Training module

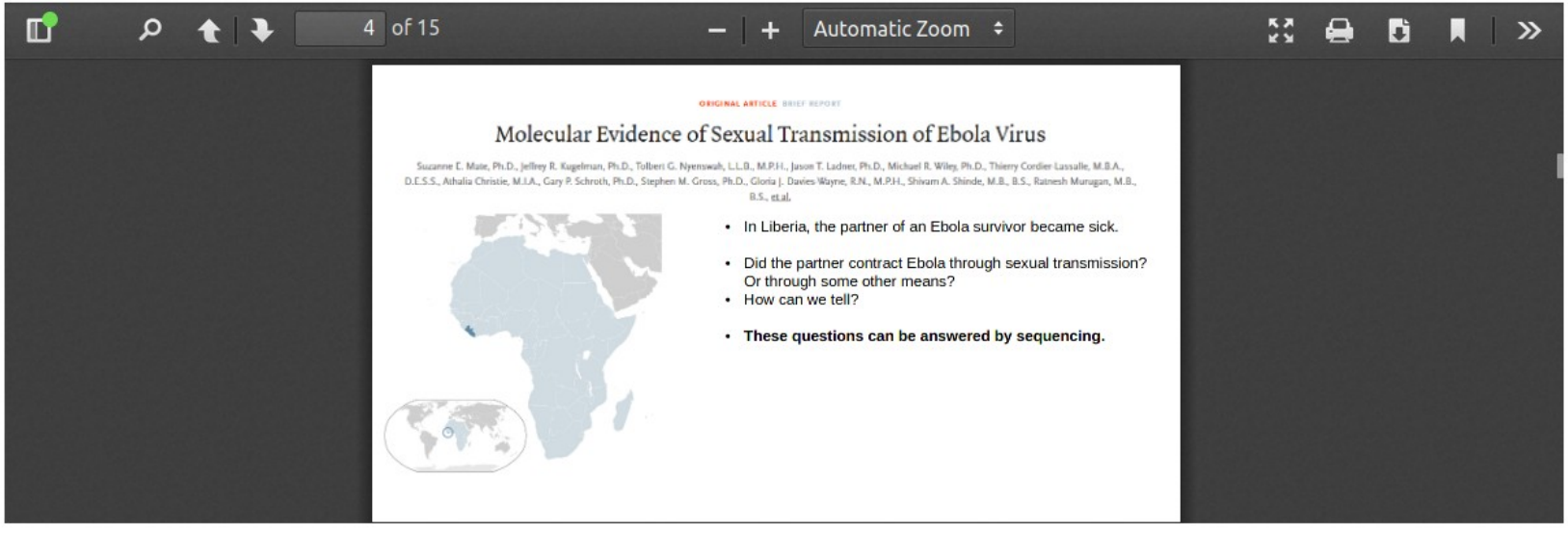
```
##############################################################################
# Genomic molecular characterization for viral strains using informatics tools    #
# CGS, USAMRIID                                                                    #
# Authors: Raina Kumar (code and training module pipeline),                        #
#          Joushua Richardson (documentation and presentations)                    #
# Contact: raina.kumar.ctr@mail.mil                                                #
##############################################################################
```

Objective

The training module will provide the complete bioinformatics workflow for analyzing genomics data using open source tools. The training module uses sequence reads generated using genomics tools such as genomic DNA or RNA sequencing using next generation sequencing technology with objective of characterization of viral strains in outbreak setting.

```
from IPython.display import IFrame
IFrame('documentation/final_pdfs/1_training_mod_013120_intro.pdf', width=900, height=300)
```

Out[4]:

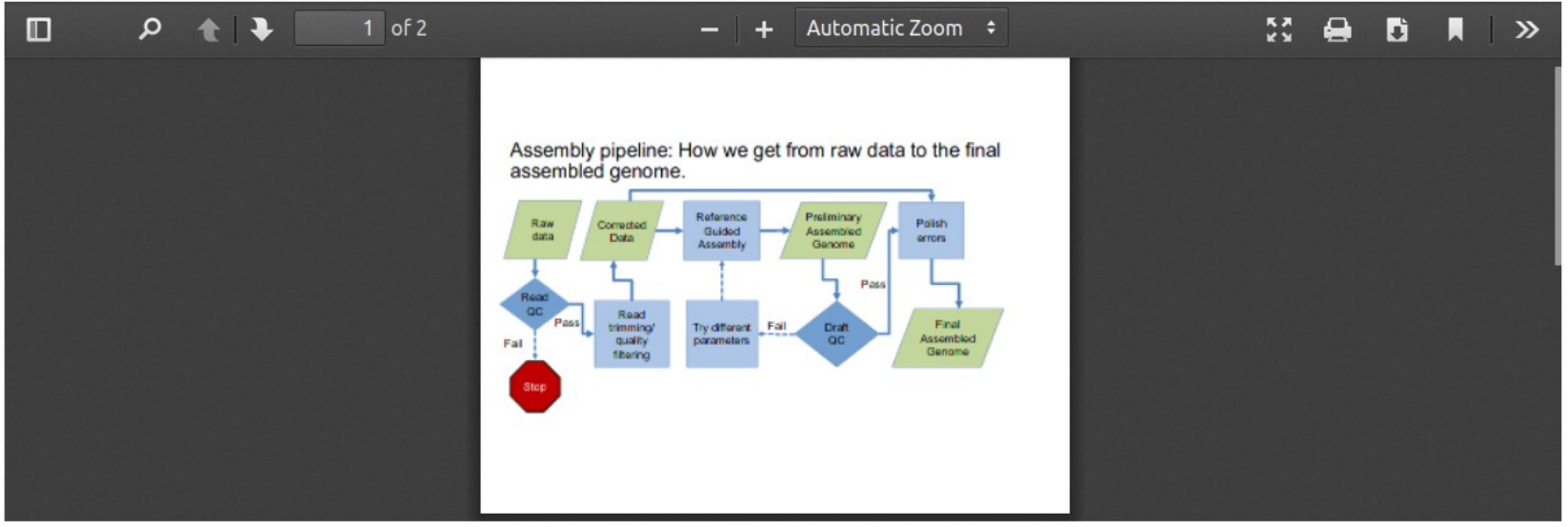


In [3]:

Edit Metadata

```
## Introduction to genomics assembly workflow
from IPython.display import IFrame
IFrame('documentation/final_pdfs/2_training_mod_013120_AssemblyPipe.pdf', width=900, height=300)
```

Out[3]:



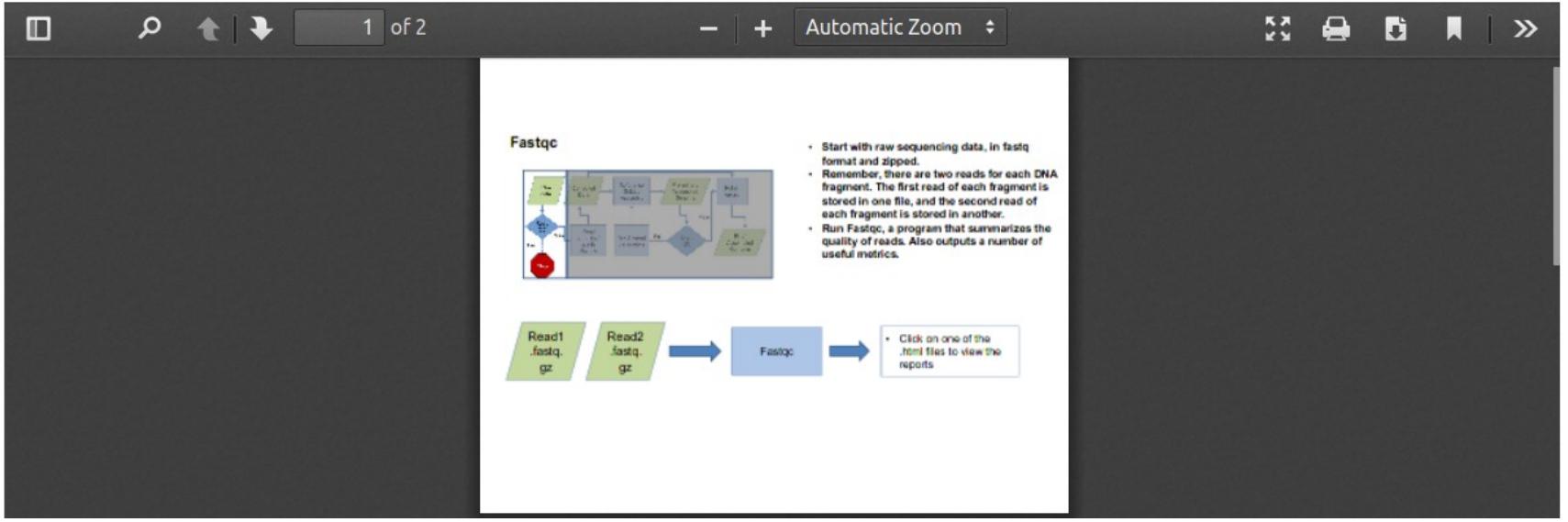
In [2]:

Edit Metadata

```
# Step 1
# Define paths for input base directory, work directory and result directory in config.yaml for any new datasets
#

base_dir ="/home/guest/projects/"
work_dir  = "/home/guest/projects/makono"
result_dir = "/home/guest/projects/results/"
reference_dir ="/home/guest/projects/makona/references/"
srefindex="/home/guest/projects/makona/seqindex/"
sreference="/home/guest/projects/makona/references/GCF_000848505.1_ViralProj14703_genomic.fna"
pri_adaptors="/home/guest/projects/makona/references/pri_adaptors.fa"
```

```
## Step 2

##Run following:
##
## shell command
## For paired end data
##   test fastqc read.R1_001.fastq.gz read.R2_001.fastq.gz -f fastq -o results/fastqc > log.txt

from IPython.display import IFrame
IFrame('documentation/final_pdfs/3_training_mod_013120__Fastqc.pdf', width=900, height=300)
```

Out[4]:



```
!snakemake -s "popgen_fastqc.smk"
```

```
/home/guest/projects//makona/results directory exists
|--- Results directory is: /home/guest/projects//makona/results
|--- The current working directory is /home/guest/projects//makona
['Brett424_1_S4_L001']
|--- Number of samples to analyze: 1
|--- Sample Brett424_1_S4_L001 (reads: Brett424_1_S4_L001_R1_001.fastq.gz & Brett424_1_S4_L001_R2_001.fastq.gz) will be process
ed
Building DAG of jobs...
Using shell: /bin/bash
Provided cores: 3
Rules claiming more threads will be scaled down.
Job counts:
        count   jobs
        1       all
        1       raw_fastqc
        2

[Wed Jan 29 09:54:16 2020]
rule raw_fastqc:
    input: samples/raw/Brett424_1_S4_L001_R1_001.fastq.gz, samples/raw/Brett424_1_S4_L001_R2_001.fastq.gz
    output: results/fastqc/Brett424_1_S4_L001_R1_001_fastqc.html, results/fastqc/Brett424_1_S4_L001_R1_001_fastqc.zip, results/
fastqc/Brett424_1_S4_L001_R2_001_fastqc.html, results/fastqc/Brett424_1_S4_L001_R2_001_fastqc.zip, results/fastqc/Brett424_1_S4
_L001_fastqc.logfc.txt
    jobid: 1
```

```
    input: samples/raw/Brett424_1_S4_L001_R1_001.fastq.gz, samples/raw/Brett424_1_S4_L001_R2_001.fastq.gz, results/fastqc/Brett
424_1_S4_L001_R1_001_fastqc.html, results/fastqc/Brett424_1_S4_L001_R1_001_fastqc.zip, results/fastqc/Brett424_1_S4_L001_R2_001
_fastqc.html, results/fastqc/Brett424_1_S4_L001_R2_001_fastqc.zip, results/fastqc/Brett424_1_S4_L001_fastqc.logfc.txt
    jobid: 0

[Wed Jan 29 09:54:30 2020]
Finished job 0.
2 of 2 steps (100%) done
Complete log: /home/guest/projects/.snakemake/log/2020-01-29T095416.164303.snakemake.log
Workflow finished, no error
```

In [14]:

Edit Metadata

```
# Step 2 Fastqc results
from IPython.display import FileLink, FileLinks
FileLinks('makona/results/fastqc/.')
```

Out[14]:

makona/results/fastqc/./

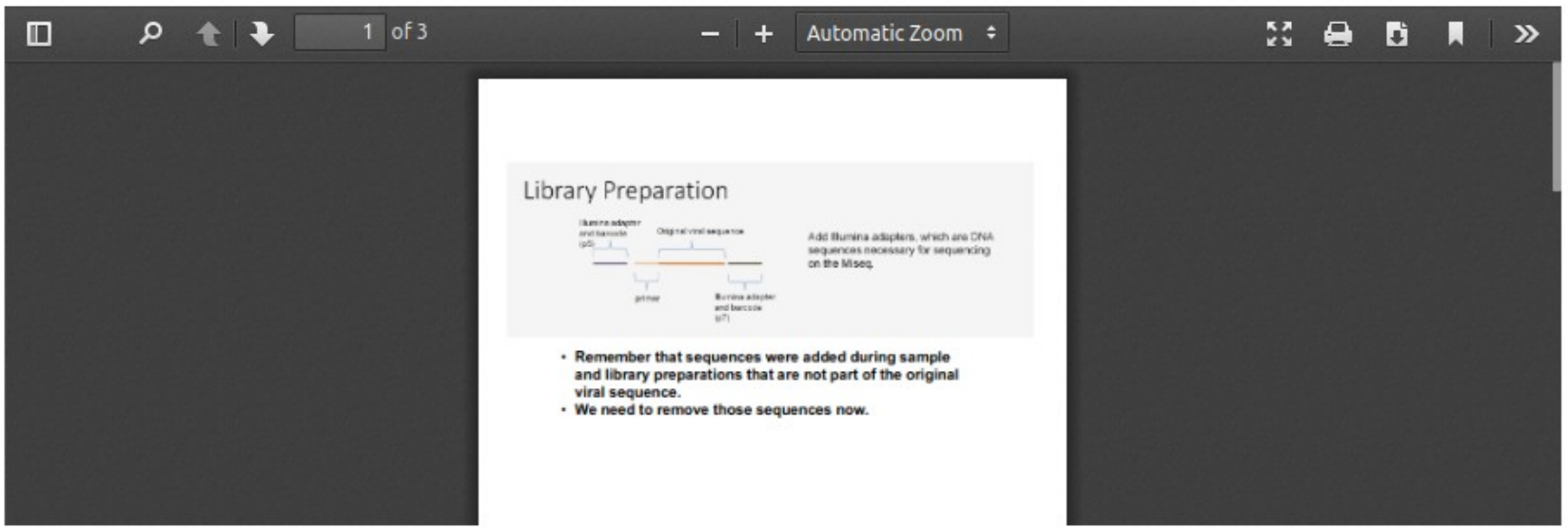
Brett424_1_S4_L001_fastqc.logfc.txt
Brett424_1_S4_L001_R2_001_fastqc.html
Brett424_1_S4_L001_R1_001_fastqc.zip
Brett424_1_S4_L001_R2_001_fastqc.zip
Brett424_1_S4_L001_R1_001_fastqc.html

In [15]:

Edit Metadata

```
# Step 3
## Trimming the bait illumina adaptors and primers from Illumina sequencing protocol using tool trimmomatic

##
## shell command
## For Paired end data
# "time java -jar trimmomatic-0.33.jar PE -threads 3 -trimlog logprefix input.read.R1_001.fastq.gz input.read.R2_001.fa
##


from IPython.display import IFrame
IFrame('documentation/final_pdfs/4_training_mod_013120__Trimv2.pdf', width=900, height=300)
```

Out[15]:



```
# Step 3 Run Trimmomatic on sequence reads using snakemake rule trimmomatics

!snakemake -s "popgen_trimmomatics.smk"
```

```
/home/guest/projects//makona/results directory exists
|--- Results directory is: /home/guest/projects//makona/results
|--- The current working directory is /home/guest/projects//makona
['Brett424_1_S4_L001']
|--- Number of samples to analyze: 1
|--- Sample Brett424_1_S4_L001 (reads: Brett424_1_S4_L001_R1_001.fastq.gz & Brett424_1_S4_L001_R2_001.fastq.gz) will be process
ed
Building DAG of jobs...
Nothing to be done.
```

In [16]:

Edit Metadata

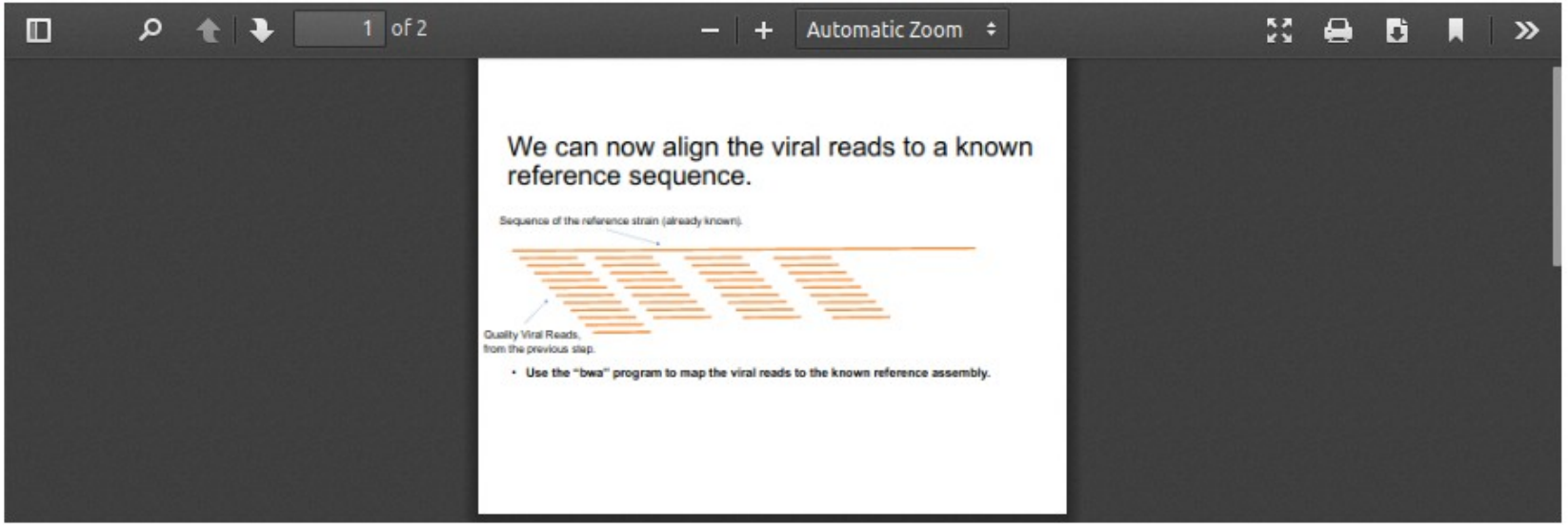
```
# Sequence read summary after trimming adaptors and primers

# Reports

from IPython.display import FileLink, FileLinks
FileLinks('makona/results/primer_adapt_removed/.')
```

Out[16]:

makona/results/primer_adapt_removed/./
Brett424_1_S4_L001_R1_unpaired.fastq
Brett424_1_S4_L001_trimmolog.txt
Brett424_1_S4_L001_R2_unpaired.fastq
Brett424_1_S4_L001_R2_paired.fastq
Brett424_1_S4_L001_R1_paired.fastq

```
# Step 4

## Reference mapping for Read correction
## Align reads to makona viral genome assembly fasta file

## Shell command
## time bwa mem -t 30 makona/references/GCF_000848505.1_ViralProj14703_genomic.fna input.read.1.fastq input.read.2.fast



from IPython.display import IFrame
IFrame('documentation/final_pdfs/5_training_mod_013120__Alignmentv2.pdf', width=900, height=300)
```

Out[6]:



In [17]:

Edit Metadata

```
# Run step 4 for reference mapping for read correction using snakemake rule refmapsam

!snakemake -s "popgen_refmapsam.smk" -n
```

```
/home/guest/projects//makona/results directory exists
|--- Results directory is: /home/guest/projects//makona/results
|--- The current working directory is /home/guest/projects//makona
['Brett424_1_S4_L001']
|--- Number of samples to analyze: 1
|--- Sample Brett424_1_S4_L001 (reads: Brett424_1_S4_L001_R1_001.fastq.gz & Brett424_1_S4_L001_R2_001.fastq.gz) will be process
ed
Building DAG of jobs...
Nothing to be done.
```

In [1]:

Edit Metadata

```
# Output from reference mapping
from IPython.display import FileLink, FileLinks
FileLinks('makona/results/ref_aligned/')
```

Out[1]:

makona/results/ref_aligned/

Brett424_1_S4_L001_assembly_align_mem_ref_sorted.bam

Brett424_1_S4_L001_assembly_align_mem_ref.sam

In [7]:

Edit Metadata

```
## Step 5

## Sort sam file and convert to bam format file using samtools software
## Shell command:
## "time samtools sort -O BAM makona.aligned.mem.sam > sample1.assembly_align_mem_ref_sorted.bam"
```

In [18]:

Edit Metadata

```
!snakemake -s "popgen_samsort2bam.smk" -n
```

```
/home/guest/projects//makona/results directory exists
|--- Results directory is: /home/guest/projects//makona/results
|--- The current working directory is /home/guest/projects//makona
['Brett424_1_S4_L001']
|--- Number of samples to analyze: 1
|--- Sample Brett424_1_S4_L001 (reads: Brett424_1_S4_L001_R1_001.fastq.gz & Brett424_1_S4_L001_R2_001.fastq.gz) will be process
ed
Building DAG of jobs...
Nothing to be done.
```

In [17]:

Edit Metadata

```
# Output from reference mapping


from IPython.display import FileLink, FileLinks
FileLinks('makona/results/ref_aligned/.')
```

Out[17]:

makona/results/ref_aligned/./
Brett424_1_S4_L001_assembly_align_mem_ref_sorted.bam
Brett424_1_S4_L001_assembly_align_mem_ref.sam

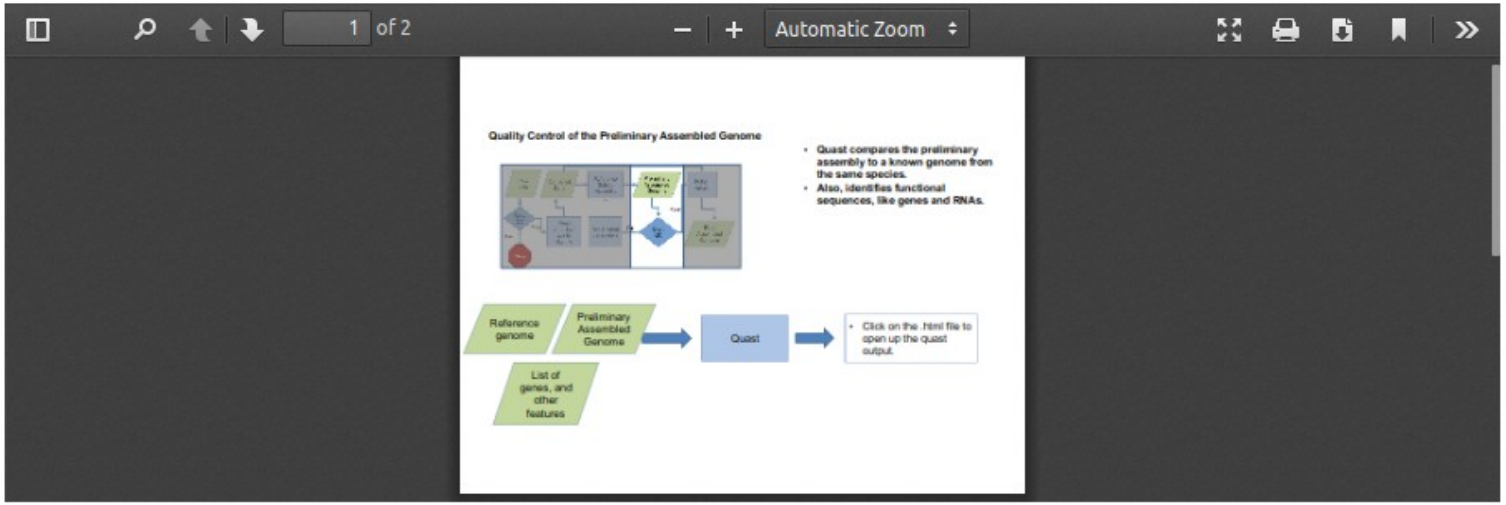
In [17]:

Edit Metadata

```
# Step 6

## Reference Guided Assembly graph using velvet assembler

## Shell Command:
## "time velveth out.assembly.dir input.kmernumber -bam -longPaired {output.assembly.dir"


from IPython.display import IFrame
IFrame('documentation/final_pdfs/', width=900, height=300)
```

```
## Reference Guided Assembly graph using velvet assembler

## Shell Command:
## "time velveth out.assembly.dir input.kmernumber -bam -longPaired {output.assembly.dir"


from IPython.display import IFrame
IFrame('documentation/final_pdfs/', width=900, height=300)
```

Out[17]:



In [19]:

Edit Metadata

```
!snakemake -s "popgen_assembly.smk" -n
```

```
/home/guest/projects//makona/results directory exists
|--- Results directory is: /home/guest/projects//makona/results
|--- The current working directory is /home/guest/projects//makona
['Brett424_1_S4_L001']
|--- Number of samples to analyze: 1
|--- Sample Brett424_1_S4_L001 (reads: Brett424_1_S4_L001_R1_001.fastq.gz & Brett424_1_S4_L001_R2_001.fastq.gz) will be process
ed
Building DAG of jobs...
Nothing to be done.
```

In [5]:

Edit Metadata

```
# Output from reference mapping


from IPython.display import FileLink, FileLinks
FileLinks('makona/results/velvet_assembly/')
```

Out[5]:

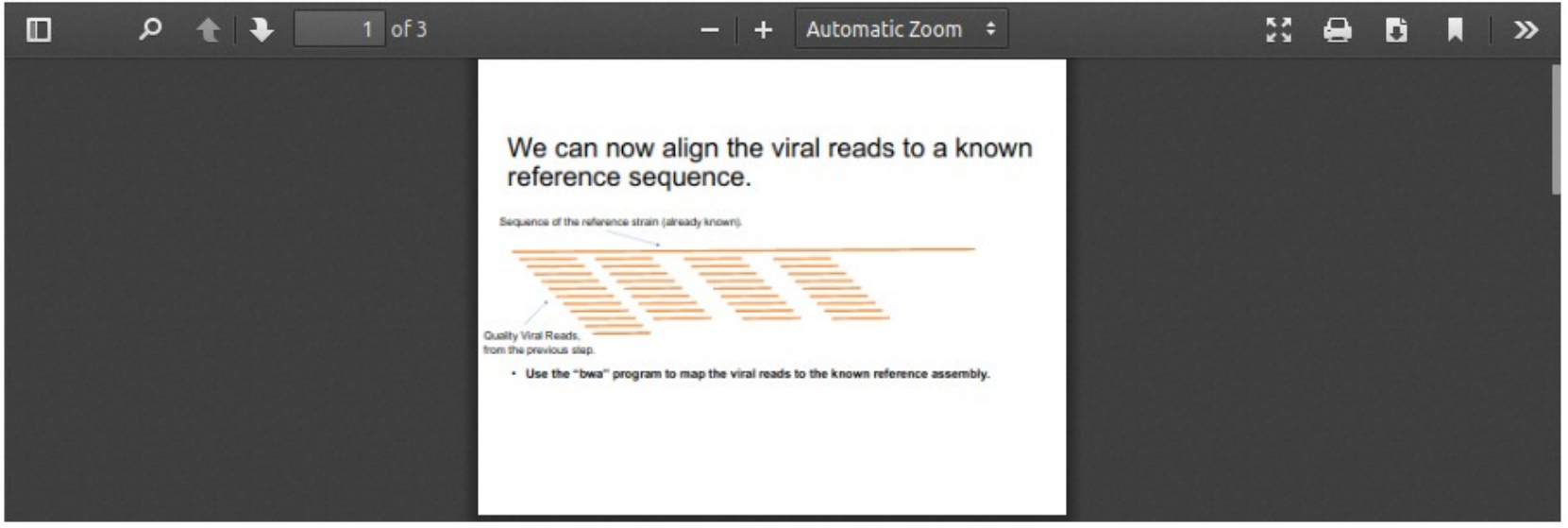
makona/results/velvet_assembly/
Brett424_1_S4_L001_assembly_log.txt
Brett424_1_S4_L001_logfile_assemref_27.txt
Brett424_1_S4_L001_logfile_cindex.txt
Brett424_1_S4_L001_reindex.log.txt
makona/results/velvet_assembly/Brett424_1_S4_L001_AssemRef/
contigs.fa.fai
contigs.fa
Log
Roadmaps
contigs.fa.bwt
contigs.fa.ann
PreGraph
contigs.fa.pac
contigs.fa.sa
velvet_asm.afg
LastGraph
Sequences
stats.txt
Graph
contigs.fa.amb

In [4]:

Edit Metadata

```
# Step 7

## Reference Guided Assembly map using velvet assembler
## Shell Command:

## "time velvetg input.out.assembly.dir -amos_file yes > output.logfile"


from IPython.display import IFrame
IFrame('documentation/final_pdfs/5_training_mod_013120__Alignmentv2.pdf', width=900, height=300)
```

Out[4]:



In [20]:

Edit Metadata

```
!snakemake -s "popgen_assembly_sgraph.smk" -n
```

```
/home/guest/projects//makona/results directory exists
|--- Results directory is: /home/guest/projects//makona/results
|--- The working directory is: /home/guest/projects//makona
['Brett424_1_S4_L001']
|--- Number of samples to analyze: 1
|--- Sample Brett424_1_S4_L001 (reads: Brett424_1_S4_L001_R1_001.fastq.gz & Brett424_1_S4_L001_R2_001.fastq.gz) will be process
ed
Building DAG of jobs...
Nothing to be done.
```

click to expand output; double click to hide output

In [6]:

Edit Metadata

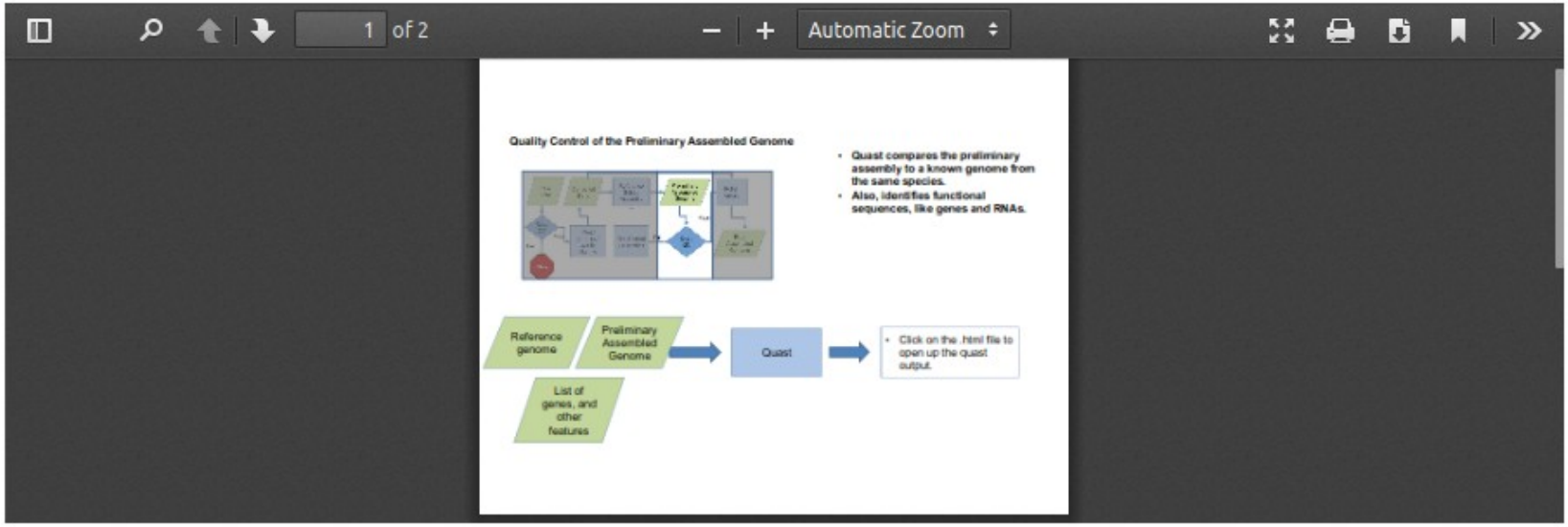
```
## Step 7 output
## # Output from velvet assembly

from IPython.display import FileLink, FileLinks
FileLinks('makona/results/velvet_assembly/.')
```

Out[6]:

makona/results/velvet_assembly/
Brett424_1_S4_L001_assembly_log.txt
Brett424_1_S4_L001_logfile_assemref_27.txt
Brett424_1_S4_L001_logfile_cindex.txt
Brett424_1_S4_L001_reindex.log.txt
makona/results/velvet_assembly/Brett424_1_S4_L001_AssemRef/
contigs.fa.fai
contigs.fa
Log
Roadmaps
contigs.fa.bwt
contigs.fa.ann
PreGraph
contigs.fa.pac
contigs.fa.sa

```
# Step 8

## Assembly quality assesment stastics and gene prediction
## Shell Command:

## "time quast.py step7.input.contig.fa -R chk.genome.fa -G chk.genome.gff -o out.assembly.stat.reports --glimmer > out



from IPython.display import IFrame
IFrame('documentation/final_pdfs/6_training_mod_013120__DraftQC.pdf', width=900, height=300)
```

Out[8]:



In [21]:

Edit Metadata

```
!snakemake -s "popgen_assembly_predictgene.smk" -n
```

In [21]:

Edit Metadata

```
!snakemake -s "popgen_assembly_predictgene.smk" -n
```

```
/home/guest/projects//makona/results directory exists
|--- Results directory is: /home/guest/projects//makona/results
|--- The current working directory is /home/guest/projects//makona
['Brett424_1_S4_L001']
|--- Number of samples to analyze: 1
|--- Sample Brett424_1_S4_L001 (reads: Brett424_1_S4_L001_R1_001.fastq.gz & Brett424_1_S4_L001_R2_001.fastq.gz) will be process
ed
Building DAG of jobs...
Nothing to be done.
```

In [10]:

Edit Metadata

```
## Step 8 Assembly reports

from IPython.display import HTML
HTML(filename="./makona/results/assembl_stats/Brett424_1_S4_L001_reference_stats/report.html")
```

Out[10]:

QUAST

Quality Assessment Tool for Genome Assemblies by CAB

24 January 2020, Friday, 02:19:26

View in Icarus contig browser

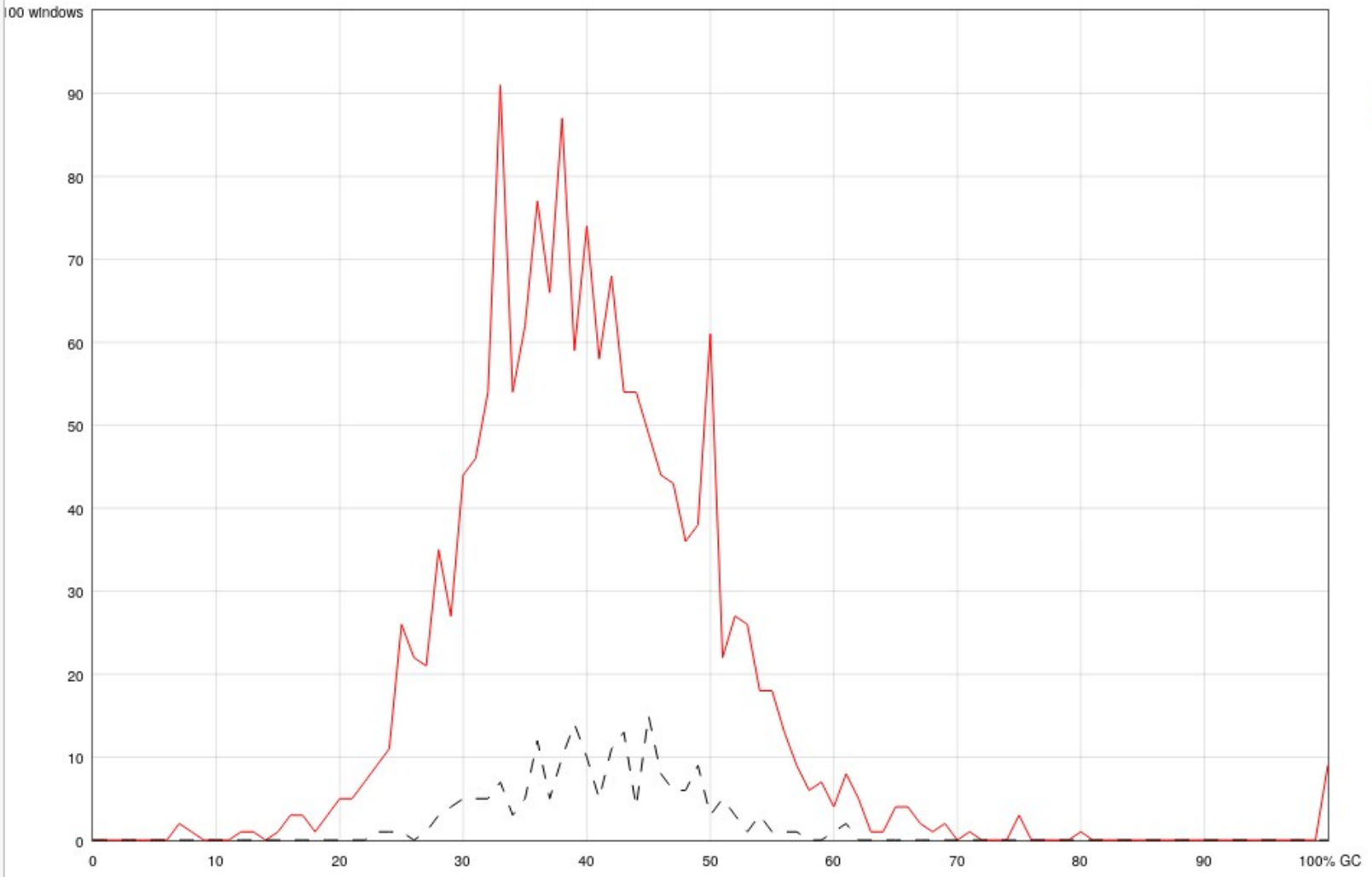
All statistics are based on contigs of size >= 300 bp, unless otherwise noted (e.g., "# contigs (>= 0 bp)" and "Total length (>= 0 bp)" include all contigs).

Aligned to "GCF_000848505.1_ViralProj14703_genomic" | 18 959 bp | 1 fragment | 41.07 % G+C

55 genomic features

click to expand output; double click to hide output

Cumulative length Nx NAx NGx NGAx Misassemblies Genomic features GC content Normal / logarithmic scale



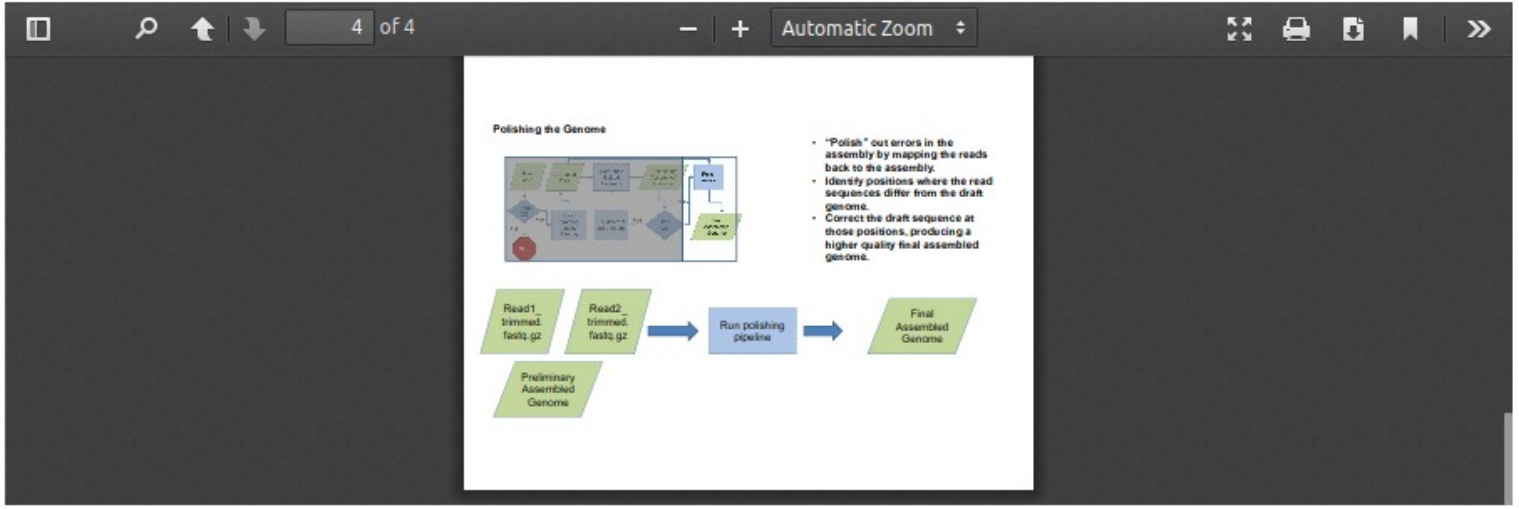
cont
by con
refere

In [9]:

Edit Metadata

```
# Step 9
## Create index of contigs and map reads back to contig
## Shell command:

## "time bwa index -a bwtsw step7.input.contig.fa > output.logfile"

from IPython.display import IFrame
IFrame('documentation/final_pdfs/7_training_mod_013120__Polishv2.pdf', width=900, height=300)
```

Out[9]:



In [22]:

Edit Metadata

```
!snakemake -s "popgen_bwaindex_contig.smk" -n
```

```
/home/guest/projects//makona/results directory exists
|--- Results directory is: /home/guest/projects//makona/results
|--- The current working directory is /home/guest/projects//makona
['Brett424_1_S4_L001']
|--- Number of samples to analyze: 1
```

```
!snakemake -s "popgen_bwaindex_contig.smk" -n
```

```
/home/guest/projects//makona/results directory exists
|--- Results directory is: /home/guest/projects//makona/results
|--- The current working directory is /home/guest/projects//makona
['Brett424_1_S4_L001']
|--- Number of samples to analyze: 1
|--- Sample Brett424_1_S4_L001 (reads: Brett424_1_S4_L001_R1_001.fastq.gz & Brett424_1_S4_L001_R2_001.fastq.gz) will be process
ed
Building DAG of jobs...
Nothing to be done.
```

In [10]:

Edit Metadata

```
# Step 10
## "time bwa mem -t 30 step8.input.contig.fa {input.read1p} {input.read2p} > {output.contigalign}"
```

In [28]:

Edit Metadata

```
!snakemake -s "popgen_alignreads2contig.smk" -n
```

```
/home/guest/projects//makona/results directory exists
|--- Results directory is: /home/guest/projects//makona/results
|--- The current working directory is /home/guest/projects//makona
['Brett424_1_S4_L001']
|--- Number of samples to analyze: 1
|--- Sample Brett424_1_S4_L001 (reads: Brett424_1_S4_L001_R1_001.fastq.gz & Brett424_1_S4_L001_R2_001.fastq.gz) will be process
ed
Building DAG of jobs...
Nothing to be done.
```

In [28]:

Edit Metadata

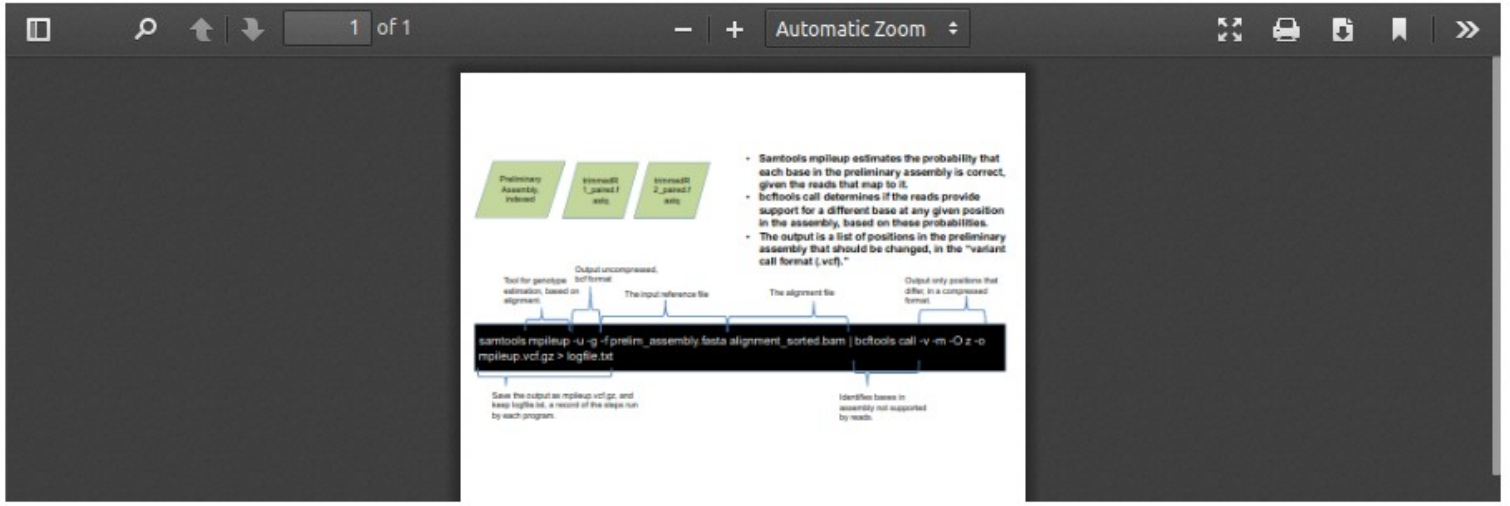
```
# Step 11
## Coordinate sort sam files and convert to bam file using samtools
```

In [30]:

Edit Metadata

```
# Step 13

## Variant Calling using samtools mpileup

## Shell Command:

## "time samtools mpileup -u -g -f step8.input.contig.fa step11.contig.read.sorted.aligned.bam | bcftools call -v -m -O

from IPython.display import IFrame
IFrame('documentation/command_pdfs/training_mod_Draft_Sl39.pdf', width=900, height=300)
```

Out[30]:



In [31]:

Edit Metadata

```
!snakemake -s "popgen_variantsCall.smk" -n
```

```
/home/guest/projects//makona/results directory exists
|--- Results directory is: /home/guest/projects//makona/results
|--- The current working directory is /home/guest/projects//makona
['Brett424_1_S4_L001']
|--- Number of samples to analyze: 1
```

```
from IPython.display import FileLink, FileLinks
FileLinks('makona/results/variants_calling/.')
```

Out[38]:

makona/results/variants_calling/./
Brett424_1_S4_L001_mpileup.vcf.gz
Brett424_1_S4_L001_mpileup.vcf.gz.csi
Brett424_1_S4_L001_vcfindex.txt
Brett424_1_S4_L001_snpcall.txt

In [32]:

Edit Metadata

```
!snakemake -s "popgen_vcfindex.smk" -n
```

```
/home/guest/projects//makona/results directory exists
|--- Results directory is: /home/guest/projects//makona/results
|--- The current working directory is /home/guest/projects//makona
['Brett424_1_S4_L001']
|--- Number of samples to analyze: 1
|--- Sample Brett424_1_S4_L001 (reads: Brett424_1_S4_L001_R1_001.fastq.gz & Brett424_1_S4_L001_R2_001.fastq.gz) will be process
ed
Building DAG of jobs...
Nothing to be done.
```

In [12]:

Edit Metadata

```
# Step 15

## Build sequences consensus

## Shell Command:

## "time cat step8.input.contig.fa | bcftools consensus output.mpileup.vcf.gz > output.consensus.fa
```

In [33]:

Edit Metadata

```
!snakemake -s "popgen_buildConsensus.smk" -n
```

```
!snakemake -s "popgen_buildConsensus.smk" -n
```

```
/home/guest/projects//makona/results directory exists
|--- Results directory is: /home/guest/projects//makona/results
|--- The current working directory is /home/guest/projects//makona
['Brett424_1_S4_L001']
|--- Number of samples to analyze: 1
|--- Sample Brett424_1_S4_L001 (reads: Brett424_1_S4_L001_R1_001.fastq.gz & Brett424_1_S4_L001_R2_001.fastq.gz) will be process
ed
Building DAG of jobs...
Nothing to be done.
```

In [39]:

Edit Metadata

```
## Reports

from IPython.display import FileLink, FileLinks
FileLinks('makona/results/consensus_seq/.')
```

Out[39]:

makona/results/consensus_seq/./
Brett424_1_S4_L001_consensus.fa

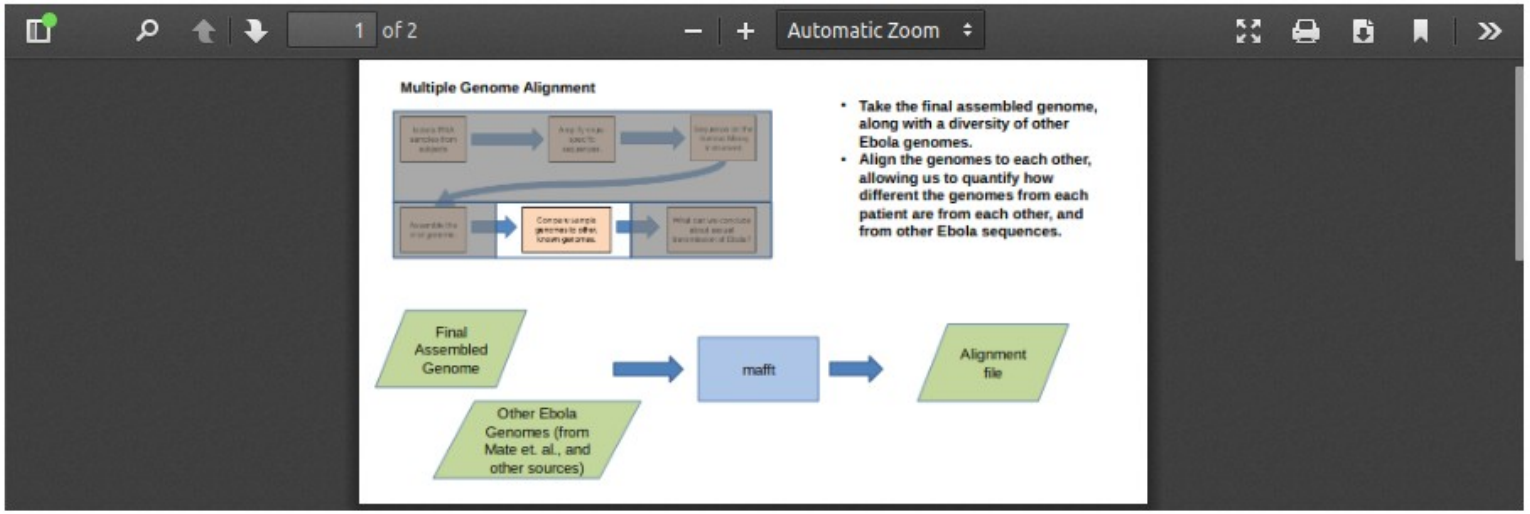
In [2]:

Edit Metadata

```
# Step 16

## Consensus muliple alignment

## Shell Command:

## cat final_assembly.fasta | mafft ebola_ref.fasta > Final_alignment.out


from IPython.display import IFrame
IFrame('documentation/final_pdfs/8_training_mod_013120__GenAlignv3.pdf', width=900, height=300)
```

Out[2]:

Out[2]:

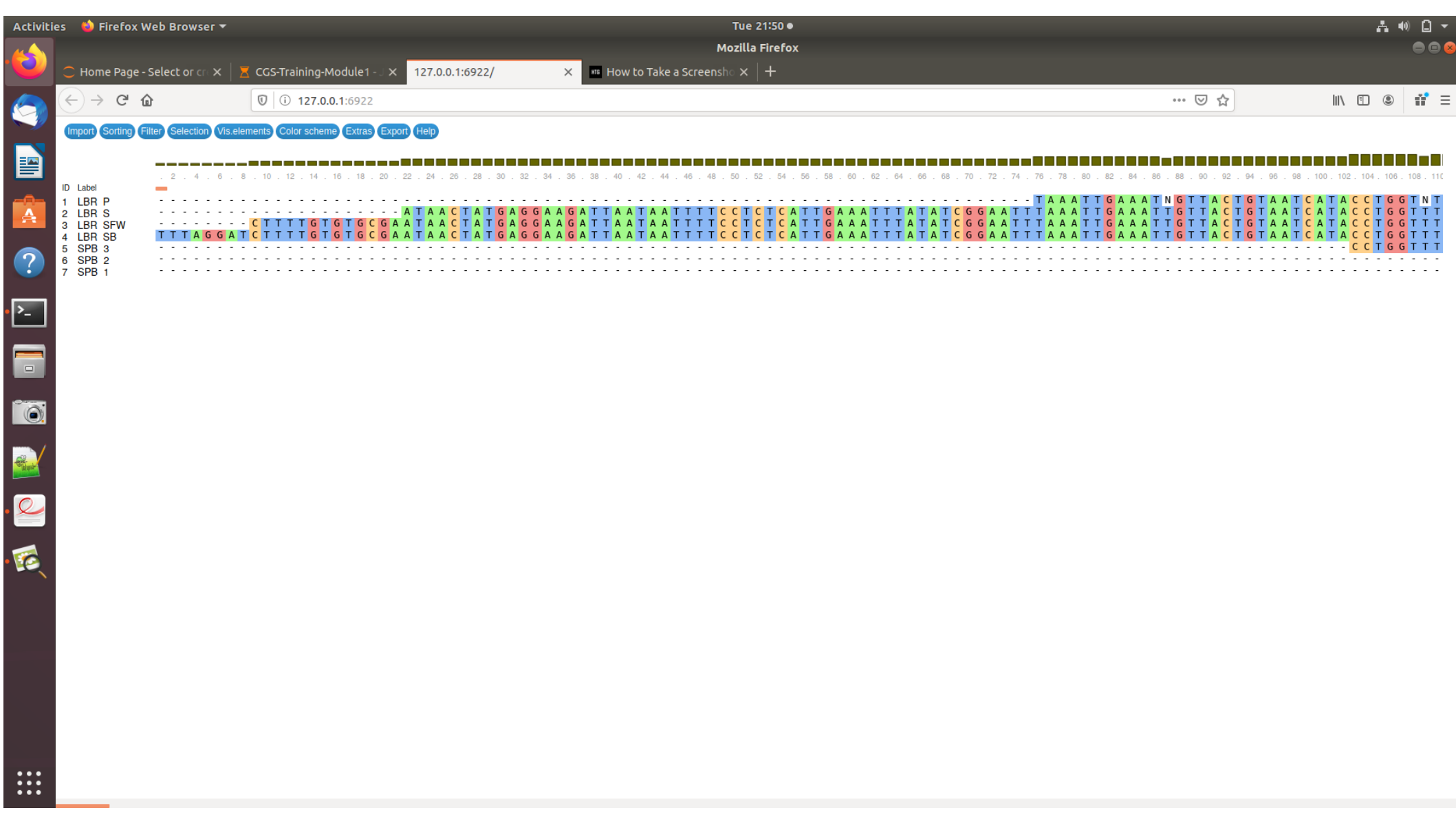


In [5]:

Edit Metadata

```
!snakemake -s "popgen_maff_alignment_view.smk"
```

```
/home/guest/projects//makona/results directory exists
|--- Results directory is: /home/guest/projects//makona/results
|--- The current working directory is /home/guest/projects//makona
['Brett424_1_S4_L001']
|--- Number of samples to analyze: 1
|--- Sample Brett424_1_S4_L001 (reads: Brett424_1_S4_L001_R1_001.fastq.gz & Brett424_1_S4_L001_R2_001.fastq.gz) will be proce
ssed
Building DAG of jobs...
Using shell: /bin/bash
Provided cores: 3
Rules claiming more threads will be scaled down.
Job counts:
```


Activities
Firefox Web Browser
Tue 21:50
Mozilla Firefox
Home Page - Select or cr
CGS-Training-Module1 -
127.0.0.1:6922/
How to Take a Screensho
127.0.0.1:6922
Import
Sorting
Filter
Selection
Vis.elements
Color scheme
Extras
Export
Help
ID Label
1 LBR P
2 LBR S
3 LBR SFW
4 LBR SB
5 SPB 3
6 SPB 2
7 SPB 1

In [*]:

Edit Metadata

```
## View MSA alignment

library(shiny)
runApp()
```

```
Listening on http://127.0.0.1:6922
```

In [2]:

Edit Metadata

```
## Reports

from IPython.display import FileLink, FileLinks
FileLinks('makona/results/maff_haplo/.')
```

Out[2]:

makona/results/maff_haplo/./
makona_muliple_alignment.out
final_contactenated_mafft.fa

In [3]:

Edit Metadata

```
## Shell Command:
# "time bcftools stats -F step8.input.contig.fa -s step11.output.mpileup.vcf.gz > output.variants.stat"

!snakemake -s "popgen_variants_stat.smk" -n
```

```
/home/guest/projects//makona/results directory exists
|--- Results directory is: /home/guest/projects//makona/results
|--- The current working directory is /home/guest/projects//makona
['Brett424_1_S4_L001']
|--- Number of samples to analyze: 1
|--- Sample Brett424_1_S4_L001 (reads: Brett424_1_S4_L001_R1_001.fastq.gz & Brett424_1_S4_L001_R2_001.fastq.gz) will be process
ed
Building DAG of jobs...
Nothing to be done.
```

Out[4]:

```
makona/results/variants_stats/./
```

click to expand output; double click to hide output

In [5]:

Edit Metadata

```
!head -100 makona/results/variants_stats/Brett424_1_S4_L001_vcf.stats
```

```
# This file was produced by bcftools stats (1.9+htslib-1.9) and can be plotted using plot-vcfstats.
# The command line was: bcftools stats  -F results/velvet_assembly/Brett424_1_S4_L001_AssemRef/contigs.fa -s - results/varian
ts_calling/Brett424_1_S4_L001_mpileup.vcf.gz
#
# Definition of sets:
# ID    [2]id   [3]tab-separated file names
ID      0       results/variants_calling/Brett424_1_S4_L001_mpileup.vcf.gz
# SN, Summary numbers:
#   number of records   .. number of data rows in the VCF
#   number of no-ALTs   .. reference-only sites, ALT is either "." or identical to REF
#   number of SNPs      .. number of rows with a SNP
#   number of MNPs      .. number of rows with a MNP, such as CC>TT
#   number of indels    .. number of rows with an indel
#   number of others    .. number of rows with other type, for example a symbolic allele or
#                          a complex substitution, such as ACT>TCGA
#   number of multiallelic sites     .. number of rows with multiple alternate alleles
#   number of multiallelic SNP sites .. number of rows with multiple alternate alleles, all SNPs
#
```

In [19]:

Edit Metadata

```
## Haplotype network and SNP analysis

## Shell

!snakemake -s "popgen_haplonetwork.smk"
```

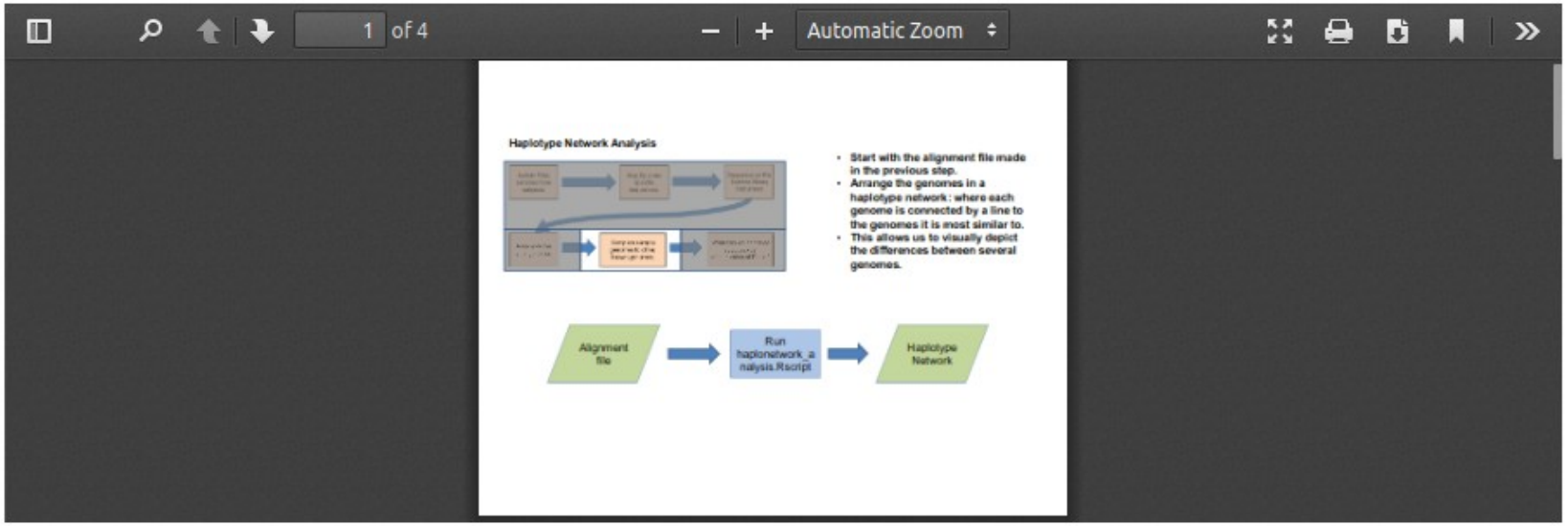
```
/home/guest/projects//makona/results directory exists
|--- Results directory is: /home/guest/projects//makona/results
|--- The current working directory is /home/guest/projects//makona
```

In [20]:

Edit Metadata

```
from IPython.display import IFrame
IFrame('documentation/final_pdfs/9_training_mod_013120__HapNetv2.pdf', width=900, height=300)
```

Out[20]:



In [21]:

Edit Metadata

```
## Reports

from IPython.display import FileLink, FileLinks
FileLinks('makona/results/haplotype_network/')
```

Out[21]:

makona/results/haplotype_network/
Brett424_1_S4_L001_logfileR.txt
study_haplonetwork.png

```
## Reports

from IPython.display import FileLink, FileLinks
FileLinks('makona/results/haplotype_network/')
```

Out[21]:

makona/results/haplotype_network/
Brett424_1_S4_L001_logfileR.txt
study_haplonetwork.png

In [11]:

Edit Metadata

```
from IPython.display import HTML
HTML(filename="./rscript_haplo.nb.html")
```

Out[11]:

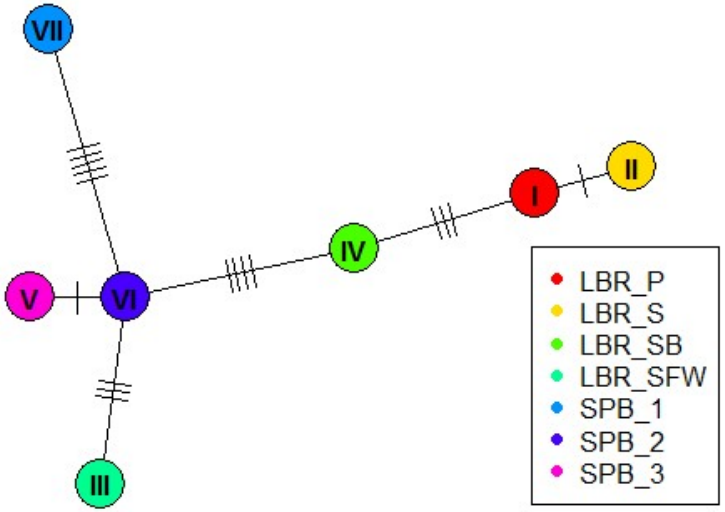
R Notebook

Code

Setting up the enviornment and reading the data and metadata

```
Haplotype network with:
  7 haplotypes
  9 links
  link lengths between 1 and 5 steps

Use print.default() to display all elements.
```

Plot the haplonetwork

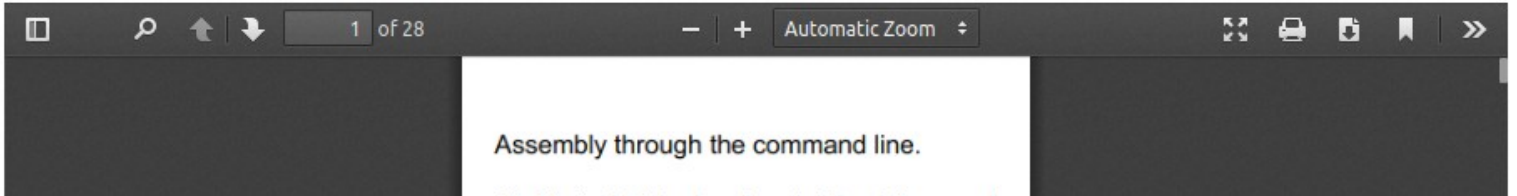


In [10]:

Edit Metadata

```
## References

## Shell

from IPython.display import IFrame
IFrame('documentation/final_pdfs/10_training_mod_013120__CommandLine.pdf', width=900, height=300)
```

Out[10]:

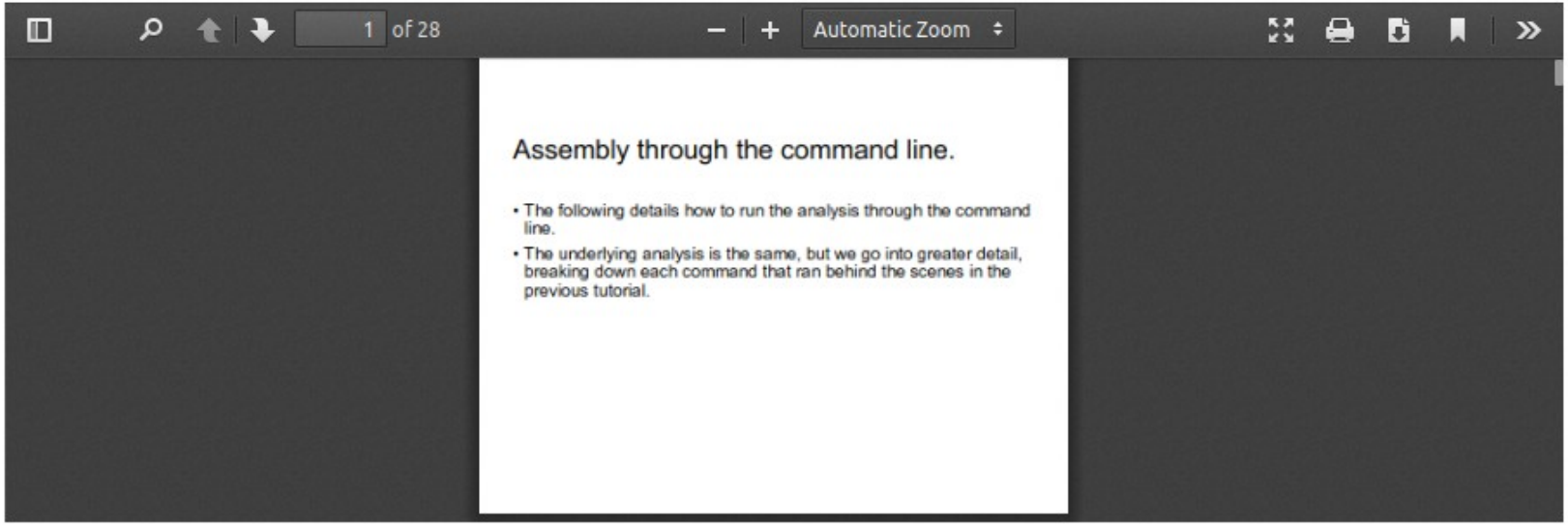


In [10]:

Edit Metadata

```
## References

## Shell

from IPython.display import IFrame
IFrame('documentation/final_pdfs/10_training_mod_013120__CommandLine.pdf', width=900, height=300)
```

Out[10]:



In [ ]:

Edit Metadata