Alignments, Stats and Visualization Tutorial

For this tutorial we will work in the folder called **alignment_tutorial**

```
$ cd
$ mkdir alignment_tutorial
$ cp ~/Course_Data/{PG15-BW001347.sam,PG15-BW001432.bam,PG16-
BW002087.bam,Ref.C.IN.95.95IN21068.AF067155.fa} ~/alignment_tutorial
$ cd alignment_tutorial
$ ls -lh
```

Alignments

To work with an NGS alignment we will primarily use **Samtools**

```
$ ~/software/samtools/samtools --help
or
$ samtools --help
```

Converting SAM to BAM file

```
$ samtools view -b PG15-BW001347.sam > PG15-BW001347.bam
```

view : Viewing an alignment file

-b : Output in BAM format

Sorting a BAM file

```
$ samtools sort PG15-BW001347.bam -o PG15-BW001347sorted.bam
```

sort : Sorts a BAM file

-o : Output file

Indexing a BAM file

```
$ samtools index PG15-BW001347sorted.bam
$ ls -lh
```

This will create a file with .bam.bai extension which will be the index file for the corresponding BAM file

The SAM file can be removed (using the **rm** command) at this stage to save space as the BAM file contains all the information that the SAM file does.

Samtools can also be used to quickly query number of reads mapped/unmapped in an alignment. We can use the Flags for each read in an alignment. If the flag includes the number 4 then the read is unmapped

Number of Mapped Reads

```
$ samtools view -c -F4 PG15-BW001347sorted.bam
```

view : Viewing the BAM file

-c : Count

-F4 : skip reads with FLAG 4

How many reads map to the HIV ?

Number of Unmapped Reads

```
$ samtools view -c -f4 PG15-BW001347sorted.bam
```

-f4 : Only include reads with FLAG 4

The reason why we have 0 reads mapped to the reference is because the BAM file was exported without the unmapped reads. You can try the command with any BAM file generated in the previous session

Stats

We are going to use weeSAM to generate Statistics for a SAM or a BAM file.

Note: The index file need to be in the same directory

```
$ ~/software/weeSAM/legacy_versions/weeSAMv1.4 --help
or
$ weeSAMv1.4 --help
```

weeSAM outputs a tab delimited file with summary stats and a plot of the coverage. The summary stats are

ReferenceID : Reference

RefLength : The length of the reference

MappedReads : Number of reads mapped to the reference

Breadth : Number of sites covered by reads

PercentageCovered : The percent of sites with coverage

MinDepth : Minimum read depth observed

MaxDepth : Max read depth observed

AverageDepth : Mean read depth observed

StandardDeviation : Standard deviation of depth

```
$ weeSAMv1.4 -b PG15-BW001347sorted.bam --out 1347.txt --plot 1347.jpg
$ nano 1347.txt
$ eog 1347.jpg
```

-b : Input in bam format

--out : Tab delimited text file of output

--plot : Coverage plot output. Can be a pdf

What is the average coverage and percentage coverage?

Samtools also provides basic coverage stats which can add to weeSAM

```
$ samtools flagstat PG15-BW001347sorted.bam
```

How many reads map in pairs?

Now repeat these steps for the other two BAM files in the folder

- **PG15-BW001432.bam**
- **PG16-BW002087.bam**

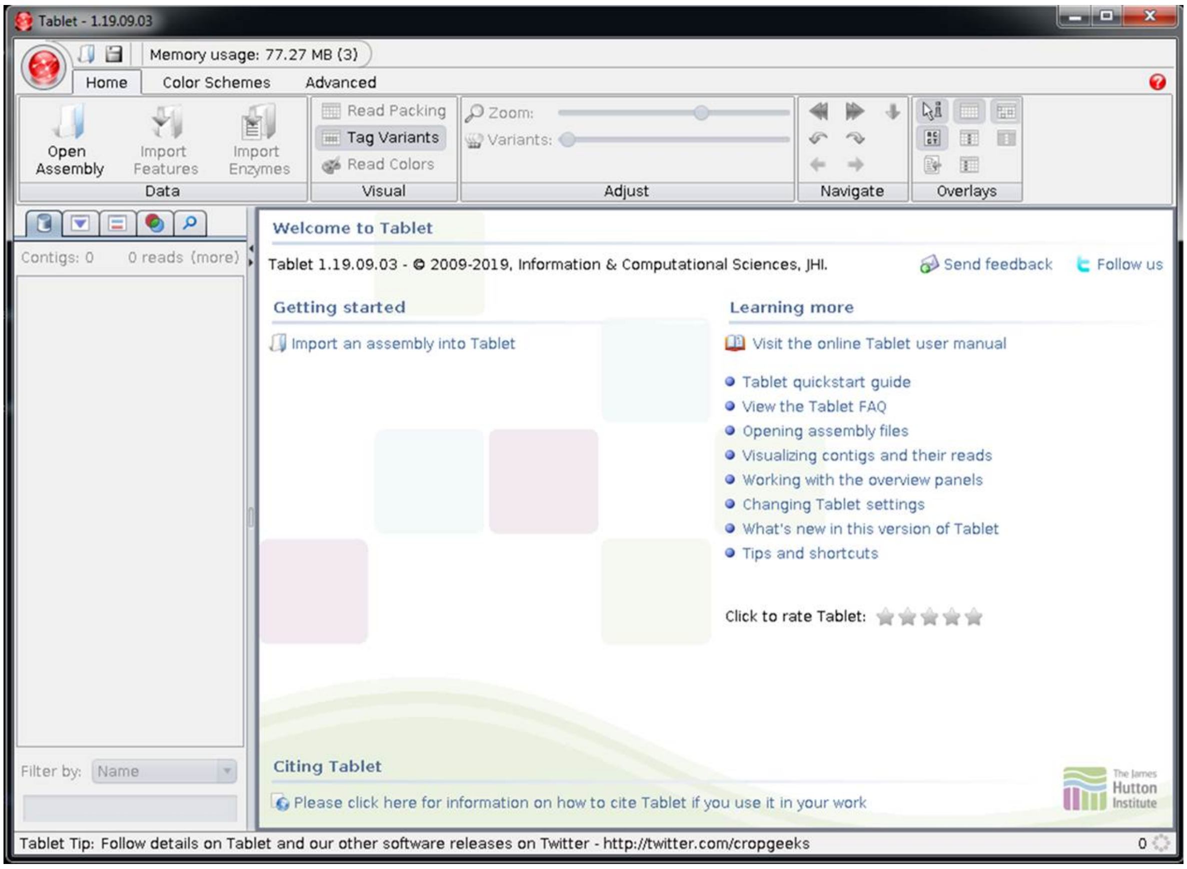
1. **How many reads map to each alignment?**
2. **What is the average coverage and the % coverage?**
3. **How do the coverage plots differ?**
4. **What is the main reason for this difference?**

Visualization

For visualizing a SAM/BAM file we are going to use Tablet

```
$ ~/software/Tablet/tablet
```

This opens up a GUI



Welcome to Tablet

Tablet 1.19.09.03 - © 2009-2019, Information & Computational Sciences, JHI.

Send feedback
Follow us

Getting started

Import an assembly into Tablet

Learning more

Visit the online Tablet user manual

- Tablet quickstart guide
- View the Tablet FAQ
- Opening assembly files
- Visualizing contigs and their reads
- Working with the overview panels
- Changing Tablet settings
- What's new in this version of Tablet
- Tips and shortcuts

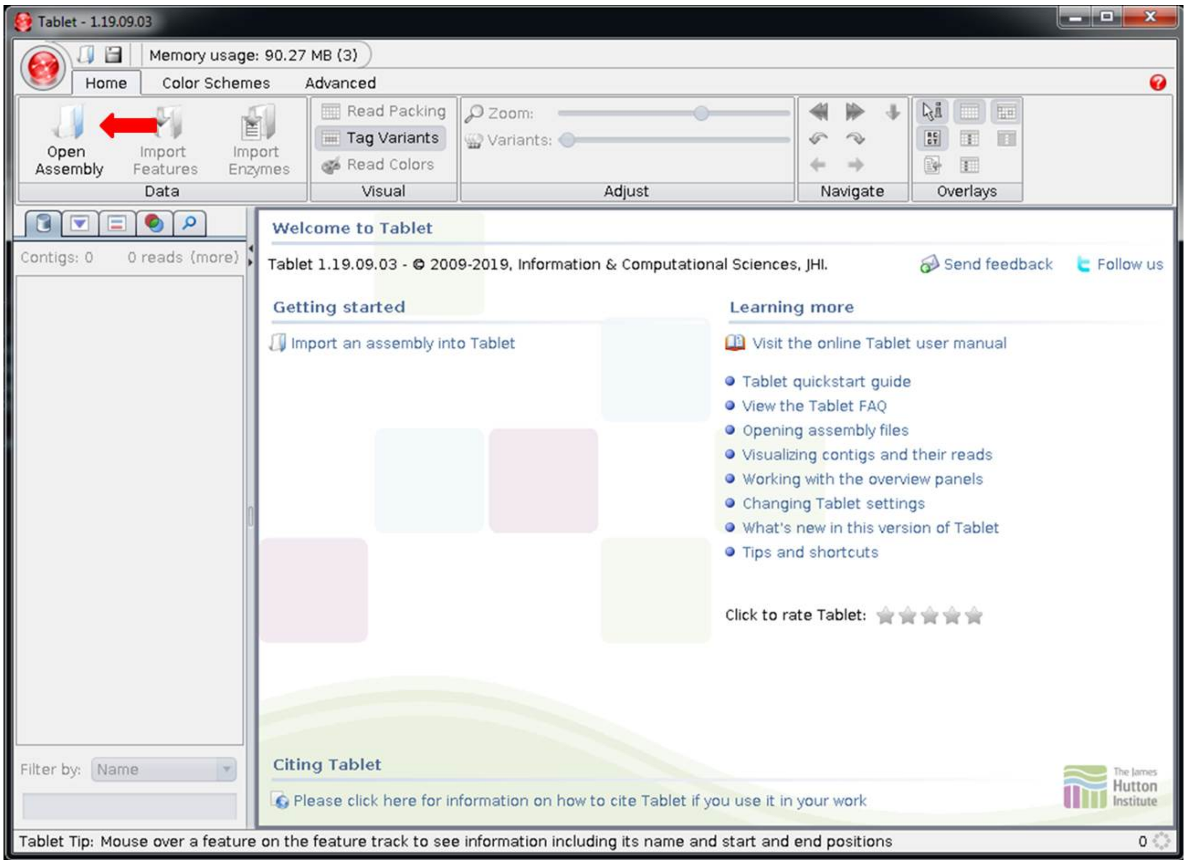
Click to rate Tablet:

Citing Tablet

Please click here for information on how to cite Tablet if you use it in your work

Tablet Tip: Follow details on Tablet and our other software releases on Twitter - http://twitter.com/cropgeeks

To open a SAM/BAM file we click on **Open Assembly**



Welcome to Tablet

Tablet 1.19.09.03 - © 2009-2019, Information & Computational Sciences, JHI.

Send feedback
Follow us

Getting started

Import an assembly into Tablet

Learning more

Visit the online Tablet user manual

- Tablet quickstart guide
- View the Tablet FAQ
- Opening assembly files
- Visualizing contigs and their reads
- Working with the overview panels
- Changing Tablet settings
- What's new in this version of Tablet
- Tips and shortcuts

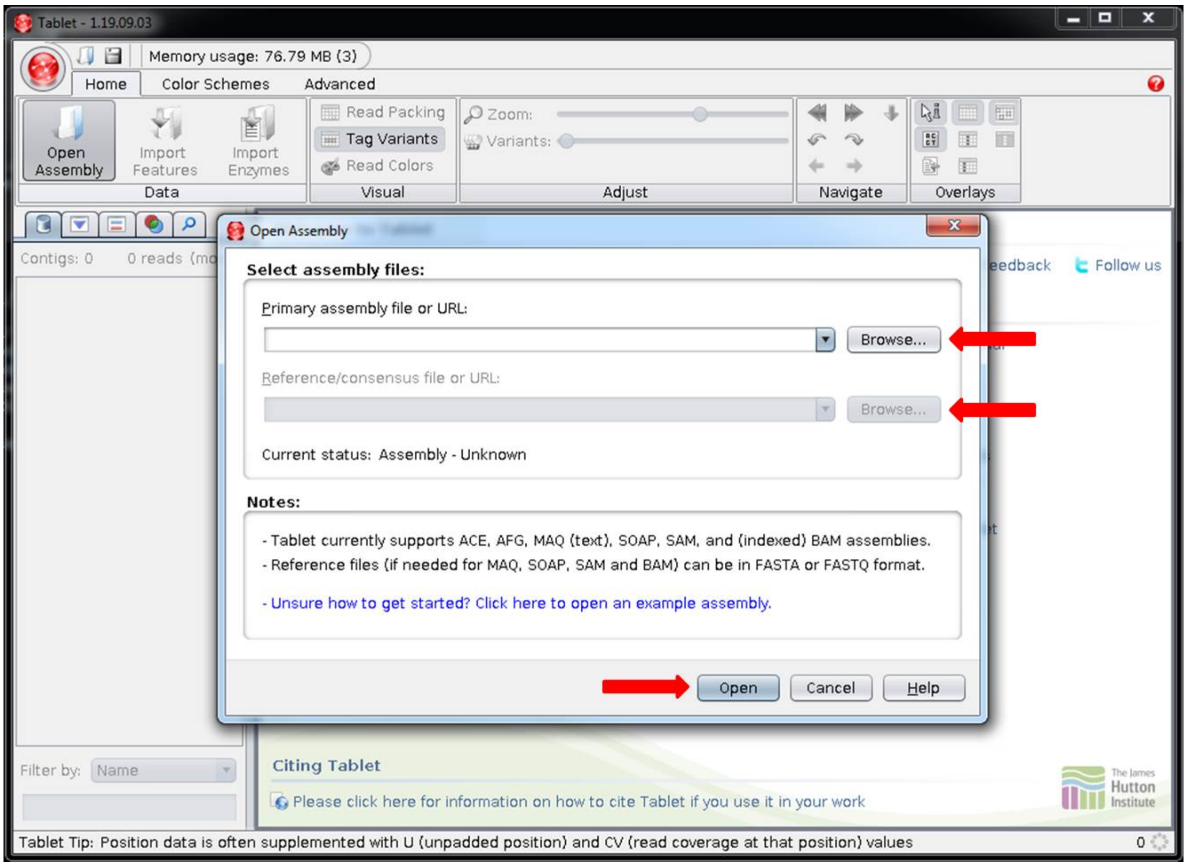
Click to rate Tablet:

Citing Tablet

Please click here for information on how to cite Tablet if you use it in your work

Tablet Tip: Mouse over a feature on the feature track to see information including its name and start and end positions

We are going to use the file **PG15-BW001347sorted.bam**. The reference sequence will be **Ref.C.IN.95.95IN21068.AF067155.fa**. After both these files have been selected Click open

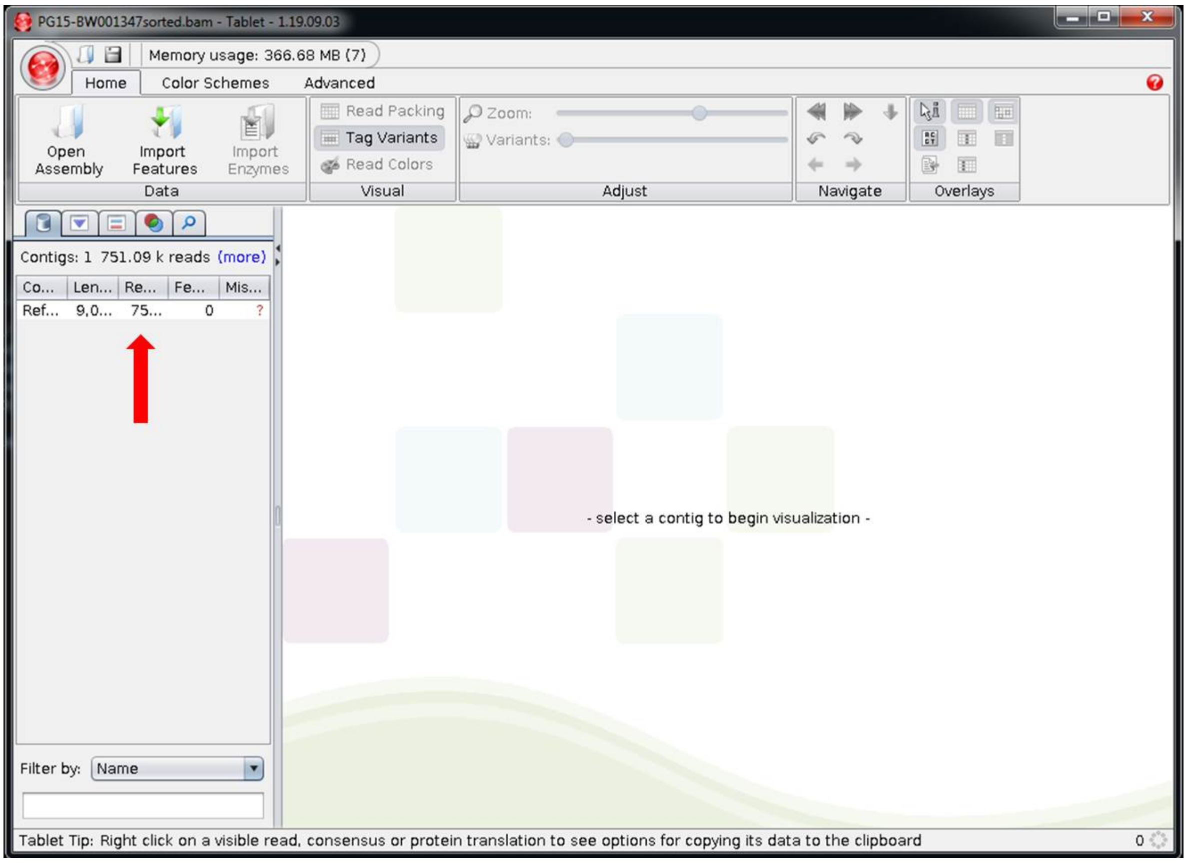


- Tablet currently supports ACE, AFG, MAQ (text), SOAP, SAM, and (indexed) BAM assemblies.
- Reference files (if needed for MAQ, SOAP, SAM and BAM) can be in FASTA or FASTQ format.
- Unsure how to get started? Click here to open an example assembly.

Please click here for information on how to cite Tablet if you use it in your work

Tablet Tip: Position data is often supplemented with U (unpadded position) and CV (read coverage at that position) values

After loading you should see a message **- select a contig to begin visualization -** along with a list of contig in the left hand panel. If you have multiple mapping like the Multiref Mapping all contigs will show up. In this case because we have a single reference only one contig shows up. Click on this entry to load the mapping.



Contigs: 1 751.09 k reads (more)

Co...	Len...	Re...	Fe...	Mis...
Ref...	9,0...	75...	0	?

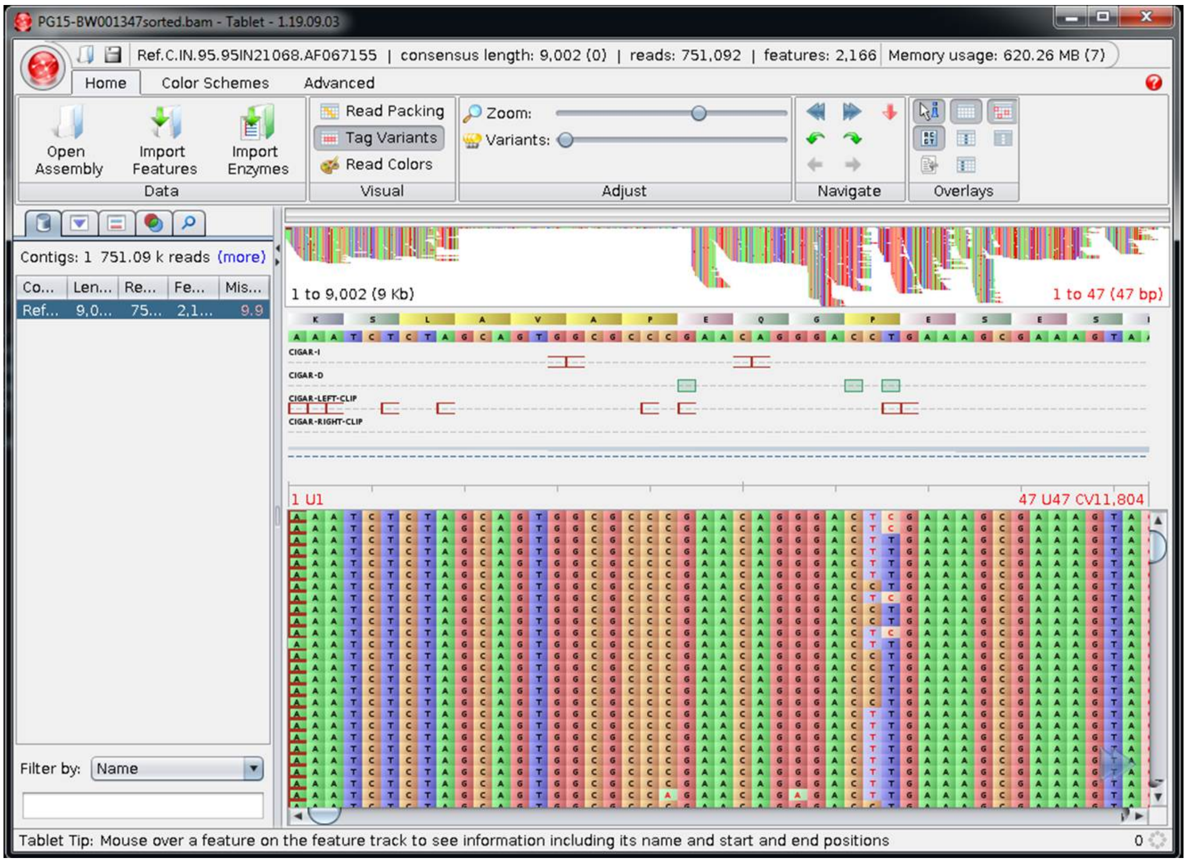
- select a contig to begin visualization -

Tablet Tip: Right click on a visible read, consensus or protein translation to see options for copying its data to the clipboard

We can now see the complete BAM file and we can scroll across to see how the reads have mapped. On top in the tool panel we have a few important tools that we can play around with

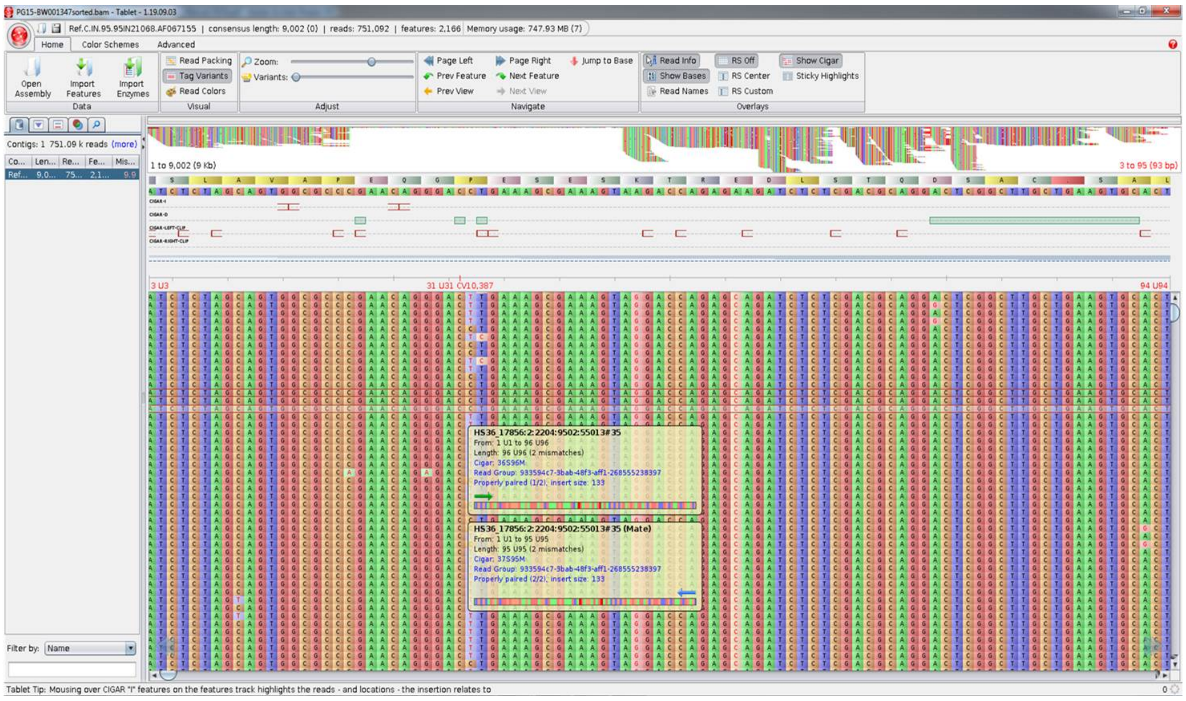
- **Zoom** : Helps you zoom in to the base level and out to the complete genome level
- **Read Colors** : Helps color the reads in many different ways
- **Variants** : Helps highlight only the variants of the alignment

Just below the tool panel we have the coverage spread across the genome followed by the reference sequence. Below that we have information on insertion, deletions and soft clips and finally the read display. Deletions are marked by a red asterisk and Insertions highlighted with red boxes



Tablet Tip: Mouse over a feature on the feature track to see information including its name and start and end positions

Hovering over the reads also gives us the read information

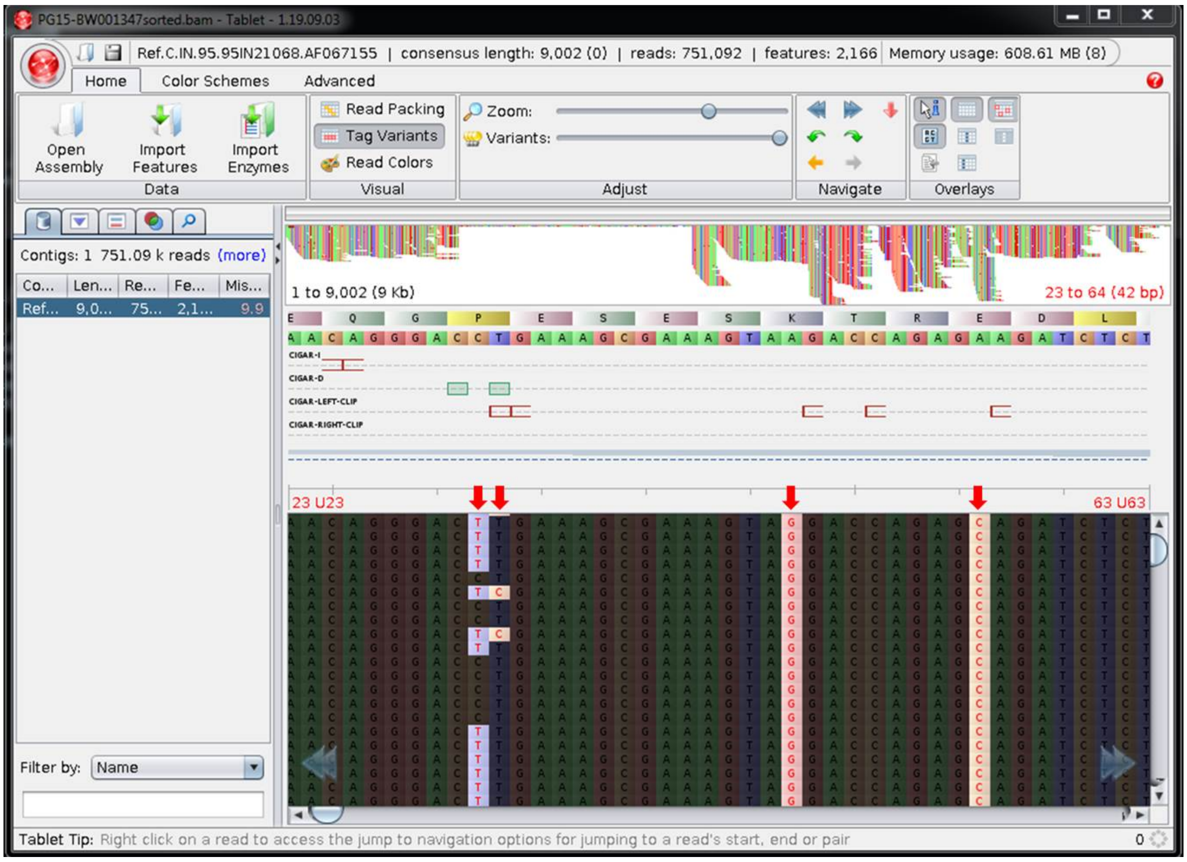

PG15-BW001347sorted.bam - Tablet - 1.19.09.03
Ref.C.IN.95.95IN21068.AF067155 | consensus length: 9,002 (0) | reads: 751,092 | features: 2,166 Memory usage: 747.93 MB (7)

HS36_17856:2:2204:9502:55013#35
From: 1 U1 to 96 U96
Length: 96 U96 (2 mismatches)
Cigar: 36S96M
Read Group: 933594c7-3bab-48f3-aff1-268555238397
Properly paired (1/2), insert size: 133

HS36_17856:2:2204:9502:55013#35 (Mate)
From: 1 U1 to 95 U95
Length: 95 U95 (2 mismatches)
Cigar: 37S95M
Read Group: 933594c7-3bab-48f3-aff1-268555238397
Properly paired (2/2), insert size: 133

Tablet Tip: Mousing over CIGAR "I" features on the features track highlights the reads - and locations - the insertion relates to

If we slide the Variant slider all the way to the right we highlight the variants in the SAM/BAM file. You can identify both consensus level and minority mutations. Minority variants can arise because of viral diversity and due to sequencing errors



Ref.C.IN.95.95IN21068.AF067155 | consensus length: 9,002 (0) | reads: 751,092 | features: 2,166 | Memory usage: 608.61 MB (8)

Tablet Tip: Right click on a read to access the jump to navigation options for jumping to a read's start, end or pair

Removing duplicates

We will use PICARD Tools to identify and mark duplicate reads.

Note: This mainly holds true for PCR duplicates generated during the library preparation process. Can also remove optical duplicates.

Note: Duplicate removal in this case carried out on a sorted BAM file

```
$ java -jar ~/software/picard-tools-1.119/picard.jar -h
```

PICARD can be used for multiple analyses of NGS data. We primarily use the **MarkDuplicates** command to remove duplicates.

```
$ java -jar ~/software/picard-tools-1.119/picard.jar MarkDuplicates INPUT=PG15-
BW001347sorted.bam OUTPUT=PG15-BW001347sorteddedup.bam
METRICS_FILE=1347_metrics.txt REMOVE_DUPLICATES=true ASSUME_SORTED=true
VERBOSITY=ERROR
$ nano 1347_metrics.txt
$ samtools index PG15-BW001347sorteddedup.bam
$ weeSAMv1.4 -b PG15-BW001347sorteddedup.bam --out 1347dedup.txt --plot
1347dedup.jpg
```

INPUT : Input file (Sorted BAM)

OUTPUT : Output file

METRICS_FILE : Output file with de-duplication metrics

REMOVE_DUPLICATES : if true removes duplicates from BAM file

ASSUME_SORTED : If input file is sorted

VERBOSITY : Output to screen

What percentage reads were marked as duplicates?

How does duplicate removal change the average depth?

Now repeat this on the sorted BAM files created from

- **PG15-BW001432.bam**
- **PG16-BW002087.bam**

What percentage reads were marked as duplicates in each case?

How does duplicate removal change the average depth in each case?

The output after duplicate removal is taken forward to Variant Calling