) which suggest falsely heterozygous sites in the assembly based on DeepVariant homozygous non-reference calls on alignments of long reads to just one assembly haplotype (maternal or paternal). The document describes how these anomalies were detected, and the methods that we use to determine whether they are genuine errors in the assembly or false calls by DeepVariant.

This document is a work in progress, and available to edit at https://docs.google.com/document/d/1Iv1pysvu_Fy0VrnHVBxKrdwMzEkgzCTCew4CjaE-x_s/edit#. Please feel free to suggest changes or additions as they occur to you.

HOW ARE THESE ANOMALIES DETECTED?

Like many assembly issues (as of April, 2023), these potentially false heterozygous sites are discovered as high quality DeepVariant (DV) calls made on HG002 reads aligned to the HG002 assembly. In this case, the DV calls were made on alignments of long reads (HiFi reads called with DeepConsensus v1.1, with DV v1.5 calls here, ONT R10 duplex reads, with DV v1.5 calls here, and ONT R10 simplex reads, with DV v1.5 calls here). This issue set includes all homozygous non-reference (HNR) “PASS” DV v1.5 calls with genotype quality (GQ) of 10 or greater in the intersection of the three long read call sets. These calls are usually heterozygous locations in the assembly (assuming they are on autosomes) where all of the reads agree, indicating that the haplotype that disagrees with the reads is wrong. If the assembly consensus on this haplotype were correct, we expect to see at least some correctly-aligning reads agree with it. We call errors in the assembly called by DeepVariant “true positives” and erroneous DeepVariant calls on a correct assembly “false positives”. To assure a high true positive rate for this type of potential issue, as of April 2023, our plan is to curate 15 of these variants to determine the reliability of five calls in each of three categories: DeepVariant GQ between 10 and 20, GQ between 20 and 30, and GQ between 30 and 40. The curation results will aid us in deciding which of DeepVariant’s suggested corrections to apply to the current and/or future assemblies that we attempt to polish.

ACCESSING YOUR ASSIGNED ISSUES

If you have volunteered to curate assembly issues for this April 2023 round, and we have your github username, a number of these issues will have been assigned to you for curation. You can view your issues at https://github.com/marbl/HG002-issues/issues (assuming you are logged in to github) by using the “Assignee” drop down menu to select yourself. To restrict to viewing just

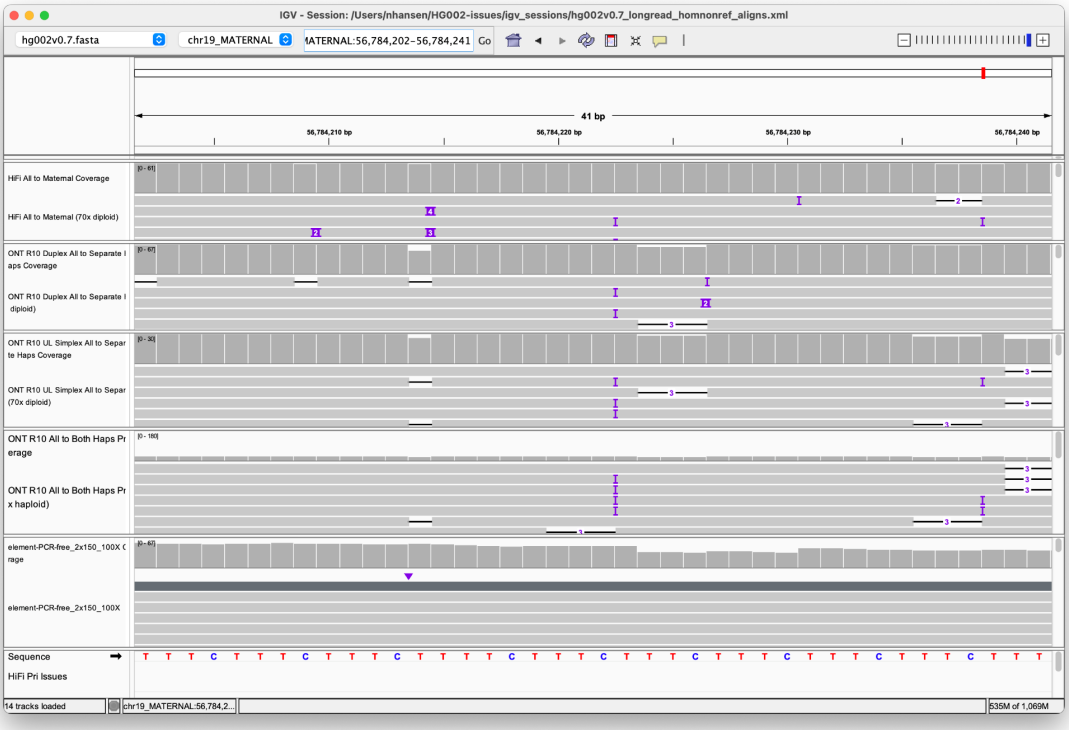
the issues described in this document, filter the issues to include only those with the label "**longread_homnonref**".

SESSION FILES AND VIEWING LONG READ INTERSECTED HNR CALLS IN IGV

The best way to determine the underlying evidence for a DeepVariant call's reliability is to view as much relevant data as possible for that region. One easy way to view groups of relevant data tracks in IGV is to load IGV session files. For curating this type of anomaly, we have created an IGV session file and put it on the HG002-issues github site at hg002v0.7_longread_homnonref_aligns.xml. Once the file is on your computer (if you clone the HG002-issues repository, you can update it to new versions easily), its tracks can be loaded into IGV by selecting "File…Open Session" and selecting the file's location on your computer.

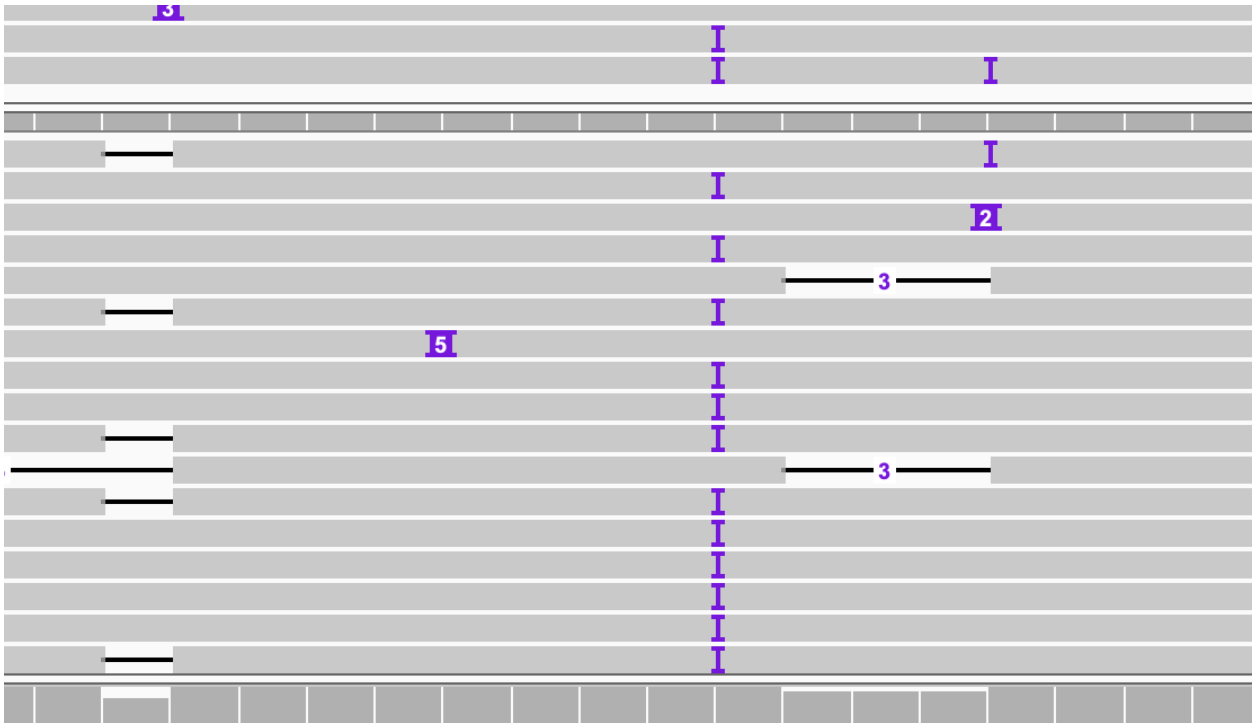
A CURATION EXAMPLE

Once you have loaded the IGV session file, copy and paste the location of one of your assigned issues (here we will walk through the example of chr19_MATERNAL:56784222-56784222) into the position window in IGV. IGV will automatically widen any region less than 41 base pairs to 41 base pairs to show the surrounding region. For our example region, our IGV window looks like this:

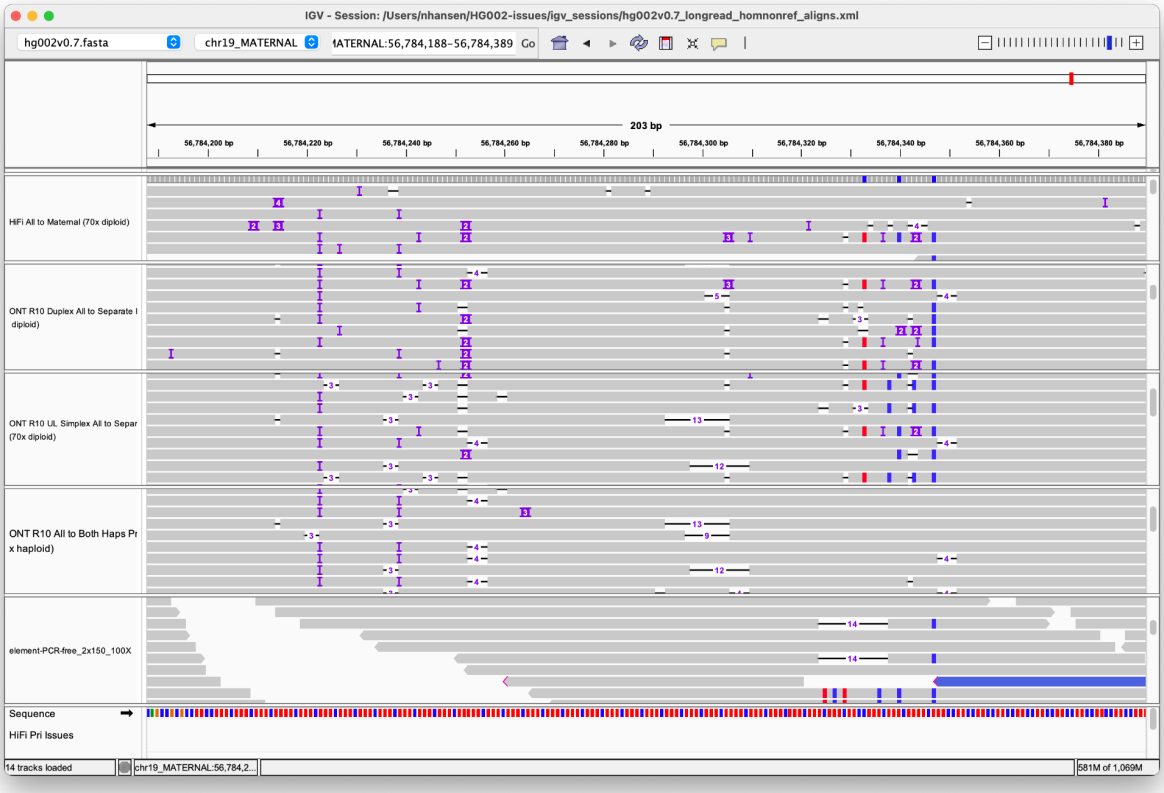


The tracks in this session file show all indels because they've all been configured to not "hide small indels". If you load tracks yourself while curating rather than using the provided IGV session file, **be sure that you also have not set each track to "hide small indels"**. Because all indels are visible, we can see the DeepVariant call, a one-base insertion of a T in one of the many repeated CTTT's, as a vertical line of insertion symbols just to the right of the middle of the IGV window.

But not all of the reads aligned at this position show this insertion. Scrolling through each of the tracks with a wider view shows that, while some of the reads don't align with an insertion at position 56784222, the many copies of CTTT or CTTTT in the surrounding sequence allow different positions of the insertion to be plausible.



Would inserting a T into the consensus at the position of the insertion allow all the reads to align in a way that matches? Maybe zooming out to view the entire CT region will help us to understand things better:



This region has a complex, repetitive sequence of C's and T's, and while the all-to-maternal-haplotype read alignments seem to show what look like two haplotypes, there are no reads that align across the region with just the proposed insertion we are curating. So it is very difficult to determine what the true maternal haplotype is here.

In general with these issues, simple repeats can result in reads not showing an insertion or deletion called as homozygous non-reference because those reads aren't long enough to span the entire length of the repeat. Remember that **alignments which fail to extend entirely across a simple repeat are not informative about the length of the run.**

VIEWING THE ALTERNATE HAPLOTYPE

When a difficult region like this appears to have two different haplotypes (i.e., the region isn't homozygous, even though the DeepVariant call is homozygous at the particular site you are curating), you can also look at the opposite haplotype to see what the assembler did there and how the read alignments look there. To find the region's coordinates on the other haplotype, open a second instance of IGV and load the session file called "hg002v0.7_issues_haps_and_markers.xml". This session has a track called "Coordinates on ALT haplotype" that will give you the 1000-base window on the paternal haplotype that corresponds to this maternal one (for this example, it is

chr19_PATERNAL:56883151-56884156). If you open a third instance of IGV and load the read alignments in the hg002v0.7_longread_homnonref_aligns.xml again, and go to chr19_PATERNAL:56883151-56884156, you'll see this:

LABELING AN ISSUE WITH YOUR DECISION

In cases like the example here, it's not clear what's going on, so a "help_wanted" label can be applied to the issue in github to mark the issue as a difficult one. In general, if you've convinced yourself that an issue is either a true error in the assembly (i.e., the reads show a true difference from the consensus) or that the DeepVariant call was wrong and the assembly is correct, you should add a **priority** or a **false_positive** label to the issue on github, and add a few comments with screenshots of what you saw that made you make the decision you did. You will, in general, have a "co-curator" who may have already made comments or applied labels, but even if they did, it's helpful for you to add your independent assessment as well, so thank you for your efforts!

See Evaluating with IGV for further tips & tricks regarding IGV.