Arima-SV Pipeline

Bioinformatics User Guide for Structural Variant Detection

Material Part Number: A101060
Document Part Number: A160602 v01
Release Date: February 2022

This product is intended for research use only. This product is not intended for diagnostic purposes.

This document and its contents are proprietary to Arima Genomics, Inc ("Arima Genomics"). Use of this document is intended solely for Arima Genomics customers for use with the Arima-HiC$^+$ Kit, P/N A101060, and for no other purpose. This document and its contents shall not be used, distributed or reproduced in whole or in part and/or otherwise communicated or disclosed without the prior written consent of Arima Genomics.

This user manual must be read in advance of using the product and strictly followed by qualified and properly trained personnel to ensure proper use of the Arima-HiC$^+$ kit. Failure to do so may result in damage to the product, injury to persons, and/or damage to other property. Arima Genomics does not assume any liability resulting from improper use of its products or others referenced herein.

U.S. Patent No. US 9,434,985 and 9,708,648 pertains to the use of this product.

TRADEMARKS
Illumina®, MiSeq®, MiniSeq®, NextSeq®, HiSeq®, and NovaSeq™ are trademarks of Illumina, Inc.
Sylabs™ is a trademark of Sylabs™.

Revision History

Document	Date	Description of Change
Material Part Number: A101060 Document Part Number: A160602 v01	February 2022	Initial Release

Table of Contents

1. Introduction

1.1 Arima-SV Pipeline Overview

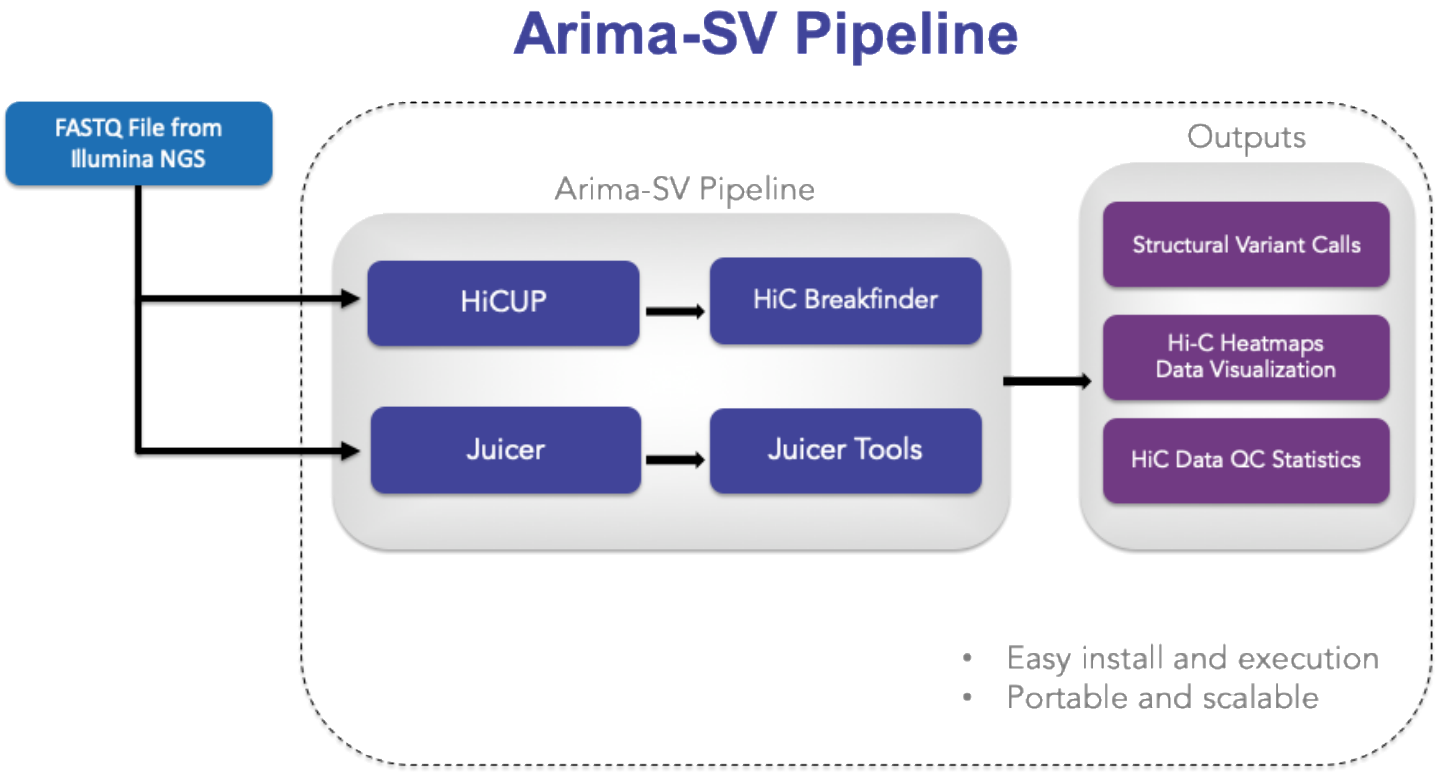


Figure 1. Schematic of the Arima-SV Pipeline.

The Arima-SV Pipeline (**Figure 1**) is contained in a Singularity (https://github.com/sylabs/singularity) image which includes all programs, dependencies, references, and accessory files. Users can download the image from the Arima FTP site. Once the image is downloaded, the user runs the pipeline by specifying the location of Illumina NGS sequencing data and a few parameters. The pipeline will run HiCUP (Wingett et al. 2015) to align the Illumina sequencing reads of the Arima Hi-C library. The alignments will be passed to hic_breakfinder (Dixon et al. 2018) which will call structural variants using the Hi-C data. The Arima-SV Pipeline will also run Juicer (Durand et al. 2016b) which will generate a .hic file for visualization. The SV calls can be overlaid on top of the Hi-C heatmap using Juicebox (Durand et al. 2016a) to aid in data interpretation. In addition, the pipeline will output QC tables for shallow and deeply sequenced data sets which have QC metrics useful for assessing the quality of the Arima-HiC data.

1.2 hic_breakfinder Overview

The Arima-SV Pipeline runs the hic_breakfinder program to call structural variants (SVs) from the Arima Hi-C data. The program is called iteratively to optimize the resolution around the break point for SV calls. To call structural variants, hic_breakfinder normalizes for local chromatin structure and factors affecting Hi-C counts such as GC content, mapping rate and restriction enzyme cut site frequency (Dixon et al. 2018). Hic_breakfinder uses precomputed and empirically derived background models for calibrating the number of

expected Hi-C contacts both intra-chromosomally and inter-chromosomally. This approach is geared towards enabling the program to distinguish structural variants from the normal biological structure of chromosomes, mainly from Compartments, Topologically Associating Domains (TAD), and Looping structures. Intra- and inter-chromosomal breakpoints that are cause by structural variants are called when the observed signal is statistically higher than expected, given the background.

2. Getting Started

2.1 Installing Singularity

2.1.1. Singularity may already be installed on your HPC or cloud environment. On systems using the TORQUE scheduler, you can load singularity by running:

```
> module load singularity
```

2.1.2. To install Singularity refer to the documentations at https://apptainer.org/user-docs/master/quick_start.html or follow the steps below to install on an Ubuntu system.

2.1.3. Install and update required packages for singularity.

```
> sudo apt-get update && sudo apt-get install -y build-essential
libssl-dev uuid-dev libgpgme11-dev squashfs-tools libseccomp-dev
wget pkg-config git cryptsetup
```

2.1.4. Download the Go Installer for Linux

```
> wget https://go.dev/dl/go1.17.6.linux-amd64.tar.gz
```

2.1.5. Remove any previous installation of Go.

```
> sudo rm -rf /usr/local/go
```

2.1.6. Extract Go and put it in the directory '/usr/local'.

```
> sudo tar -C /usr/local -xzf go1.17.6.linux-amd64.tar.gz
```

2.1.7. Add Go to the $PATH variable and to the user profile.

```
> export PATH=$PATH:/usr/local/go/bin
```

```
>  echo 'export PATH=/usr/local/go/bin:$PATH' >> ~/.bashrc && \
  source ~/.bashrc
```

2.1.8. Verify the Go installation. The command below should print out the Go version number.

```
> go version
```

2.1.9. Create a variable with the version of Singularity to be downloaded.

```
> export VERSION=3.8.5
```

2.1.10. Download the Singularity source code.

```
> wget
https://github.com/hpcng/singularity/releases/download/v${VERSIO
N}/singularity-${VERSION}.tar.gz
```

2.1.11. Extract the Singularity code.

```
> tar -xzf singularity-${VERSION}.tar.gz
```

2.1.12. Change into the Singularity directory.

```
> cd singularity-${VERSION}
```

2.1.13. Compile the Singularity source code.

```
> ./mconfig
> make -C builddir
> sudo make -C builddir install
```

2.1.14. Verify Singularity installation. The command below should print the version number of the Singularity source code that was downloaded.

```
> singularity version
```

2.2 Downloading the Arima-SV Pipeline Singularity Image

The Arima-SV Pipeline along with all associated accessory files required for the analysis of human samples in hg38 coordinates are contained in a single Singularity image located at the Arima FTP site and can be downloaded with a single command:

```
> wget ftp://ftp-arimagenomics.sdsc.edu/pub/singularity/Arima-SV-Pipeline-
singularity-v1.sif
```

The Singularity image includes all tools, dependencies, accessory files, and test data needed to run the Arima-SV Pipeline and only needs to be downloaded once.

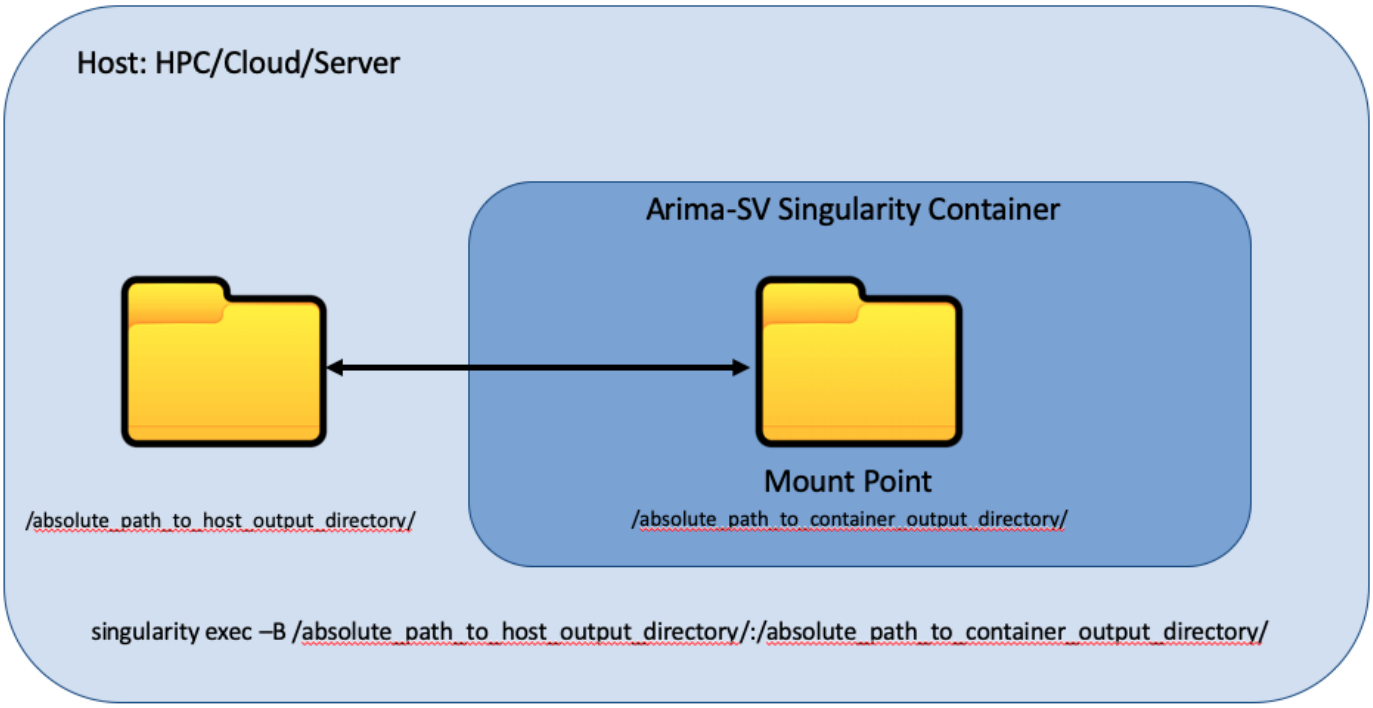


Figure 2. Schematic of relationship between the host and the container while running the Arima-SV Pipeline.

Containers such as Singularity containers package all the code and resources together so that download, installation and execution are simplified and streamlined. When the pipeline is run it will create a container, which is like a virtual environment on the "Host" machine (**Figure 2**). The "*Host*" is the physical hardware that the container is running in. To avoid execution errors always use absolute paths (relative to the root directory "/"). Do not use paths relative to a working directory nor paths to symbolic links. When the Arima-SV Pipeline is run, a user specified directory on the *Host* machine will be bound to another user-specified folder in the *Container*. The user-specific *Host* directory should be relative to the root directory on the *Host* machine and the user-specified container output directory (Mount Point) should be relative to the root directory in the *Container*. Data output in the container Mount Point will be replicated in the *Host* output folder. See **Figure 3** below for an example of binding directories between the *Host* and *Container* works in practice.

A.

```
[QuantumLooper1 /home/ubuntu/Arima-SV-Pipeline] >> ls /home/ubuntu/Arima-SV-Pipeline/tes
t_output/
[QuantumLooper1 /home/ubuntu/Arima-SV-Pipeline] >>
```

B.

```
[QuantumLooper1 /home/ubuntu/Arima-SV-Pipeline] >> singularity exec Arima-SV-Pipeline-
singularity-v1.sif ls /FFPE/mydata/
/bin/ls: cannot access '/FFPE/mydata/': No such file or directory
[QuantumLooper1 /home/ubuntu/Arima-SV-Pipeline] >>
```

C.

```
[QuantumLooper1 /home/ubuntu/Arima-SV-Pipeline] >> singularity exec -B /home/ubuntu/Ar
ima-SV-Pipeline/test_output/:/FFPE/mydata/ Arima-SV-Pipeline-singularity-v1.sif touch
/FFPE/mydata/binding_test.txt
[QuantumLooper1 /home/ubuntu/Arima-SV-Pipeline] >>
```

D.

```
[QuantumLooper1 /home/ubuntu/Arima-SV-Pipeline] >> ls /home/ubuntu/Arima-SV-Pipeline/tes
t_output/
binding_test.txt
[QuantumLooper1 /home/ubuntu/Arima-SV-Pipeline] >>
```

Figure 3. Example of binding Host and Container directories using the Arima-SV Pipeline Singularity image. **A**) listing (ls) the contents of the *Host* directory "/home/ubuntu/Arima-SV-Pipeline/test_output/" reveals that this *Host* folder is empty. **B**) Running the Arima-SV Pipeline Singularity image and listing the files in the *Container* directory "/FFPE/mydata/" reveals that this *Container* directory does not exist. **C**) An example of running a "touch" command in the Arima-SV-Pipeline Singularity image with a binding "-B" of the *Host* directory "/home/ubuntu/Arima-SV-Pipeline/test_output/" and the *Container* directory "/FFPE/mydata/". At run time the Container creates the directory "/FFPE/mydata/" inside itself and runs the touch command to create the file "binding_test.txt". The file "binding_test.txt" is copied to the *Host* directory "/home/ubuntu/Arima-SV-Pipeline/test_output/" and **D**) can be viewed on the *Host* by running an 'ls' command to list the files in /home/ubuntu/Arima-SV-Pipeline/test_output/.

2.3 Overview of Command line Arguments

Table 1. Overview of Required Command Line Arguments.

Argument	Required	Default Value	Description
-I	Yes	**User Input Required**	Absolute path to a pair of FASTQ files (*.fastq or *.fastq.gz) separated by "," (no space is allowed). Ex: "-I sample1_R1.fastq.gz,sample1_R2.fastq.gz" Ex: "-I sample1_R1.fastq,sample1_R2.fastq"
-o	Yes	**User Input Required**	Absolute path to the output directory. This is the directory where all of the output files will go.
-p	Yes	**User Input Required**	Output file prefix (filename only, not including the path nor file extension). Ex: "-p sample1"
-t	Yes	**User Input Required**	Number of threads to run HiCUP and Juicer
-W	Yes	1	Run HiCUP pipeline, "0" to skip. This flag calls HiCUP which aligns the sequencing reads to the reference genome.
-B	Yes	1	Run hic_breakfinder, "0" to skip. This flag calls hic_breakfinder which calls structural variants from the Arima Hi-C data.
-J	Yes	1	Run Juicer, "0" to skip. This flag calls Juicer which generates a .hic file for visualizing the Arima Hi-C data using the tool Juicebox.
-H	Yes	0	Run HiCCUPS, "0" to skip. This flag calls HiCCUPS which calls statistically significant loops from the Arima Hi-C data.
-r	Yes	Hg38 reference file	Absolute path to the reference genome FASTA file for aligning reads (Accessory file included in the Singularity Image)
-s	Yes	Hg38 chromosome sizes file	Absolute path to the chromosome sizes for reference genome (Accessory file included in the Singularity Image)

-c	Yes	Arima 2 enzyme cut sites file	Absolute path to the cut site file used by Juicer pipeline (Accessory file included in the Singularity Image)
-a	Yes	Pre-specified	bowtie2 tool location reads (Tool included in the Singularity Image)
-x	Yes	Hg38 index	bowtie2 index file prefix (Accessory file included in the Singularity Image)
-d	Yes	Arima 2 Enzyme Digest file	Absolute path to the genome digest file produced by hicup_digester (Accessory file included in the Singularity Image)
-w	Yes	Pre-specified	Absolute path to the directory of the HiCUP tool (Tool included in the Singularity Image)
-b	Yes	Pre-specified	Absolute path to the directory of the hic_breakfinder tool (Tool included in the Singularity Image)
-j	Yes	Pre-specified	Absolute path to the directory of the Juicer tool (Tool included in the Singularity Image)
-e	Yes	Hg38 intra-chromosomal expectation file	Hic_breakfinder intra-chromosomal background model file for normalization (Accessory file included in the Singularity Image)
-E	Yes	Hg38 inter-chromosomal expectation file	Hic_breakfinder inter-chromosomal background model file for normalization (Accessory file included in the Singularity Image)

Table 2. Overview of Optional Command Line Arguments.

Argument	Required	Default Value	Description
-v	No	N/A	Print version number and exit
-h	No	N/A	Print help and exit

2.4 Arima-HiC Data QC and Sequencing Recommendations

Prior to deep sequencing Arima-HiC libraries (defined here as >50M reads), we recommend performing a shallow sequencing run using a low throughput platform such as on the Illumina® MiniSeq® or MiSeq® to obtain approximately 0.5-2M read-pairs for QC assessment of the libraries and Arima-HiChIP data. The shallow sequencing QC metrics output by the Arima-SV Pipeline are important for assessing library quality the percentage of long-range chromatin interactions, which in turn is used to estimate the required deep sequencing depth needed for robust and reproducible SV discovery. The Arima-SV Pipeline is executed the same for either shallow or deep sequencing data, with the exception of the computational resources required. See Section 2.5 below for further discussion on computational resources.

2.5 Compute Resources

For shallow sequencing (0.5 - 2 million raw read-pairs), the Arima-SV Pipeline requires 8-12 CPU cores with 32-48 GB RAM. The shallow sequencing analysis should complete in less than 2 hours, depending on hardware. For deep sequencing (50 - 500 million raw read-pairs), we recommend 16-20 CPU cores with at least 64-80 GB RAM. Samples with 100-150 million raw read-pairs will run through the Arima-SV Pipeline in about 3 days with the recommended computational resources. Additional resources can be added to decrease the analysis time.

2.6 How to Cite the Arima-SV Pipeline in Publications

When citing the Arima-SV Pipeline please use: “Structural Variants were called using the Arima-SV Pipeline (https://github.com/ArimaGenomics/Arima-SV-Pipeline)”.

3. Running the Arima-SV Pipeline

3.1 Overview

Inputs:

- Arima-SV Pipeline Singularity image
- Paired-end Arima Hi-C shallow or deep sequencing data in .fastq or .fastq.gz format
- The name of the output directory
- A prefix for naming the outputs
- An HPC or cloud computing environment (see Section 2.5 for computing resources)

Outputs:

- Arima-SV shallow and deep sequencing QC tables for use with the **Worksheet Arima-HiC Quality Control For Structural Variants.xls (doc#: A160431)**.
- SV Calls generated from calling hic_breakfinder in .bedpe format
- Hi-C heatmap file in .hic format used for visualizing the Hi-C data for the sample using Juicebox.

This section walks the user step-by-step through running the Arima-SV pipeline. This section also, introduces containers and how to work with them and it introduces the key output files and where they are located in the output directory.

3.2 Arima-SV Analysis Workflow

3.2.1. Open a terminal or command prompt and connect to a server, HPC, or cloud environment of choice. Navigate to the desired working directory.

3.2.2. Download the Singularity image of the Arima-SV Pipeline to the directory where tools and programs are stored on the system (see Section 2.1, above).

3.2.3. Source the FASTQ files that will be run through the Arima-SV Pipeline.

```
[QuantumLooper1 /home/ubuntu/Arima-SV-Pipeline] >> ls
Arima-SV-Pipeline-singularity-v1.sif  fastq  test_output
[QuantumLooper1 /home/ubuntu/Arima-SV-Pipeline] >> ls fastq/
test_1M_R1.fastq.gz  test_1M_R2.fastq.gz
[QuantumLooper1 /home/ubuntu/Arima-SV-Pipeline] >>
```

Figure 4. Shows the location of test data used for demonstration purposes in this demo.

3.2.4. Run the Arima-SV Pipeline on Arima Hi-C data. The below command shows a generic example of how to run the analysis and **Figure 5** shows the specific command that was run for analysis for this demo. The generic command for running the Arima-SV Pipeline can be copied from the Arima Genomics GitHub page: https://github.com/ArimaGenomics/Arima-SV-Pipeline or from the text below:

```
> singularity exec -B [HOST_OUTPUT_DIR]:[CONTAINER_OUTPUT_DIR]
  Arima-SV-Pipeline-singularity-v1.sif \
  bash /FFPE/Arima-SV-Pipeline-v1.sh \
  -I
  [ABSOLUTE_PATH_TO_FASTQ1_IN_CONTAINER],[ABSOLUTE_PATH_TO_FASTQ1
  _IN_CONTAINER] \
  -o [ABSOLUTE_PATH_TO_OUTPUT_DIR_IN_CONTAINER] \
  -p [PREFIX] \
  -t [THREADS] \
  -W 1 \
  -B 1 \
  -J 1 \
  -H 0 \
  -a /root/anaconda3/bin/bowtie2 \
  -b /usr/local/bin/ \
  -w /FFPE/HiCUP-0.8.0/ \
  -j /FFPE/juicer-1.6/ \
  -r /FFPE/Arima_files/reference/hg38/hg38.fa \
  -s /FFPE/Arima_files/Juicer/hg38.chrom.sizes \
  -c /FFPE/Arima_files/Juicer/hg38_GATC_GANTC.txt \
  -x /FFPE/Arima_files/reference/hg38/hg38 \
  -d /FFPE/Arima_files/HiCUP/Digest_hg38_Arima.txt \
  -e
  /FFPE/Arima_files/hic_breakfinder/intra_expect_100kb.hg38.txt \
  -E /FFPE/Arima_files/hic_breakfinder/inter_expect_1Mb.hg38.txt
  \
  &> log.txt
```

In the command above, Singularity is called to run the Arima-SV Pipeline Singularity image. The User Specified directory from the *Host* "[HOST_OUTPUT_DIR]" is bound to the Container directory "[CONTAINER_OUTPUT_DIR]" using the -B option. The script that runs the Arima-SV Pipeline "`Arima-SV-Pipeline-v1.sh`" is called inside the Singularity container with the options indicated. Absolute path to Arima Hi-C data is specified using the "-I" option. The 2 reads (R1 and R2) are separated by commas without any spaces. Absolute path to the output directory is specified using the "-o" option. The prefix of the output file names is specified using the "-p" option. The number of threads to use when running the Arima-SV Pipeline is specified using the "-t" option. See Section 2.5 above for a description of computing resources needed for shallow and deeply sequenced datasets. The location of tools and accessory files needed for running the Arima-SV Pipeline are specified with options "-a" through "-E". No input is required from the user for these inputs. The last line writes the standard output from the pipeline to a log file.

When running the Arima-SV Pipeline on the test data from **Figure 4**, the *Host* directory "/home/ubuntu/Arima-SV-Pipeline/" is bound to the *Container* directory "/FFPE/mydata/" (**Figure 5**). It can be useful to think of this binding of the *Host* and *Container* directories as an equivalence, for practical purposes. In this example one should consider that the *Container* directory "/FFPE/mydata/" equals the *Host* directory "/home/ubuntu/Arima-SV-Pipeline/"at run time and has access to the files in the *Host* directory. The Arima Hi-C sequencing data: test_1M_R1.fastq.gz and test_1M_R2.fastq.gz which are located on the *Host* machine in the

directory “/home/ubuntu/Arima-SV-Pipeline/fastq/” are passed to “-I” using the absolute path to those files from within the *Container* directory “/FFPE/mydata/fastq/”. The output directory is specified with the “-o” flag and the *Container* directory “/FFPE/mydata/test_output/” passed to it. As the program runs, output files in the *Container* directory “/FFPE/mydata/test_output/” will be placed in the host directory “/home/ubuntu/Arima-SV-Pipeline/test_output/”. The file prefix for the output files “test_1M” are passed to the “-p” flag and the number 12 was passed to the “-t” flag to set the number of threads for the Arima-SV Pipeline to use.

```
[QuantumLooper1 /home/ubuntu/Arima-SV-Pipeline] >> singularity exec -B /home/ubuntu/Arim
a-SV-Pipeline/:/FFPE/mydata/ Arima-SV-Pipeline-singularity-v0.sif \
> bash /FFPE/Arima-FFPE-v0.1.sh \
> -I /FFPE/mydata/fastq/test_1M_R1.fastq.gz,/FFPE/mydata/fastq/test_1M_R2.fastq.gz \
> -o /FFPE/mydata/test_output/ \
> -p test_1M \
> -t 12 \
> -W 1 \
> -B 1 \
> -J 1 \
> -H 0 \
> -a /root/anaconda3/bin/bowtie2 \
> -b /usr/local/bin/ \
> -w /FFPE/HiCUP-0.8.0/ \
> -j /FFPE/juicer-1.6/ \
> -r /FFPE/Arima_files/reference/hg38/hg38.fa \
> -s /FFPE/Arima_files/Juicer/hg38.chrom.sizes \
> -c /FFPE/Arima_files/Juicer/hg38_GATC_GANTC.txt \
> -x /FFPE/Arima_files/reference/hg38/hg38 \
> -d /FFPE/Arima_files/HiCUP/Digest_hg38_Arima.txt \
> -e /FFPE/Arima_files/hic_breakfinder/intra_expect_100kb.hg38.txt \
> -E /FFPE/Arima_files/hic_breakfinder/inter_expect_1Mb.hg38.txt \
> &> log.txt
[QuantumLooper1 /home/ubuntu/Arima-SV-Pipeline] >>
```

Figure 5. A specific example of how to run the Arima-SV Pipeline.

3.2.5. Open the log.txt file to see if any errors occurred while running the Arima-SV Pipeline. **Figure 6** shows examples of the log file outputs when a run completes successfully. If technical assistance is required, please send the log file (log.txt) with the detailed run script to techsupport@arimagenomics.com.

A.

```
Running HiCUP [2022/02/17 05:24:03] ...
/FFPE/HiCUP-0.8.0//hicup --config /FFPE/mydata/test_output//hicup//hicup.conf &> /FFPE/mydata/test_output//hicup//
hicup.log
Finished running HiCUP! [2022/02/17 05:38:23]

Output BAM file from HiCUP: /FFPE/mydata/test_output//hicup//test_1M_R1_2.hicup.bam

Running hic_breakfinder [2022/02/17 05:38:23] ...
/usr/local/bin//hic_breakfinder --bam-file /FFPE/mydata/test_output//hicup//test_1M_R1_2.hicup.bam --exp-file-
inter /FFPE/Arima_files/hic_breakfinder/inter_expect_1Mb.hg38.txt --exp-file-intra /FFPE/Arima_files/
hic_breakfinder/intra_expect_100kb.hg38.txt --name /FFPE/mydata/test_output//hic_breakfinder//test_1M --min-1kb &> /
FFPE/mydata/test_output//hic_breakfinder//hic_breakfinder.log
Finished running hic_breakfinder! [2022/02/17 06:16:27]

Output SV .txt file from hic_breakfinder: /FFPE/mydata/test_output//hic_breakfinder//test_1M.breaks.txt

Output SV .bedpe file from hic_breakfinder: /FFPE/mydata/test_output//hic_breakfinder//test_1M.breaks.bedpe
```

B.

```
Running Juicer [2022/02/17 06:16:32] ...
/FFPE/juicer-1.6//scripts/juicer.sh -d /FFPE/mydata/test_output//juicer/ -p /FFPE/Arima_files/Juicer/
hg38.chrom.sizes -s Arima -y /FFPE/Arima_files/Juicer/hg38_GATC_GANTC.txt -z /FFPE/Arima_files/reference/hg38/
hg38.fa -D /FFPE/juicer-1.6/ -f -t 12 &> /FFPE/mydata/test_output//juicer//juicer.log
Finished running Juicer! [2022/02/17 07:28:38]

Output filtered .hic file from Juicer: /FFPE/mydata/test_output//juicer//aligned/inter_30.hic

Arima FFPE pipeline finished successfully! [2022/02/17 07:28:38]

Please download the QC result from: /FFPE/mydata/test_output//test_1M_Arima_QC_deep.txt and /FFPE/mydata/
test_output//test_1M_Arima_QC_shallow.txt and then copy the contents to the corresponding tables in the QC
worksheet.

The filtered .hic file from Juicer pipeline for visualization using Juicebox is located at: /FFPE/mydata/
test_output//juicer//aligned/inter_30.hic
The SV .txt file from hic_breakfinder is located at: /FFPE/mydata/test_output//hic_breakfinder//test_1M.breaks.txt
The SV .bedpe file from hic_breakfinder is located at: /FFPE/mydata/test_output//hic_breakfinder//
test_1M.breaks.bedpe
```

Figure 6. Key sections from the output logs. A) Selected output from "log.txt" which shows the successful running of HiCUP and hic_breakfinder. **B**) Selected output from "log.txt" which shows the successful running of Juicer and successful completion of the Arima-SV Pipeline.

3.2.6. Navigate to the output directory on the Host machine to view the outputs of the program. **Figure 7** shows screenshots of the output directories and key output files from the Arima-SV Pipeline.

3.2.7. These files should be used for viewing the structural variant calls in Juicebox along with the Hi-C heatmap to aid in the interpretation of the biology of the sample.

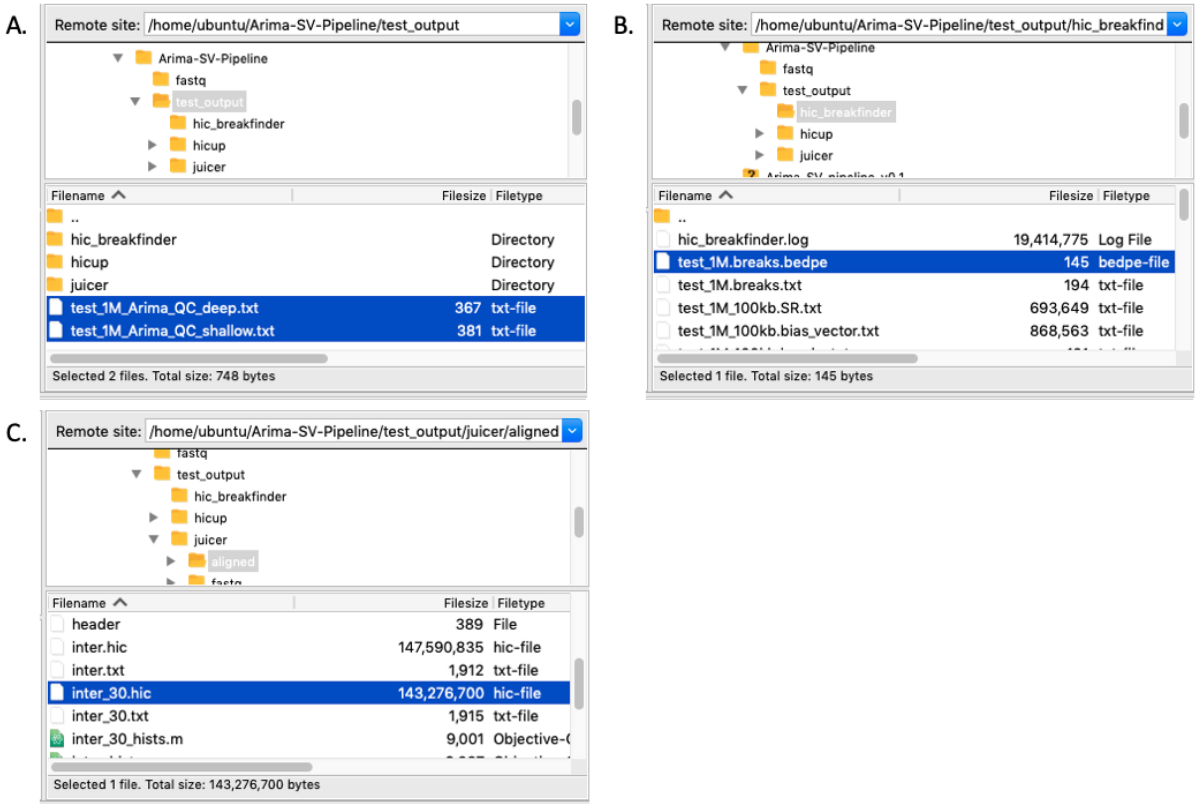


Figure 7. Screenshots of key files in the Arima-SV Pipeline output folders. **A**) The Arima_QC_shallow.txt and Arima_QC_deep.txt contain QC metrics for the Arima Hi-C libraries the pipeline was run on. **B**) The "hic_breakfinder" directory contains output files from the hic_breakfinder program. The key output file in this directory is the *.breaks.bedpe which contains all of the SV calls from the pipeline in a .bedpe format. **C**) The "juicer" directory

contains the outputs from juicer. The key file in this directory is nested inside the “aligned” directory and is called “inter_30.hic”. The “inter_30.hic” file is used for viewing the Hi-C heatmap using Juicebox and contains all Hi-C interactions that have a mapping alignment quality score of Q30 or greater. Alternately, the “inter.hic” contains all Hi-C interactions without with thresholding the alignment quality and may be appropriate for some applications.

4. Using and Viewing Pipeline Outputs

4.1 Introduction

The Arima-SV Pipeline creates several key output files:

1. The Arima Shallow QC file
2. The Arima Deep Seq QC file
3. SV calls from hic_breakfinder
4. .hic files from Juicer

This section walks through how to view the data in each of these files, what the primary metrics mean, how to interpret outputs, and how to visualize the results.

4.2 Arima Shallow and Deep Sequencing QC metrics files

The Arima-SV Pipeline outputs QC tables for use with the **Worksheet Arima-HiC Quality Control For Structural Variants.xls (doc#: A160431)** to aid in interpretation of the Arima Hi-C data quality. These two tables with the suffix "_Arima_QC_shallow.txt" and "_Arima_QC_deep.txt", are created in the output directory specified by "-o" from the command line. The "_Arima_QC_shallow.txt" file includes metrics related to raw sequencing depth, mapping rate, duplication rate, the target raw reads for deep sequencing, a summary of the reads used for loop calling, a summary of the Hi-C characteristics of the data. The "_Arima_QC_deep.txt" file has the same QC metrics as the shallow sequencing file with the exception of target sequencing depth. Instead, the "_Arima_QC_deep.txt" file reports the number of Structural Variants discovered by hic_breakinder. These tables should be copied and pasted into the **Worksheet Arima-HiC Quality Control For Structural Variants.xls (doc#: A160431)** that accompanies the **User Guide: Arima-HiC for Formalin Fixed Paraffin Embedded (FFPE) Tissues (doc#: A160172)**. **Table 3** describes the metrics in the QC output files and the expected ranges for the metrics.

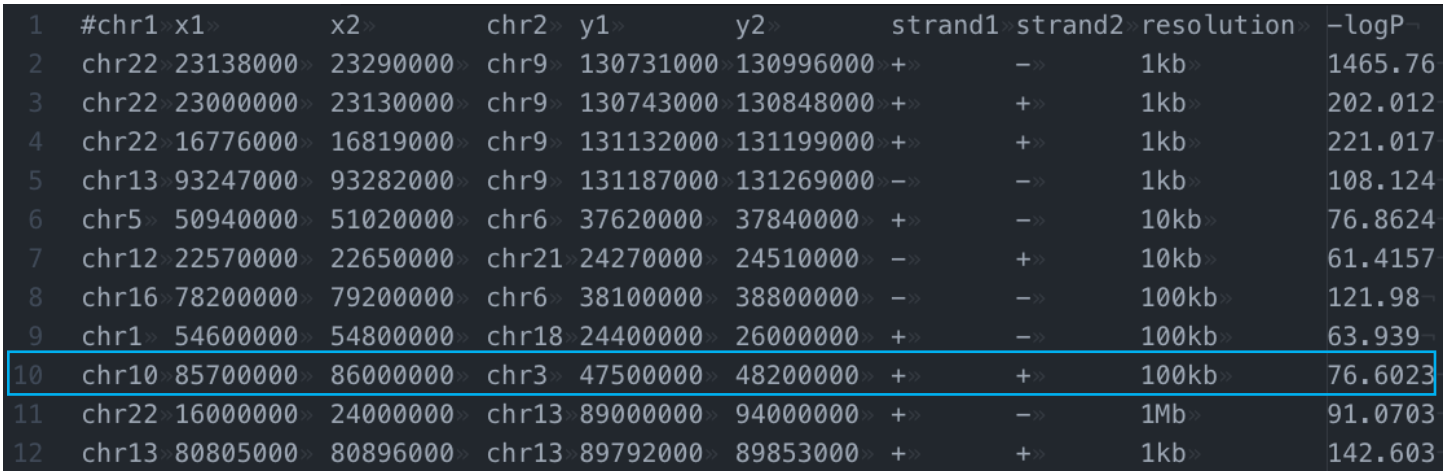
Table 3. Description of Arima Hi-C QC Metric.

Metric	Definition	Target Spec.
Raw PE Reads	Raw paired-end (PE) sequence reads	0.5-2M Shallow seq, 100-200M Deep seq
Mapped SE Reads	Single-end reads aligned to the genome with a mapping quality >= 30	Library specific
% Mapped SE Reads	Percentage of all raw single-end reads that align to the genome relative to the total raw single-end read count	≥80%
Duplicates	Duplicate read-pairs aligned to the genome	Library specific

% Duplicates	Percentage of all mapped paired-end reads that are PCR duplicates	≤2% for shallow seq, ≤30% for deep seq
Unique Valid Pairs	HiC read-pairs which are not derived from artifacts such as self-circles and dangling-ends, and which contain spatial proximity information	Library specific
% Unique Valid Pairs	% of mapped reads that are unique valid pairs	Library specific
Library Complexity	Theoretical number of unique molecules in a Hi-C library.	100M - 1B of read pairs
INTRA pairs	Unique valid pairs where both read ends align to the same chromosome	Library specific
% INTRA pairs	Percentage of all Unique Valid Pairs that have both read-ends aligning to the same chromosome	≥80%
INTRA > 15kb pairs (Lcis)	Unique Valid Pairs where both read-ends align to the same chromosome and have an insert size >=15kb	Library specific
% INTRA > 15kb pairs (Lcis)	Percentage of all Unique Valid Pairs that have both read-ends aligning to the same chromosome and have an insert size >=15kb	≥25%
INTER pairs (Trans)	Unique Valid Pairs where each read-end aligns to a different chromosome	Library specific
% INTER pairs (Trans)	Percentage of all Unique Valid Pairs where each read-end aligns to a different chromosome	≤20%
Lcis / Trans	The ratio of Lcis to Trans data. This is the signal to noise ratio for translocation calling.	≥1
Target Raw Reads for Deep Seq	Number of Raw PE Reads to obtain high sensitivity SV calls Based on bench marking against ground truth datasets. This value is calibrated to balance Sensitivity and Specificity of the SV calls from hic_breakfinder. This value uses conservative values for the mapping rate at 80% and the duplicate rate at 30%.	100-200M paired reads
Number of SV Calls	The number of SV calls made by the Arima-SV Pipeline.	Library specific

4.3 hic_breakfinder SV calls

The SV calls from hic_breakfinder are located in the "*.breaks.bedpe" file in the "hic_breakfinder" output folder (**Figure 7**). This file is a 10-column tab-delimited file that represents the genomic coordinates of the partner sequences involved in a structural variant. The first column, "chr1" is the chromosome ID for the first partner, the second column, "x1" is the starting coordinate of the first partner and the third column, "x2" is the end coordinate of the first partner. Likewise, columns 4-6 ("chr2", "y1", "y2") have the chromosome ID, starting coordinate, and ending coordinate of the second partner. This format is different from traditional SV calls which specify a precise break point and is due to the resolution of the SV call from hic_breakfinder and can be found in column 9 "resolution". This results in a square being visible when overlaid on a heatmap in Juicebox. Columns 7, "strand1" and column 8, "strand2" indicate which corner of the square SV call is likely to contain the breakpoint. A "+" indicates that the breakpoint is predicted near the end coordinate (3') and a "-" predicts that the breakpoint is predicted at the start coordinate (5') for a given partner. The 10th column is the -log scaled p-value of the SV call. For instance, Line 10 in **Figure 7** has a translocation between chromosome 10 and chromosome 3 with the breakpoint estimated to be at chr3:48,200,000 and chr10:86,000,000 based on the strand information.



	#chr1	x1	x2	chr2	y1	y2	strand1	strand2	resolution	-logP
2	chr22	23138000	23290000	chr9	130731000	130996000	+	-	1kb	1465.76
3	chr22	23000000	23130000	chr9	130743000	130848000	+	+	1kb	202.012
4	chr22	16776000	16819000	chr9	131132000	131199000	+	+	1kb	221.017
5	chr13	93247000	93282000	chr9	131187000	131269000	-	-	1kb	108.124
6	chr5	50940000	51020000	chr6	37620000	37840000	+	-	10kb	76.8624
7	chr12	22570000	22650000	chr21	24270000	24510000	-	+	10kb	61.4157
8	chr16	78200000	79200000	chr6	38100000	38800000	-	-	100kb	121.98
9	chr1	54600000	54800000	chr18	24400000	26000000	+	-	100kb	63.939
10	chr10	85700000	86000000	chr3	47500000	48200000	+	+	100kb	76.6023
11	chr22	16000000	24000000	chr13	89000000	94000000	+	-	1Mb	91.0703
12	chr13	80805000	80896000	chr13	89792000	89853000	+	+	1kb	142.603

Figure 7. Screenshot of the hic_breakfinder's *.breaks.bedpe output file. The cyan box highlights the translocation between chromosome 3 and chromosome 10.

4.4 Viewing .hic files in Juicebox

4.4.1. You can view the .hic file using the online portal located at: https://aidenlab.org/juicebox/. Or you may install a local version of the Juicebox.

4.4.2. Install Juicebox locally by following the developer's instructions https://github.com/aidenlab/Juicebox/wiki/Download.

4.4.3. Copy the .hic file and the *.breaks.bedpe file to the local computer where the visualization will take place.

4.4.4. Open the Juicebox application, click on the file tab, and then click on "Open" to pull up the dialogue box for importing data (**Figure 8**).

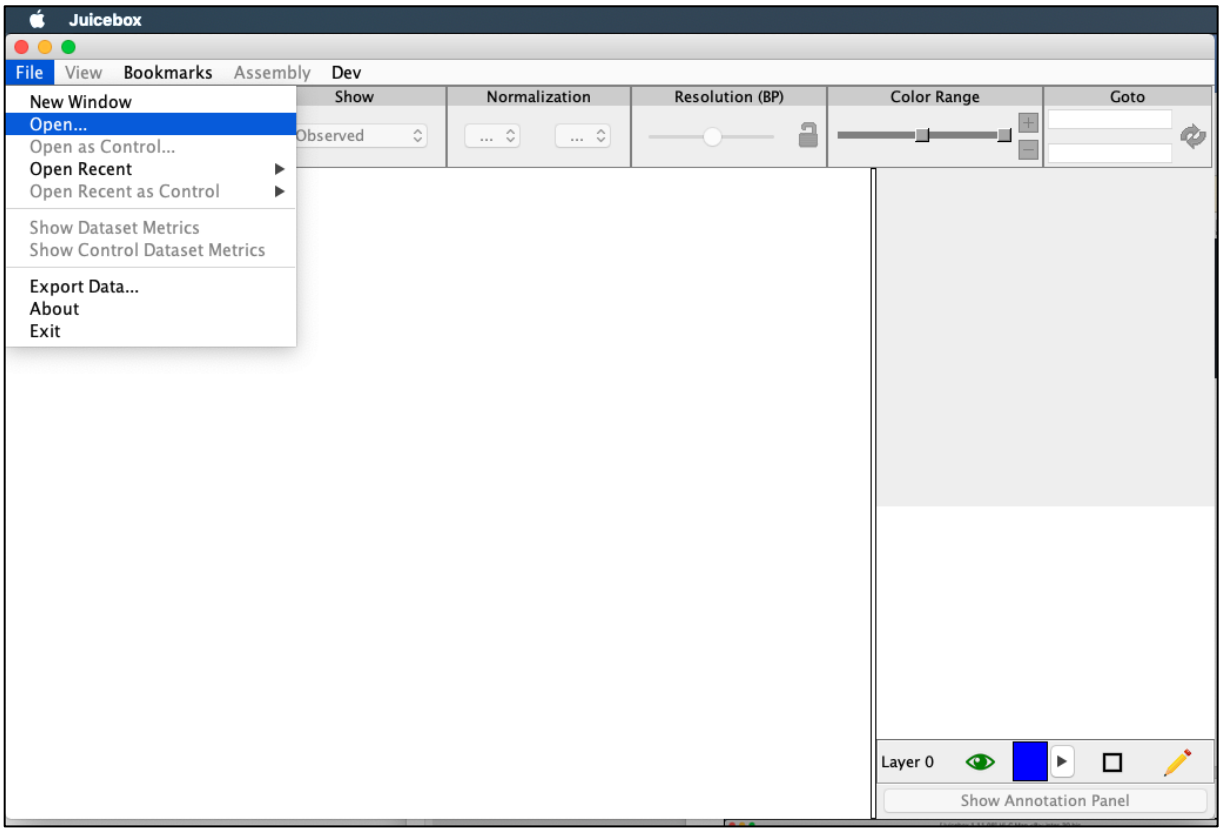


Figure 8. Screenshot of Juicebox dialogue box for importing data.

4.4.5. Select "Local" from the dialogue box and select the .hic file copied to the local computer (**Figure 9**).

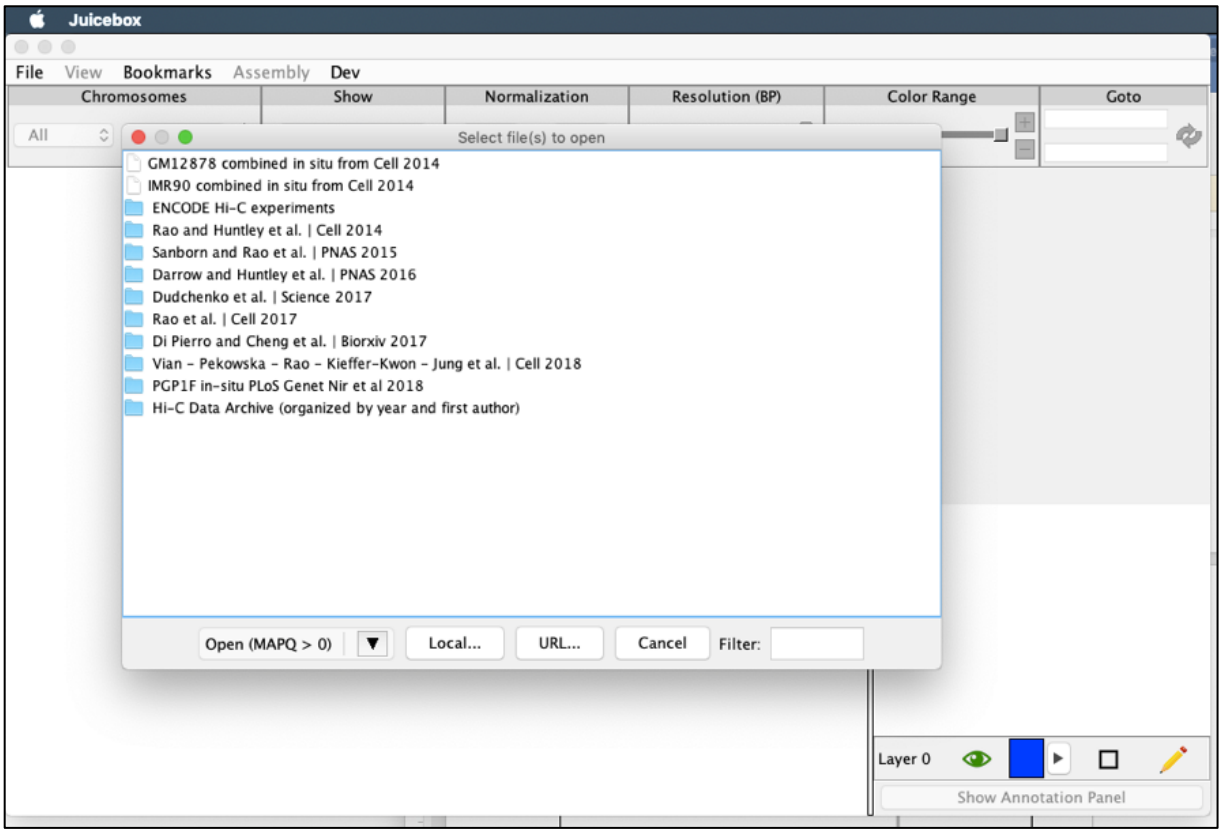


Figure 9. Screenshot of the Juicebox dialogue screen for opening data.

4.4.6. The data will be loaded into Juicebox and will initially present a genome-wide view of the .hic data at low resolution (**Figure 10**)

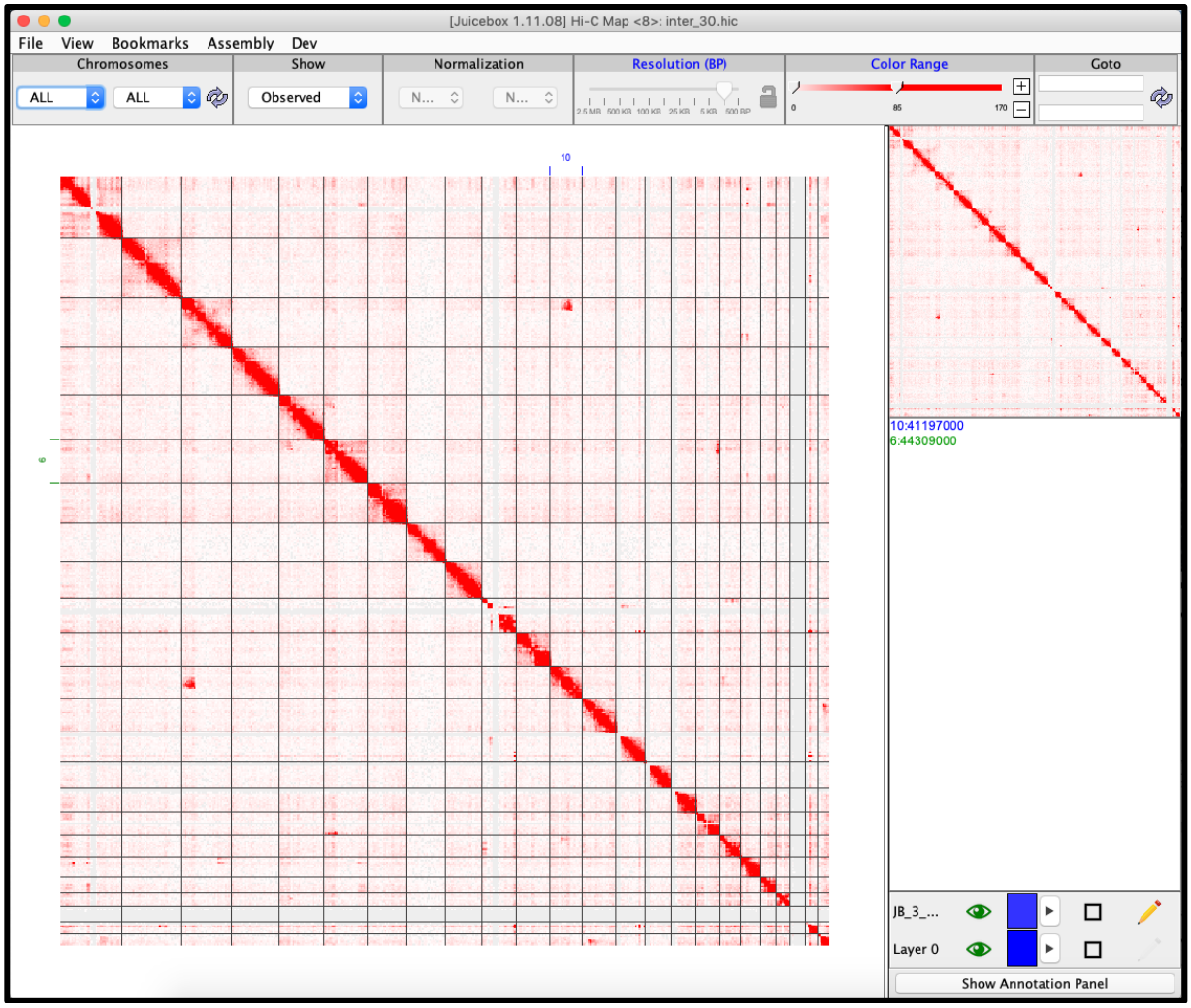


Figure 10. Screenshot of .hic file form the Arima-SV Pipeline visualized in Juicebox. The chromosomes are laid end-to-end across the x-axis and across the y-axis. The intensity of each pixel represents the number of Arima Hi-C counts that were sequenced between two genomic loci and the frequency with which those two loci interact with each other in the cell nucleus. Notice the large translocation between chromosome 3 and chromosome 10. The Arima Hi-C library in this figure was subsampled to 5M paired-end reads prior to running through the pipeline.

4.4.7. Load the *.breaks.bedpe file into Juicebox by clicking on the "Show Annotation Panel" in the lower right hand corner of the application (**Figure 11**). Click on "2D Annotations" in the top of the dialogue box. Then click "Add Local" to upload the *.breaks.bedpe file (**Figure 11**). The file is now added to the Annotations Panel in Juicebox but is not open. **Note: 1D annotations such as gene tracks or epigenetic marks can be added by clicking on the "1D Annotations" tab at the top and uploading the desired data.**

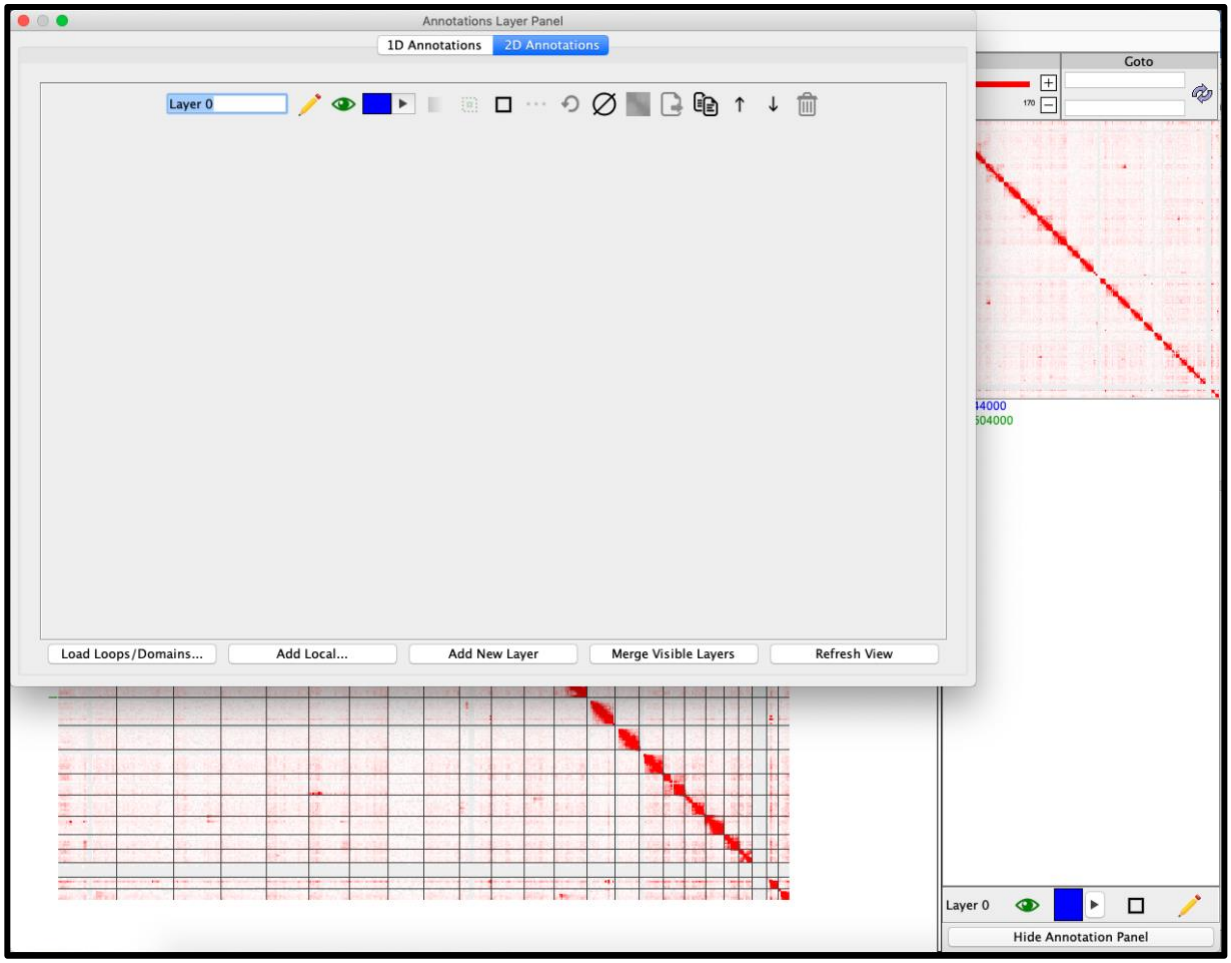


Figure 11. Screenshot of the Annotation Panel

4.4.8. In the annotation dialogue box, click on the *.breaks.bedpe file just uploaded then click “open” (**Figure 12**). The file is now open in Juicebox and added to an “Annotation Layer” (**Figure 13**).

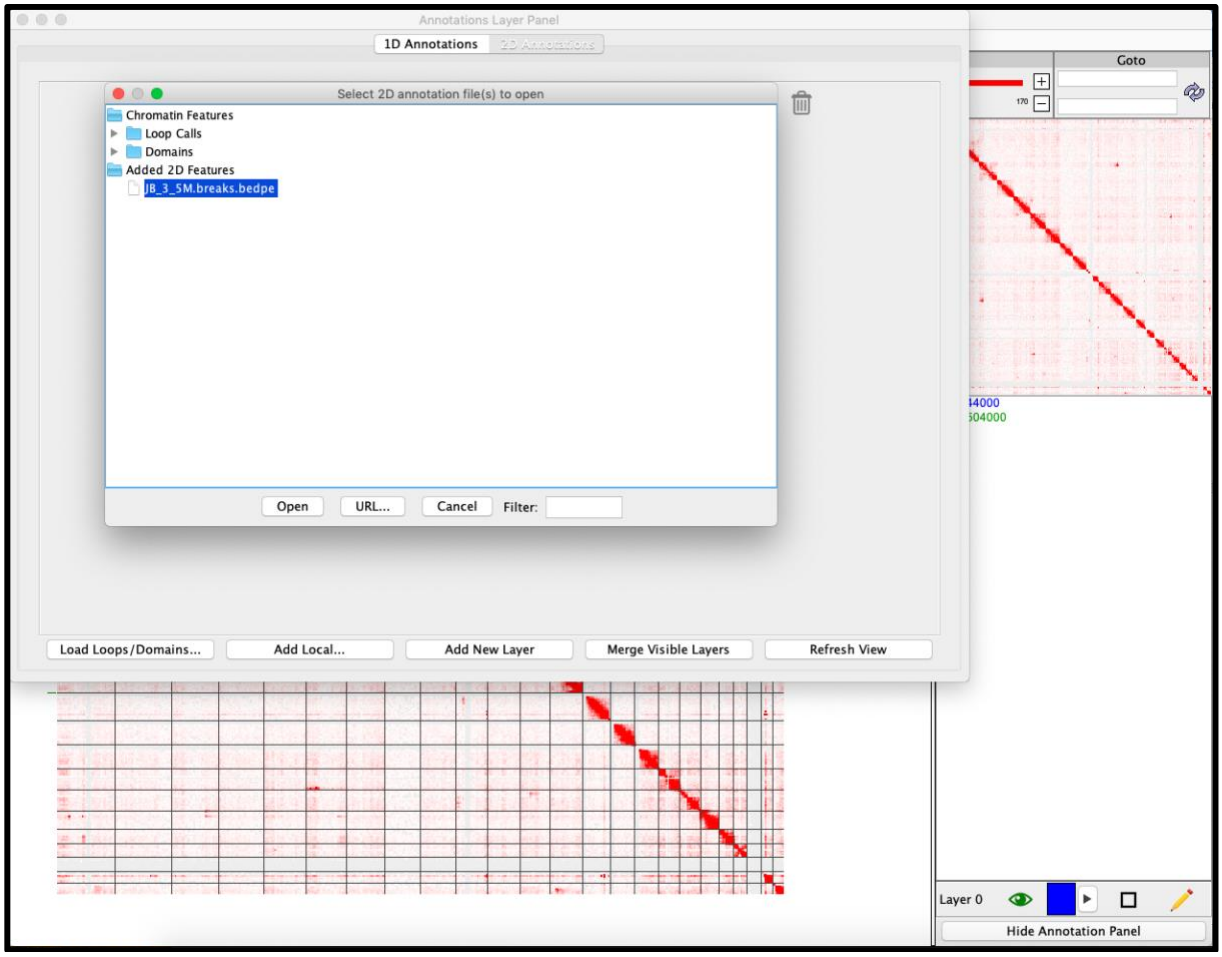


Figure 12. Screenshot of the dialogue panel in Juicebox to add the *.breaks.bedpe file to an annotation layer.

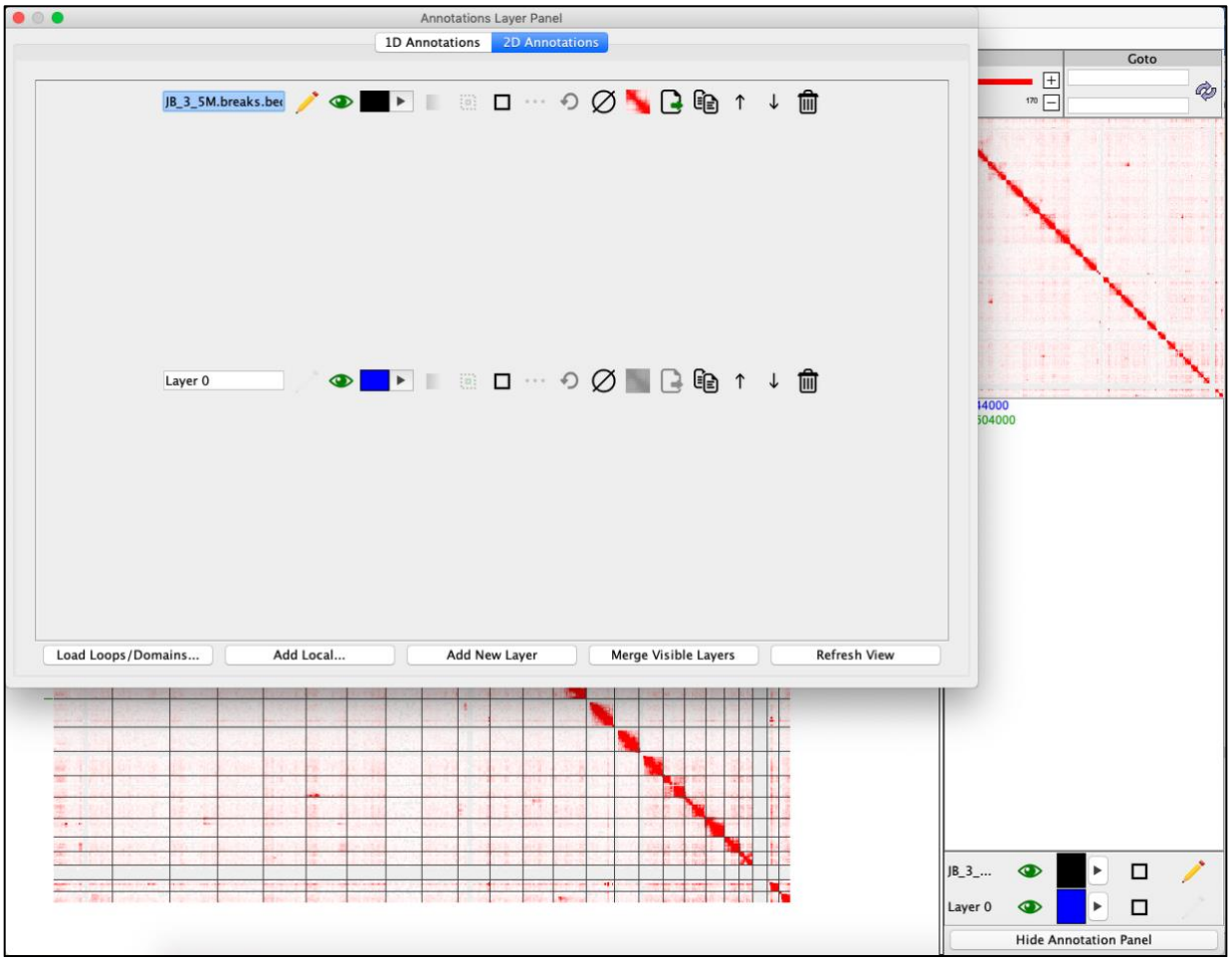


Figure 13. Screenshot of the Annotation Panel with a *.breaks.bedpe file loaded.

4.4.9. Click on "refresh view" then close the Annotation panel and return to the heatmap.

4.4.10. Click on any intra- or inter-chromosomal region in the heatmap to zoom in. For this demo, we will zoom into the inter-chromosomal region between chromosome 3 and chromosome 10 which we previously showed to have a translocation called in the hic_breakfinder output in **Figure 7** (**Figure 14**).

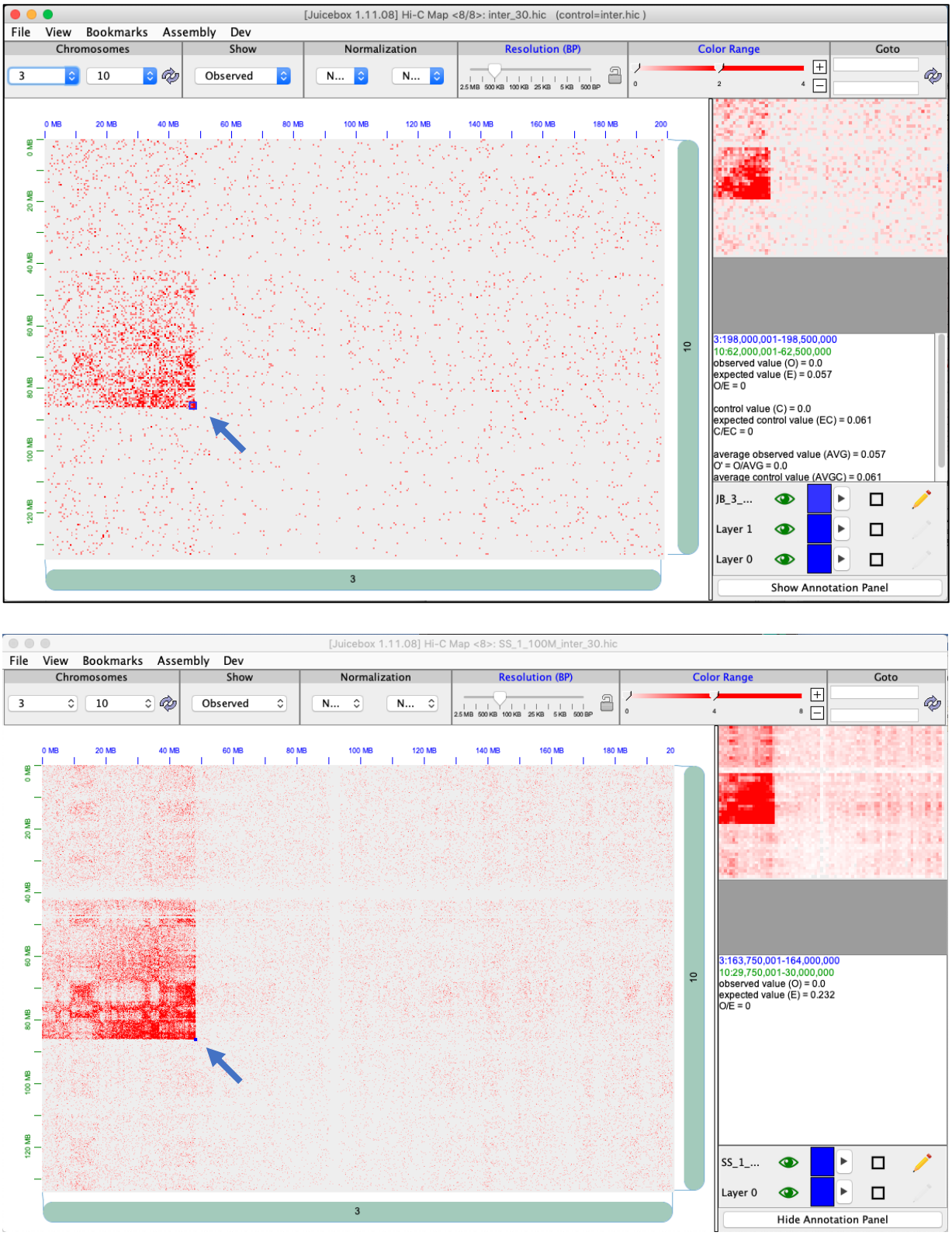


Figure 14. Screenshot of the translocation on chromosome 3 and chromosome 10. Top. Heatmap from a sample with 5M Pair-end reads. The blue box indicates the SV call from hic_breakfinder. Blue arrow is pointing to the SV call breakpoint. **Bottom.** Heatmap from an independent replicate with 100M paired-end reads.

4.4.11. Additional annotations such as gene tracks or epigenetic marks can be added to the heatmap using the steps above to aid in interpretation of the biology of the sample.

References

Dixon JR, Xu J, Dileep V, Zhan Y, Song F, Le VT, Yardımcı GG, Chakraborty A, Bann D V, Wang Y, et al. 2018. Integrative detection and analysis of structural variation in cancer genomes. *Nat Genet* **50**: 1388–1398. https://pubmed.ncbi.nlm.nih.gov/30202056.

Durand NC, Robinson JT, Shamim MS, Machol I, Mesirov JP, Lander ES, Durand NC, Robinson JT, Shamim MS, Machol I, et al. 2016a. Juicebox Provides a Visualization System for Hi-C Contact Maps with Unlimited Zoom Tool Juicebox Provides a Visualization System for Hi-C Contact Maps with Unlimited Zoom. *Cell Syst* **3**: 99–101. http://dx.doi.org/10.1016/j.cels.2015.07.012.

Durand NC, Shamim MS, Machol I, Rao SSP, Huntley MH, Lander ES, Aiden EL. 2016b. Juicer Provides a One-Click System for Analyzing Loop-Resolution Hi-C Experiments. *Cell Syst* **3**: 95–98. https://www.sciencedirect.com/science/article/pii/S2405471216302198.

Wingett S, Ewels P, Furlan-magaril M, Nagano T, Schoenfelder S, Fraser P, Andrews S. 2015. HiCUP : pipeline for mapping and processing Hi-C data [version 1 ; referees : 2 approved , 1 approved with reservations] Referee Status : **1310**: 1–12.

Warranty and Contact Info

WARRANTY

All warranties are personal to the Purchaser and may not be transferred or assigned to a third-party, including an affiliate of the Purchaser. The warranty described below excludes any stand-alone third-party goods that may be acquired or used with the product. Arima Genomics only warrants that the kit reagents will be made and tested in accordance with Arima Genomics manufacturing and quality control processes. Arima Genomics makes no warranty that the reagents provided in this kit will work as intended by the Purchaser's or for the Purchaser's intended uses. ARIMA GENOMICS MAKES NO OTHER WARRANTY, EXPRESSED OR IMPLIED. THERE IS NO WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. THERE IS NO WARRANTY OF NON_INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS. The warranty provided herein and the data and descriptions of Arima Genomics products appearing in Arima Genomics product literature and website may not be altered except by express written agreement signed by an officer of Arima Genomics. Representations, oral or written, which are inconsistent with this warranty, or such publications are not authorized and if given, should not be relied upon.

The foregoing warranties do not apply to the extent a non-conformance is due to (i) abuse, misuse, neglect, negligence, accident, improper storage, or use contrary to the Documentation or Specifications, (ii) use that is an Excluded Use, (iii) improper handling, (iv) unauthorized alterations, (v) natural disasters, or (vi) third-party's use with a third-party's good that is not specified in the product documentation. In the event of a breach of the foregoing warranty, customer shall promptly contact Arima Genomics customer support to report the non-conformance and shall cooperate with Arima Genomics in confirming or diagnosing the non-conformance. Additionally, Arima Genomics may request return shipment of the non-conforming product at Arima Genomics cost. Arima Genomics sole obligation shall be to replace the applicable product or part thereof, provided the customer notifies Arima Genomics within 90 days of any such breach. If after exercising reasonable efforts, Arima Genomics is unable to replace the product, then Arima Genomics shall refund to the Purchaser all monies paid for such applicable product.

WARRANTY DISCLAIMERS

THE EXPRESS WARRANTIES AND THE REMEDIES SET FORTH ABOVE ARE IN LIEU OF, AND ARIMA GENOMICS AND ITS LICENSORS, SUPPLIERS AND REPRESENTATIVES HEREBY DISCLAIM, ALL OTHER REMEDIES AND WARRANTIES, EXPRESS, STATUTORY, IMPLIED, OR OTHERWISE, INCLUDING, BUT NOT LIMITED TO, ANY WARRANTIES OF MERCHANTABILITY, SATISFACTORY QUALITY, NONINFRINGEMENT OR FITNESS FOR A PARTICULAR PURPOSE, OR REGARDING RESULTS OBTAINED THROUGH THE USE OF ANY PRODUCT OR SERVICE (INCLUDING, WITHOUT LIMITATION, ANY CLAIM OF INACCURATE, INVALID OR INCOMPLETE RESULTS), IN EACH CASE HOWEVER ARISING, INCLUDING WITHOUT LIMITATION FROM A COURSE OF PERFORMANCE, DEALING OR USAGE OF TRADE, OR OTHERWISE. TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, ARIMA AND ITS LICENSORS, SUPPLIERS AND REPRESENTATIVES SHALL NOT BE LIABLE FOR LOSS OF USE, PROFITS, REVENUE, GOODWILL, BUSINESS OR OTHER FINANCIAL LOSS OR BUSINESS INTERRUPTION, OR COSTS OF SUBSTITUTE GOODS OR SERVICES, OR FOR ANY SPECIAL, CONSEQUENTIAL, INCIDENTAL, EXEMPLARY OR INDIRECT DAMAGES FOR BREACH OF WARRANTY.

CONTACT US

Technical Support: techsupport@arimagenomics.com

Order Support: ordersupport@arimagenomics.com

VISIT

www.arimagenomics.com