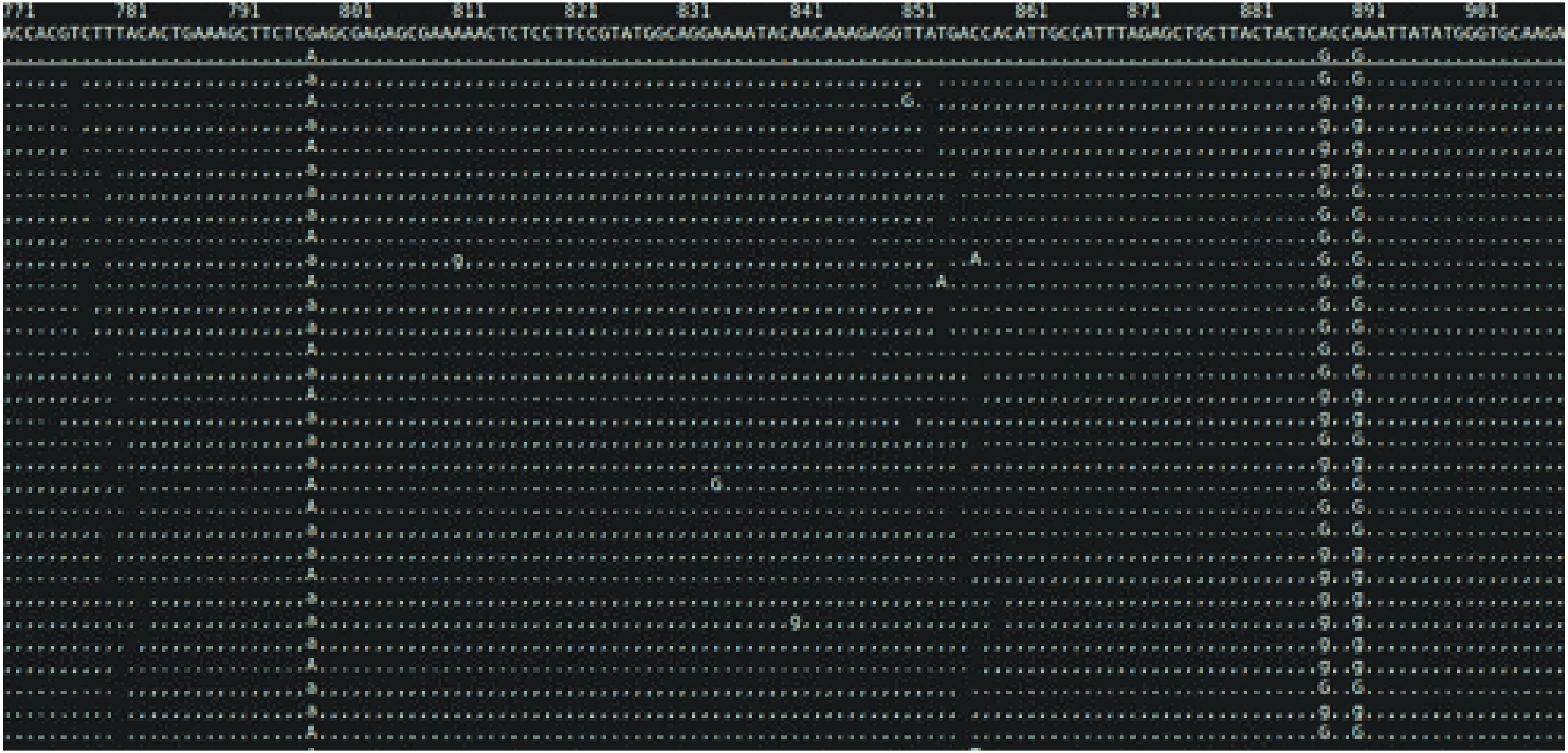
Variation among genomes


771 781 791 801 811 821 831 841 851 861 871 881 891 901
ACCACGTCTTTACACTGAAAGCTTCTCGAGCGAGAGCGAAAAACTCTCCTTCCGTATGGCAGGAAAATACAACAAAGAGGTTATGACCACATTGCCATTTAGAGCTGCTTACTACTCACCAAATTATATGGGTGCAAGA

Comparing genomes

The reference genome

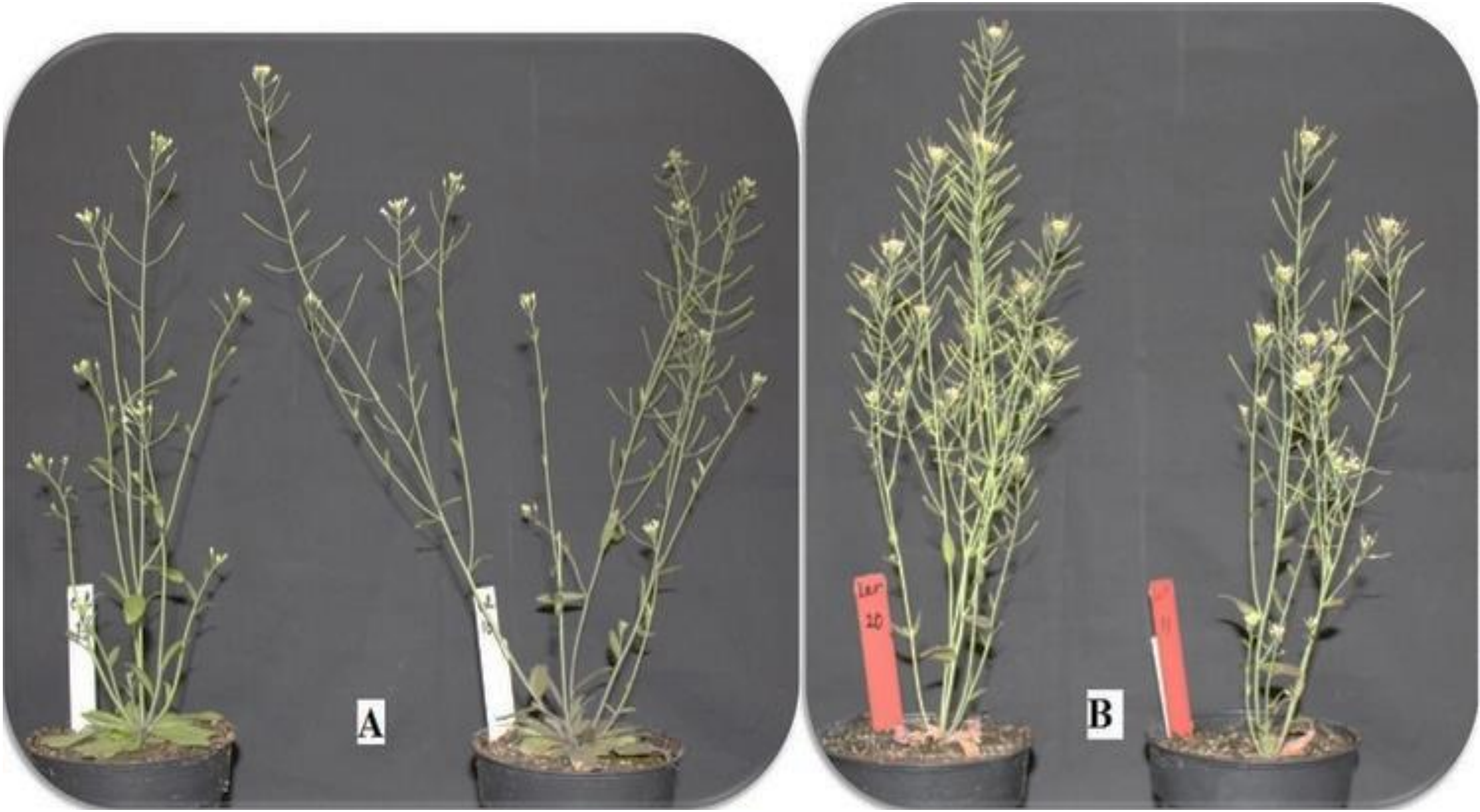
http://www.ncbi.nlm.nih.gov/nuccore/26556996

Arabidopsis thaliana, a model plant

Col-0 variety is from Landsberg, Germany

Ler is a mutant variety from the same population

Col vs. Ler



Moving files between computers

Macs or UNIX –

```
scp genomics2016@128.138.220.248:~/mt.fa ./
```

Windows:

https://winscp.net/eng/index.php

Or other program, e.g. see:

http://www.thegeekstuff.com/2011/06/windows-sftp-scp-clients/

How do we identify differences?

How do we identify differences in the genome of two organisms?

How do we identify differences in the genome of two organisms?

If you know the sequence of one genome...

How do we identify differences in the genome of two organisms?

If you know the sequence of one genome...

1. The first step is to sequence the other genome

How do we identify differences in the genome of two organisms?

If you know the sequence of one genome...

1. The first step is to sequence the other genome

2. The next steps are either

–Assemble that second genome, then compare the two assembled genomes

–OR

–Using the first genome, align the sequences and identify variants

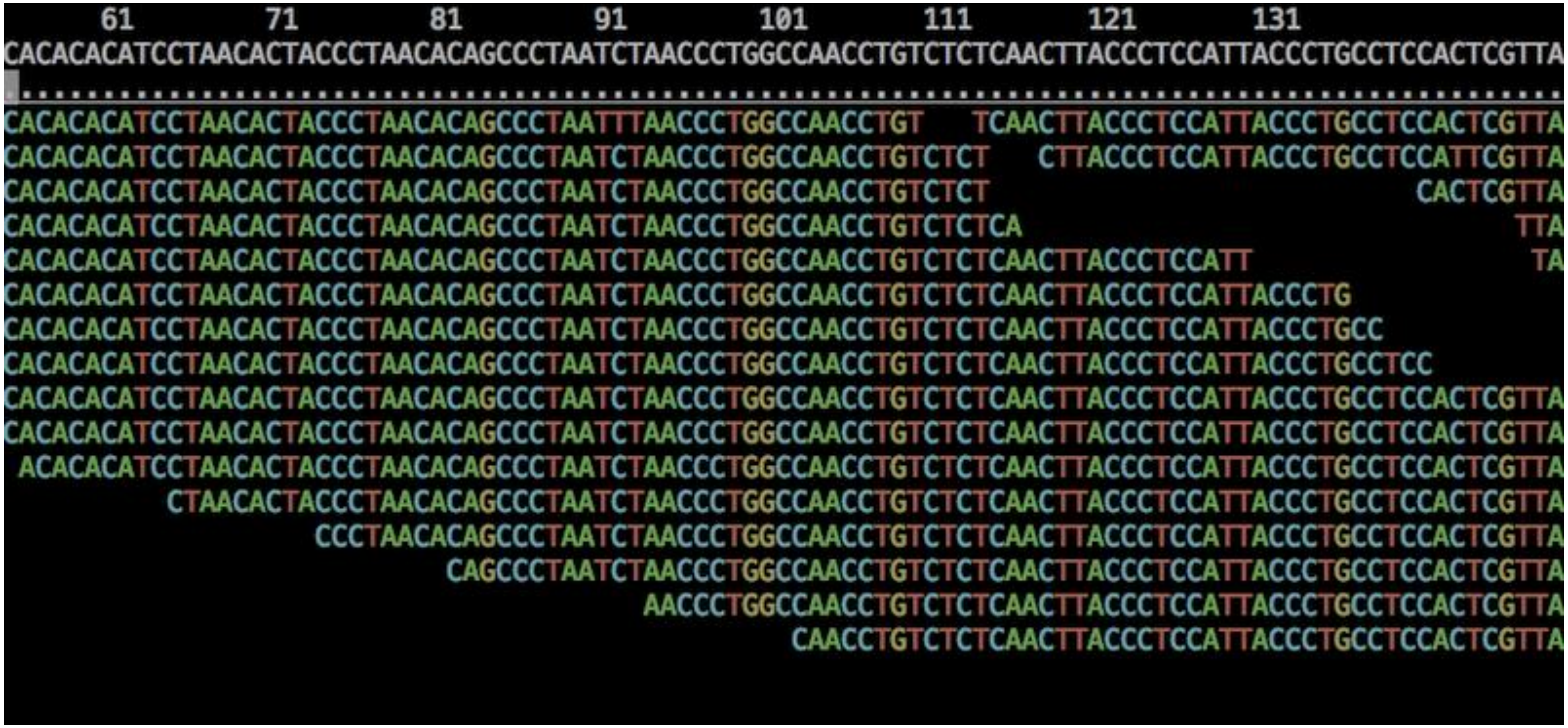
Trimming and cleaning Illumina

- http://www.usadellab.org/cms/index.php?page=trimmomatic

-
```
java -jar /home/nkane/Trimmomatic-0.32/trimmomatic-0.32.jar SE -threads 4 -phred33 ler.fq ler_trimmed.fq LEADING:30 TRAILING:30 MINLEN:35
```

Aligning reads to a reference

The idea is – we have a good genome we can use as a 'reference', and many reads of another related organism we can align to that reference, with the goal of identifying variation



```
      61        71        81        91        101       111       121       131
CACACACATCCTAACACTACCCTAACACAGCCCTAATCTAACCCTGGCCAACCTGTCTCTCAACTTACCCTCCATTACCCTGCCTCCACTCGTTA
...............................................................................................
CACACACATCCTAACACTACCCTAACACAGCCCTAATTTAACCCTGGCCAACCTGT   TCAACTTACCCTCCATTACCCTGCCTCCACTCGTTA
CACACACATCCTAACACTACCCTAACACAGCCCTAATCTAACCCTGGCCAACCTGTCTCT   CTTACCCTCCATTACCCTGCCTCCATTCGTTA
CACACACATCCTAACACTACCCTAACACAGCCCTAATCTAACCCTGGCCAACCTGTCTCT                          CACTCGTTA
CACACACATCCTAACACTACCCTAACACAGCCCTAATCTAACCCTGGCCAACCTGTCTCTCA                              TTA
CACACACATCCTAACACTACCCTAACACAGCCCTAATCTAACCCTGGCCAACCTGTCTCTCAACTTACCCTCCATT                 TA
CACACACATCCTAACACTACCCTAACACAGCCCTAATCTAACCCTGGCCAACCTGTCTCTCAACTTACCCTCCATTACCCTG
CACACACATCCTAACACTACCCTAACACAGCCCTAATCTAACCCTGGCCAACCTGTCTCTCAACTTACCCTCCATTACCCTGCC
CACACACATCCTAACACTACCCTAACACAGCCCTAATCTAACCCTGGCCAACCTGTCTCTCAACTTACCCTCCATTACCCTGCCTCC
CACACACATCCTAACACTACCCTAACACAGCCCTAATCTAACCCTGGCCAACCTGTCTCTCAACTTACCCTCCATTACCCTGCCTCCACTCGTTA
CACACACATCCTAACACTACCCTAACACAGCCCTAATCTAACCCTGGCCAACCTGTCTCTCAACTTACCCTCCATTACCCTGCCTCCACTCGTTA
 ACACACATCCTAACACTACCCTAACACAGCCCTAATCTAACCCTGGCCAACCTGTCTCTCAACTTACCCTCCATTACCCTGCCTCCACTCGTTA
          CTAACACTACCCTAACACAGCCCTAATCTAACCCTGGCCAACCTGTCTCTCAACTTACCCTCCATTACCCTGCCTCCACTCGTTA
                   CCCTAACACAGCCCTAATCTAACCCTGGCCAACCTGTCTCTCAACTTACCCTCCATTACCCTGCCTCCACTCGTTA
                           CAGCCCTAATCTAACCCTGGCCAACCTGTCTCTCAACTTACCCTCCATTACCCTGCCTCCACTCGTTA
                                       AACCCTGGCCAACCTGTCTCTCAACTTACCCTCCATTACCCTGCCTCCACTCGTTA
                                                CAACCTGTCTCTCAACTTACCCTCCATTACCCTGCCTCCACTCGTTA
```

Aligning reads to a reference

- We will be using the program BWA

Aligning reads to a reference

- We will be using the program BWA
- man bwa

BWA

- http://bio-bwa.sourceforge.net/bwa.shtml
- BWA is a software package for mapping low-divergent sequences against a large reference genome, such as the human genome. It consists of three algorithms: BWA-backtrack, BWA-SW and BWA-MEM. The first algorithm is designed for Illumina sequence reads up to 100bp, while the rest two for longer sequences ranged from 70bp to 1Mbp. BWA-MEM and BWA-SW share similar features such as long-read support and split alignment, but BWA-MEM, which is the latest, is generally recommended for high-quality queries as it is faster and more accurate. BWA-MEM also has better performance than BWA-backtrack for 70-100bp Illumina reads.
- For all the algorithms, BWA first needs to construct the FM-index for the reference genome (the index command). Alignment algorithms are invoked with different sub-commands: aln/samse/sampe for BWA-backtrack, bwasw for BWA-SW and mem for the BWA-MEM algorithm.

Aligning reads to a reference

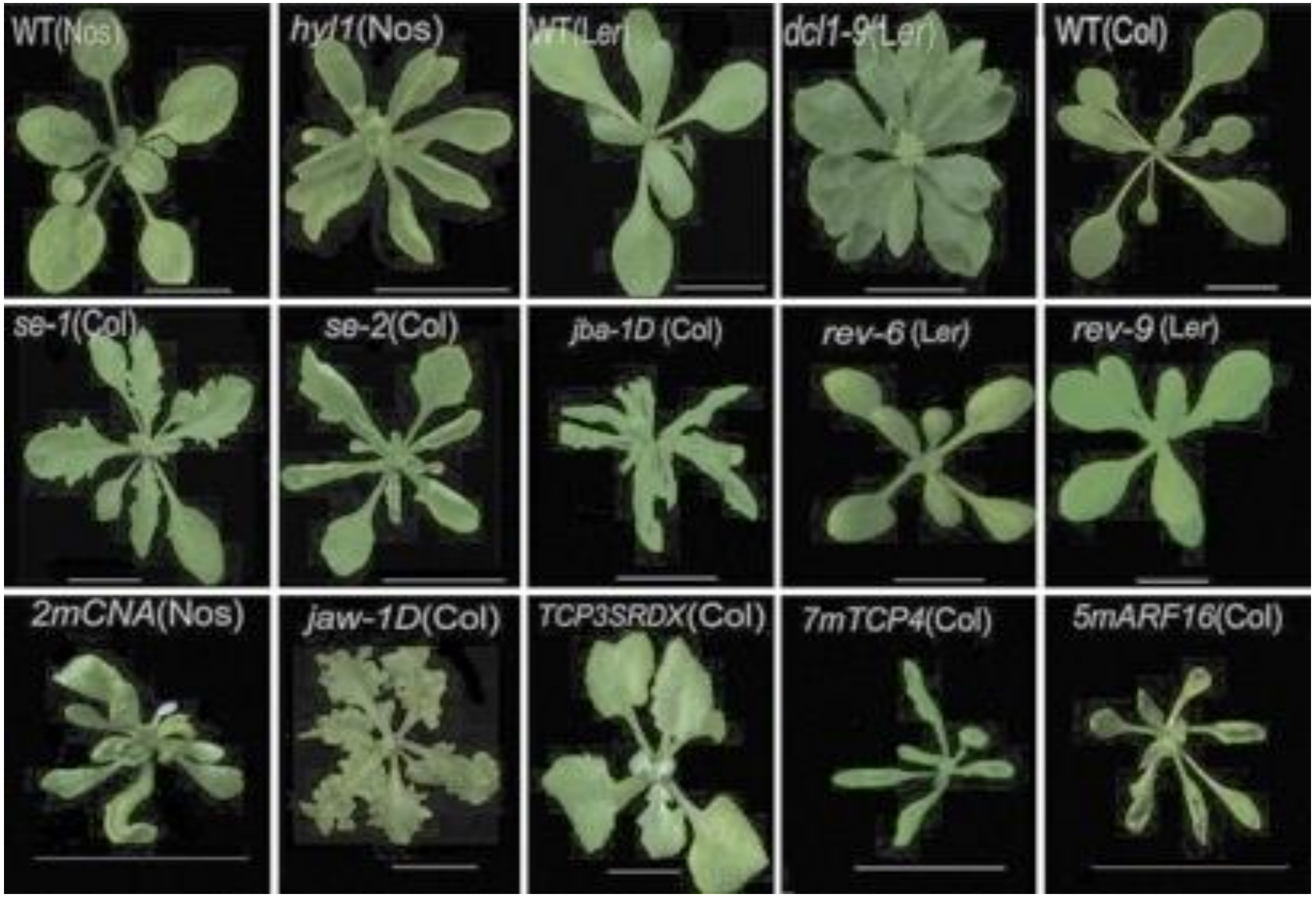
- What are the commands we need to run to do this?

Aligning reads to a reference

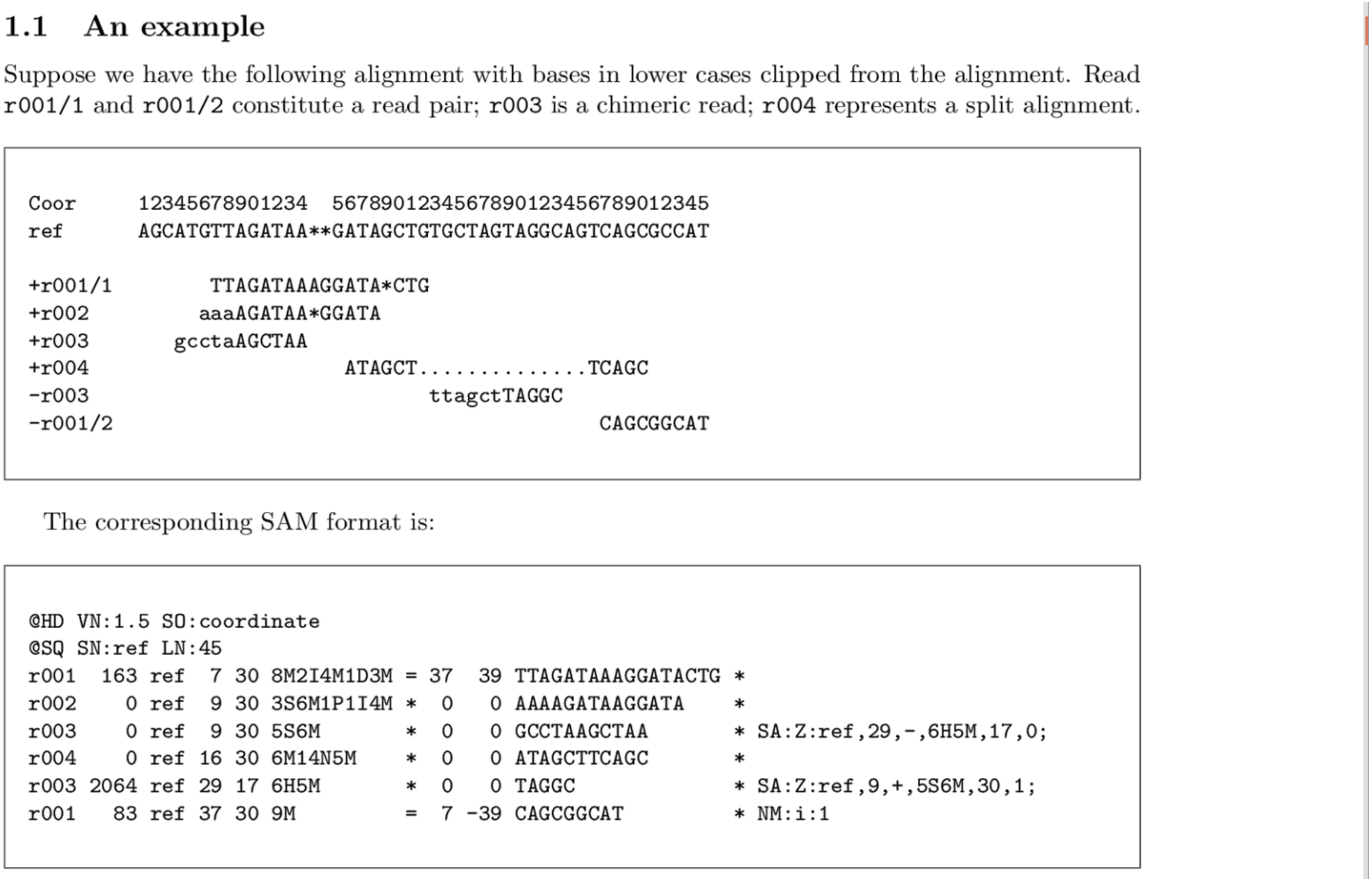
```
bwa index mt.fa

bwa mem mt.fa ler.fq > ler.sam
```

mt.fa is the Col genotype (reference genome)
The second dataset of fastq reads is sequence from the Ler genotype



SAM file



1.1 An example

Suppose we have the following alignment with bases in lower cases clipped from the alignment. Read r001/1 and r001/2 constitute a read pair; r003 is a chimeric read; r004 represents a split alignment.

```
Coor     12345678901234  5678901234567890123456789012345
ref      AGCATGTTAGATAA**GATAGCTGTGCTAGTAGGCAGTCAGCGCCAT

+r001/1        TTAGATAAAGGATA*CTG
+r002         aaaAGATAA*GGATA
+r003       gcctaAGCTAA
+r004                     ATAGCT..............TCAGC
-r003                            ttagctTAGGC
-r001/2                                        CAGCGGCAT
```

The corresponding SAM format is:

```
@HD VN:1.5 SO:coordinate
@SQ SN:ref LN:45
r001   163 ref  7 30 8M2I4M1D3M = 37  39 TTAGATAAAGGATACTG *
r002     0 ref  9 30 3S6M1P1I4M *  0   0 AAAAGATAAGGATA    *
r003     0 ref  9 30 5S6M       *  0   0 GCCTAAGCTAA       * SA:Z:ref,29,-,6H5M,17,0;
r004     0 ref 16 30 6M14N5M    *  0   0 ATAGCTTCAGC       *
r003  2064 ref 29 17 6H5M       *  0   0 TAGGC             * SA:Z:ref,9,+,5S6M,30,1;
r001    83 ref 37 30 9M         =  7 -39 CAGCGGCAT         * NM:i:1
```

SAM file

Basically, a giant table listing each sequence from your fastq file, it's quality scores, header information, and where it aligns to your reference genome

Identifying differences

Now we want to use the aligned sequences to identify and visualize similarities and differences in the fastq file as compared to the reference genome

samtools

http://samtools.sourceforge.net/samtools.shtml

samtools

1. convert sam to bam
2. sort the bam file
3. index the bam file and reference file
4. call SNPs

More Unix!!

wc	word count
wc -l	count lines
ls -thor	list files in reverse order, sorted by time
df -h	how much space is left on my drives
du -h	how much space are directories taking
less -NS	look at files, line numbers, scrolling
mv f1 f2	rename file f1 to file f2

SNP calling against a reference

- BWA
  - Align reads against a reference sequence
- Samtools
  - SNP and indel calling

samtools

1. convert sam to bam

2. sort the bam file

3. index the bam file and reference file

4. call SNPs

SNP and indel calling using samtools

- samtools view -b -o ler.bam -S ler.sam

SNP and indel calling using samtools

```
samtools view -b -o ler.bam -S ler.sam
samtools sort ler.bam ler.sorted
samtools index ler.sorted.bam
samtools faidx mt.fa
```

samtools tview

Allows you to view your reads aligned to the reference!

```
samtools tview ler.sorted.bam mt.fa
```

to navigate in tview:

- –Space bar moves you one screen forward
- –Backspace one screen backwards
- –Arrows scroll up,down,right, left
- –“g” go to a particular position, e.g. by typing “=1000” go to base 1000 in the current

samtools mpileup

Allows you to view alignment information on each base of the reference genome

Can be used to call SNPs or INDELs, or also identify regions of similarity.

mpileup

- samtools mpileup -gf mt.fa ler.sorted.bam > mt.bcf
- bcftools view -vc mt.bcf > snps_indels.vcf

SNP and indel calling using samtoolsq

- samtools view -b -o ler.bam -S ler.sam
- samtools sort ler.bam ler.sorted
- samtools index ler.sorted.bam
- samtools faidx mt.fa
- samtools tview ler.sorted.bam mt.fa
- samtools mpileup -uf mt.fa ler.sorted.bam | bcftools vie
-vcg - > ler_snps_indels.vcf
- less -S ler_snps_indels.vcf

Now we have some large files

- cat
  - Print a file or files line by line
- less
  - Display a file so you can scroll through it
- head -n X
  - Print X lines from the beginning of the file
- tail -n X
  - Prints X lines from the end of the file

Finding what you want in a large file

- grep
  - Search for a string of characters
  - e.g.

```
grep 'word' filename
```

Finding what you want in a large file

- grep
  - Search for a string of characters

```
        grep '#' ler_snps_indels.vcf
        grep -c '#' ler_snps_indels.vcf
     grep -v '##' ler_snps_indels.vcf > ler_snps_indels.txt
```

  - Useful for filtering out lines that you want / don't want in a file, as well as counting, etc.

- Use grep to remove INDELS from ler_snps_indels.txt
- Save it as allsnps.txt

Even more Unix!!!!

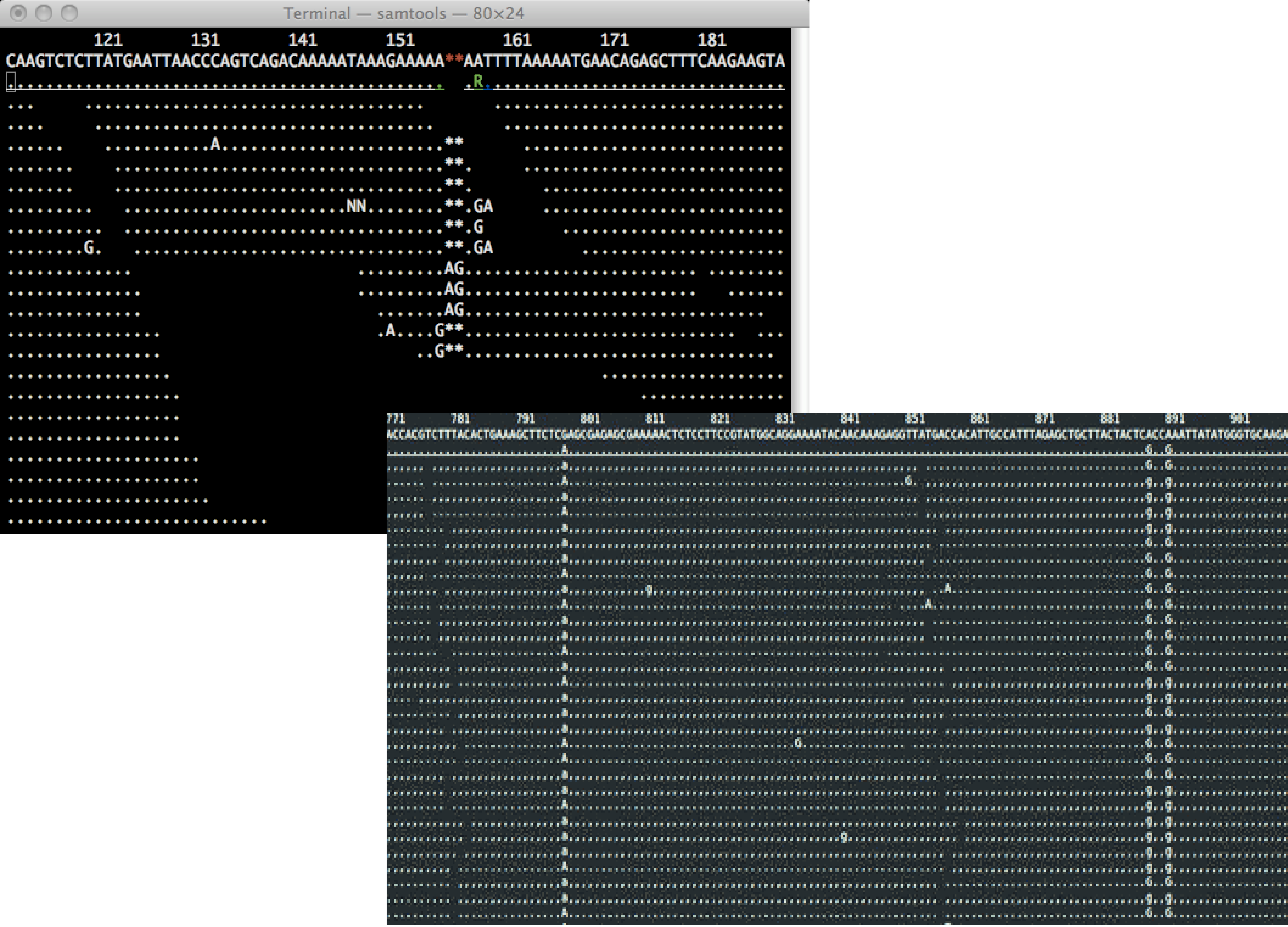
- head -n X
  - Print X lines from the beginning of the file
- tail -n X
  - Prints X lines from the end of the file
- grep
  - Search for a string of characters

```
grep '#' ler_snps_indels.vcf
grep -c '#' ler_snps_indels.vcf
grep -v '##' ler_snps_indels.vcf > allsnps.txt
```

  - Useful for filtering out lines that you want / don't want in a file, as well as counting, etc.

SNP and indel calling using samtoolsq

- samtools view -b -o ler.bam -S ler.sam
- samtools sort ler.bam ler.sorted
- samtools index ler.sorted.bam
- samtools faidx mt.fa
- samtools tview ler.sorted.bam mt.fa
- samtools mpileup -uf mt.fa ler.sorted.bam | bcftools vie
-vcg - > ler_snps_indels.vcf
- less -S ler_snps_indels.vcf
- grep -v '##' ler_snps_indels.vcf > ler_snps_indels.txt



Terminal — samtools — 80×24

```
         121       131       141       151         161       171       181
CAAGTCTCTTATGAATTAACCCAGTCAGACAAAAATAAAGAAAAA**AATTTTAAAAATGAACAGAGCTTTCAAGAAGTA
.............................................  .R...............................
...     ...................................       ..............................
....     ...................................       .............................
......    ...........A.......................**      ...........................
.......    ..................................**.     ...........................
.......    ..................................**.       .........................
.........   .......................NN........**.GA     .........................
..........  .................................**.G        .......................
........G.   ................................**.GA         .....................
.............                       .........AG........................  .......
..............                      .........AG........................   ......
..............                        .......AG...............................
................                      .A....G**............................  ...
................                          ..G**................................
.................                                            ...................
..................                                               ...............
..................
..................
....................
....................
.....................
...........................
```

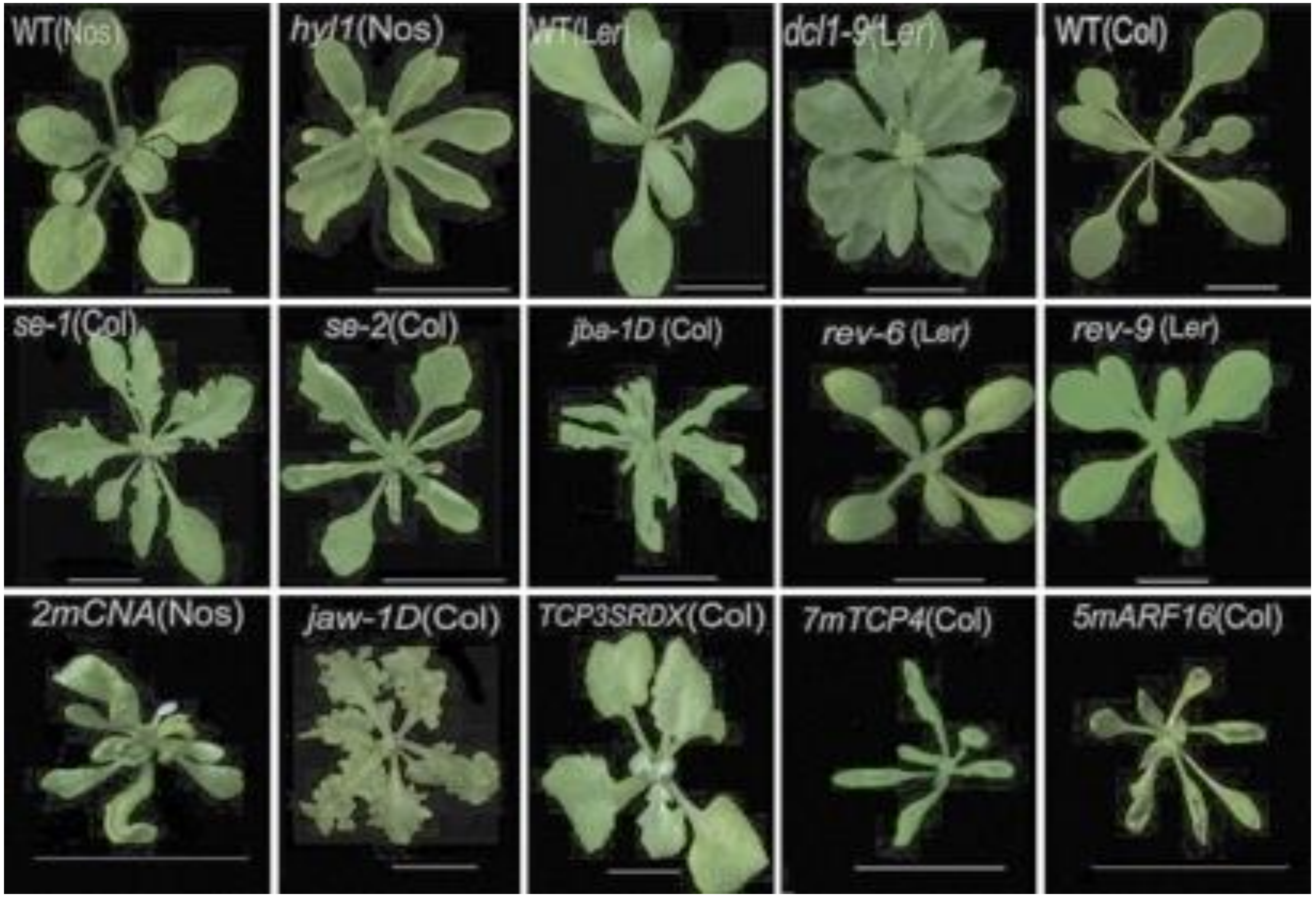
```
771       781       791       801       811       821       831       841       851       861       871       881       891       901
ACCACGTCTTTACACTGAAAGCTTCTCGAGCGAGAGCGAAAAACTCTCCTTCCGTATGGCAGGAAAATACAACAAAGAGGTTATGACCACATTGCCATTTAGAGCTGCTTACTACTCACCAAATTATATGGGTGCAAGA
...........................A..........................................................................................G..G.................
```

Aligning reads to a reference

```
bwa index mt.fa

bwa mem mt.fa trimmed.fq > ler.sam
```

mt.fa is the Col genotype (reference genome)
The second dataset of fastq reads is sequence from the Ler genotype



SNP and indel calling using samtoolsq

- samtools view -b -o ler.bam -S ler.sam
- samtools sort ler.bam ler.sorted
- samtools index ler.sorted.bam
- samtools faidx mt.fa
- samtools tview ler.sorted.bam mt.fa
- samtools mpileup -uf mt.fa ler.sorted.bam | bcftools vie
-vcg - > ler_snps_indels.vcf
- less -S ler_snps_indels.vcf

Even more Unix!!!!

- head -n X
  - Print X lines from the beginning of the file
- tail -n X
  - Prints X lines from the end of the file
- grep
  - Search for a string of characters

```
grep '#' ler_snps_indels.vcf
grep -c '#' ler_snps_indels.vcf
grep -v '##' ler_snps_indels.vcf > allsnps.txt
```

  - Useful for filtering out lines that you want / don't want in a file, as well as counting, etc.