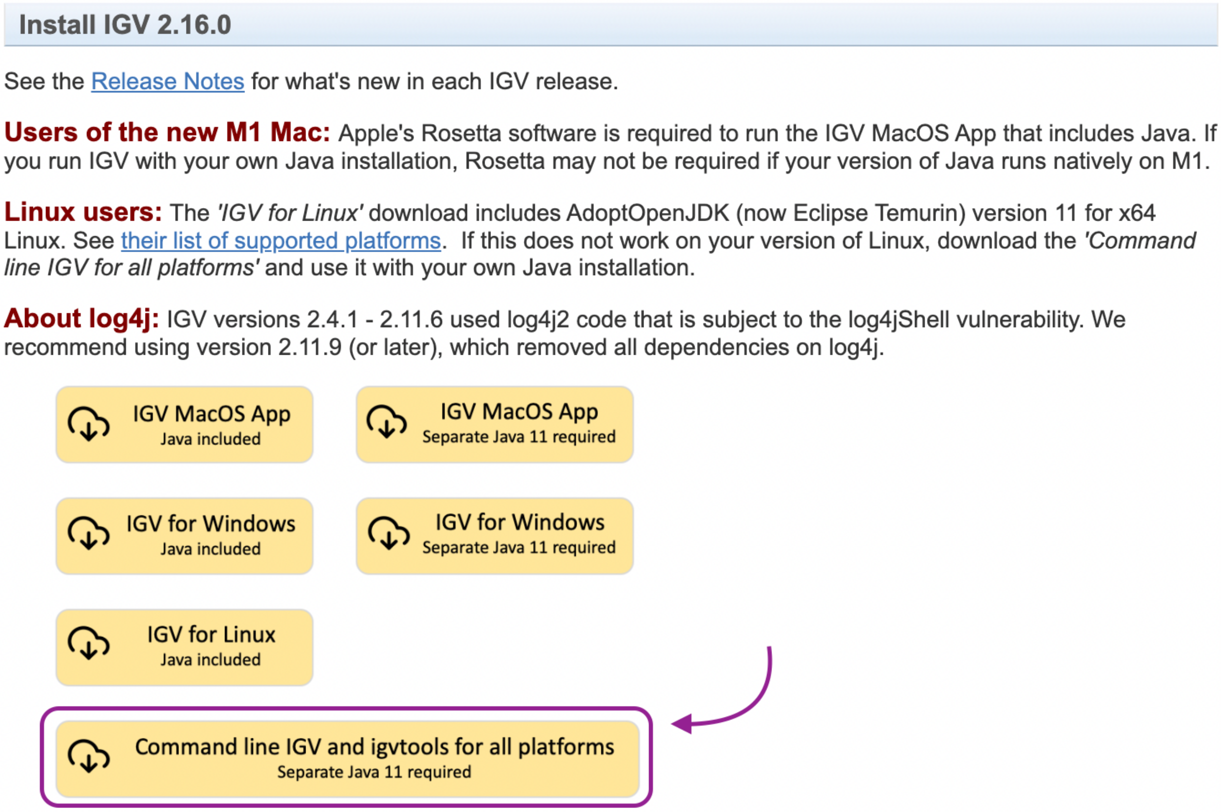
Evaluating with IGV

Downloading and Settings

1. Download the “Command line IGV and igvtools for all platforms”



Install IGV 2.16.0

See the Release Notes for what's new in each IGV release.

Users of the new M1 Mac: Apple's Rosetta software is required to run the IGV MacOS App that includes Java. If you run IGV with your own Java installation, Rosetta may not be required if your version of Java runs natively on M1.

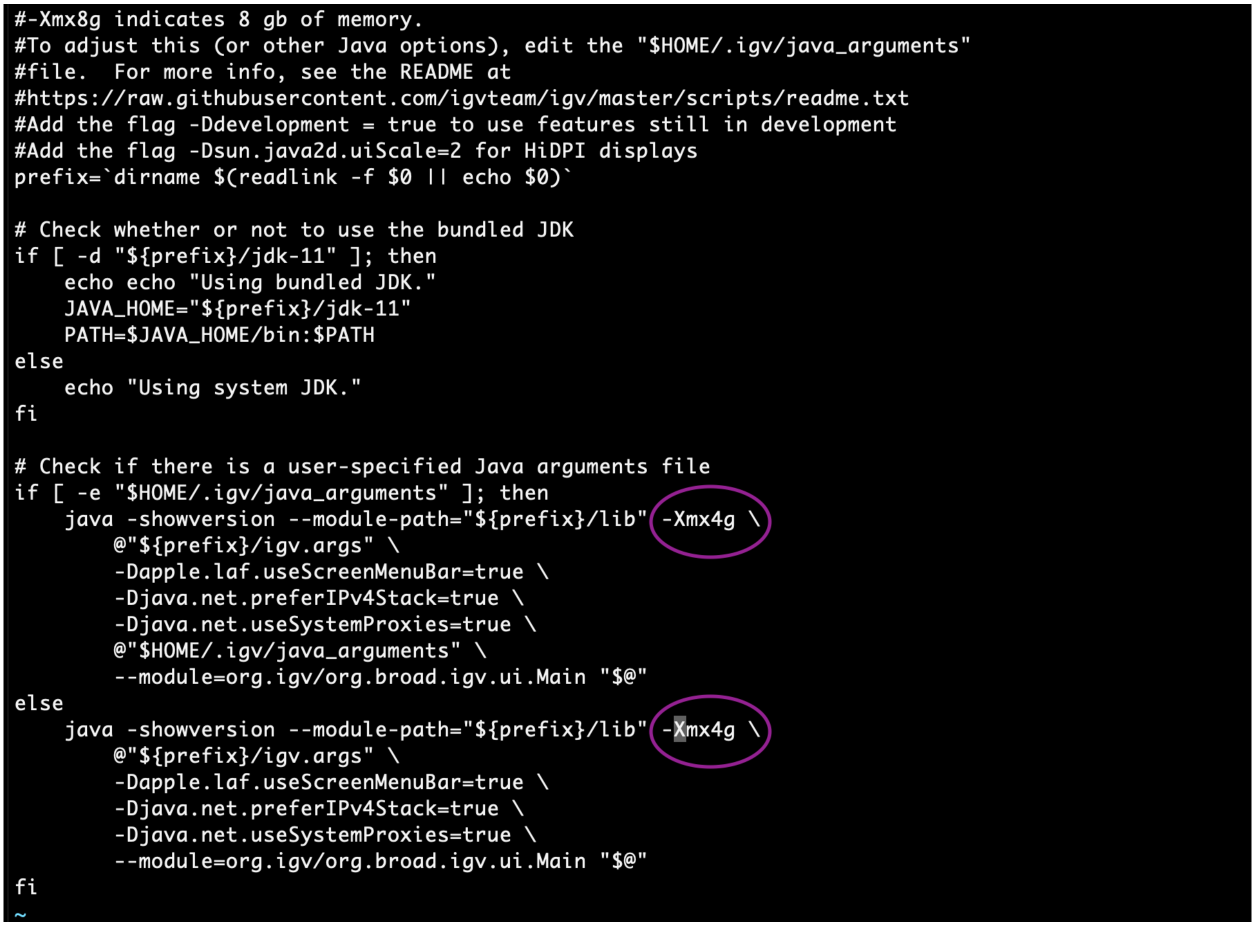
Linux users: The *'IGV for Linux'* download includes AdoptOpenJDK (now Eclipse Temurin) version 11 for x64 Linux. See their list of supported platforms. If this does not work on your version of Linux, download the *'Command line IGV for all platforms'* and use it with your own Java installation.

About log4j: IGV versions 2.4.1 - 2.11.6 used log4j2 code that is subject to the log4jShell vulnerability. We recommend using version 2.11.9 (or later), which removed all dependencies on log4j.

IGV MacOS App
Java included
IGV MacOS App
Separate Java 11 required
IGV for Windows
Java included
IGV for Windows
Separate Java 11 required
IGV for Linux
Java included
Command line IGV and igvtools for all platforms
Separate Java 11 required

2. Unzip and edit igv.sh

   “-Xmx8g” part: -Xmx8g indicates 8g of memory (this is the default).
   There are 2 places in IGV 2.16.0 where you can edit.
   As we are usually opening 2 IGVs, depending on your laptop memory, two 8g apps might overkill and cause some strange issues. Adjust it to match your memory. Usually, we wouldn’t use more than 2g.



```
#-Xmx8g indicates 8 gb of memory.
#To adjust this (or other Java options), edit the "$HOME/.igv/java_arguments"
#file.  For more info, see the README at
#https://raw.githubusercontent.com/igvteam/igv/master/scripts/readme.txt
#Add the flag -Ddevelopment = true to use features still in development
#Add the flag -Dsun.java2d.uiScale=2 for HiDPI displays
prefix=`dirname $(readlink -f $0 || echo $0)`

# Check whether or not to use the bundled JDK
if [ -d "${prefix}/jdk-11" ]; then
    echo echo "Using bundled JDK."
    JAVA_HOME="${prefix}/jdk-11"
    PATH=$JAVA_HOME/bin:$PATH
else
    echo "Using system JDK."
fi

# Check if there is a user-specified Java arguments file
if [ -e "$HOME/.igv/java_arguments" ]; then
    java -showversion --module-path="${prefix}/lib" -Xmx4g \
        @"${prefix}/igv.args" \
        -Dapple.laf.useScreenMenuBar=true \
        -Djava.net.preferIPv4Stack=true \
        -Djava.net.useSystemProxies=true \
        @"$HOME/.igv/java_arguments" \
        --module=org.igv/org.broad.igv.ui.Main "$@"
else
    java -showversion --module-path="${prefix}/lib" -Xmx4g \
        @"${prefix}/igv.args" \
        -Dapple.laf.useScreenMenuBar=true \
        -Djava.net.preferIPv4Stack=true \
        -Djava.net.useSystemProxies=true \
        --module=org.igv/org.broad.igv.ui.Main "$@"
fi
~
```

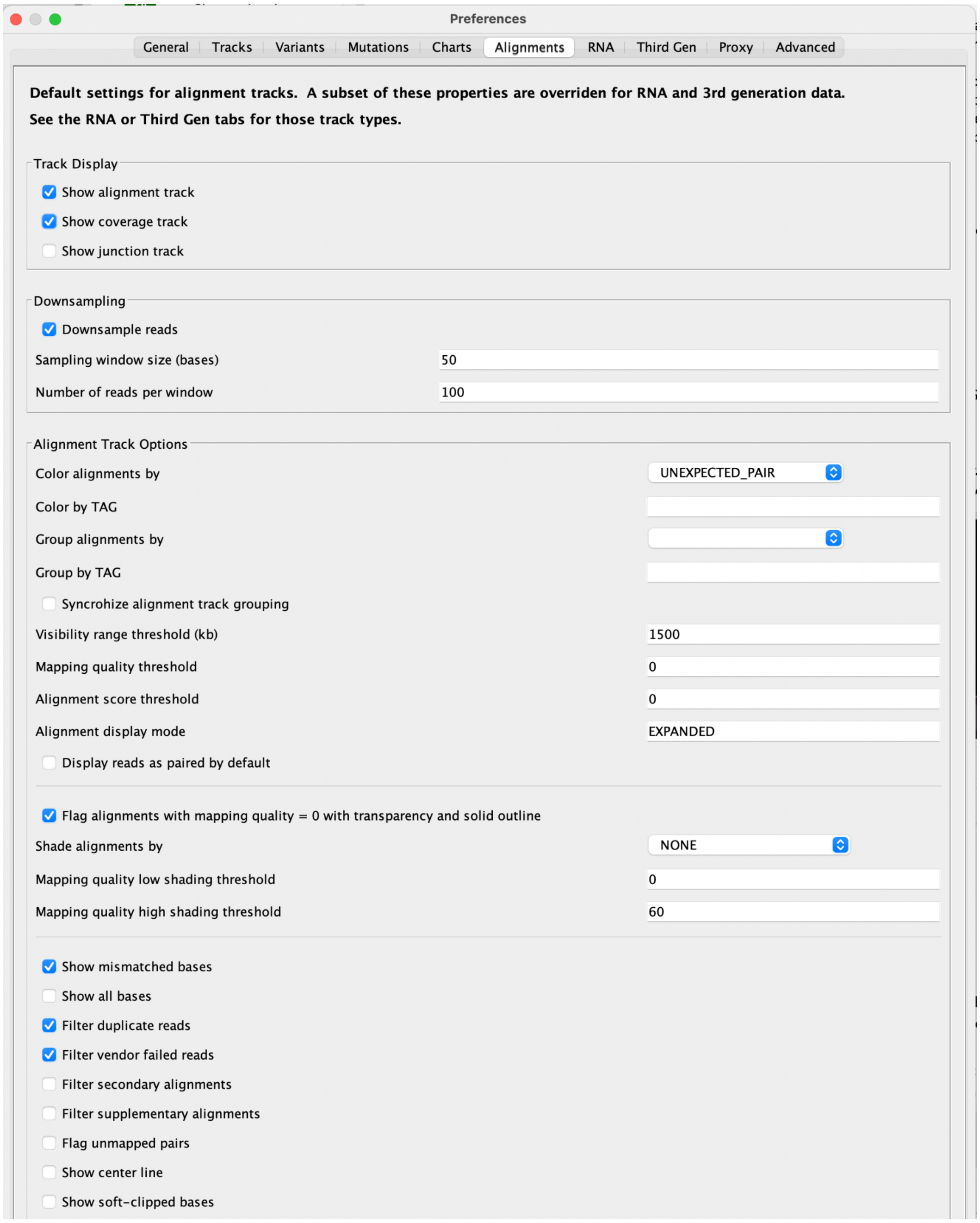
3. Launch IGV from terminal: ./igv.sh

    This way you can see the logs in case IGV is just loading or an error occurred.

4. IGV Settings under View > Preferences

    We need two places to update: [Alignments] and [Third Gen].

[Alignments]



Default settings for alignment tracks. A subset of these properties are overriden for RNA and 3rd generation data. See the RNA or Third Gen tabs for those track types.

Track Display

☑ Show alignment track

☑ Show coverage track

☐ Show junction track

Downsampling

☑ Downsample reads

Sampling window size (bases)	50
Number of reads per window	100

Alignment Track Options

Color alignments by	UNEXPECTED_PAIR
Color by TAG	
Group alignments by	
Group by TAG	

☐ Syncrohize alignment track grouping

Visibility range threshold (kb)	1500
Mapping quality threshold	0
Alignment score threshold	0
Alignment display mode	EXPANDED

☐ Display reads as paired by default

☑ Flag alignments with mapping quality = 0 with transparency and solid outline

Shade alignments by	NONE
Mapping quality low shading threshold	0
Mapping quality high shading threshold	60

☑ Show mismatched bases

☐ Show all bases

☑ Filter duplicate reads

☑ Filter vendor failed reads

☐ Filter secondary alignments

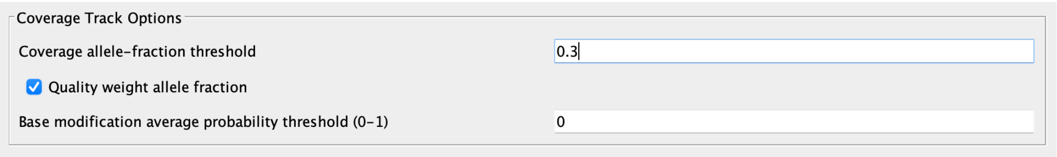
☐ Filter supplementary alignments

☐ Flag unmapped pairs

☐ Show center line

☐ Show soft-clipped bases

...



Coverage Track Options

Coverage allele-fraction threshold	0.3

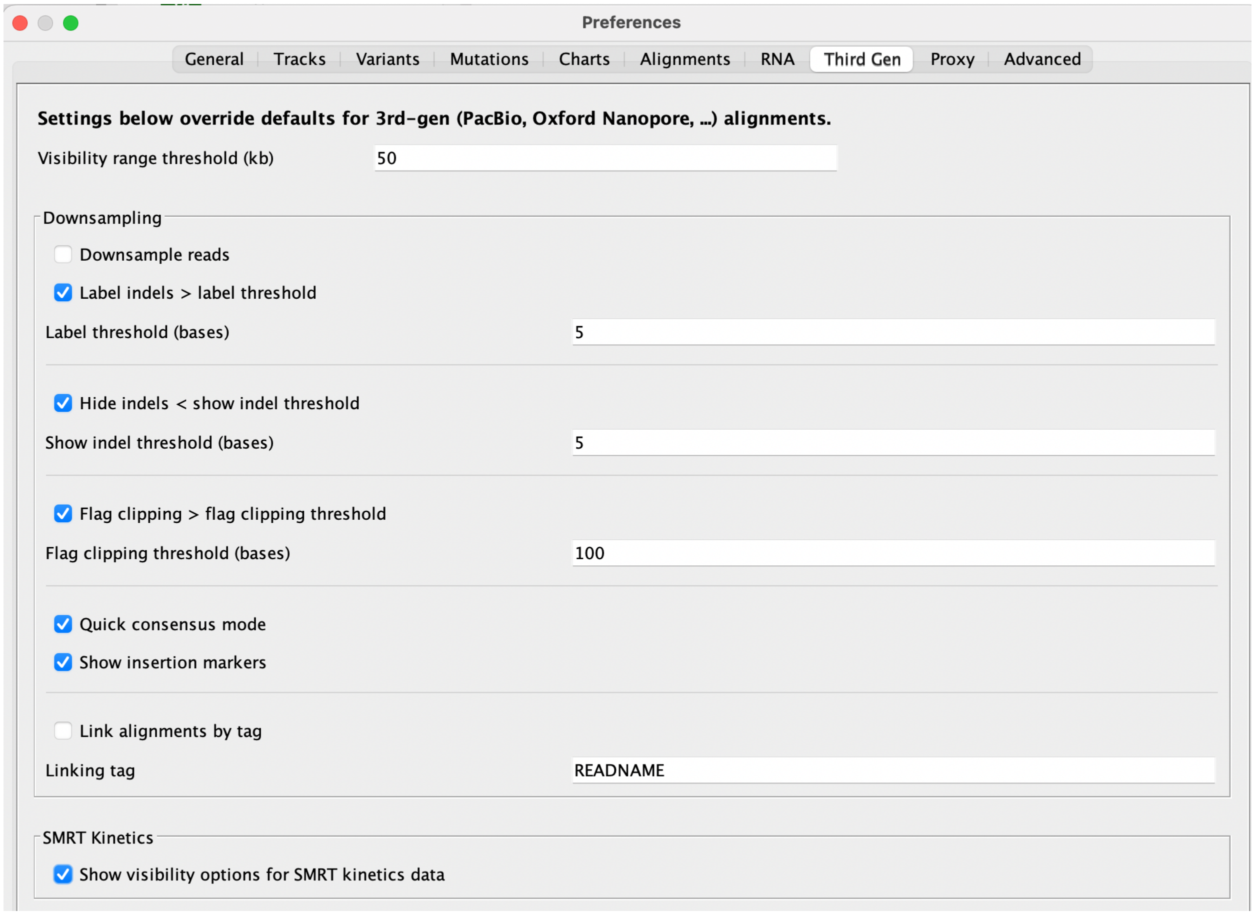
☑ Quality weight allele fraction

Base modification average probability threshold (0-1)	0

- This option is largely applied to non-bam views
- Uncheck the “Filter supplementary alignments”
- Adjust “Visibility range threshold (kb): 0 will display all ranges

- Set “Coverage allele-fraction threshold” to 0.3

[Third Gen]



Settings below override defaults for 3rd-gen (PacBio, Oxford Nanopore, …) alignments.

Visibility range threshold (kb): 50

Downsampling

☐ Downsample reads

☑ Label indels > label threshold

Label threshold (bases): 5

☑ Hide indels < show indel threshold

Show indel threshold (bases): 5

☑ Flag clipping > flag clipping threshold

Flag clipping threshold (bases): 100

☑ Quick consensus mode

☑ Show insertion markers

☐ Link alignments by tag

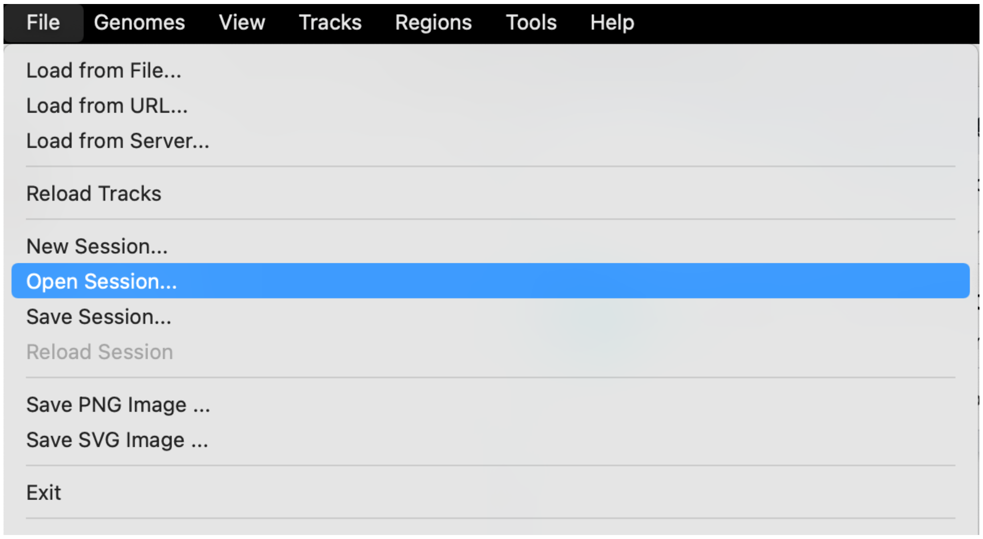
Linking tag: READNAME

SMRT Kinetics

☑ Show visibility options for SMRT kinetics data

- This option is applied to long-read bams
- Set “Visibility range threshold (kb) to something smaller than 100 kb (I use 60 kb)
- Depending on the variant, adjust indel or clipping base threshold to show or hide

Using provided session files



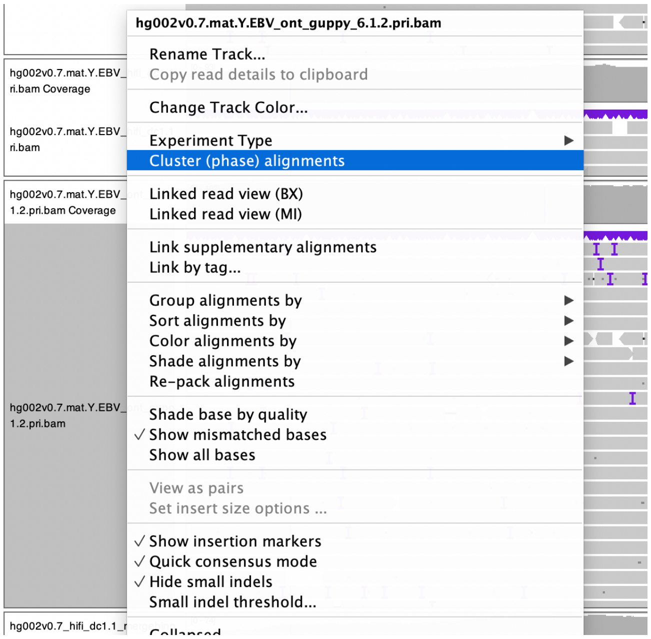
Some predefined session files (.xml) are available on https://github.com/marbl/HG002-issues/tree/main/igv_sessions.

- **hg002v0.7_phase_switch_view.xml**: Overview. Contains coverage wiggles + markers, and bed formatted annotations. The only bam here is ALT haplotype. Suitable for zooming out over 100 kb or more.
- …

Tracks can be also added manually by selecting “File…Load From URL” and entering any of the URLs from Description of AWS-hosted IGV Tracks for Polishers.

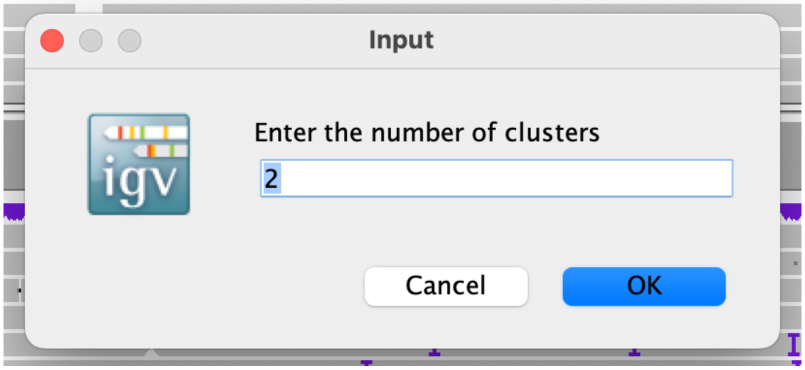
Using the “Phase” feature

It is possible to phase bam alignments on-the-fly, as long as the region contains at least 1 heterozygous variant visible on the coverage track.



- Right-click on the bam read pileup name panel
Select “Cluster (phase) alignments”

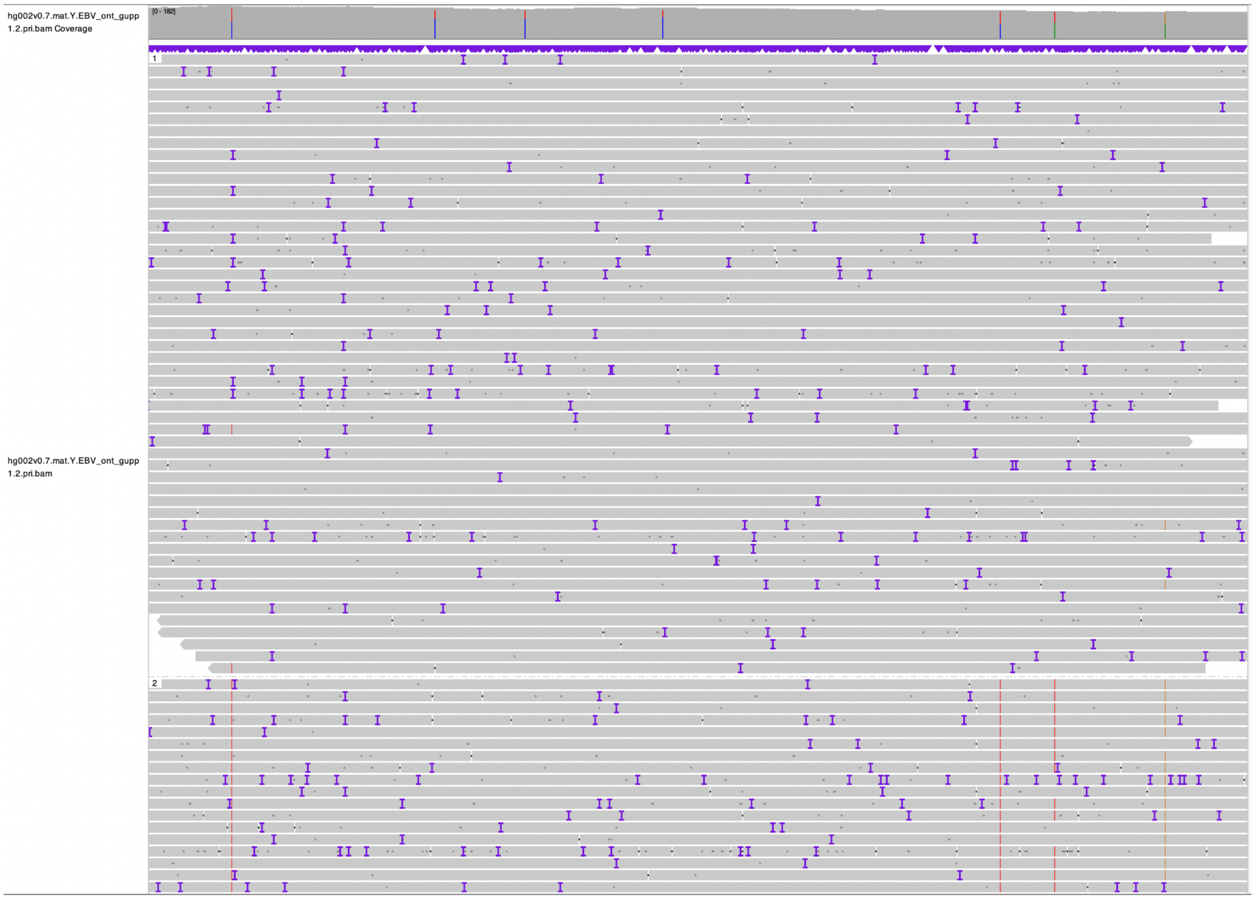
- Centering the het variant in the Coverage track helps if IGV complains



Enter the number of clusters: 2

Setting it as 2 will cluster the reads to 1, 2, and “None”

This is how an example region looks like after “Clustering”


hg002v0.7.mat.Y.EBV_ont_gupp 1.2.pri.bam Coverage
[0 - 182]
1
hg002v0.7.mat.Y.EBV_ont_gupp 1.2.pri.bam
2

Setting the cluster to 0 will bin the reads back to group “None”.

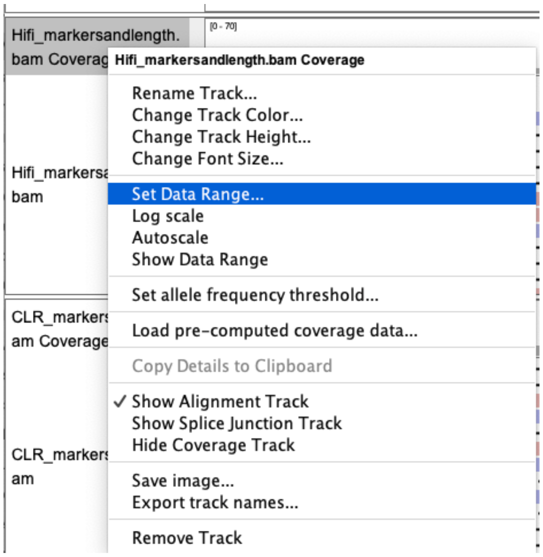
Taking Screenshots

Please take a screenshot of each issue region from the “Overview” and the “Read alignments”. Navigate to the region from the overview session, then zoom out until at least one marker k-mer

(Marker or MAT PAT) is within view, up until ~50 kbp. Copy the locus coordinates, and paste it on the “Read alignments” session.
Then take screenshots of both, and upload them to the assigned issue.

Other tips

Format bam Coverage tracks



1) Set Data Range
    a) HiFi: Max to 100
    b) ONT: Max to 200
2) Set allele frequency threshold… to 0.3