



Cold  
Spring  
Harbor  
Laboratory

# Advanced Sequencing Technologies & Applications

<http://meetings.cshl.edu/courses.html>



Cold  
Spring  
Harbor  
Laboratory

## Introduction to IGV The Integrative Genomics Viewer

Kelsy Cotto, Obi Griffith, Malachi Griffith,  
Alex Wagner, Jason Walker

Advanced Sequencing Technologies & Applications

November 6- 18, 2018



# Visualization Tools in Genomics

- there are **over 40 different genome browsers**, which to use?
- depends on
  - task at hand
  - kind and size of data
  - data privacy

# HT-seq Genome Browsers



Integrative  
Genome  
Viewer



UCSC  
Genome Browser  
Cancer Genome Browser



Trackster  
(part of Galaxy)

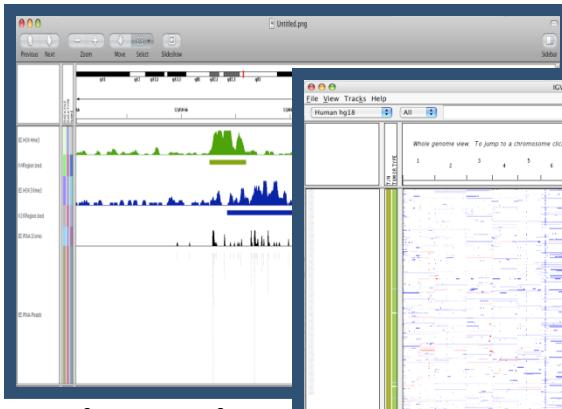


Savant  
Genome  
Browser

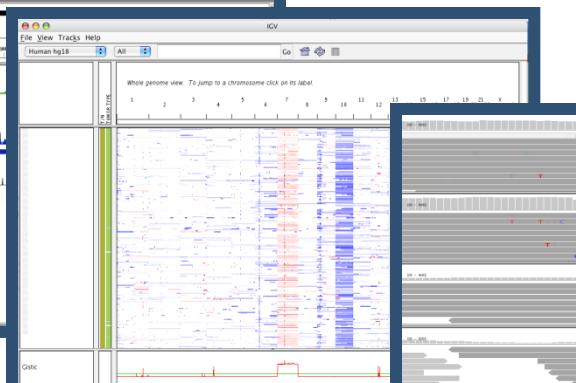
- task at hand : visualizing HT-seq reads, especially good for inspecting variants
- kind and size of data : large BAM files, stored locally or remotely
- data privacy : run on the desktop, can keep all data private
- UCSC Genome Browser has been retro-fitted to display BAM files
- Trackster is a genome browser that can perform visual analytics on small windows of the genome, deploy full analysis with Galaxy

# Integrative Genomics Viewer (IGV)

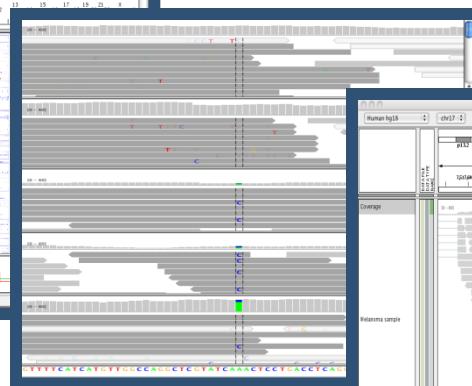
*Desktop application for the interactive visual exploration of integrated genomic datasets*



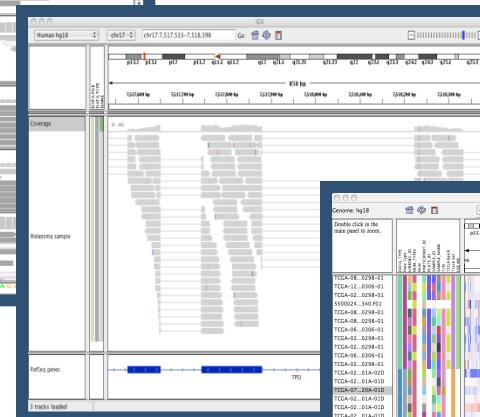
**Epigenomics**



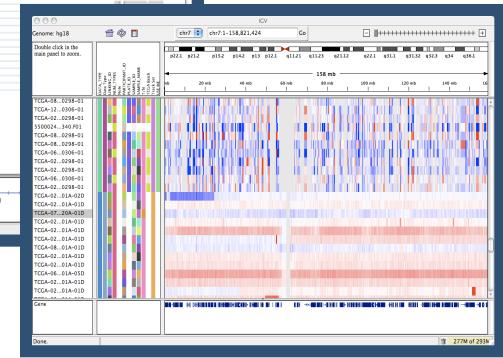
**Microarrays**



**NGS alignments**



**RNA-Seq**



**mRNA, CNV, Seq**

<http://www.broadinstitute.org/igv>

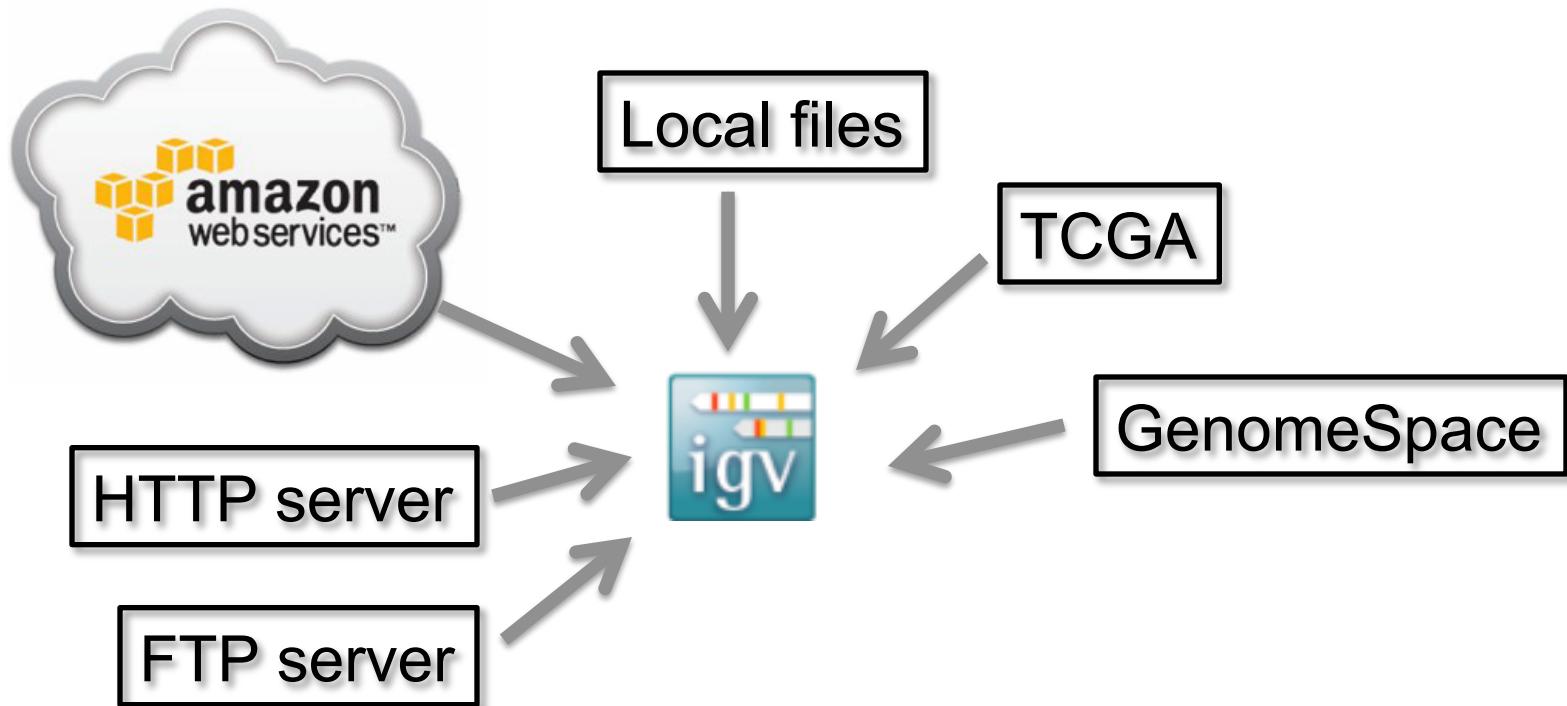
>85,000 registrations (2014)

# Features

With IGV you can...

- Explore large genomic datasets with an intuitive, easy-to-use interface.
- Integrate multiple data types with clinical and other sample information.
- View data from multiple sources:
  - local, remote, and “cloud-based”.
- Automation of specific tasks using command-line interface

# IGV data sources

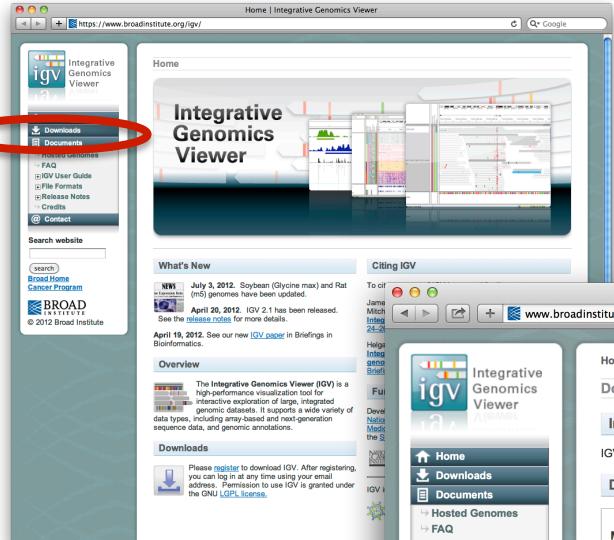
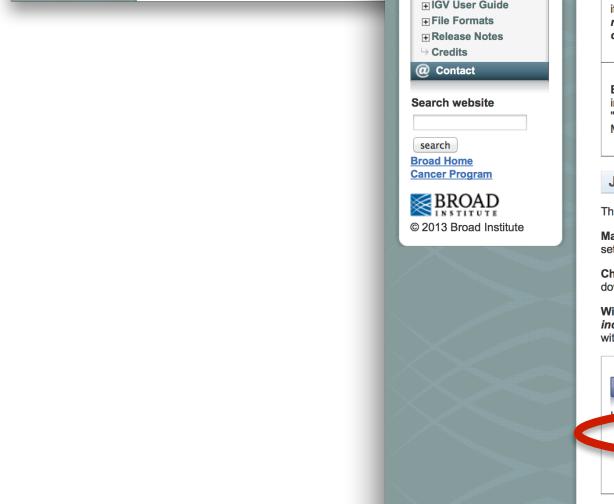
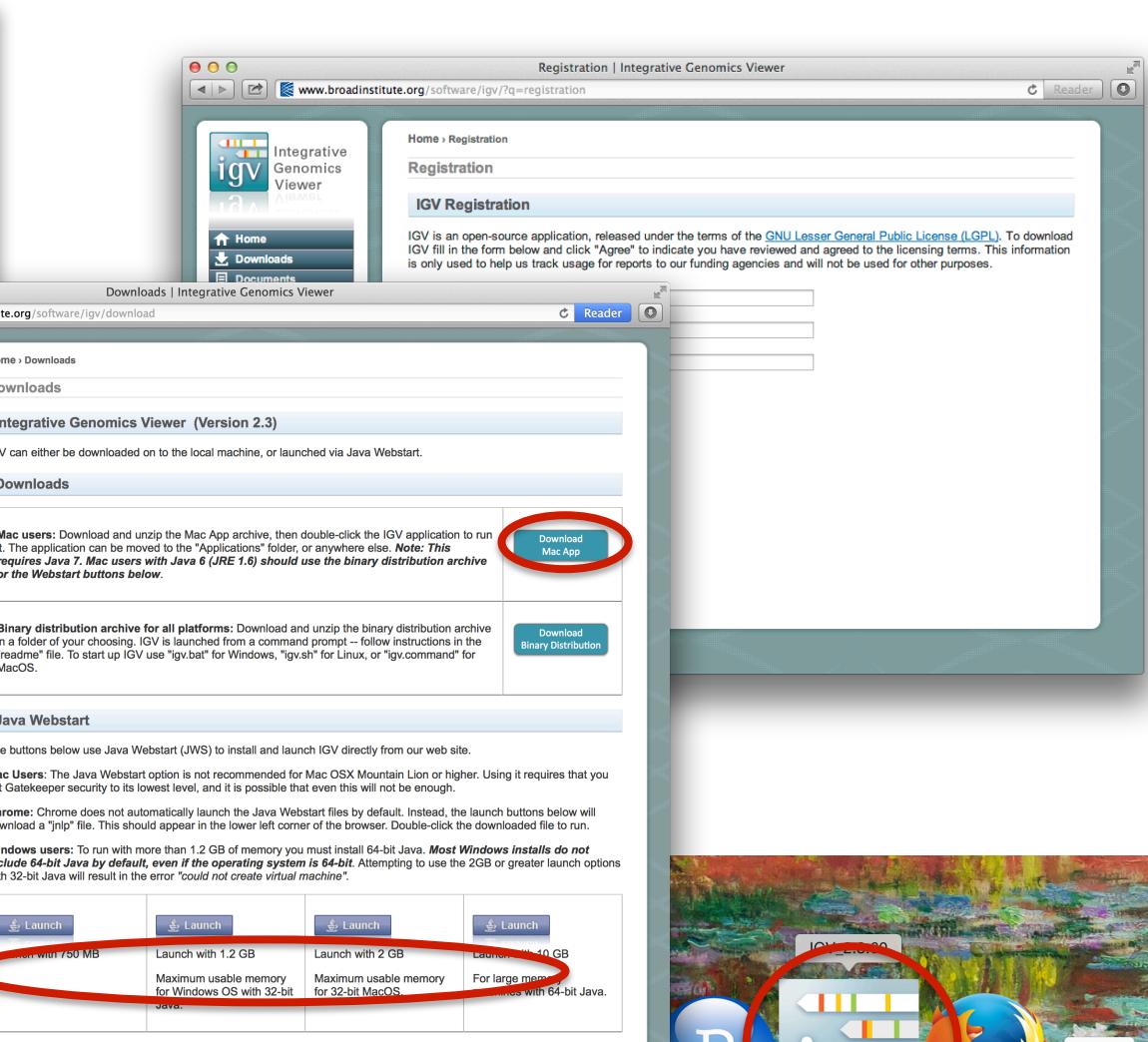


- View **local** files without uploading.
- View **remote** files without downloading the whole dataset.

# Using IGV: the basics

- Launch IGV
- Select a reference genome
- Load data
- Navigate through the data
  - WGS data
    - SNVs
    - structural variations

# Launch IGV

**IGV Registration**

IGV is an open-source application, released under the terms of the [GNU Lesser General Public License \(LGPL\)](#). To download IGV fill in the form below and click "Agree" to indicate you have reviewed and agreed to the licensing terms. This information is only used to help us track usage for reports to our funding agencies and will not be used for other purposes.

**IGV Registration**

Please register to download IGV. After registering, you can log in at any time using your email address and password to use IGV as granted under the [GNU GPL license](#).

**Downloads**

**Integrative Genomics Viewer (Version 2.3)**

IGV can either be downloaded onto the local machine, or launched via Java Webstart.

**Downloads**

**Mac users:** Download and unzip the Mac App archive, then double-click the IGV application to run it. The application can be moved to the "Applications" folder, or anywhere else. **Note: This requires Java 7.** **Mac users with Java 6 (JRE 1.6) should use the binary distribution archive or the Webstart buttons below.**

**Binary distribution archive for all platforms:** Download and unzip the binary distribution archive in a folder of your choosing. IGV is launched from a command prompt -- follow instructions in the "readme" file. To start up IGV use "igv.bat" for Windows, "igv.sh" for Linux, or "igv.command" for MacOS.

**Java Webstart**

The buttons below use Java Webstart (JWS) to install and launch IGV directly from our web site.

**Mac Users:** The Java Webstart option is not recommended for Mac OSX Mountain Lion or higher. Using it requires that you set Gatekeeper security to its lowest level, and it is possible that even this will not be enough.

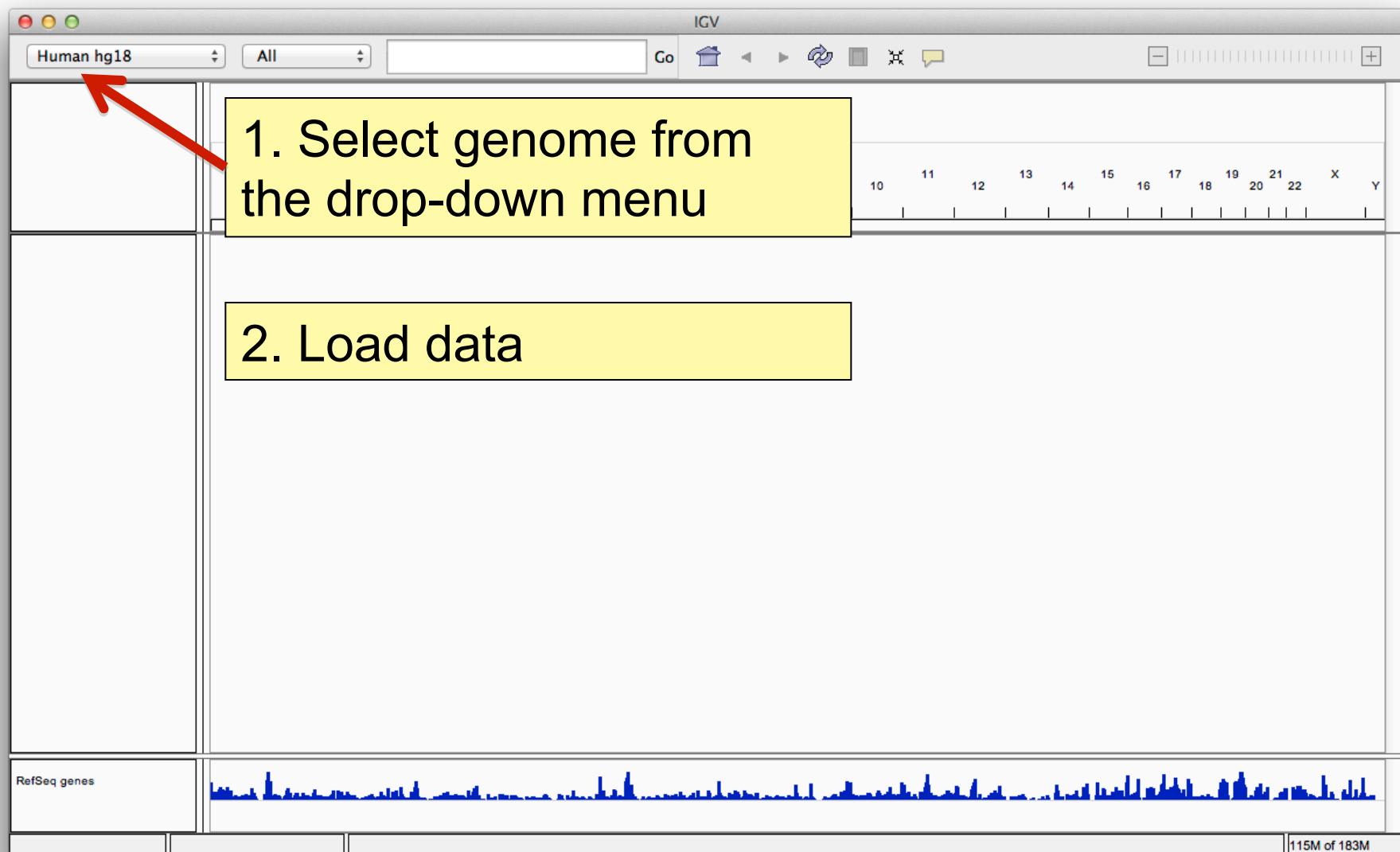
**Chrome:** Chrome does not automatically launch the Java Webstart files by default. Instead, the launch buttons below will download a "jnlp" file. This should appear in the lower left corner of the browser. Double-click the downloaded file to run.

**Windows users:** To run with more than 1.2 GB of memory you must install 64-bit Java. **Most Windows installs do not include 64-bit Java by default, even if the operating system is 64-bit.** Attempting to use the 2GB or greater launch options with 32-bit Java will result in the error "could not create virtual machine".

<a href="#">Launch</a>	<a href="#">Launch</a>	<a href="#">Launch</a>	<a href="#">Launch</a>
Launch with 750 MB	Launch with 1.2 GB	Launch with 2 GB	Launch with 10 GB
Maximum usable memory for Windows OS with 32-bit Java.	Maximum usable memory for 32-bit MacOS.	For large memory needs with 64-bit Java.	

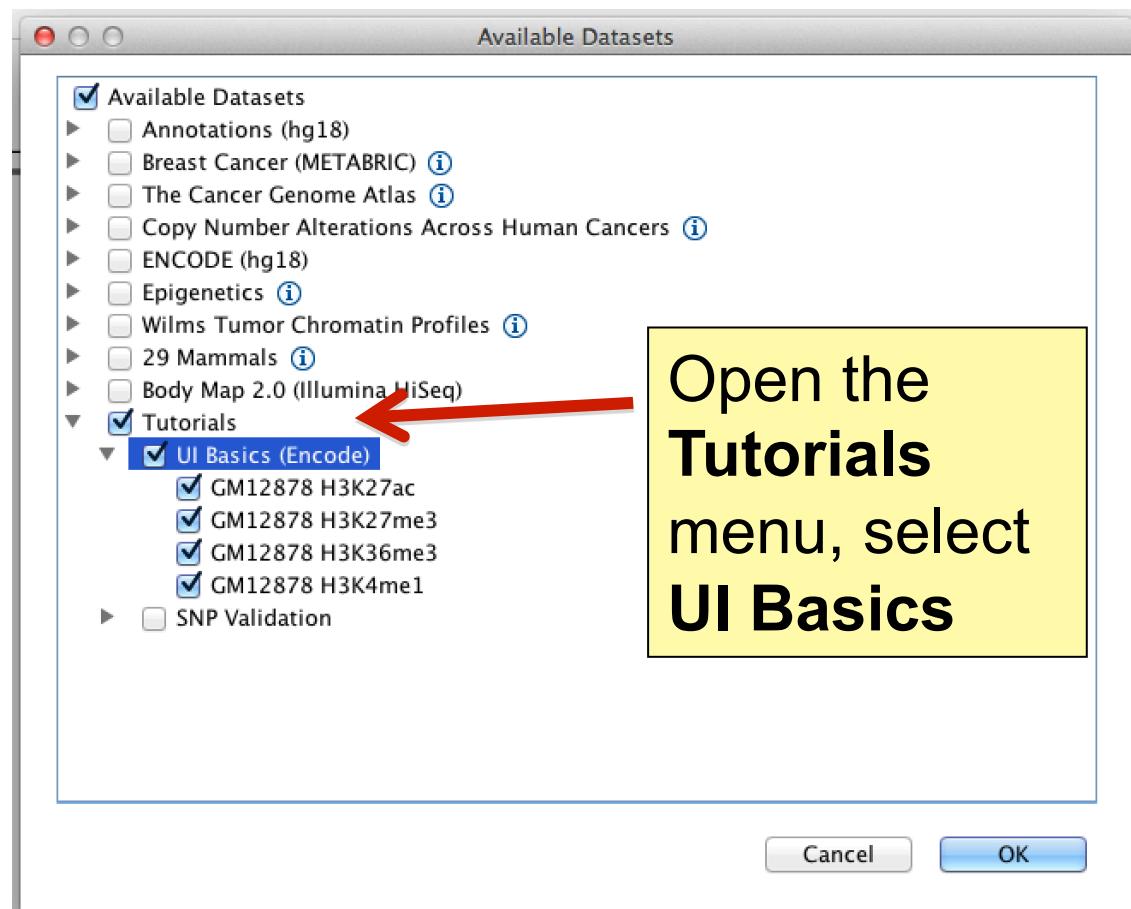
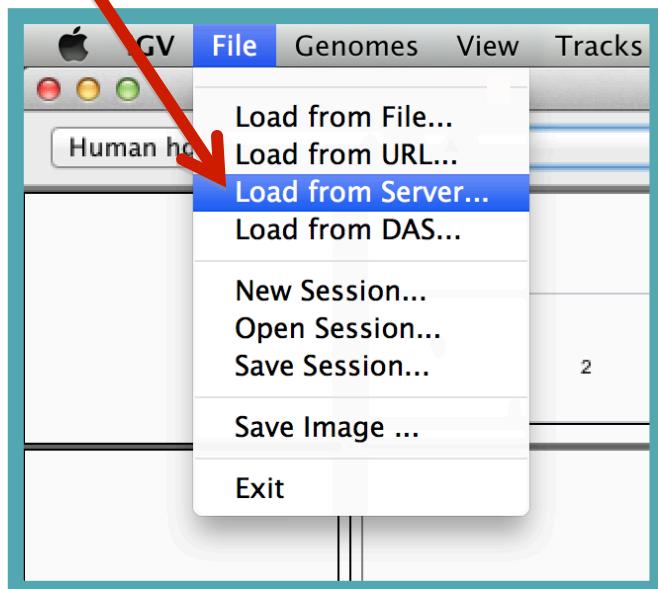


# Launch IGV



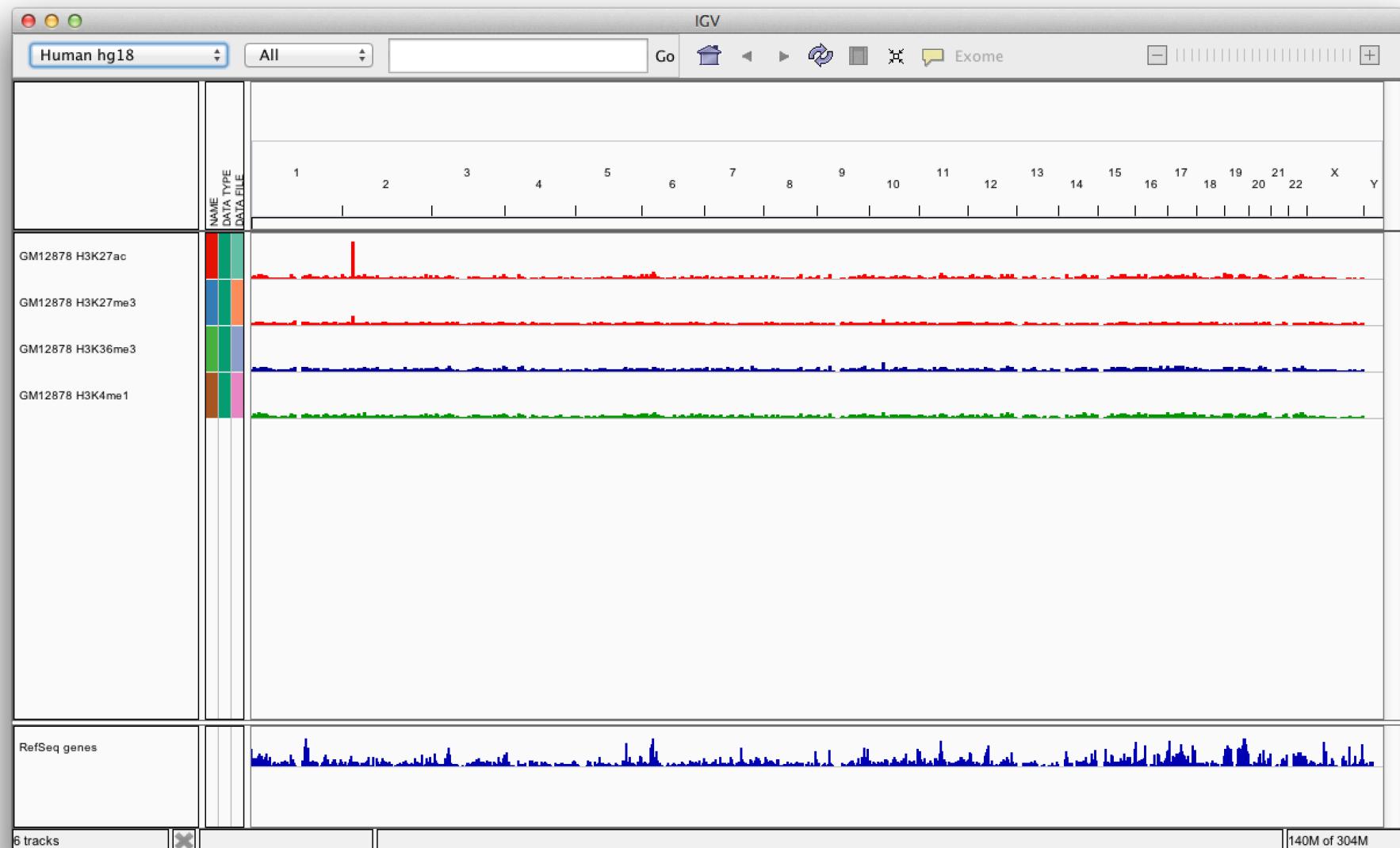
# Load data

Select File > Load from Server...

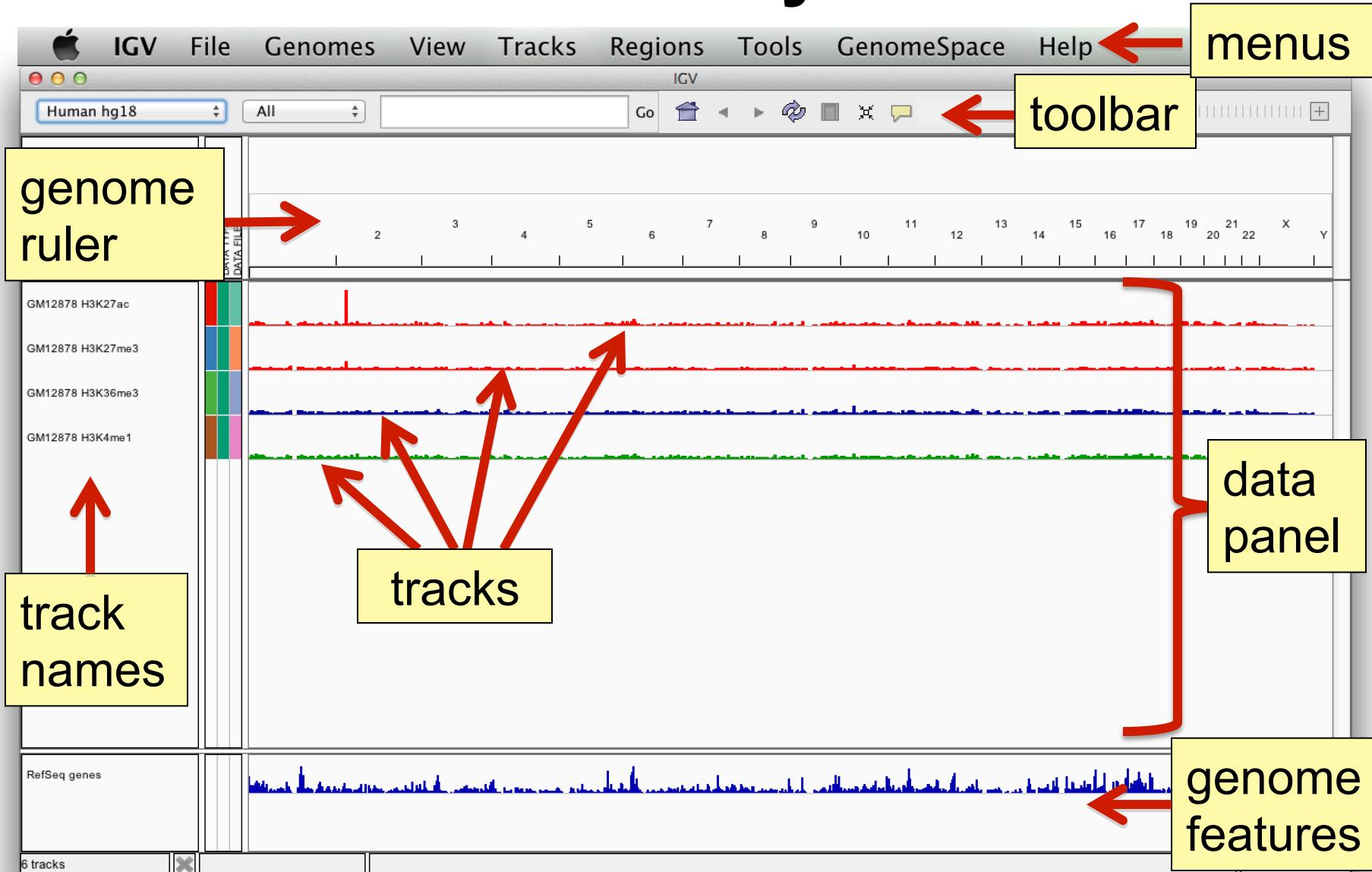


Open the Tutorials menu, select UI Basics

# Screen layout



# Screen layout



# File formats and track types

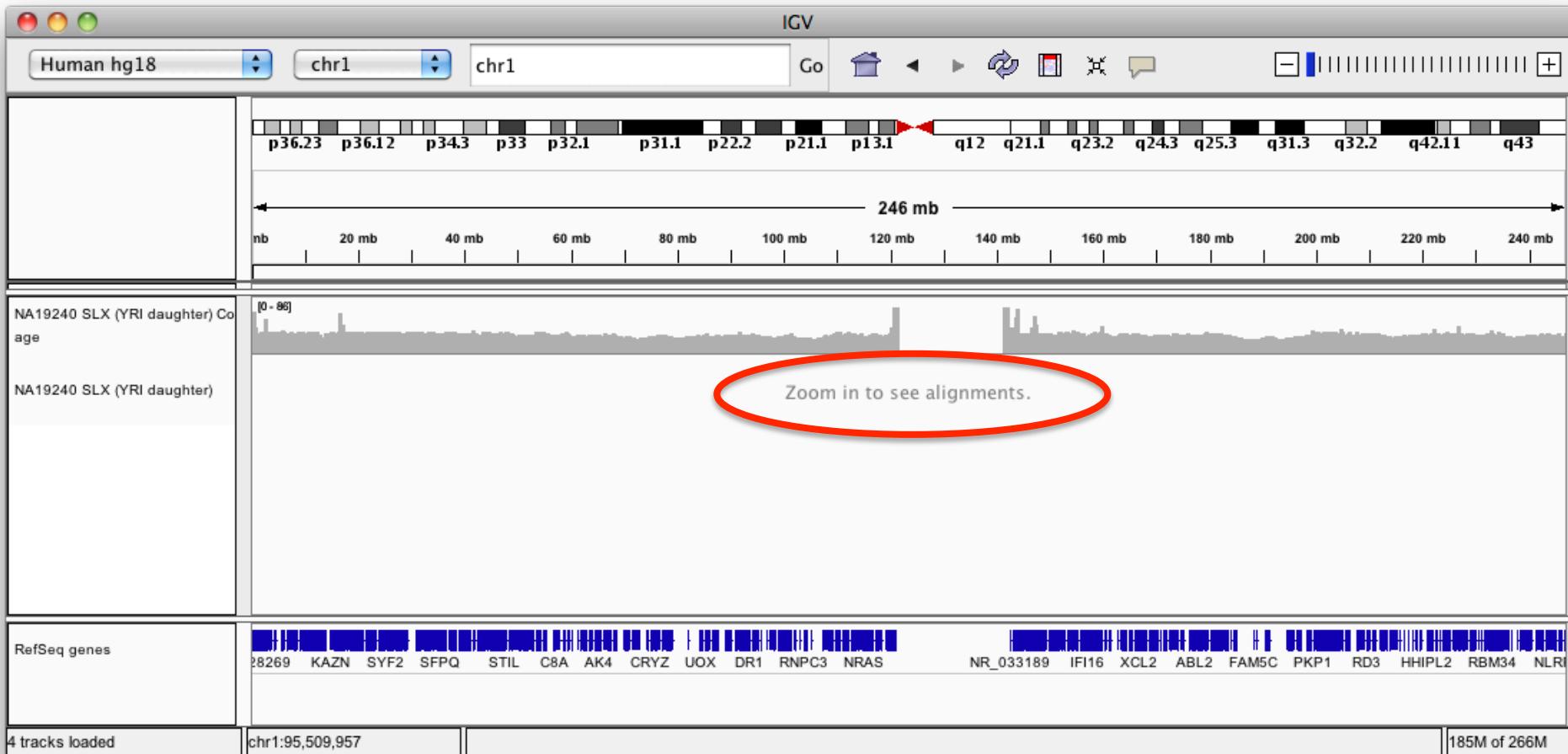
- The **file format** defines the track type.
- The **track type** determines the display options

- [BAM](#)
- [BED](#)
- [BedGraph](#)
- [bigBed](#)
- [bigWig](#)
- [Birdsuite Files](#)
- [CBS](#)
- [CN](#)
- [Cufflinks Files](#)
- [Custom File Formats](#)
- [Cytoband](#)
- [FASTA](#)
- [GCT](#)
- [genePred](#)
- [GFF](#)
- [GISTIC](#)
- [Goby](#)
- [GWAS](#)
- [IGV](#)
- [LOH](#)
- [MAF](#)
- [Merged BAM File \(.bam.list\)](#)
- [MUT](#)
- [PSL](#)
- [RES](#)
- [SAM](#)
- [Sample Information](#)
- [SEG](#)
- [SNP](#)
- [TAB](#)
- [TDF](#)
- [Track Line](#)
- [Type Line](#)
- [VCF](#)
- [WIG](#)

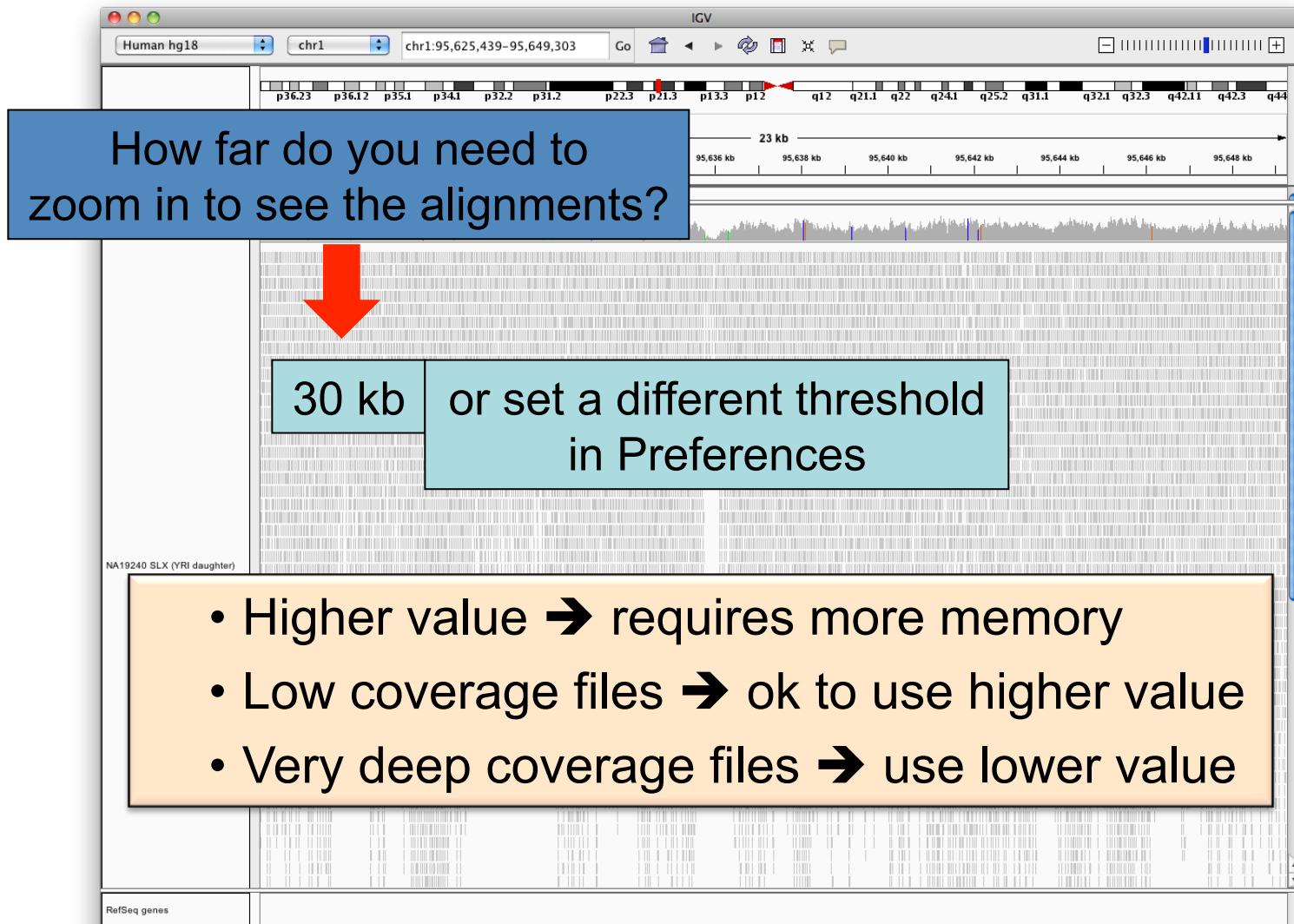
- For current list see: [www.broadinstitute.org/igv/FileFormats](http://www.broadinstitute.org/igv/FileFormats)

# Viewing alignments

## Whole chromosome view



# Viewing alignments – Zoom in



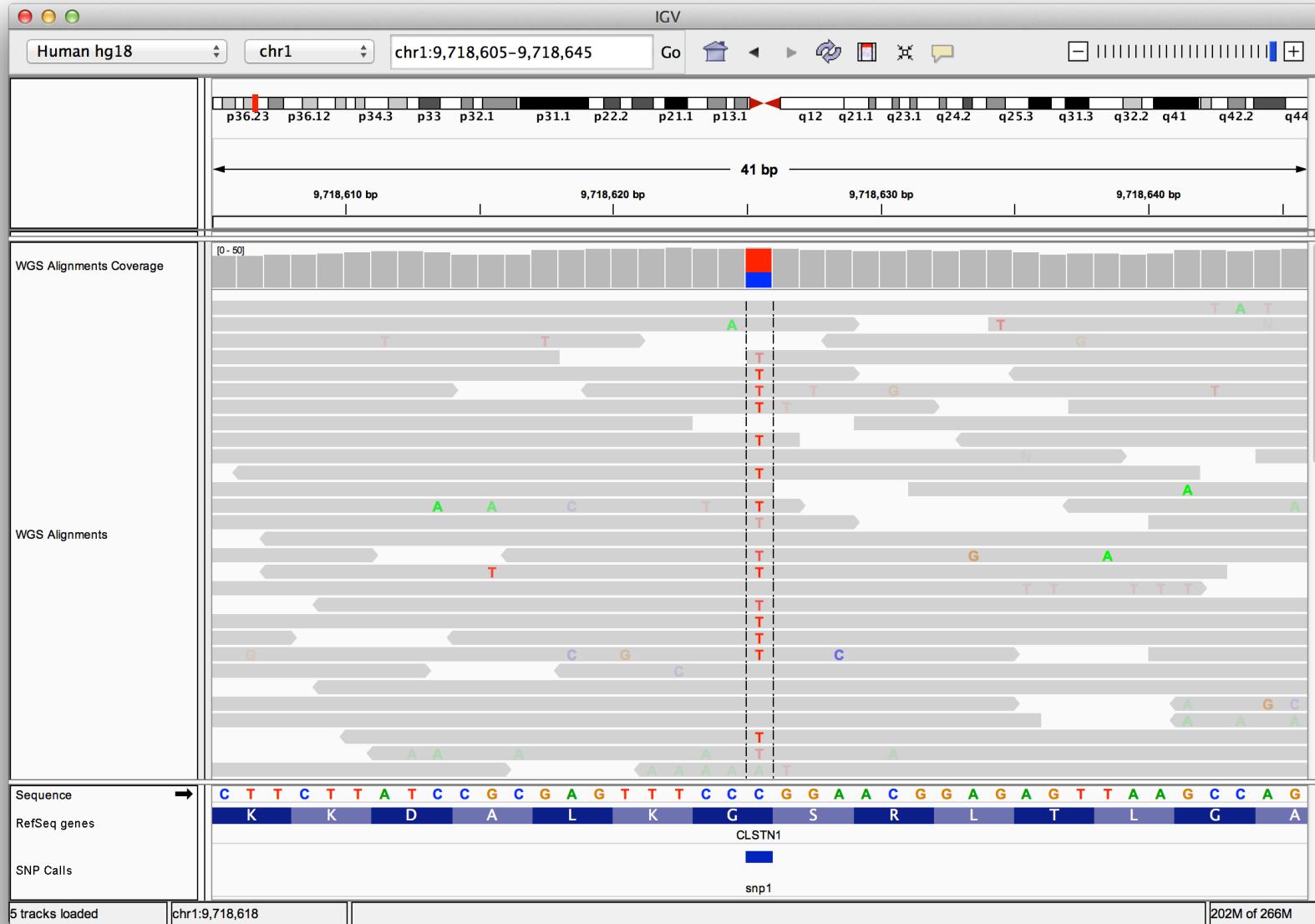
# Viewing alignments – Zoom in



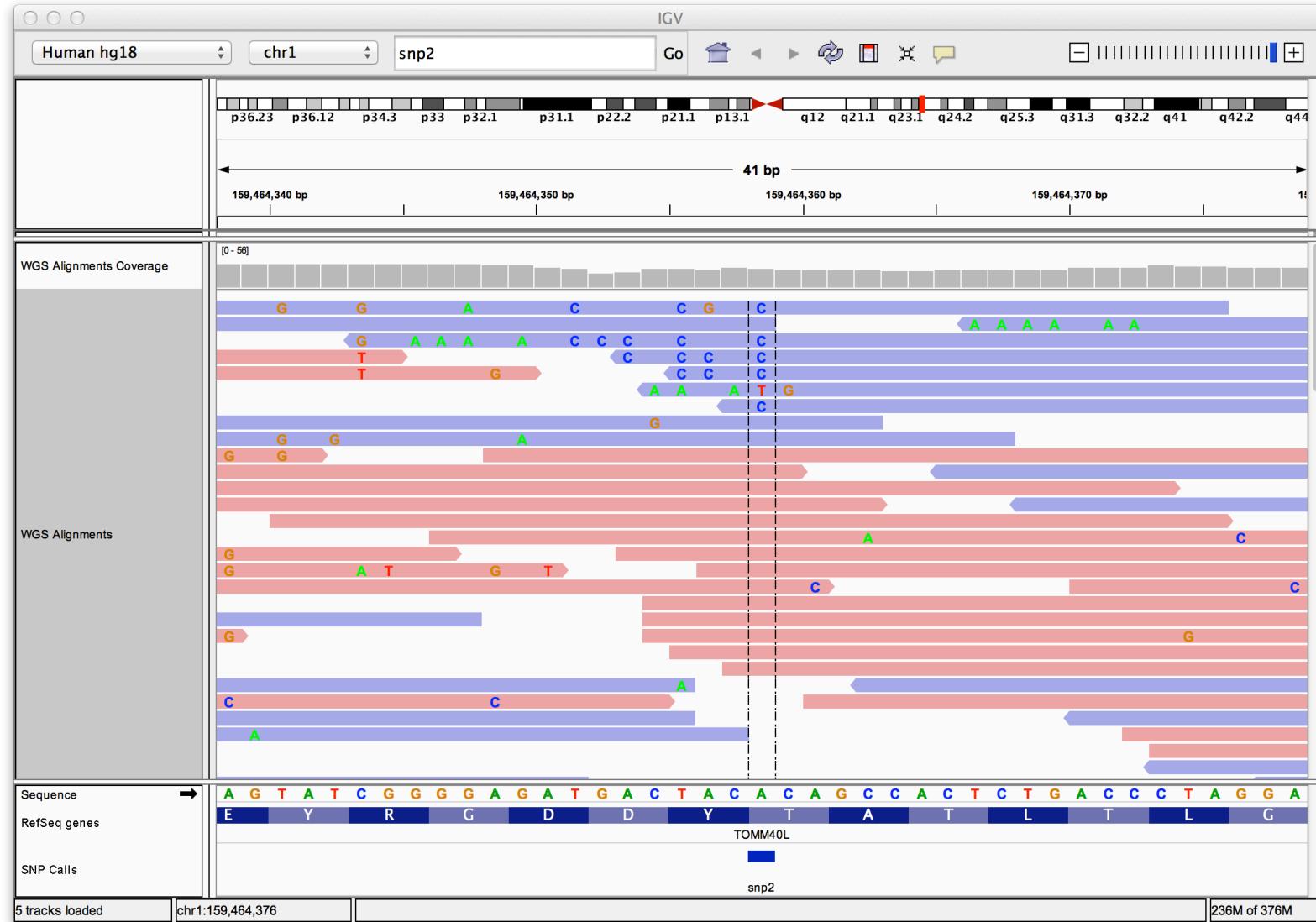
# SNVs and Structural variations

- Important metrics for evaluating the validity of SNVs:
  - Coverage
  - Amount of support
  - Strand bias / PCR artifacts
  - Mapping qualities
  - Base qualities
- Important metrics for evaluating SVs:
  - Coverage
  - Insert size
  - Read pair orientation

# Viewing SNPs and SNVs



# Viewing SNPs and SNVs



# Viewing Structural Events

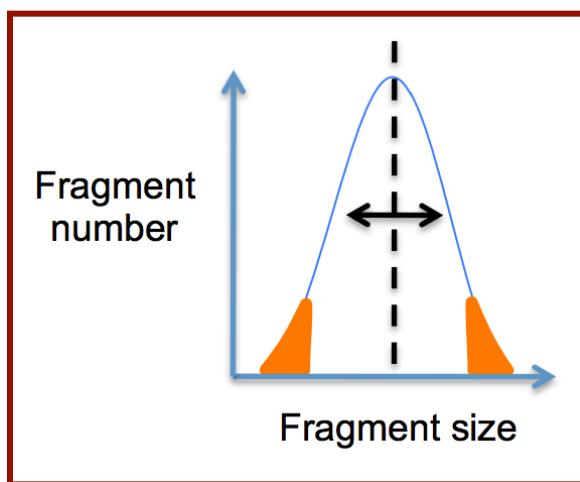
- Paired reads can yield evidence for genomic “structural events”, such as deletions, translocations, and inversions.
- Alignment coloring options help highlight these events based on:
  - Inferred insert size (template length)
  - Pair orientation (relative strand of pair)

# Paired-end sequencing

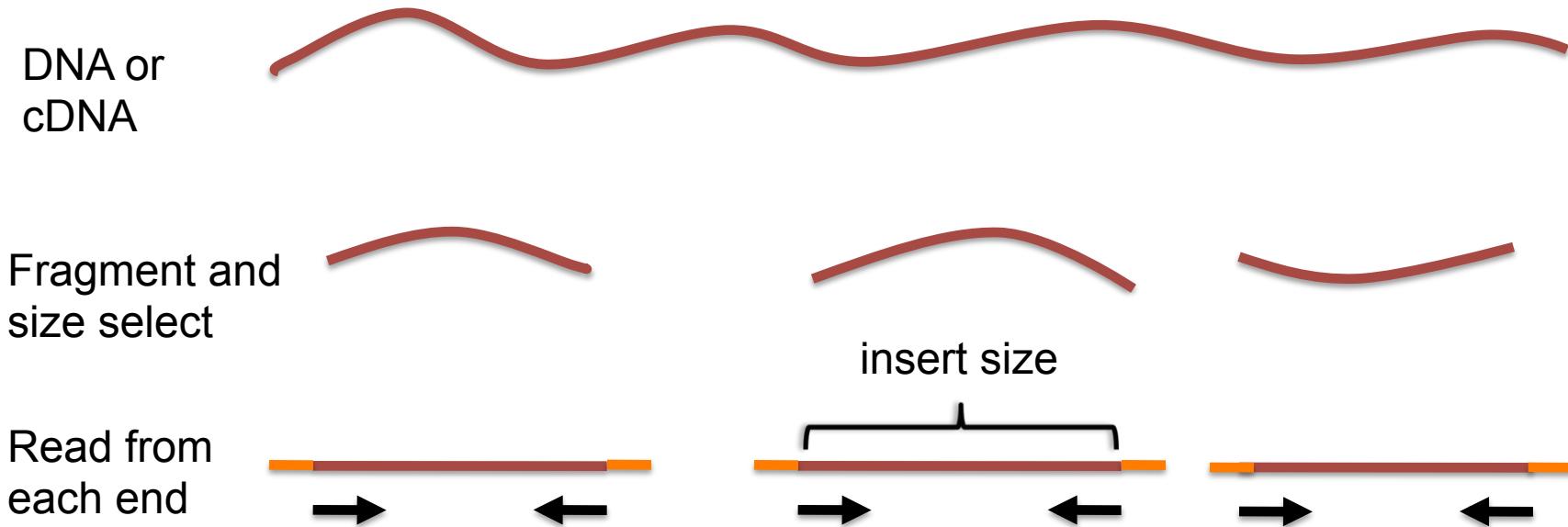
DNA or  
cDNA



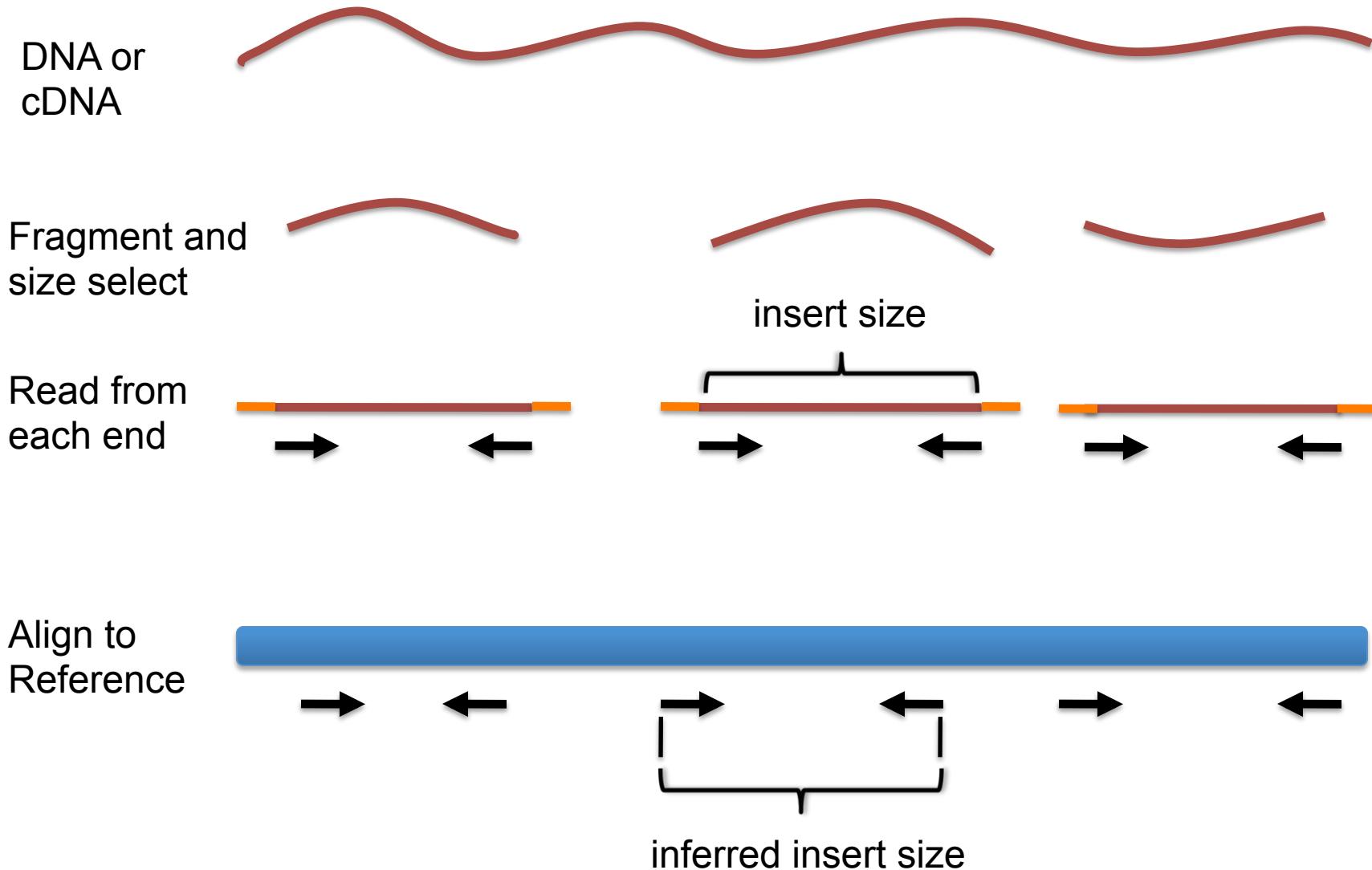
Fragment and  
size select



# Paired-end sequencing



# Paired-end sequencing



# Interpreting inferred insert size

The “inferred insert size” can be used to detect structural variants including

- Deletions
- Insertions
- Inter-chromosomal rearrangements: (Undefined insert size)

# Deletion

What is the effect of a deletion on inferred insert size?

# Deletion

Reference  
Genome



Subject



# Deletion

Reference  
Genome



Subject



# Deletion

Reference  
Genome



Subject



# Deletion

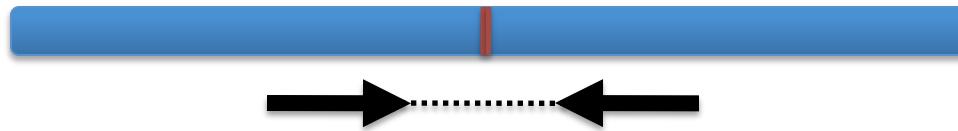
Inferred insert size is > expected value

Reference  
Genome



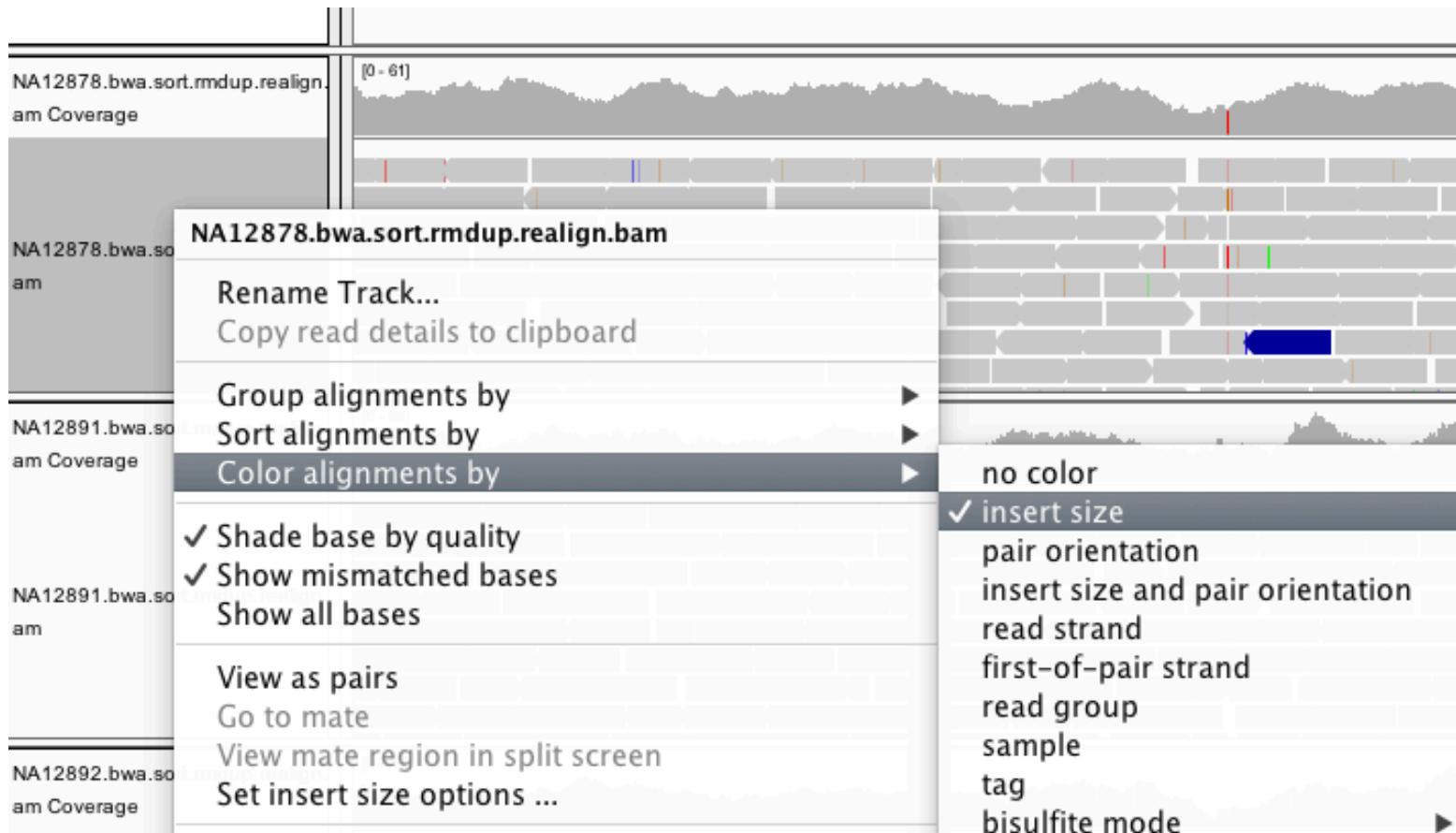
inferred insert size

Subject

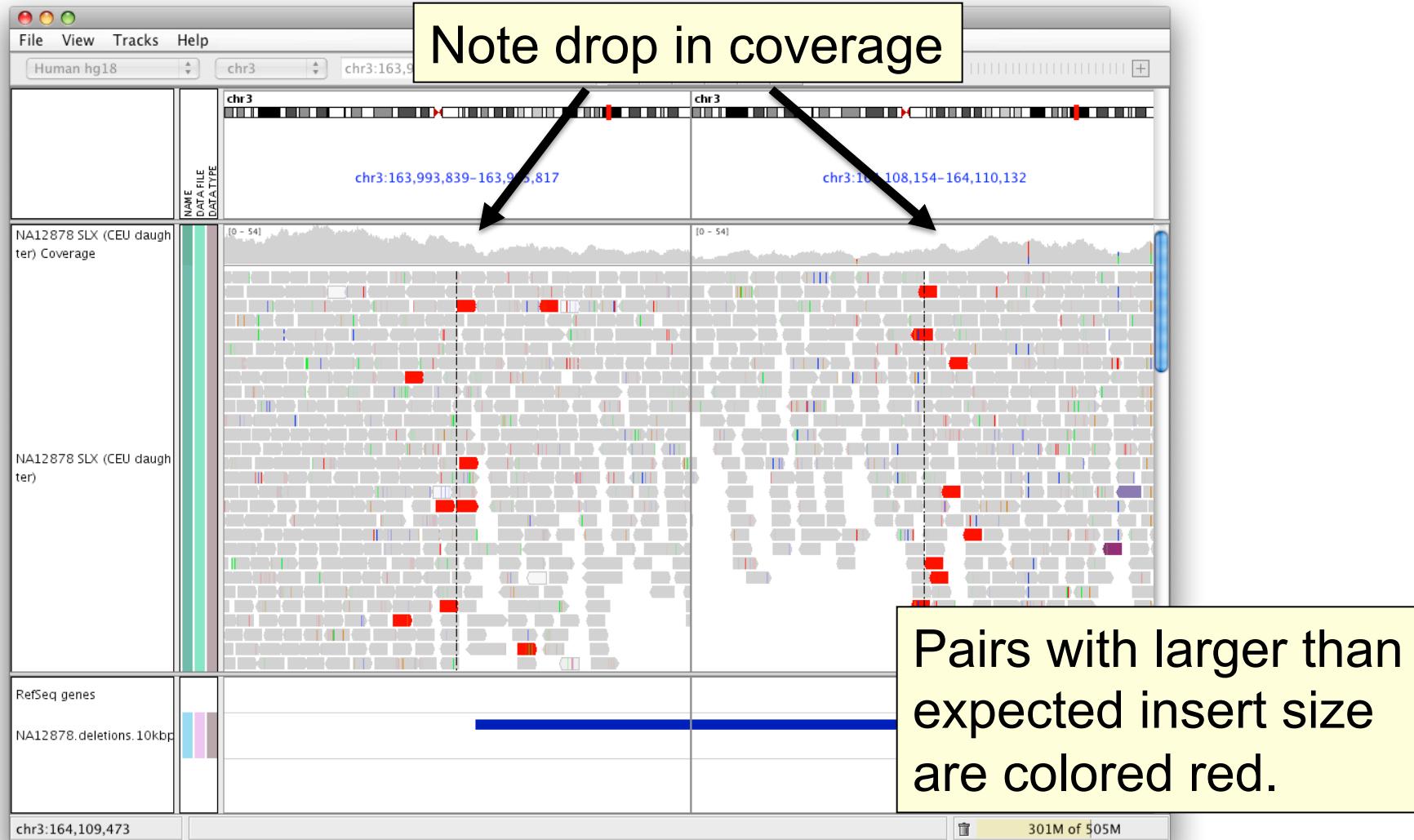


expected insert size

# Color by insert size



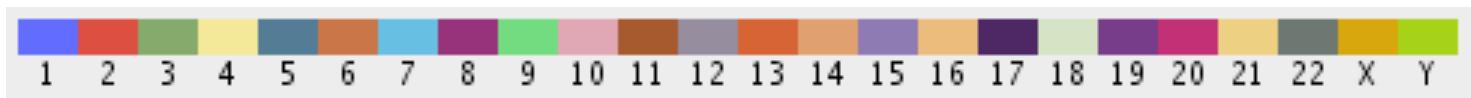
# Deletion



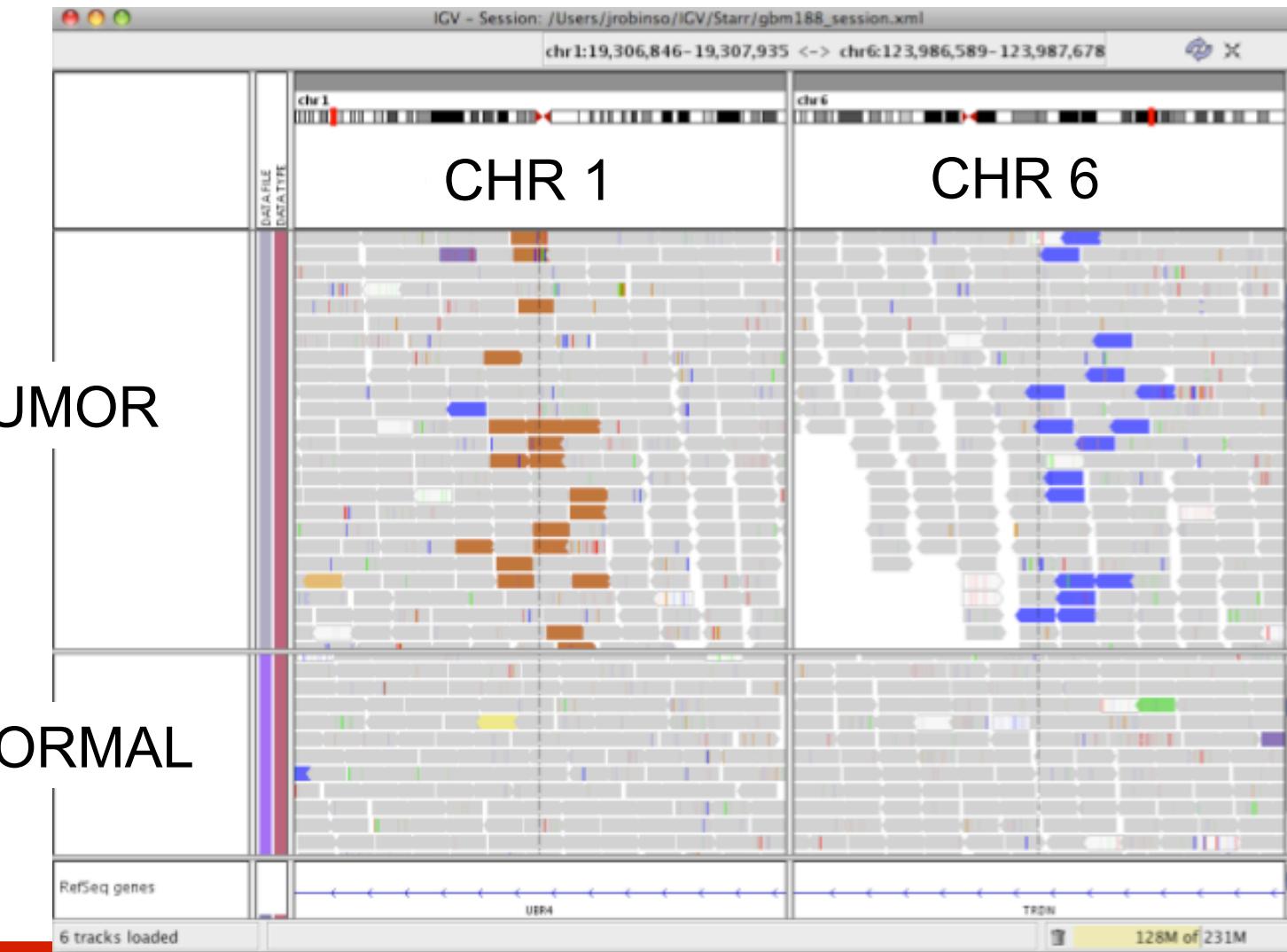
# Insert size color scheme

- Smaller than expected insert size: 
- Larger than expected insert size: 
- Pairs on different chromosomes

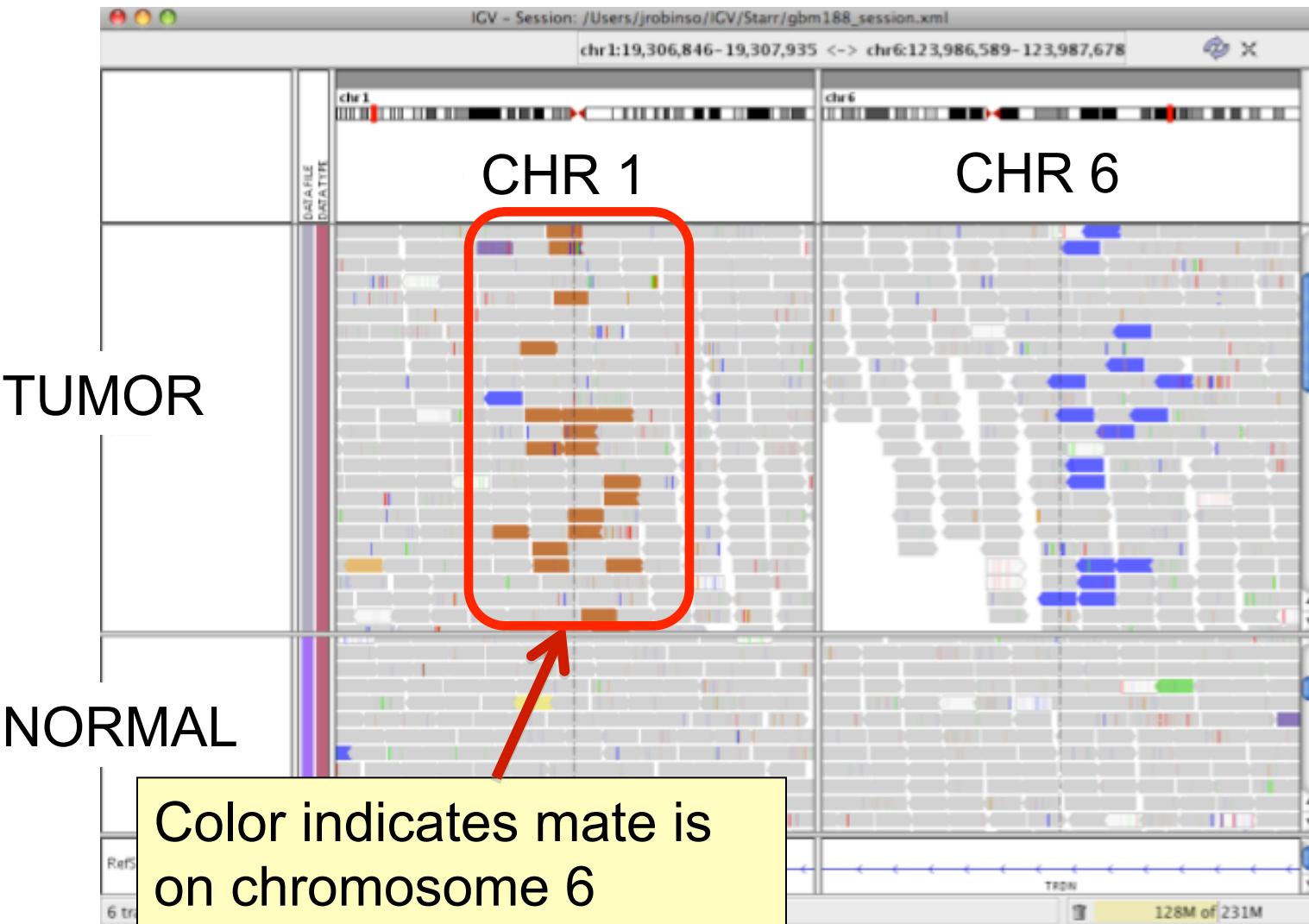
*Each end colored by chromosome of its mate*



# Rearrangement



# Rearrangement



# Interpreting Read-Pair Orientations

Orientation of paired reads can reveal structural events:

- Inversions
- Duplications
- Translocations
- Complex rearrangements

Orientation is defined in terms of

- read strand, left *vs* right, *and*
- read order, first *vs* second

# Inversion

Reference  
genome



# Inversion

Reference  
genome



# Inversion

Reference  
Genome



A

B

Subject



B

A

# Inversion

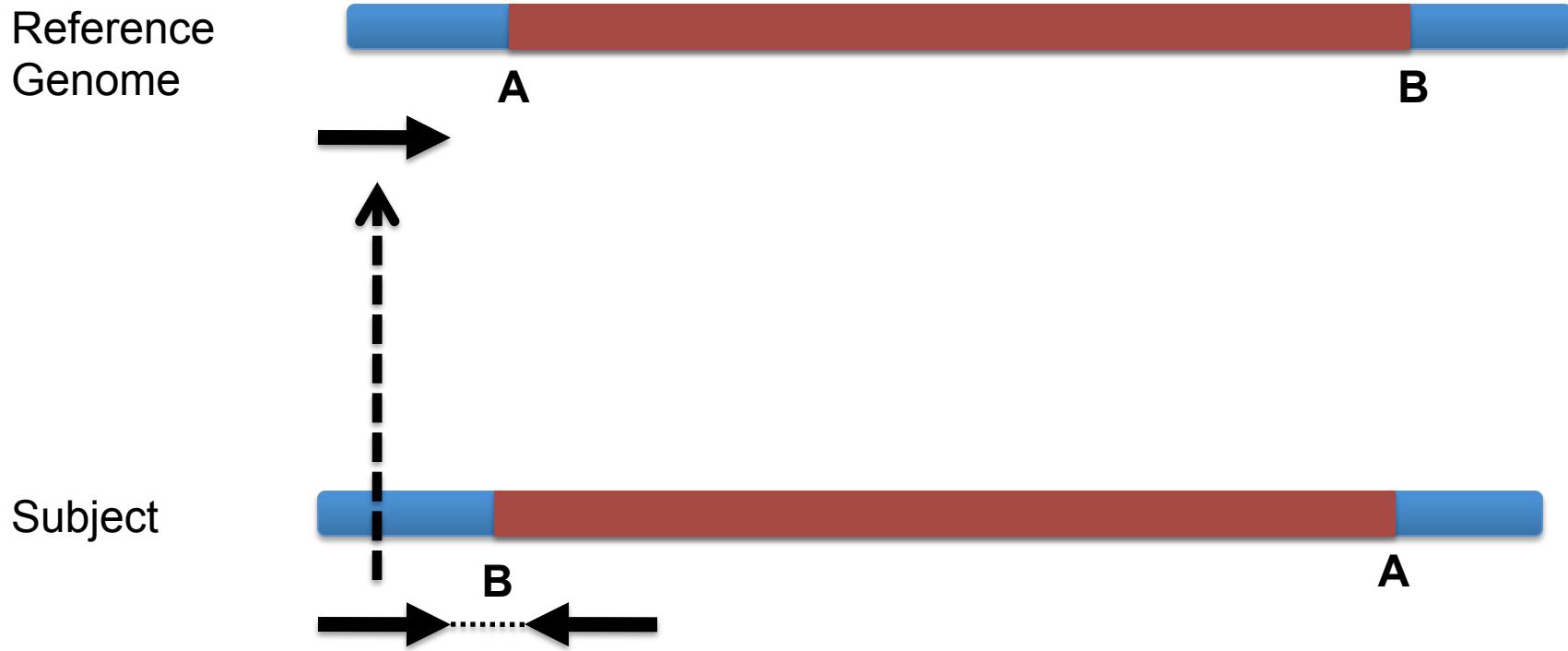
Reference  
Genome



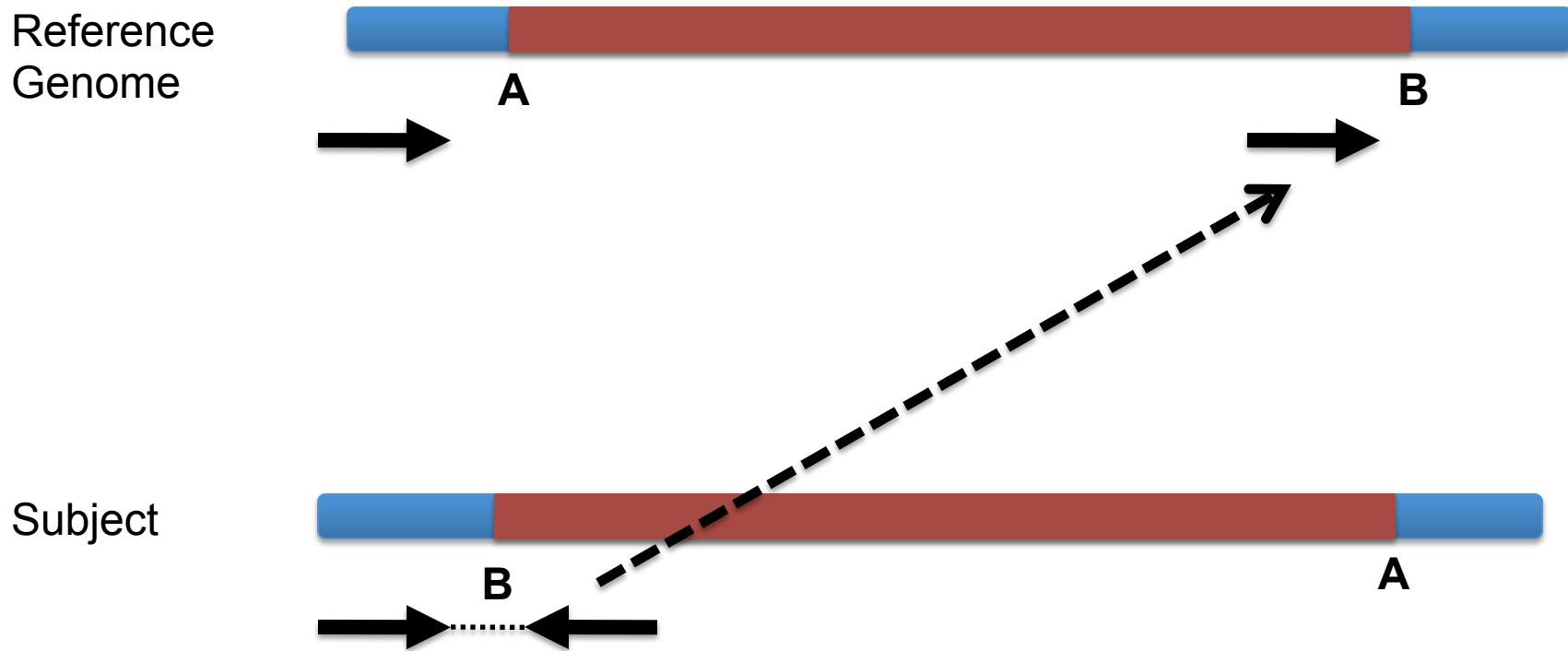
Subject



# Inversion

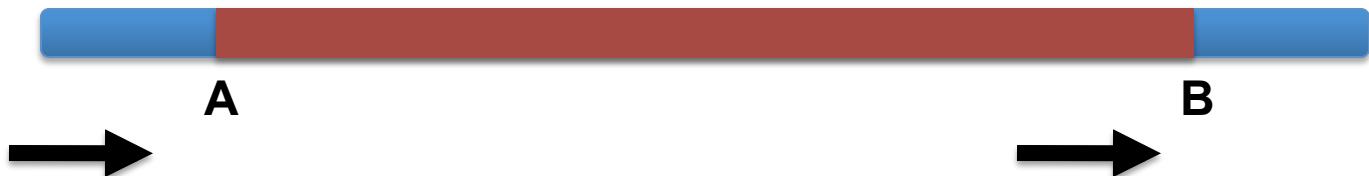


# Inversion

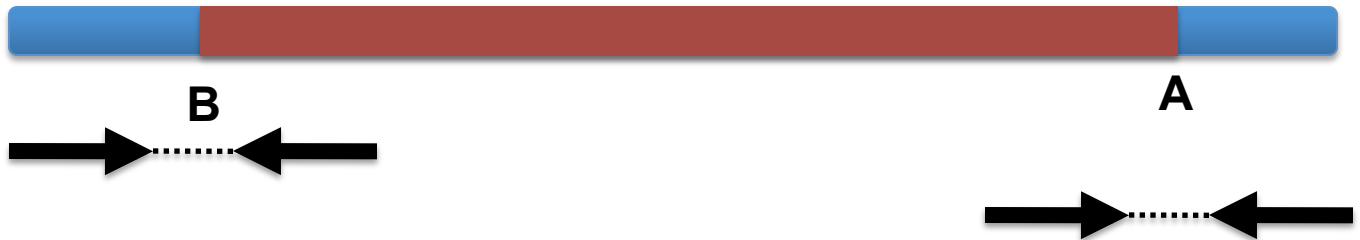


# Inversion

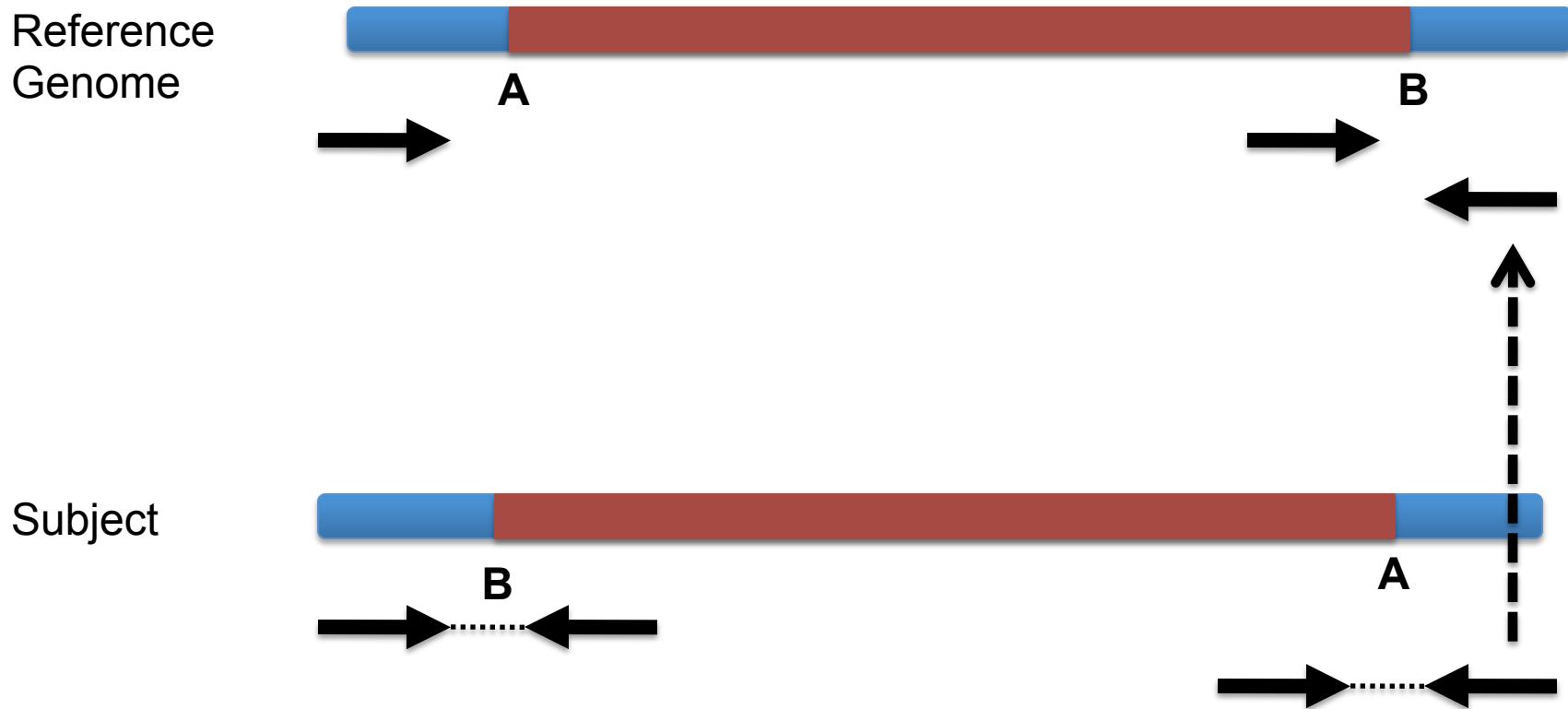
Reference  
Genome



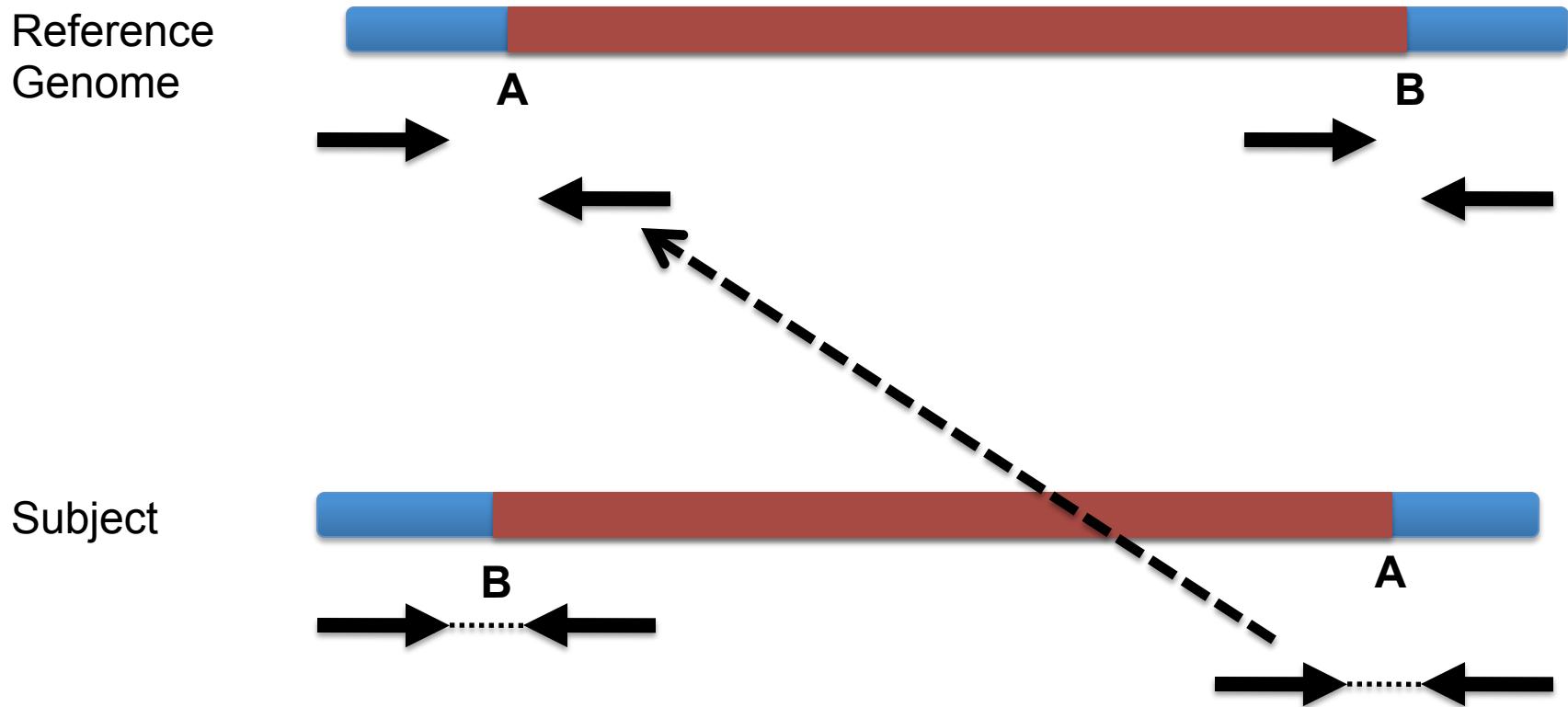
Subject



# Inversion

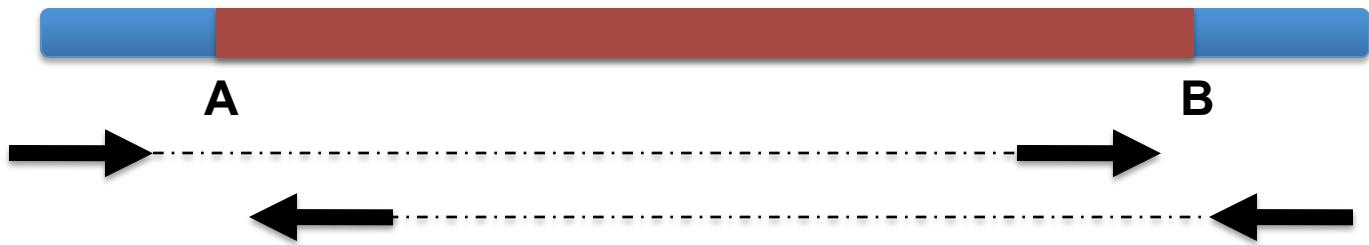


# Inversion



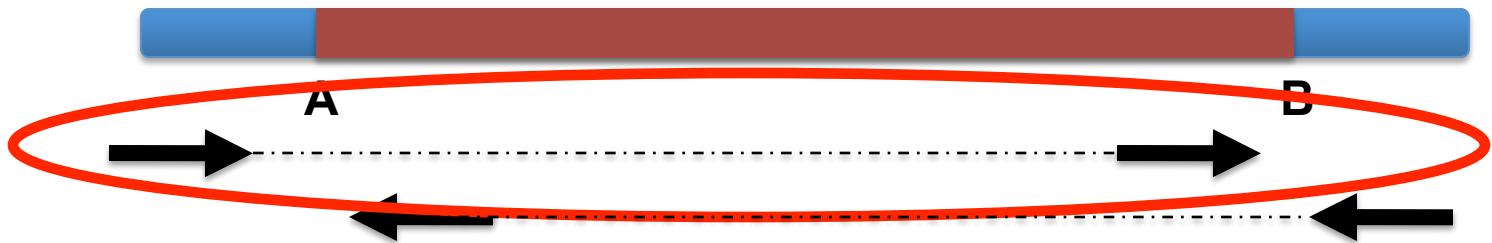
# Inversion

Reference  
Genome



# Inversion

Reference  
Genome



Anomaly: expected orientation of pair is  
inward facing ( → ← )

# Inversion

Reference  
Genome



“Left” side pair

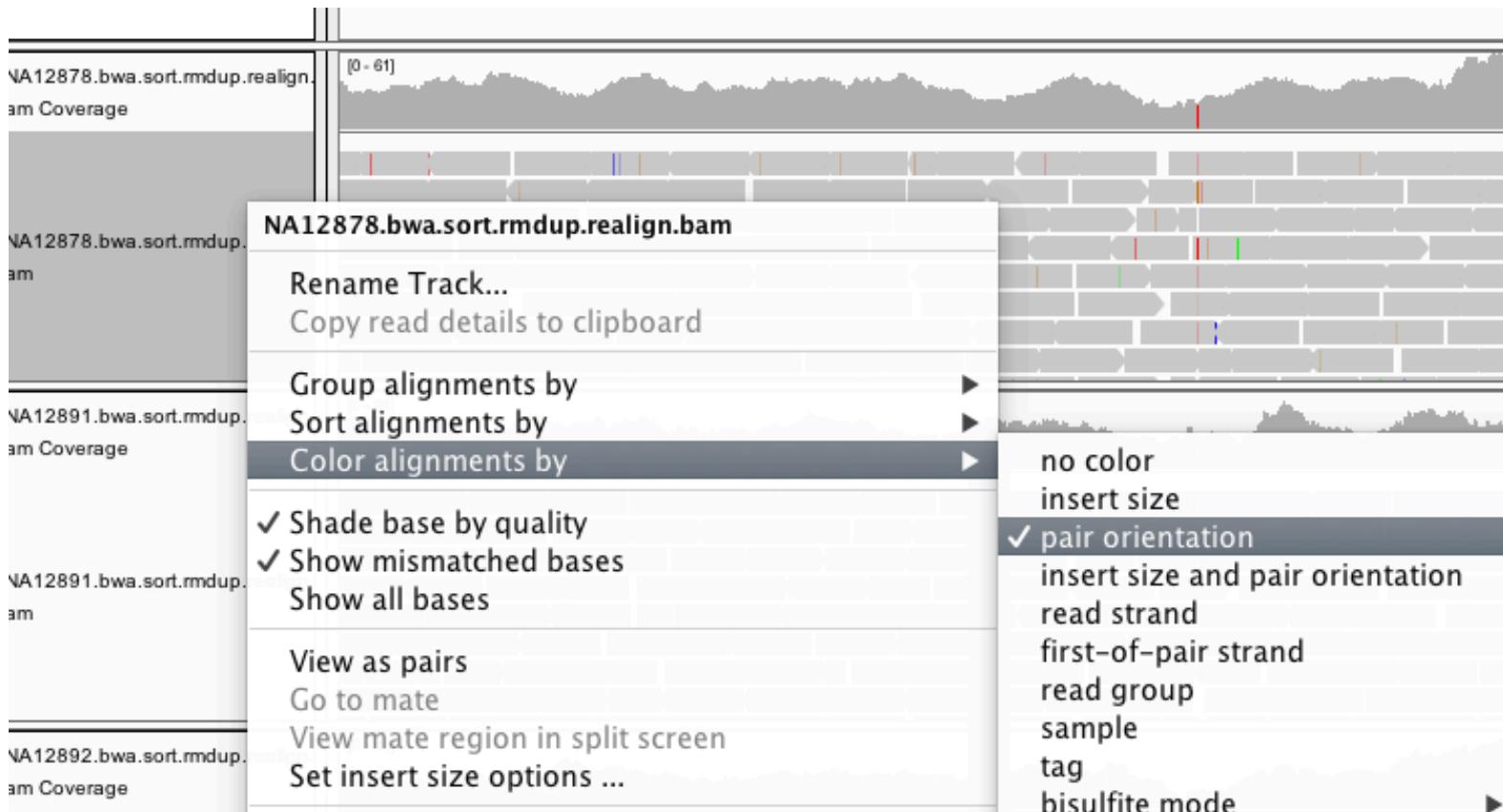
# Inversion

Reference  
Genome

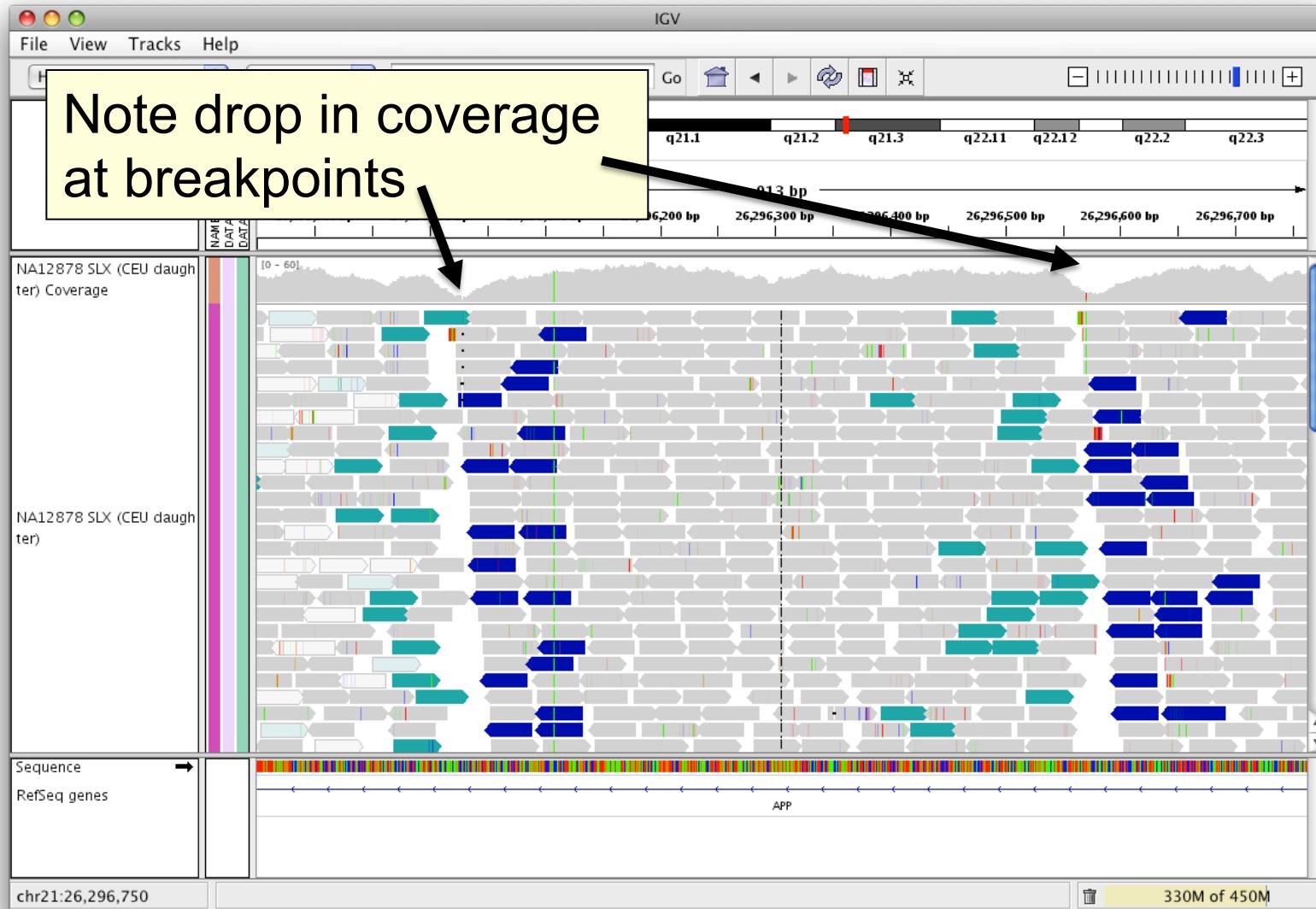


“Right” side pair

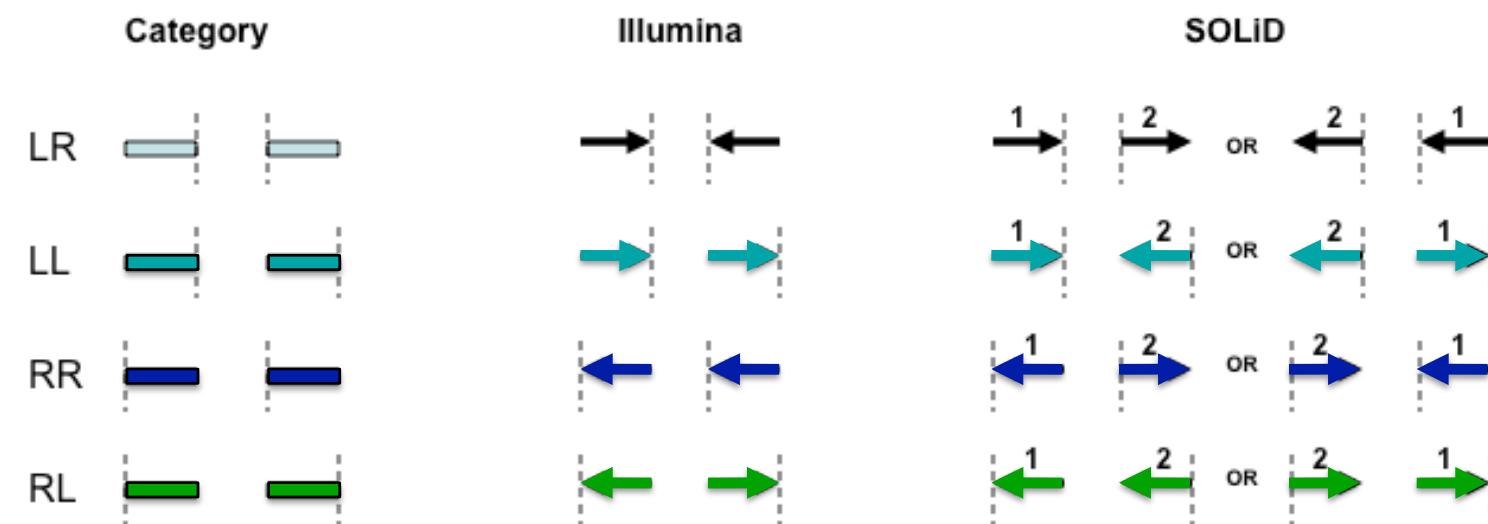
# Color by pair orientation



# Inversion



## Interpretation of read pair orientations



LR      Normal reads.  
The reads are left and right (respectively) of the unsequenced part of the sequenced DNA fragment when aligned back to the reference genome.

LL,RR    Implies inversion in sequenced DNA with respect to reference.

RL       Implies duplication or translocation with respect to reference.

These categories only apply to reads where both mates map to the same chromosome.

*Figure courtesy of Bob Handsaker*

# IGV hands-on tutorial

[https://github.com/griffithlab/  
rnaseq tutorial/wiki/IGV-Tutorial](https://github.com/griffithlab/rnaseq_tutorial/wiki/IGV-Tutorial)

# Manual Review Standard Operating Procedure (SOP) paper

© American College of Medical Genetics and Genomics

ARTICLE | Genetics  
inMedicine

Open

## Standard operating procedure for somatic variant refinement of sequencing data with paired tumor and normal samples

Erica K. Barnell, BS<sup>1</sup>, Peter Ronning, BS<sup>1</sup>, Katie M. Campbell, BS<sup>1</sup>, Kilannin Krysiak, PhD<sup>1,2</sup>, Benjamin J. Ainscough, PhD<sup>1,3</sup>, Lana M. Sheta<sup>1</sup>, Shahil P. Pema<sup>1</sup>, Alina D. Schmidt, BS<sup>1</sup>, Megan Richters, BS<sup>1</sup>, Kelsy C. Cotto, BS<sup>1</sup>, Arpad M. Danos, PhD<sup>1</sup>, Cody Ramirez, BS<sup>1</sup>, Zachary L. Skidmore, MEng<sup>1</sup>, Nicholas C. Spies, BS<sup>1</sup>, Jasreet Hundal, MS<sup>1</sup>, Malik S. Sediqzad<sup>1</sup>, Jason Kunisaki, BS<sup>1</sup>, Felicia Gomez, PhD<sup>1</sup>, Lee Trani, BS<sup>1</sup>, Matthew Matlock, BS<sup>1</sup>, Alex H. Wagner, PhD<sup>1</sup>, S. Joshua Swamidass, MD/PhD<sup>4,5</sup>, Malachi Griffith, PhD<sup>1,2,3,6</sup> and Obi L. Griffith, PhD<sup>1,2,3,6</sup>

**Purpose:** Following automated variant calling, manual review of aligned read sequences is required to identify a high-quality list of somatic variants. Despite widespread use in analyzing sequence data, methods to standardize manual review have not been described, resulting in high inter- and intralab variability.

**Methods:** This manual review standard operating procedure (SOP) consists of methods to annotate variants with four different calls and 19 tags. The calls indicate a reviewer's confidence in each variant and the tags indicate commonly observed sequencing patterns and artifacts that inform the manual review call. Four individuals were asked to classify variants prior to, and after, reading the SOP and accuracy was assessed by comparing reviewer calls with orthogonal validation sequencing.

**Results:** After reading the SOP, average accuracy in somatic variant identification increased by 16.7% ( $p$  value = 0.0298) and average interreviewer agreement increased by 12.7% ( $p$  value < 0.001). Manual review conducted after reading the SOP did not significantly increase reviewer time.

**Conclusion:** This SOP supports and enhances manual somatic variant detection by improving reviewer accuracy while reducing the interreviewer variability for variant calling and annotation.

*Genetics in Medicine* (2018) <https://doi.org/10.1038/s41436-018-0278-z>

**Keywords:** somatic variant refinement; manual review

# DeepSVR Paper

nature  
genetics

TECHNICAL REPORT

<https://doi.org/10.1038/s41588-018-0257-y>

## A deep learning approach to automate refinement of somatic variant calling from cancer sequencing data

Benjamin J. Ainscough<sup>ID 1,2,12</sup>, Erica K. Barnell<sup>ID 1,12</sup>, Peter Ronning<sup>1</sup>, Katie M. Campbell<sup>ID 1</sup>, Alex H. Wagner<sup>ID 1</sup>, Todd A. Fehniger<sup>ID 2,3</sup>, Gavin P. Dunn<sup>4</sup>, Ravindra Uppaluri<sup>5</sup>, Ramaswamy Govindan<sup>2,3</sup>, Thomas E. Rohan<sup>6</sup>, Malachi Griffith<sup>ID 1,2,3,7</sup>, Elaine R. Mardis<sup>8,9</sup>, S. Joshua Swamidass<sup>10,11\*</sup> and Obi L. Griffith<sup>ID 1,2,3,7\*</sup>

Cancer genomic analysis requires accurate identification of somatic variants in sequencing data. Manual review to refine somatic variant calls is required as a final step after automated processing. However, manual variant refinement is time-consuming, costly, poorly standardized, and non-reproducible. Here, we systematized and standardized somatic variant refinement using a machine learning approach. The final model incorporates 41,000 variants from 440 sequencing cases. This model accurately recapitulated manual refinement labels for three independent testing sets (13,579 variants) and accurately predicted somatic variants confirmed by orthogonal validation sequencing data (212,158 variants). The model improves on manual somatic refinement by reducing bias on calls otherwise subject to high inter-reviewer variability.

# Break