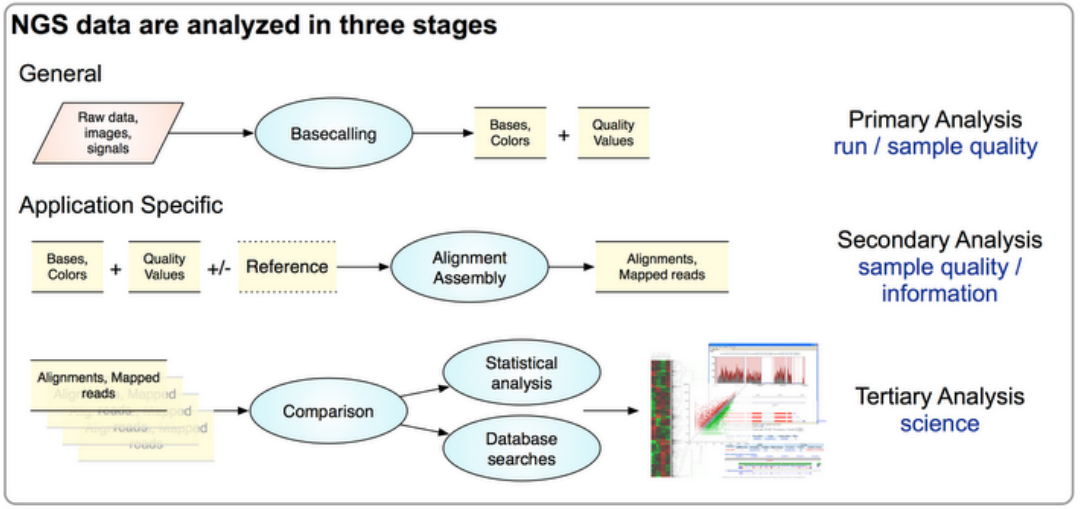
Data formats

Data formats

- There are many different types of file formats depending on
  - Type of information they contain
    - Raw Sequence files
    - Co-ordinate files
    - Parameter files
    - Annotation files
    - Metadata files
  - Sequencing platform
  - Analysis stage
  - Data source



Data formats

- Formats are designed to hold sequence data and other information about sequence
- All Sequence formats are ASCII text containing sequence ID, Quality Scores, Annotation details, comments, and other descriptions about sequence
- **FastA** format (everybody knows about it)
  - Header line starts with “>” followed by a sequence ID
  - Sequence (string of nt).

```
>gi|5524211|gb|AAD44166.1| cytochrome b [Elephas maximus maximus]
LCLYTHIGRNIYYGSYLYSETWNTGIMLLLITMATAFMGYVLPWGQMSFWGATVITNLFSAIPYIGTNLV
EWIWGGFSVDKATLNRFFAFHFILPFTMVALAGVHLTFLHETGSNNPLGLTSDSDKIPFHPYYTIKDFLG
LLILILLLLLLALLSPDMLGDPDNHMPADPLNTPLHIKPEWYFLFAYAILRSVPNKLGGVLALFLSIVIL
GLMPFLHTSKHRSMMLRPLSQALFWTLTMDLLTLTWIGSQPVEYPYTIIGQMASILYFSIILAFLPIAGX
IENY
```

Data formats

Fastq format

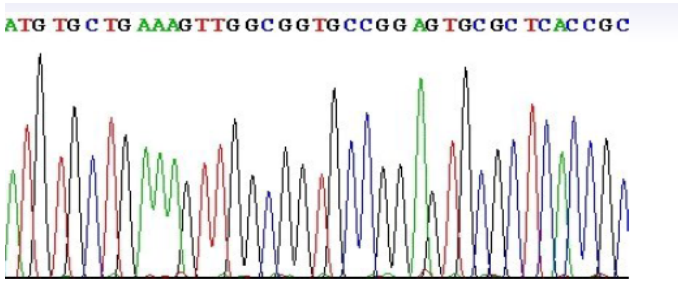
- Output of most actual sequencing platforms for raw data
- FASTQ files are uncompressed and quite large because they contain the following information for every single sequencing read:
  1. @ followed by the read ID and possibly information about the sequencing run
  2. sequenced bases
  3. + (perhaps followed by the read ID again, or some other description)
  4. quality scores for each base of the sequence (ASCII-translated Phred scores)

```
@Seq description
GATTTGGGGTTCAAAGCAGTATCGATCAAATAGTAAATCCATTTGTTCAACTCACAGTTT
+
!''*((((***+))%%%++)(%%%%).1***-+*''))**55CCF>>>>>>CCCCCCC65
```

Data formats

Fastq format

Phred Scores



- Sequencing systems assign quality scores to each peak, that represents the error probability that an individual base call is incorrect.
- Phred scores provide log(10)-transformed error probability values:

If p is probability that the base call is wrong the Phred score is

$$Q = .10 \cdot log_{10} p$$

PHRED Quality Score	Probability of incorrect base call	Base call accuracy
10	1 in 10	90 %
20	1 in 100	99 %
30	1 in 1000	99.9 %
40	1 in 10000	99.99 %
50	1 in 100000	99.999 %

- The base calling (A, T, G or C) is performed based on Phred scores.
- Ambiguous positions with Phred scores <= 20 are labeled with N.
- To assign each base a unique score identifier (instead of numbers of varying character length), Phred scores are typically represented as ASCII characters.

Data formats

BAM/SAM formats

- The Sequence Alignment/Map (SAM) format is a generic nucleotide alignment format that describes the alignment of sequencing reads to a reference.
- SAM files typically contain a short header section with information about the genomic loci of each read and a very long alignment section where each row represents a single read alignment. For each read, there are 11 mandatory fields that always appear in the same order:

  <QNAME> <FLAG> <RNAME> <POS> <MAPQ> <CIGAR> <MRNM> <MPOS> <ISIZE> <SEQ> <QUAL>

- BAM is a dedicated binary format including the compressed SAM. It enables fast access to data without having to “unzip” the whole file. For the typically large data, BAM is currently the most recommended and most “standard” format.

```
@HD    VN:1.0
@SQ    SN:chr20 LN:62435964
@RG    ID:L1 PU:SC_1_10 LB:SC_1 SM:NA12891
@RG    ID:L2 PU:SC_2_12 LB:SC_2 SM:NA12891
read_28833_29006_6945 99  chr20 28833 20 10M1D25M = 28993 195 \
       AGCTTAGCTAGCTACCTATATCTTGGTCTTGGCCG <<<<<<<<<<<<<<<<<<<<:<9/,&,22;;<<< \
       NM:i:1 RG:Z:L1
read_28701_28881_323b 147 chr20 28834 30 35M      = 28701 -168 \
       ACCTATATCTTGGCCTTGGCCGATGCGGCCTTGCA <<<<<;<<<<7;:<<<6;<<<<<<<<<<<7<<<< \
       MF:i:18 RG:Z:L2
```

Data formats

BAM/SAM formats

Mandatory Alignment Section Fields

Position	Field	Description
1	QNAME	Query template (or read) name
2	FLAG	Information about read mapping (see next section)
3	RNAME	Reference sequence name. This should match a @SQ line in the header.
4	POS	1-based leftmost mapping position of the first matching base. Set as 0 for an unmapped read without coordinate.
5	MAPQ	Mapping quality of the alignment. Based on base qualities of the mapped read.
6	CIGAR	Detailed information about the alignment (see relevant section).
7	RNEXT	Used for paired end reads. Reference sequence name of the next read. Set to "=" if the next segment has the same name.
8	PNEXT	Used for paired end reads. Position of the next read.
9	TLEN	Observed template length. Used for paired end reads and is defined by the length of the reference aligned to.
10	SEQ	The sequence of the aligned read.
11	QUAL	ASCII of base quality plus 33 (same as the quality string in the Sanger FASTQ format).
12	OPT	Optional fields (see relevant section).

Data formats

Formats for genome annotations (BED, GFF)

- One line per genomic feature
- The BED format is the simplest way to store annotation tracks. It has three required fields (chromosome, start, end) and up to 9 optional fields (name, score, strand, thickStart, thickEnd, itemRgb, blockCount, blockSizes, blockStarts).

```
# 6-column BED file defining transcript loci
chr1   66999824   67210768   NM_032291 0 +
chr1   33546713   33586132   NM_052998 0 +
chr1   25071759   25170815   NM_013943 0 +
chr1   48998526   50489626   NM_032785 0 -
```

- The General Feature Format (GFF) has nine required fields; the first three fields form the basic name, start, end tuple that allows for the identification of the location in respect to the reference genome.

```
ctg123 . operon          1300 15000  .  +  .  ID=operon001;Name=superOperon
ctg123 . mRNA            1300  9000  .  +  .  ID=mrna0001;Parent=operon001;Name=sonichedgehog
ctg123 . exon            1300  1500  .  +  .  Parent=mrna0001
ctg123 . exon            1050  1500  .  +  .  Parent=mrna0001
ctg123 . exon            3000  3902  .  +  .  Parent=mrna0001
ctg123 . exon            5000  5500  .  +  .  Parent=mrna0001
```

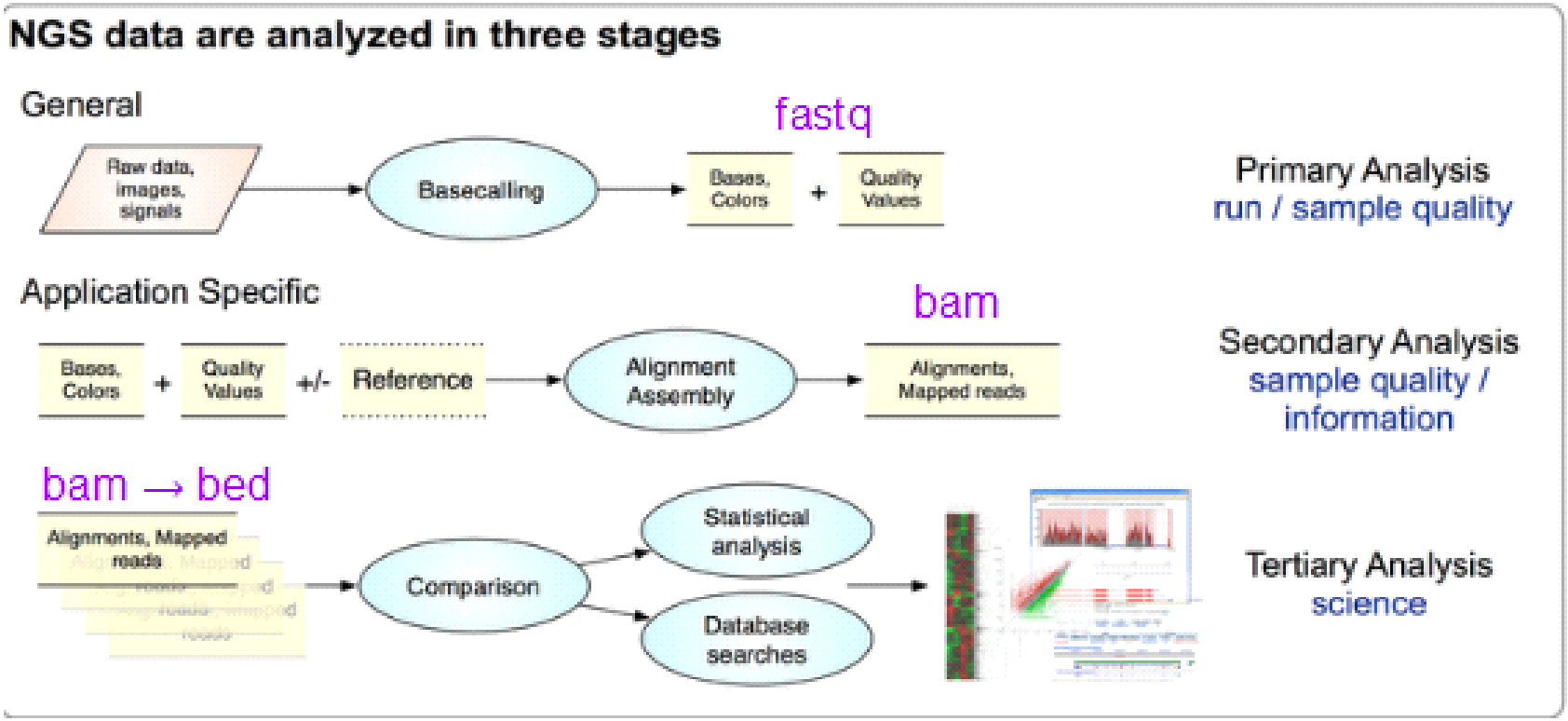
Data formats

Formats for genome annotations (BED, GFF)

1. **reference sequence**: coordinate system of the annotation (e.g., "Chr1")
2. **source**: describes how the annotation was derived (e.g., the name of the annotation software)
3. **method**: annotation type (e.g., gene)
4. **start position**: 1-based integer, always less than or equal to the stop position
5. **stop position**: for zero-length features, such as insertion sites, start equals end and the implied site is to the right of the indicated base
6. **score**: e.g., sequence identity
7. **strand**: "+" for the forward strand, "-" for the reverse strand, or "." for annotations that are not stranded
8. **phase**: codon phase for annotations linked to proteins; 0, 1, or 2, indicating the frame, or the number of bases that should be removed from the beginning of this feature to reach the first base of the next codon
9. **group**: contains the class and ID of an annotation which is the logical parent of the current one ("feature is composed of")

Data formats

So, in summary…



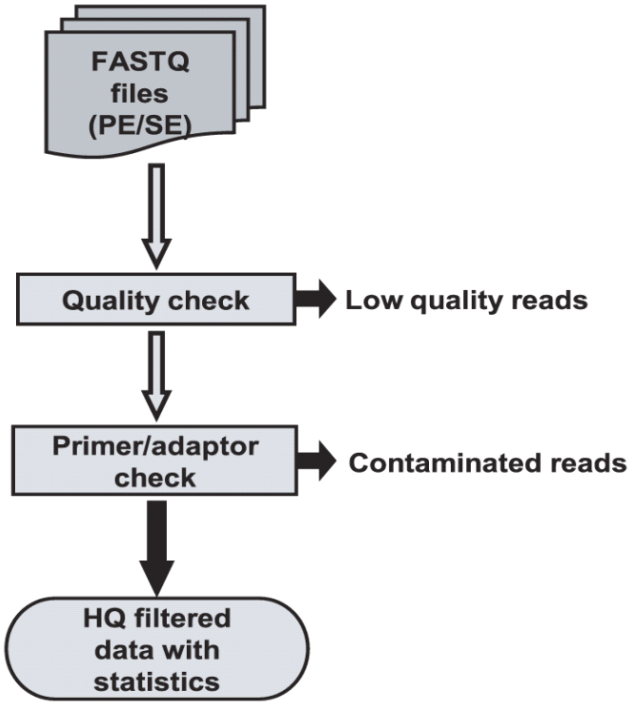
- **For base-call data, FASTQ (Sanger, Phred)**
- **For read alignments, SAM/BAM/MAQ format**
- **For annotation results, GFF or BED format**

Quality control and Preprocessing of NGS data

Quality Control and Preprocessing

QC analysis of sequence data is extremely important for meaningful downstream analysis

- To analyze problems in quality scores/ statistics of sequencing data
- To check whether further analysis with sequence is possible
- To remove redundancy (filtering)
- To remove low quality reads from analysis
- To remove adapter contamination



Highly efficient and fast processing tools are required to handle large volume of datasets

Quality Control

FastQC tool

http://www.bioinformatics.babraham.ac.uk/projects/fastqc/

- Basic statistics
- Quality- Per base position
- Per Sequence Quality Distribution
- Nucleotide content per position
- Per sequence GC distribution
- Per base GC distribution
- Per base N content
- Length Distribution
- Overrepresented/ duplicated sequences
- K-mer content

Quality Control

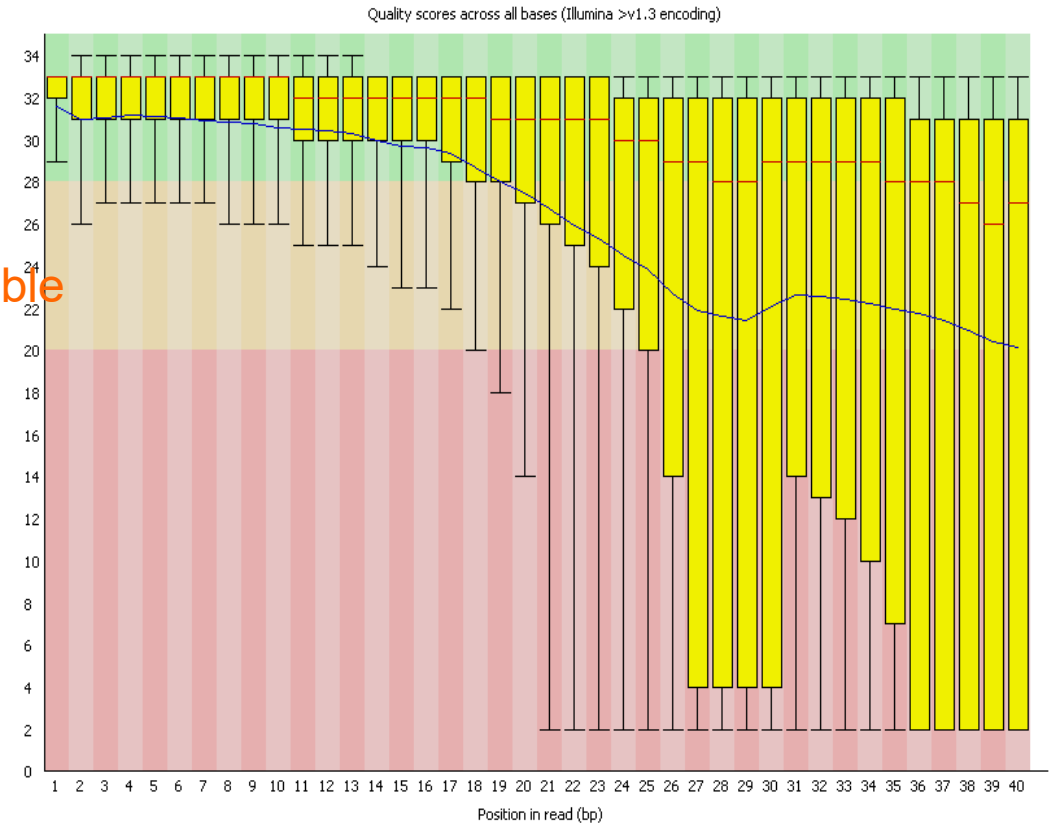
FastQC

Per base sequence quality (Boxplot)

Good

Acceptable

Bad

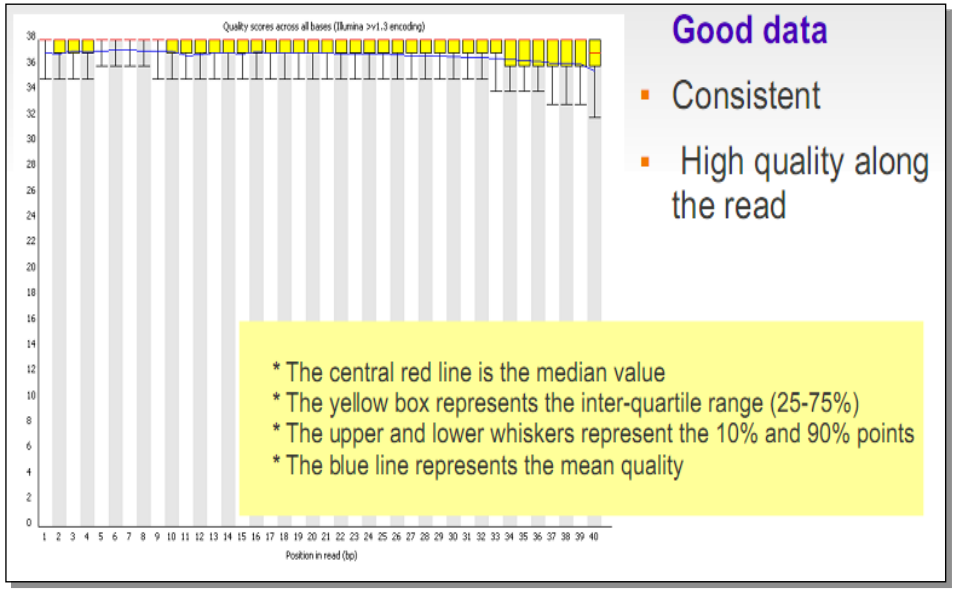


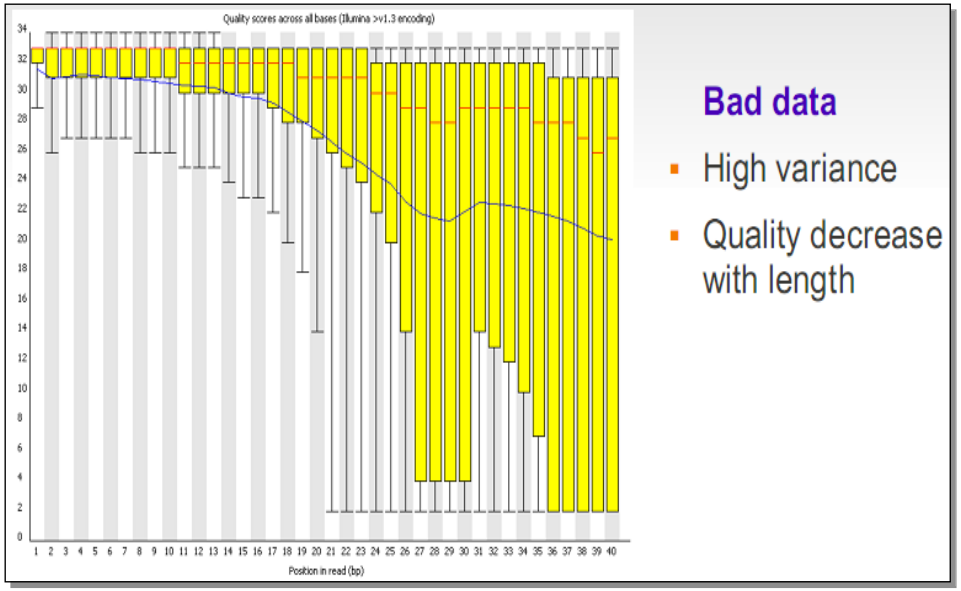
shows an overview of the range of quality values across all bases at each position in the FastQ file

Quality Control

FastQC

Per base sequence quality (Boxplot)

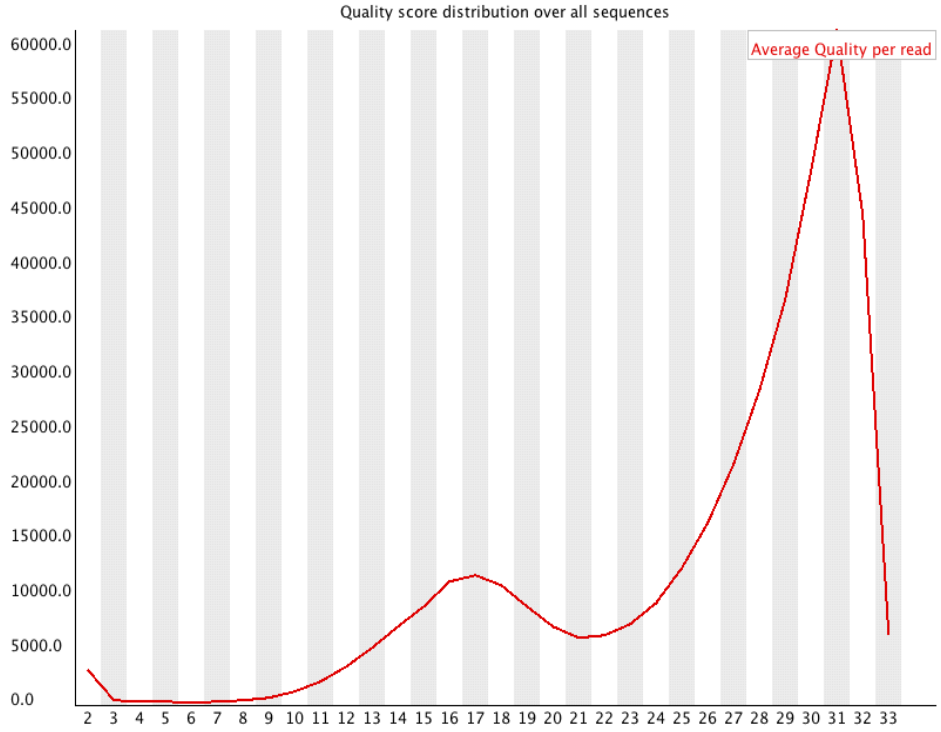




Quality Control

FastQC

Per sequence quality scores

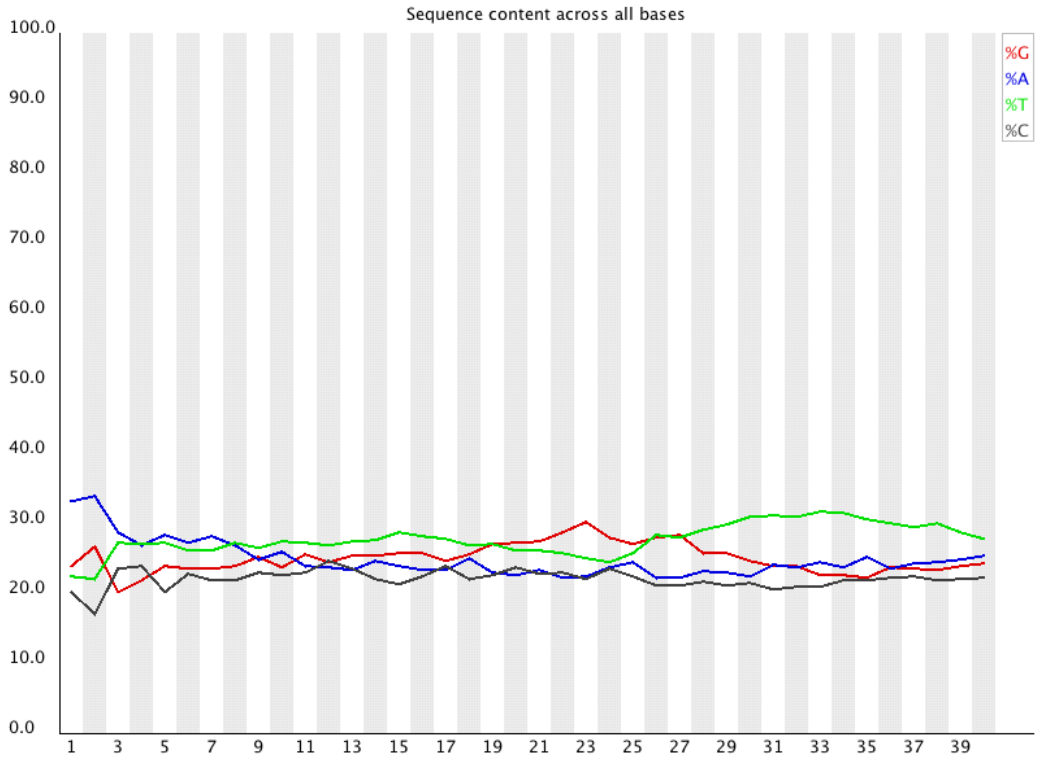


allows you to see if a subset of your sequences have universally low quality values.

Quality Control

FastQC

Per base sequence content



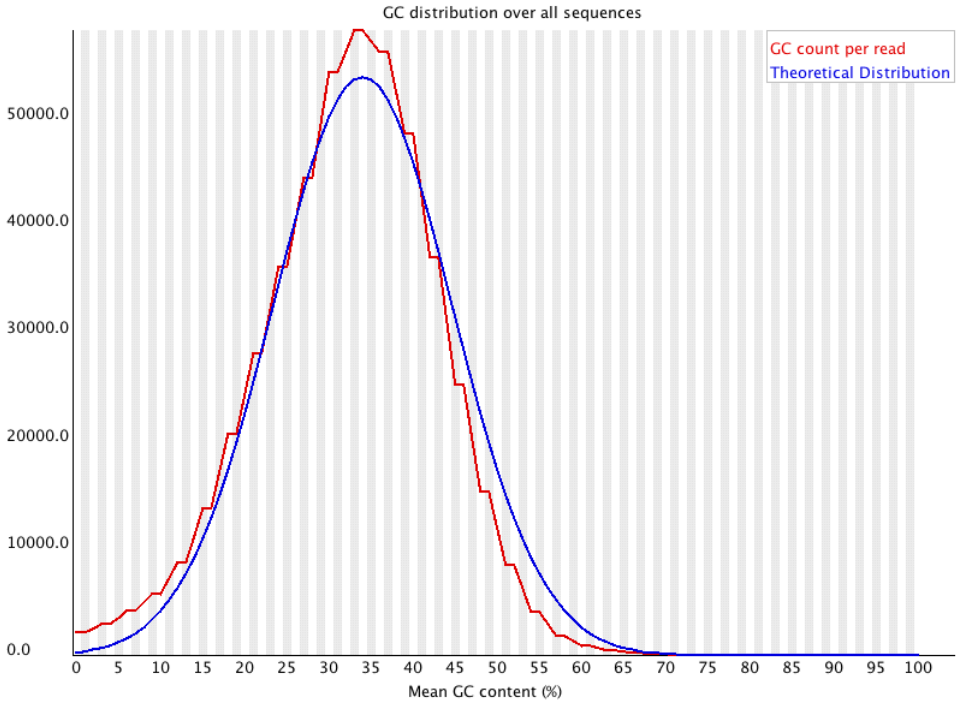
Per Base Sequence Content plots out the proportion of each base position in a file for which each of the four normal DNA bases has been called

- In a random library you would expect that there would be little to no difference between the different bases of a sequence run, so the lines in this plot should run parallel with each other.

Quality Control

FastQC

Per sequence GC content



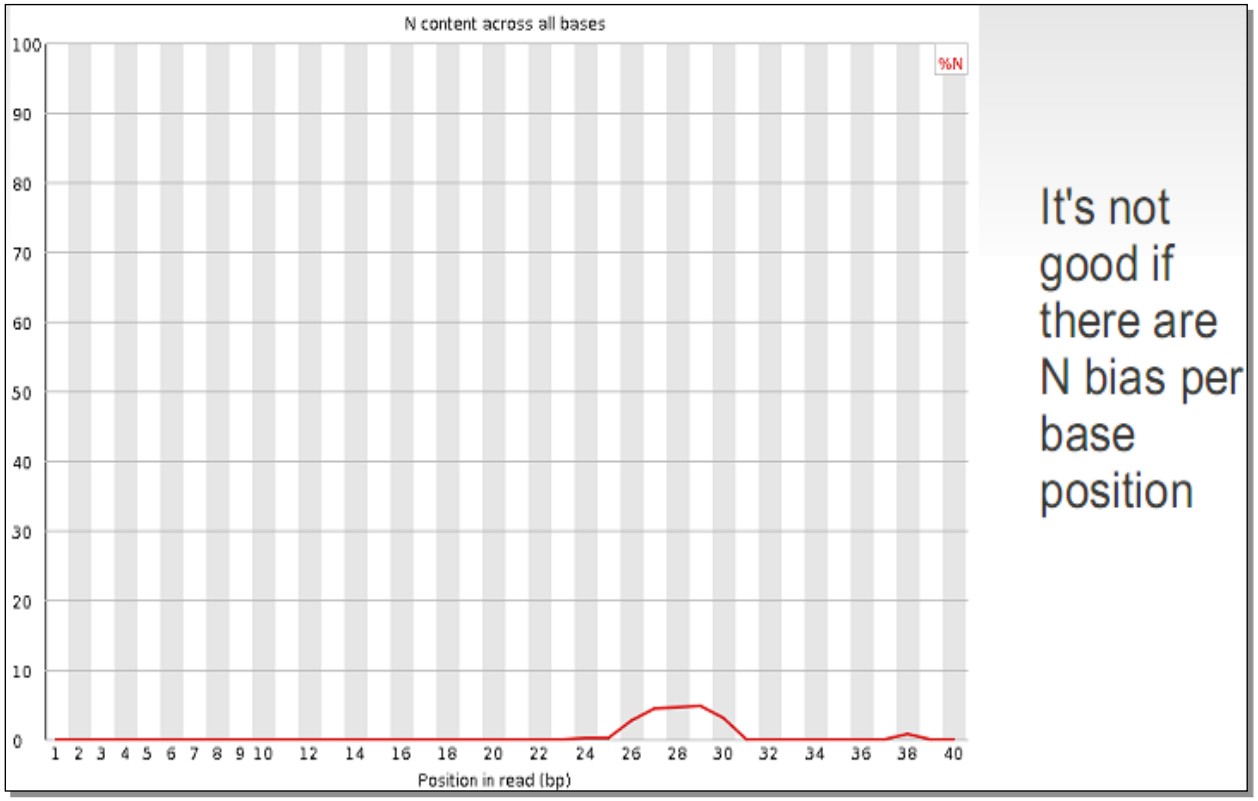
measures the GC content across the whole length of each sequence in a file and compares it to a modelled normal distribution of GC content

- you would expect to see a roughly normal distribution of GC content
- An unusually shaped distribution could indicate a contaminated library or some other kinds of biased subset.

Quality Control

FastQC

Per base N content

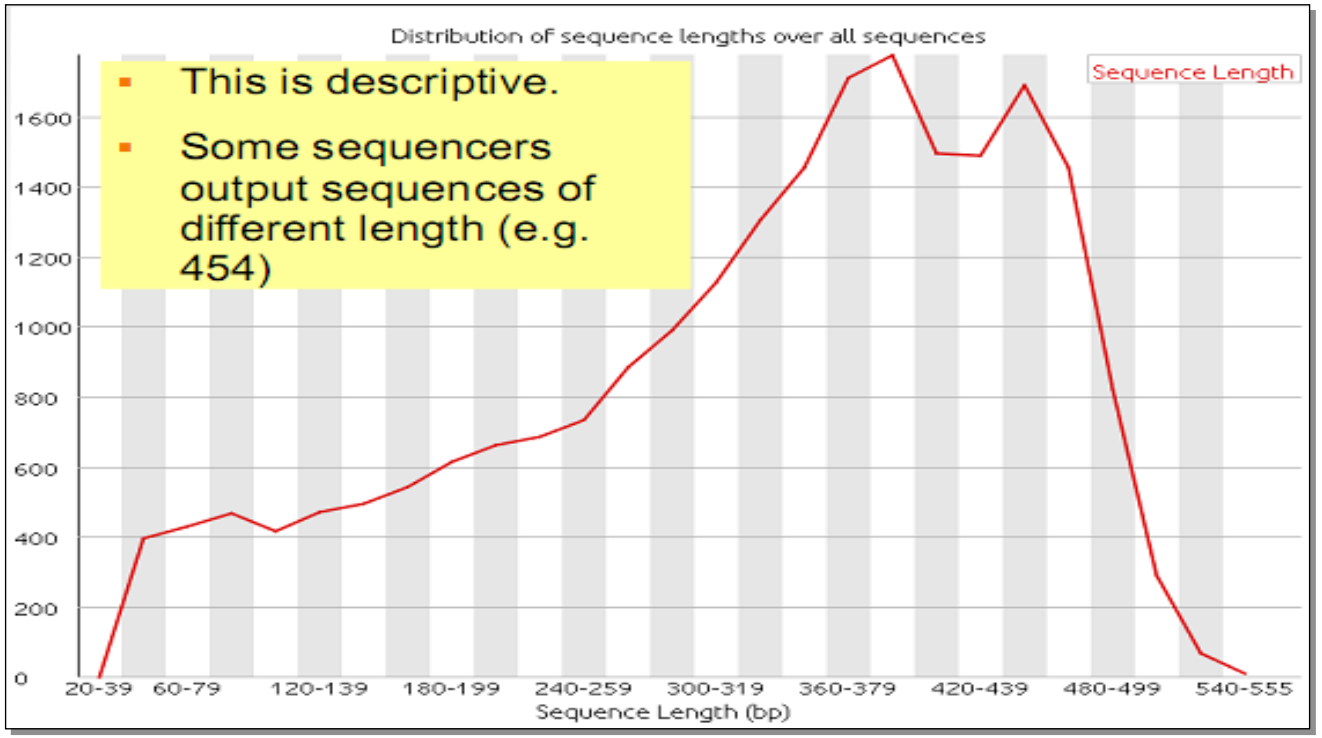


If a sequencer is unable to make a base call with sufficient confidence then it will normally substitute an N rather than a conventional base call. It plots out the percentage of base calls at each position for which an N was called.

Quality Control

FastQC

Sequence length distribution

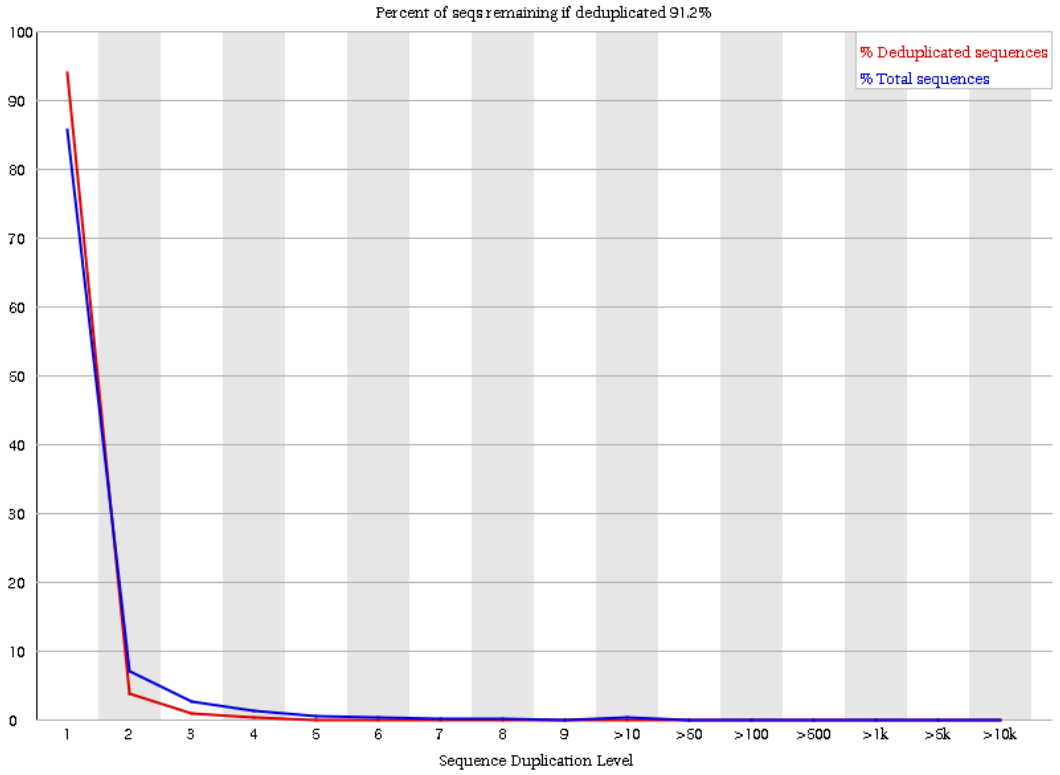


In many cases it will produce a simple graph showing a peak only at one size, but for variable length FASTQ files, it will show the relative amounts of each different size of sequence fragment.

Quality Control

FastQC

Sequence duplication level



Counts the degree of duplication for every sequence. Too many duplicate regions in the sequence may indicate contamination or technical problems

Quality Control

FastQC

Overrepresented sequences

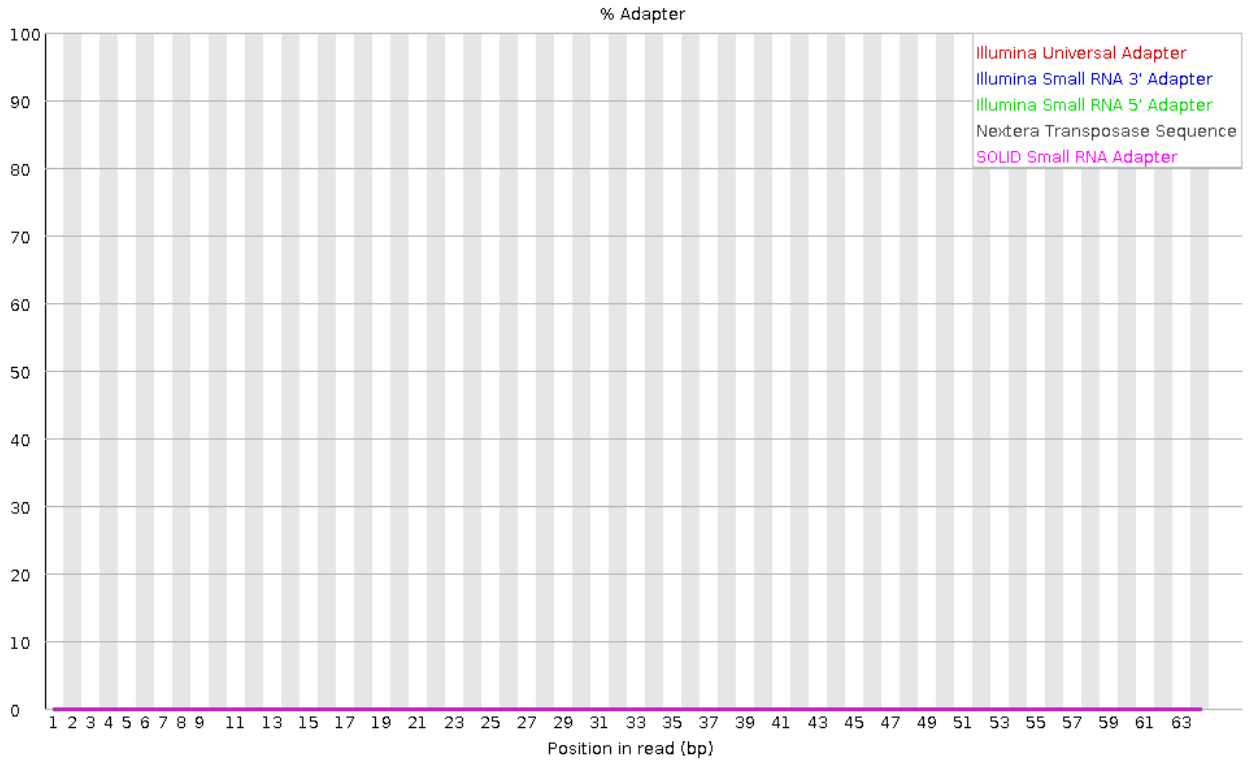
Sequence	Count	Percentage	Possible Source
AAGATCCGAGTCGTCCGGAAATCCATTGCCCGTGTTCTCACAGTTATTAA	432	0.43585733743631133	No Hit
AGATCCGAGTCGTCCGGAAATCCATTGCCCGTGTTCTCACAGTTATTAAC	335	0.33799122231750994	No Hit
TGGCAGAAGTAGAGCAGAAGAAGAAGCGGACCTTCCGCAAGTTCACCTAC	250	0.25223225546082834	No Hit
CAGAAGTAGAGCAGAAGAAGAAGCGGACCTTCCGCAAGTTCACCTACCGC	237	0.23911617817686526	No Hit
GTAGAGCAGAAGAAGAAGCGGACCTTCCGCAAGTTCACCTACCGCGGCGT	223	0.22499117187105888	No Hit
AAGAAATCTGACCCGGTCGTCTCGTACCGCGAGACGGTCAGTGAAGAGTC	204	0.2058215204560359	No Hit
AAGTAGAGCAGAAGAAGAAGCGGACCTTCCGCAAGTTCACCTACCGCGGC	151	0.1523482822983403	No Hit
CACCTGGAGATCTGCCTGAAGGACCTGGAGGAGGACCACGCCTGCATCCC	147	0.14831256621096706	No Hit
TCTGCCTGAAGGACCTGGAGGAGGACCACGCCTGCATCCCCATCAAGAAA	146	0.14730363718912376	No Hit

Lists all of the sequence which make up more than 0.1% of the total. Finding that a single sequence is very overrepresented in the set either means that is highly biologically significant, or that the library is contaminated. For each overrepresented sequence it will look for matches in a database of common contaminants.

Quality Control

FastQC

Adapter content



Does a generic analysis of all the Kmers in the library to find those that don't have even coverage through the length of the reads.

Quality Control

FastQC

- Good (Illumina) quality data:

http://www.bioinformatics.babraham.ac.uk/projects/fastqc/good_sequence_short_fastqc.html

- Bad (Illumina) quality data:

http://www.bioinformatics.babraham.ac.uk/projects/fastqc/bad_sequence_fastqc.html

Preprocessing raw data

Removing technical artifacts

Adjusting biases

etc

Preprocessing raw data

Based on the information provided by the QC graphs, the sequences may be treated to reduce bias in downstream analysis by:

- Filtering sequences
  - with low mean quality score
  - too short
  - with too many ambiguous (N) bases
  - based on their GC content
- Cutting/Trimming/masking sequences
  - from low quality score regions
  - beginning/end of sequence
  - removing adapters, primers

But, removing duplicate reads is not advised since high expressed genes can have genuine duplicate reads that are not due to the PCR amplification step.

Preprocessing raw data

After preprocessing, run QC again.

If too bad quality, it may be advisable to rerun the experiment...

Introduction to Galaxy

A web-based genome analysis platform

Galaxy

- An open, web-based platform integrating many popular tools and resources for intensive biomedical research.
- **What can be done?**
  - Obtain data from many data sources like UCSC Table Browser, Biomart, WormBase, or your own data
  - Prepare data for further analysis by rearranging or cutting data columns, filtering data and many other options
  - Analyze data by finding overlapping regions, determining statistics, preprocessing NGS data and much more
  - Share data and workflows

https://usegalaxy.org/

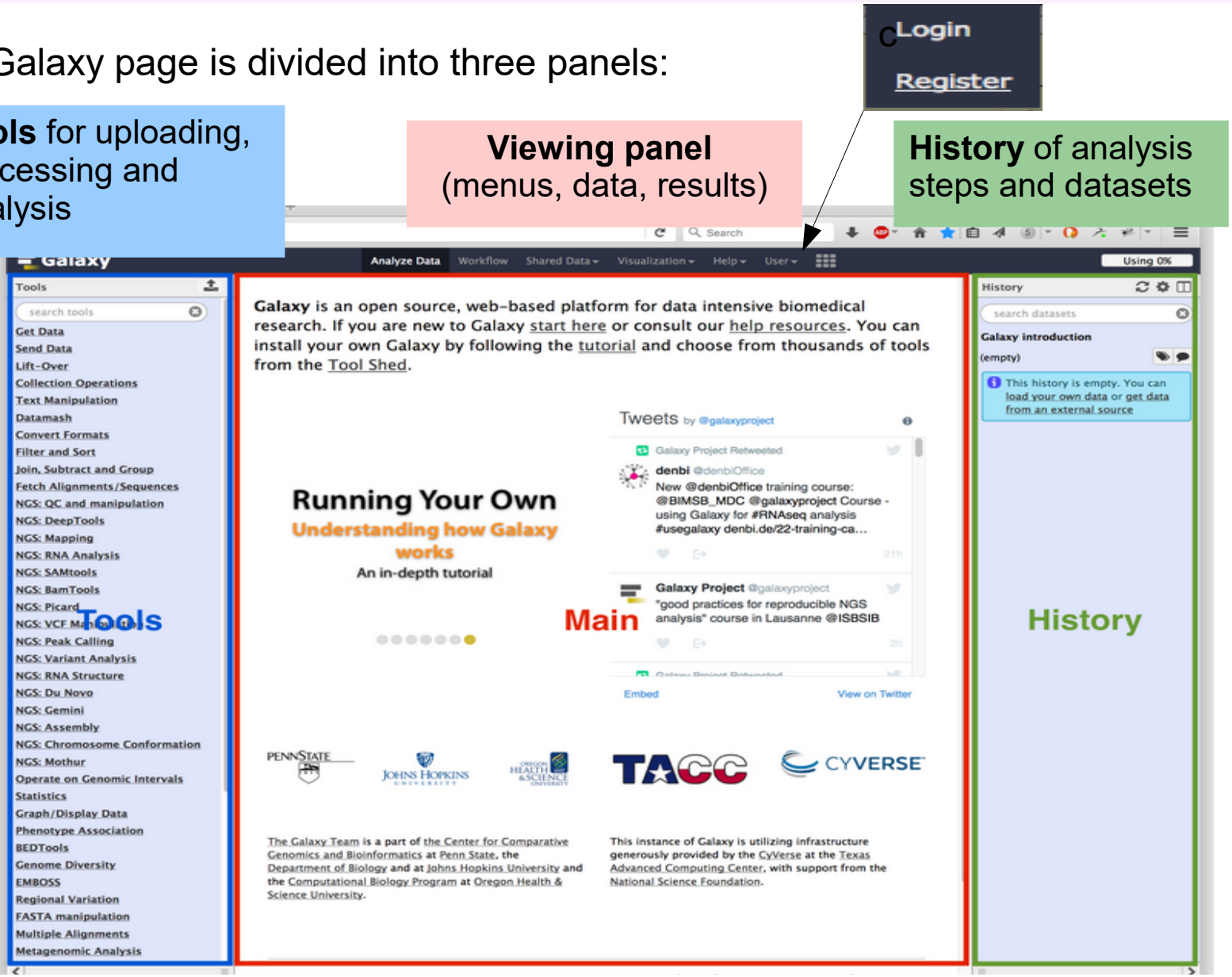
Galaxy Interface

The Galaxy page is divided into three panels:

Tools for uploading, processing and analysis

Viewing panel (menus, data, results)

History of analysis steps and datasets



Galaxy Interface

Tools for data analysis

Get Data
- From databases (UCSC Table Browser, ...)
- From uploaded files
- From urls

Text manipulation

Filter and Sort

Operate on Genomic Intervals

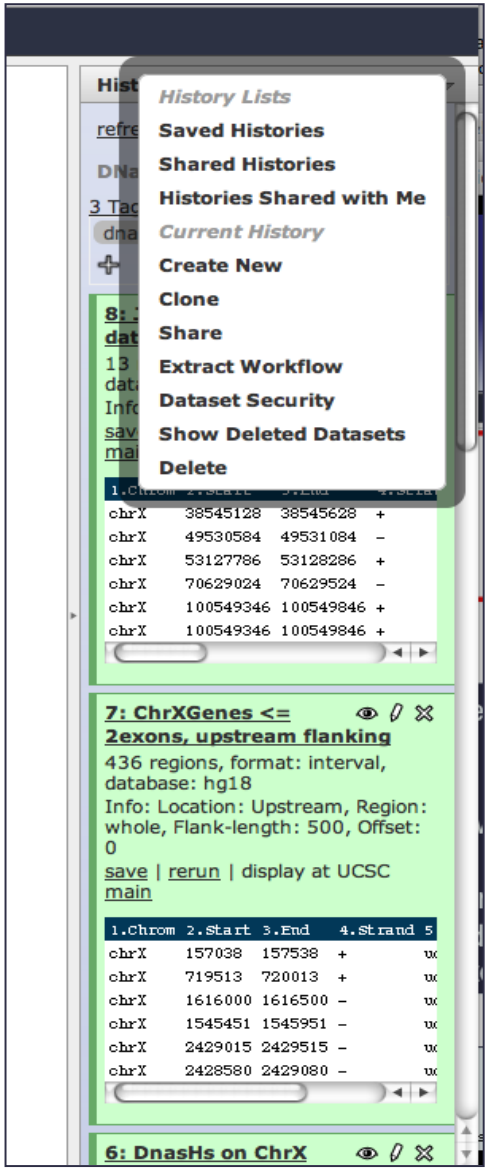
FASTA manipulation

NGS analysis
- QC
- Fastq file pre-processing
- Read Alignment / Mapping
- SAM tools

Galaxy Interface

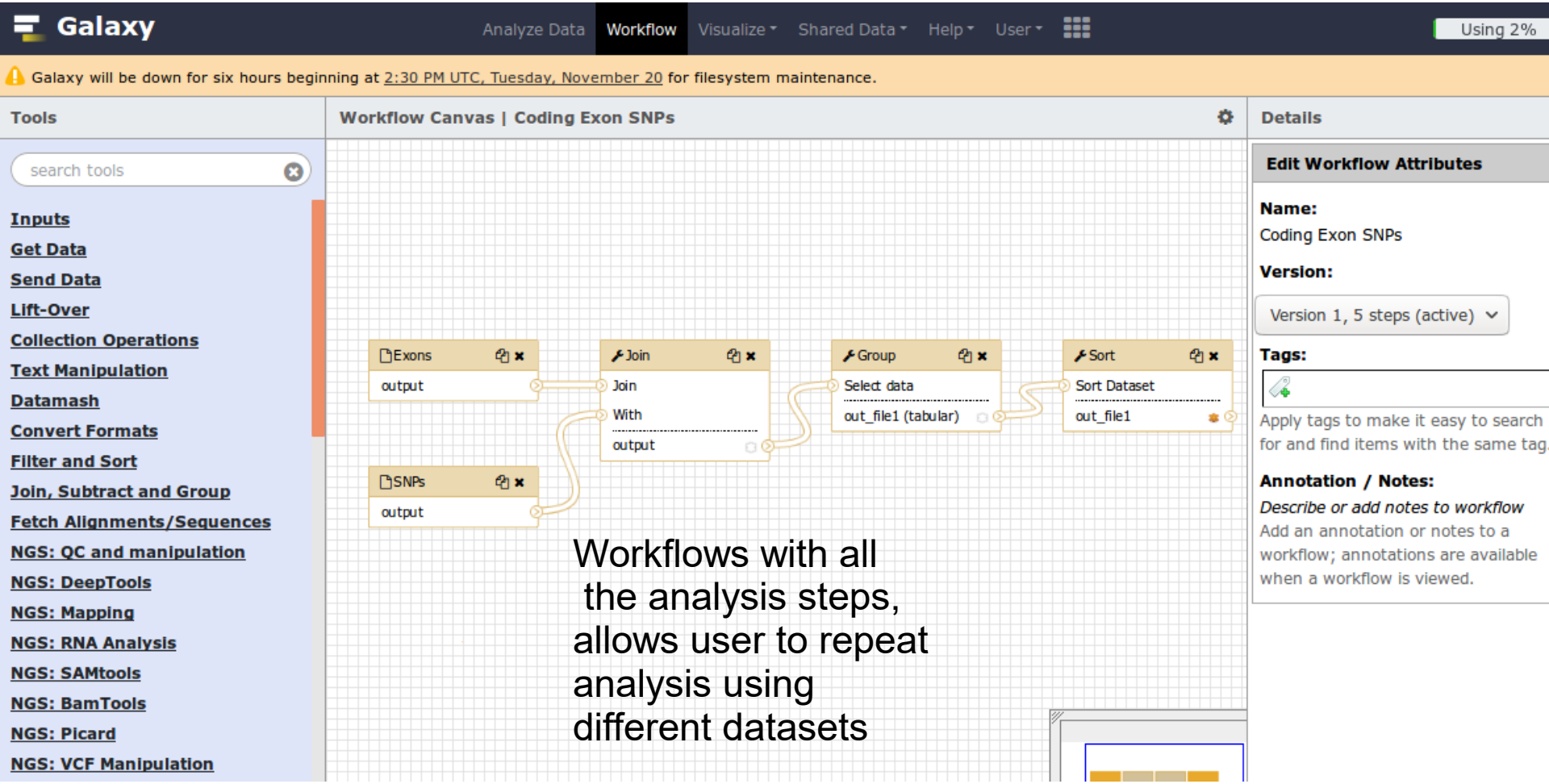
Histories

List saved histories and shared histories.
Work on Current History, create new, clone, share, create workflow, set permissions, show deleted datasets or delete history.



Galaxy Interface

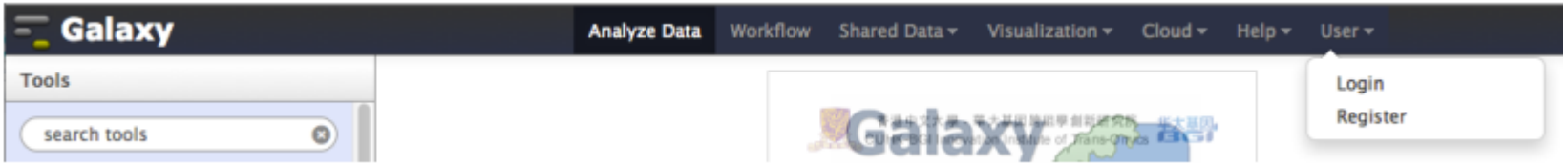
Workflows



Practicum

Register for a Galaxy account

This will only take a moment, and will allow all the work that you do to persist between sessions and allow you to name, save, share, and publish Galaxy histories, workflows, datasets and pages. It allows you to store up to 250GB of data on this public server.

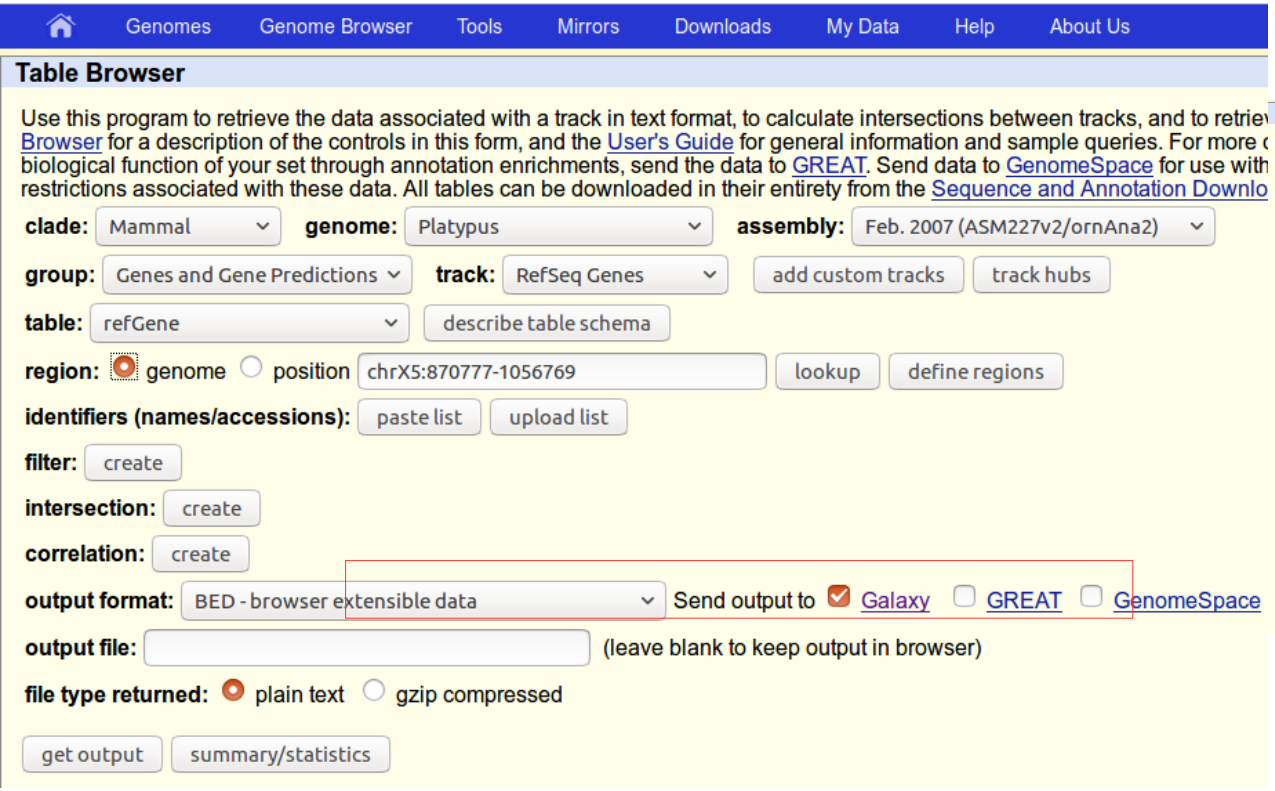


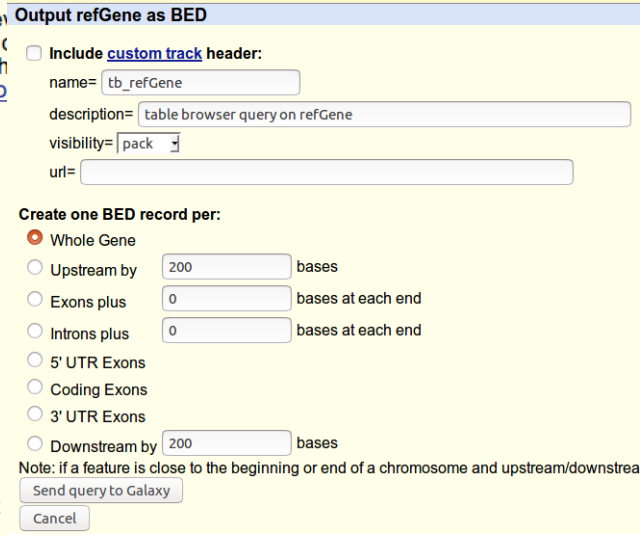
Practicum

Importing data into Galaxy

1. From database queries (eg. UCSC): obtain a BED-formatted dataset of all RefSeq genes from platypus.

   Get Data > UCSC Main – Table Browser tool
   Set genome, RefSeg Genes, and BED output format (send to Galaxy)

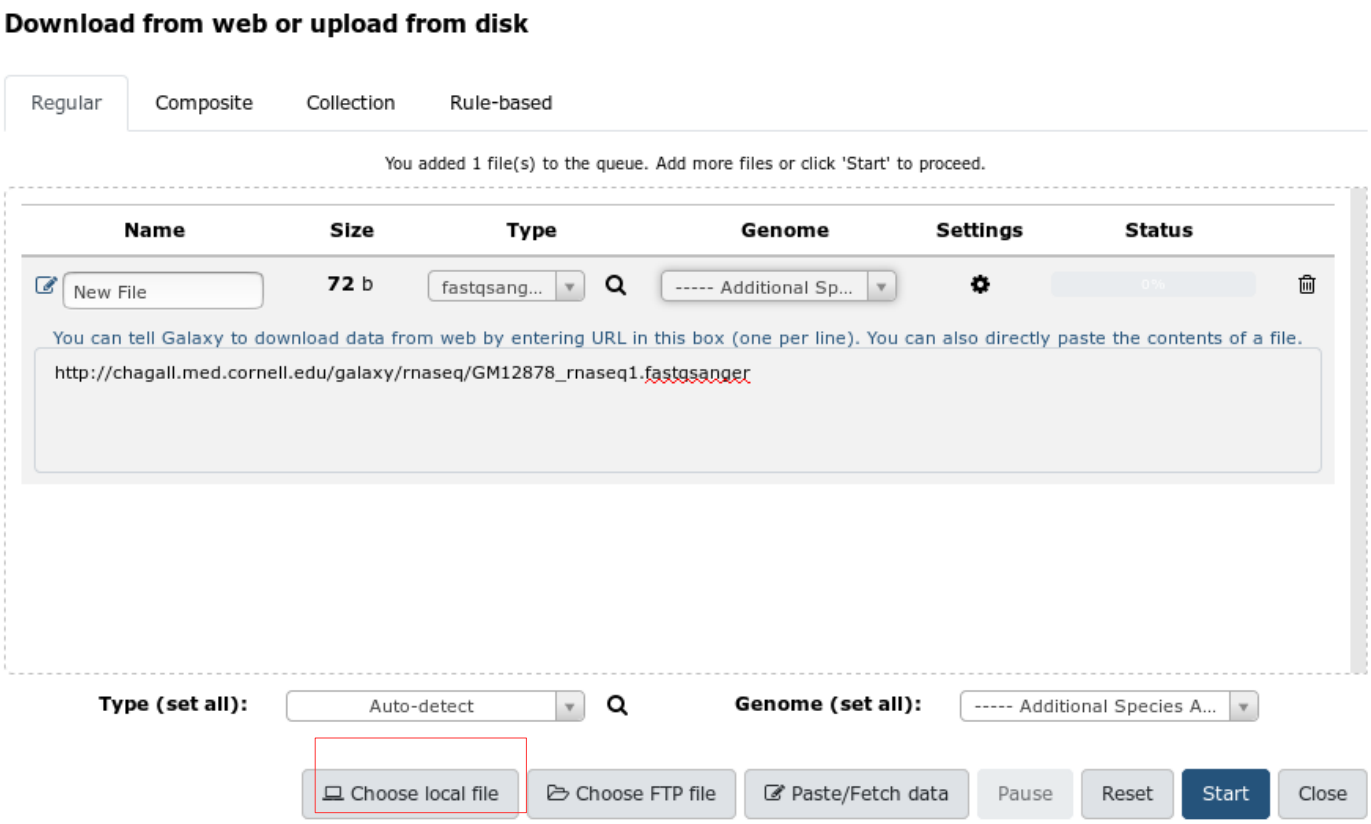




Importing data into Galaxy

2. From a File on your computer / FTP file:

Get Data > Upload File

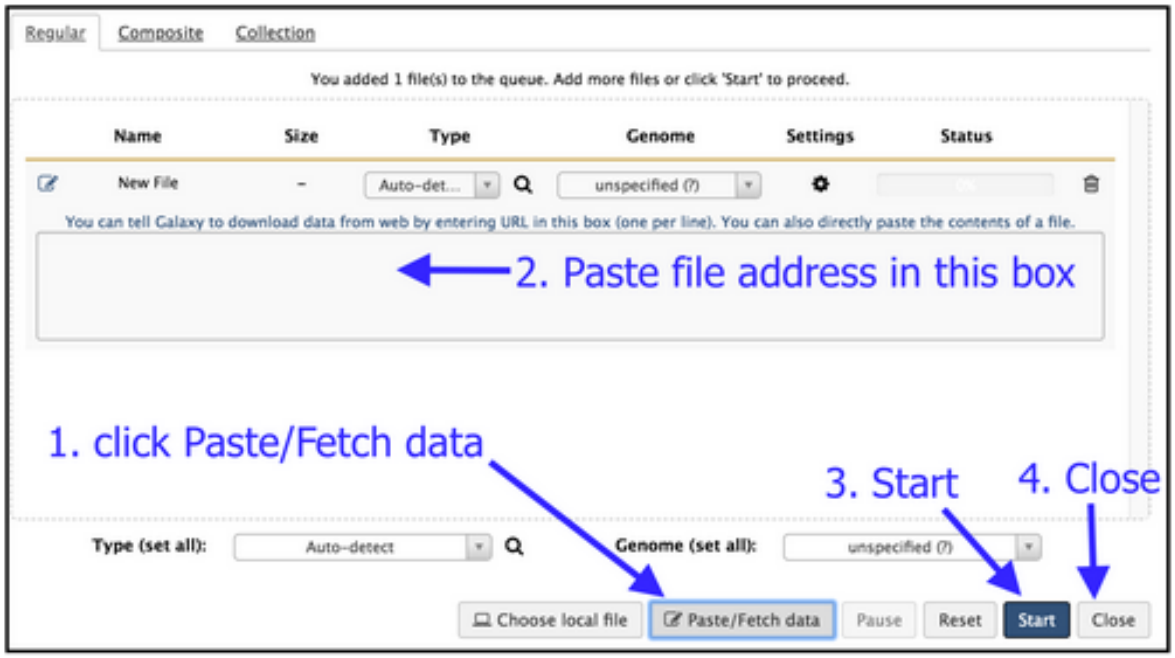


Practicum

Importing data into Galaxy

3. From a website:

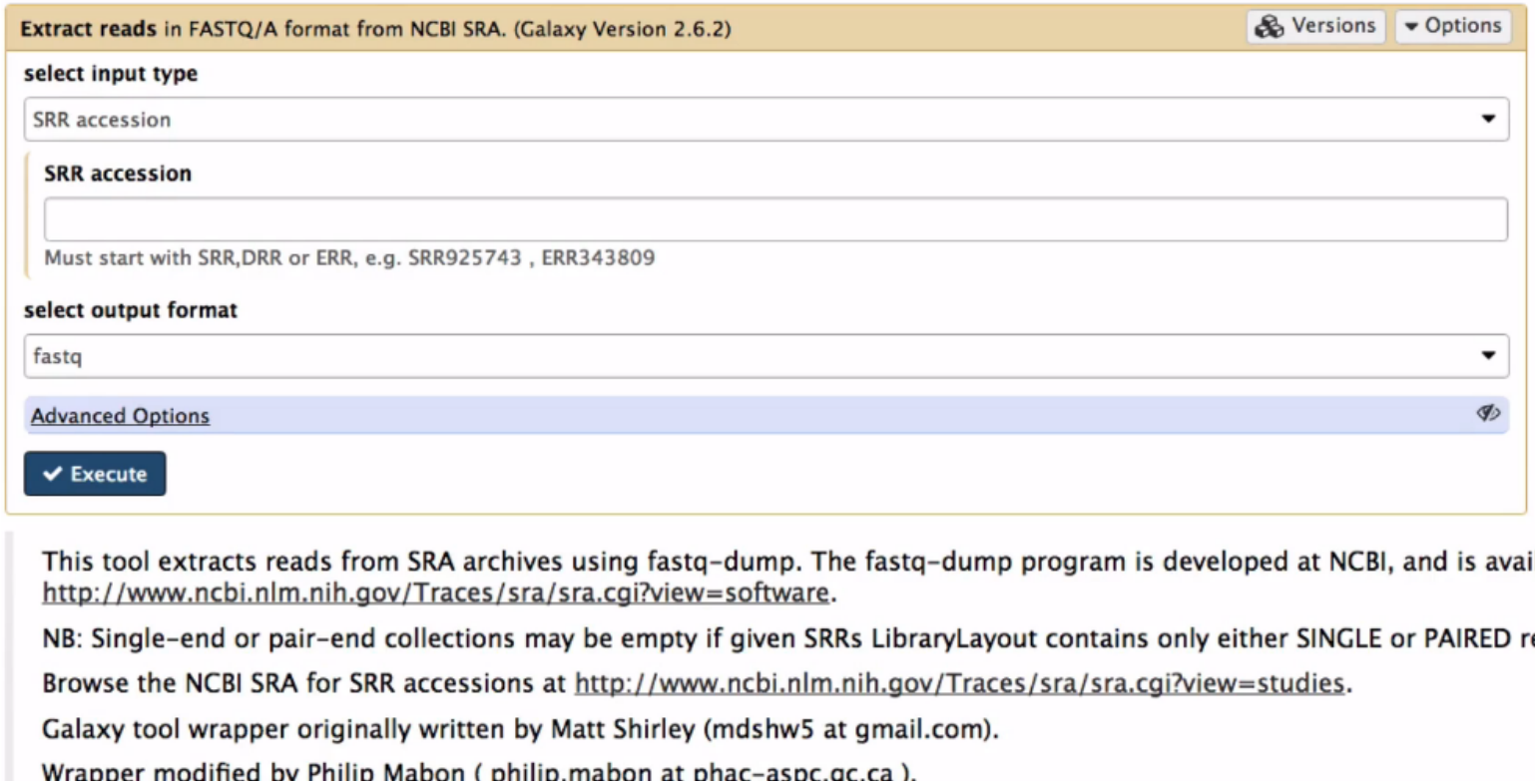
Get Data > Upload File
Copy this URL into the text-entry box:
url: https://zenodo.org/record/582600/files/mutant_R1.fastq



Importing data into Galaxy

4. From the Sequence Read Archive (EBI/NCBI): eg. SRR925743

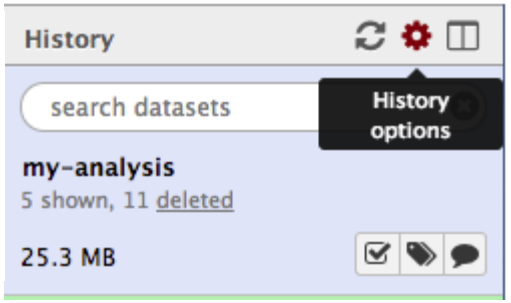
NCBI SRA Tools > Extract reads in FASTQ format



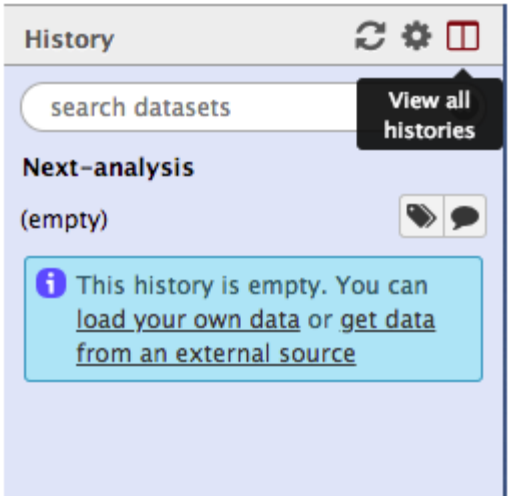
Practicum

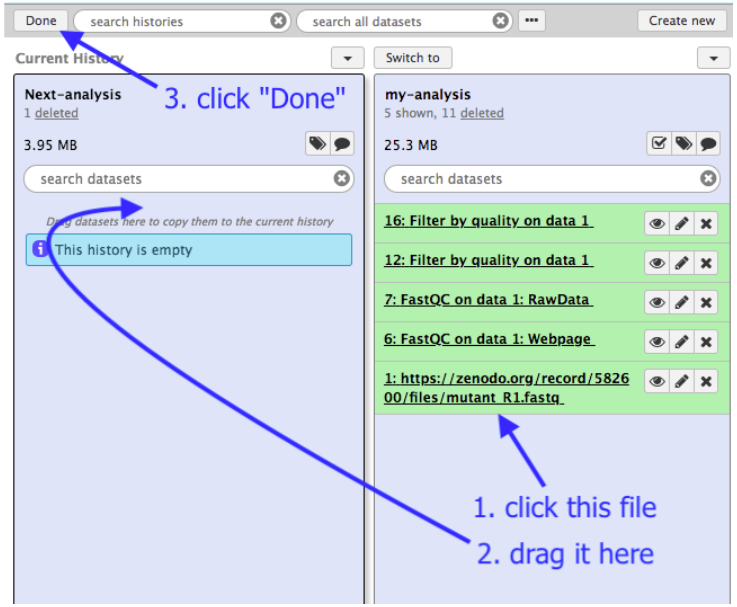
Managing histories

- Name your current history
- Create new history and rename it (eg. “pract1”)



- Manage datasets and histories:
  - View all histories
  - Drag files between histories (new history must be set to current)



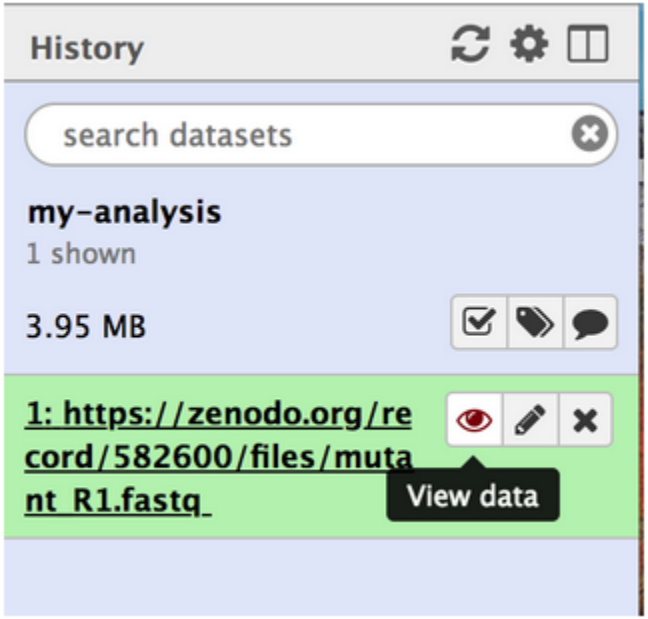


Practicum

Visualizing the dataset

- You can view file content clicking the eye icon in history.
  - The mutant_R1.fastq file contains DNA sequencing reads from a bacteria, in FASTQ format:

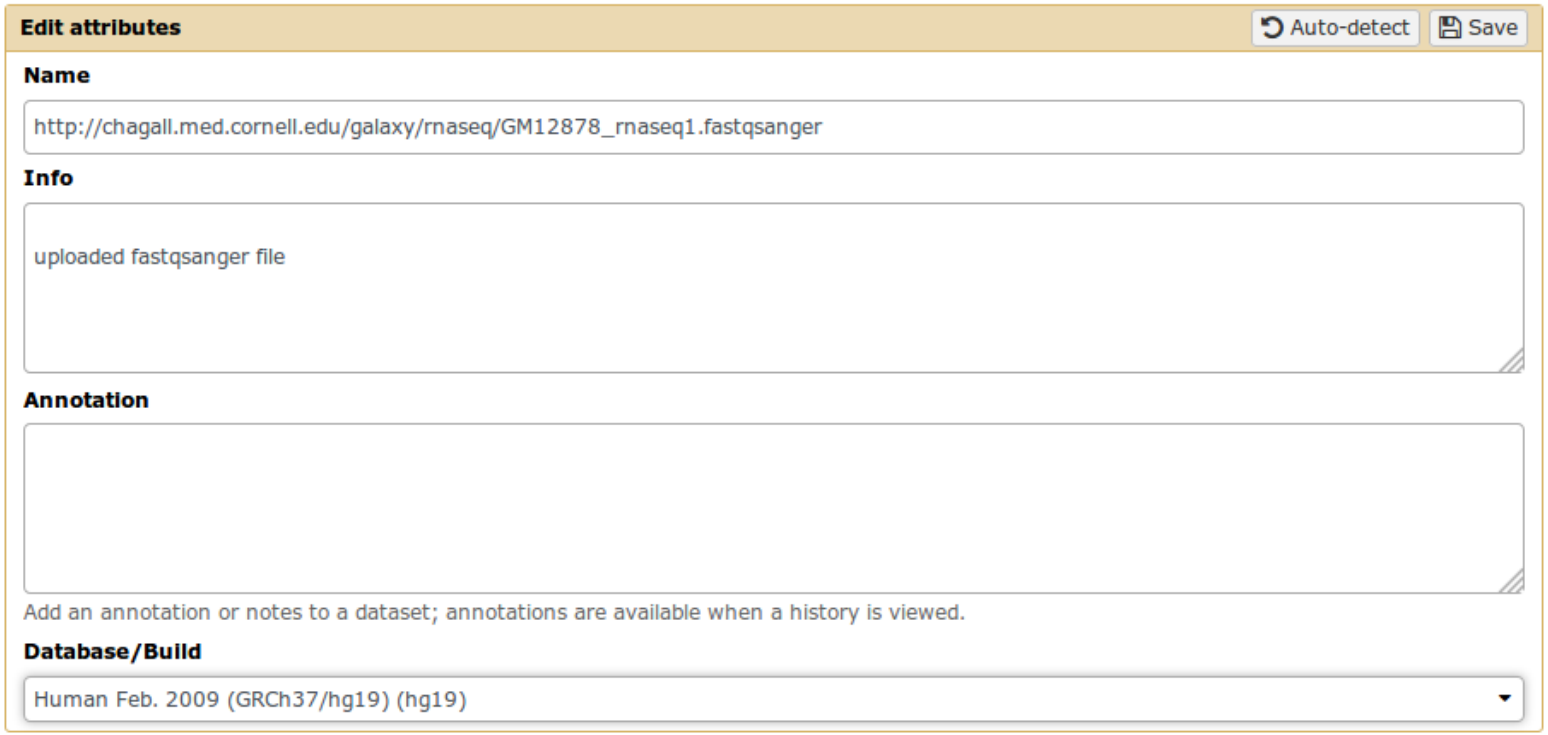
```
@mutant-no_snps.gff-24960/1                                    read 1 sequence
AATGTTGTCACTTGGATTCAAATGACATTTTAAATCTAATTATTCATGAATCGAACTAGTACGAAATGCAATGAG
+                                                              read 1 quality scores
5??A9?BBBDDDBEDDBFF+FGHHIIHHHEIHIIHIIAHDHIIHIG#IIHIFHHHFGIII*IHHHIHFIIHGICI
@mutant-no_snps.gff-24958/1
CAAAGTCGTTGGTCATATAAAAAACCGCGTACAGTCAACTATAGATACAATCAAGATAAACTCATGCACAGATTG
+
?A????@?DDDABDE9FGGGFGICFHIIIBGHIIIGICHHIFH=IHAFIHIHHHHIFCIIEIHAIFGIHIDDIHE
@mutant-no_snps.gff-24956/1
TATAAATTCAACTTTGCAACAGAACCATCTAATCTTCAACAAACTGGCCCGTTTGTTGAACTACTCTTTAATAAA
+
?????BBADD5DDDDDGFGCFFEECFBBICIII,IIHIICHIHIIFHHHHHIHIIIIIIIAHHIHHH5FHDHHH
```



Update dataset attributes

The attributes of the dataset can be updated by clicking the pencil icon next to the dataset.

- Set a name (eg. “pract1”)
- Change data type if not correct
- Add notes (eg. “url”)
- (Associate a genome to the dataset)

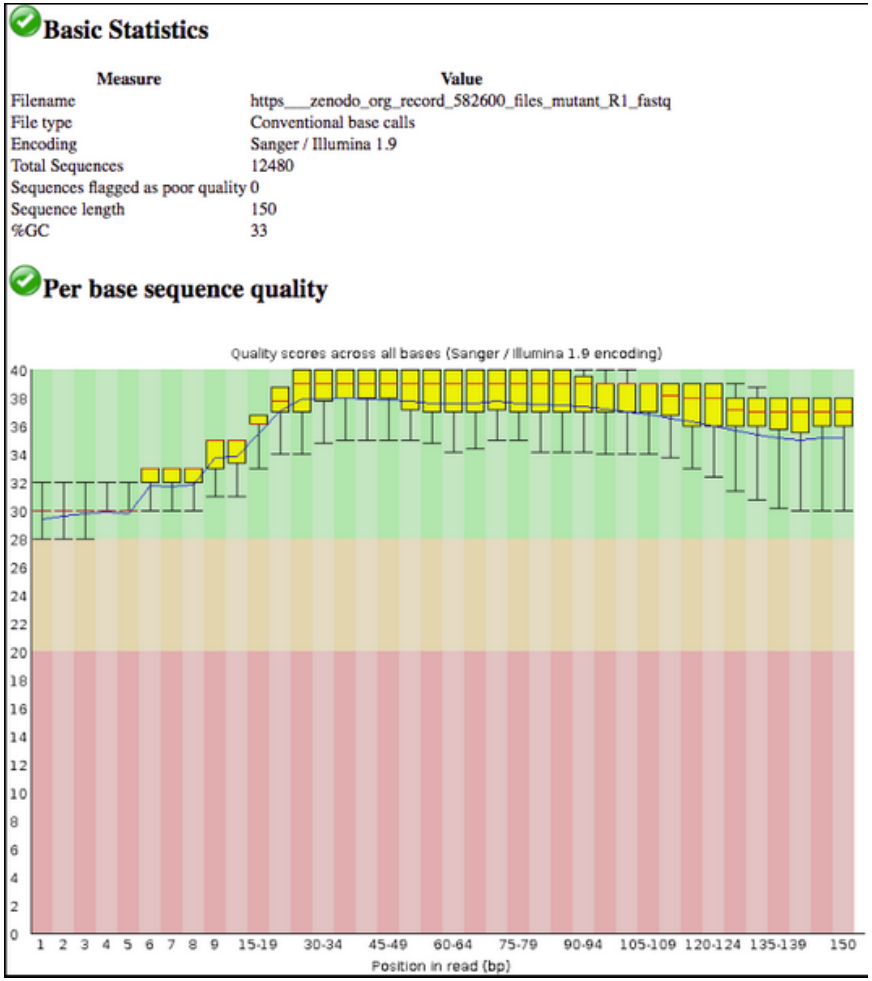


Practicum

Quality Control of Raw Data

To do FastQC in this dataset run the tool “NGS: QC and manipulation” > “FastQC”

- Look at the output file called FastQC on data 1: Webpage (click on eye icon).
- What was the length of the reads in the input FASTQ file?
- Do these reads have higher quality scores in the centre or at the ends?



Basic Statistics

Measure	Value
Filename	https___zenodo_org_record_582600_files_mutant_R1_fastq
File type	Conventional base calls
Encoding	Sanger / Illumina 1.9
Total Sequences	12480
Sequences flagged as poor quality	0
Sequence length	150
%GC	33

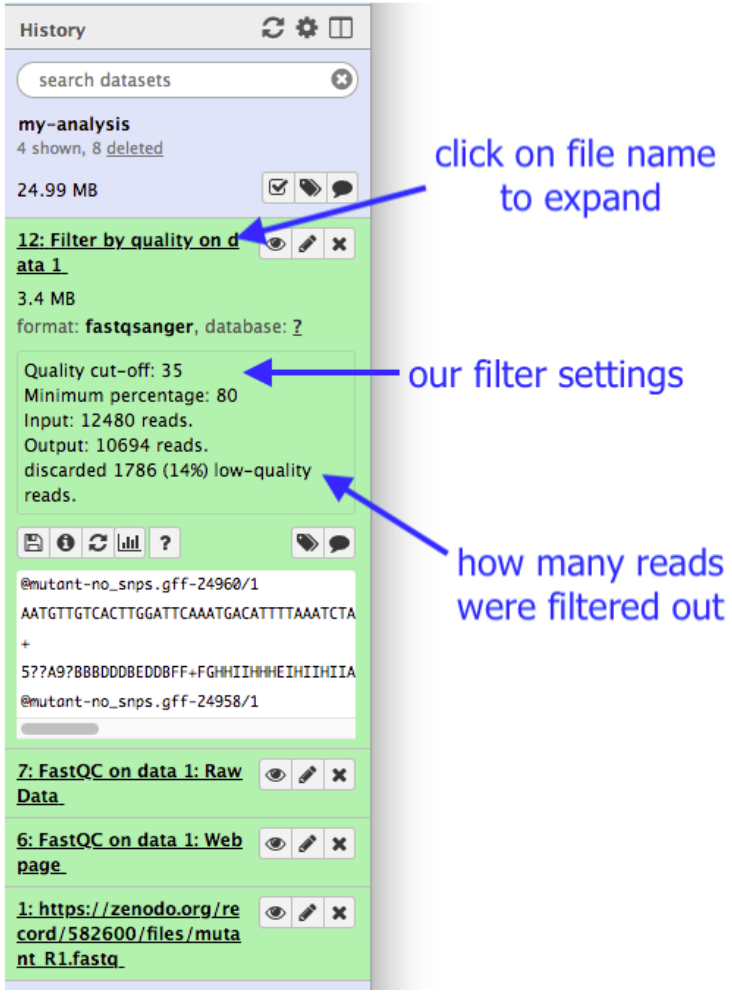
Per base sequence quality

Preprocessing of Raw Data

Filter out lower-quality reads from our FASTQ file using the tool NGS: QC > Filter by quality with the following parameters:

- Quality cutoff value: 35
- %bases in sequence with >=quality cutoff: 80

- How many reads have been discarded?



Practicum

Create workflow from history

- From history options: Export workflow

The following list contains each tool that was run to create the datasets in your current history. Please select those that you wish to include in the workflow.

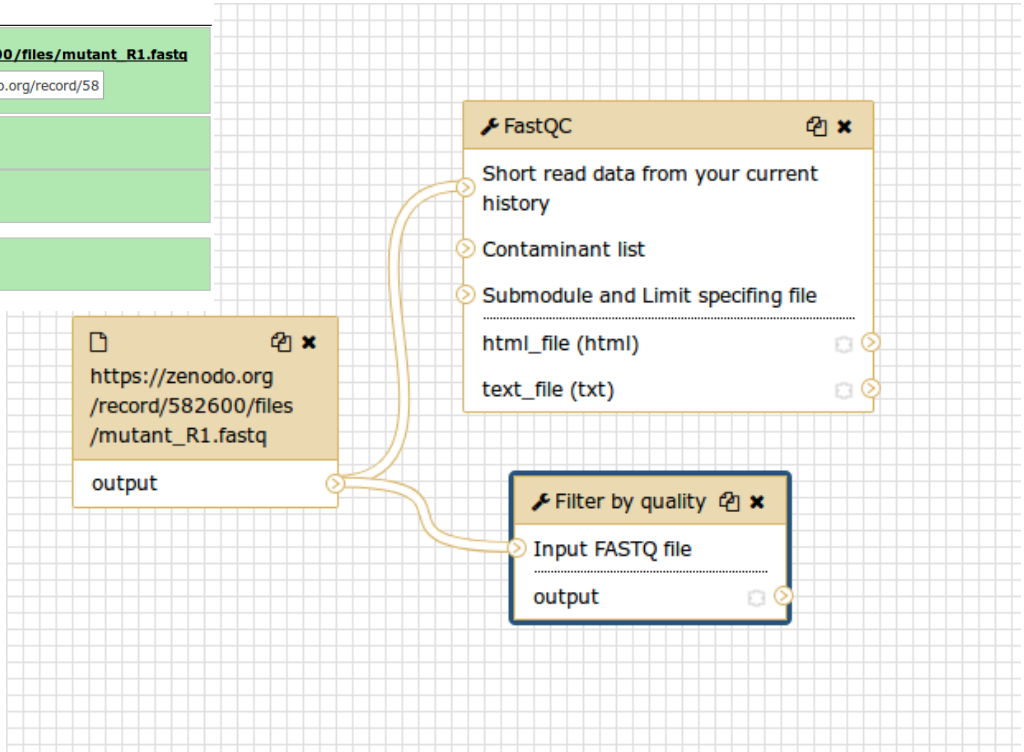
Tools which cannot be run interactively and thus cannot be incorporated into a workflow will be shown in gray.

Workflow name

Workflow constructed from history 'prova'

Create Workflow | Check all | Uncheck all

Tool	History items created
Upload File *This tool cannot be used in workflows*	1 https://zenodo.org/record/582600/files/mutant_R1.fastq Treat as input dataset https://zenodo.org/record/58
FastQC Include "FastQC" in workflow	2 FastQC on data 1: Webpage 3 FastQC on data 1: RawData
Filter by quality Include "Filter by quality" in workflow	4 Filter by quality on data 1



Practicum

Your turn!

1. Create a new history and name it “pract2”

2. Upload “raw_child-ds-1.fq” and “raw_child-ds-2.fq” files from your USB

3. Update the attributes of the two dataseta

- Change the names of the datasets to “sample-f” and “sample-r”, respectively.
- Check data type is set to “fastq”
- Associate the dataset with the human hg38 genome in the Database/Build field.

4. Run a quality control on each dataset using the FastQC tool.

- What can we say about the quality of these datasets?

Practicum

Your turn!

5. Trim the reads in the GM12878 dataset using the tool “NGS: QC and manipulation > FastQ Trimmer” tool. Set the parameters:

- Determine from the boxplot and FastQC figures where the quality of the reads begins to drop off sharply.
- Calculate how many bases have to be trimmed from the end and use that number as the Offset from 3' end.

6. Using the tool “NGS: QC and manipulation > Filter by quality”, filter out all sequences with >=80% bases that have a quality less than 20 (use as input the trimmed sequences).

- How many sequences do you have left in your dataset?

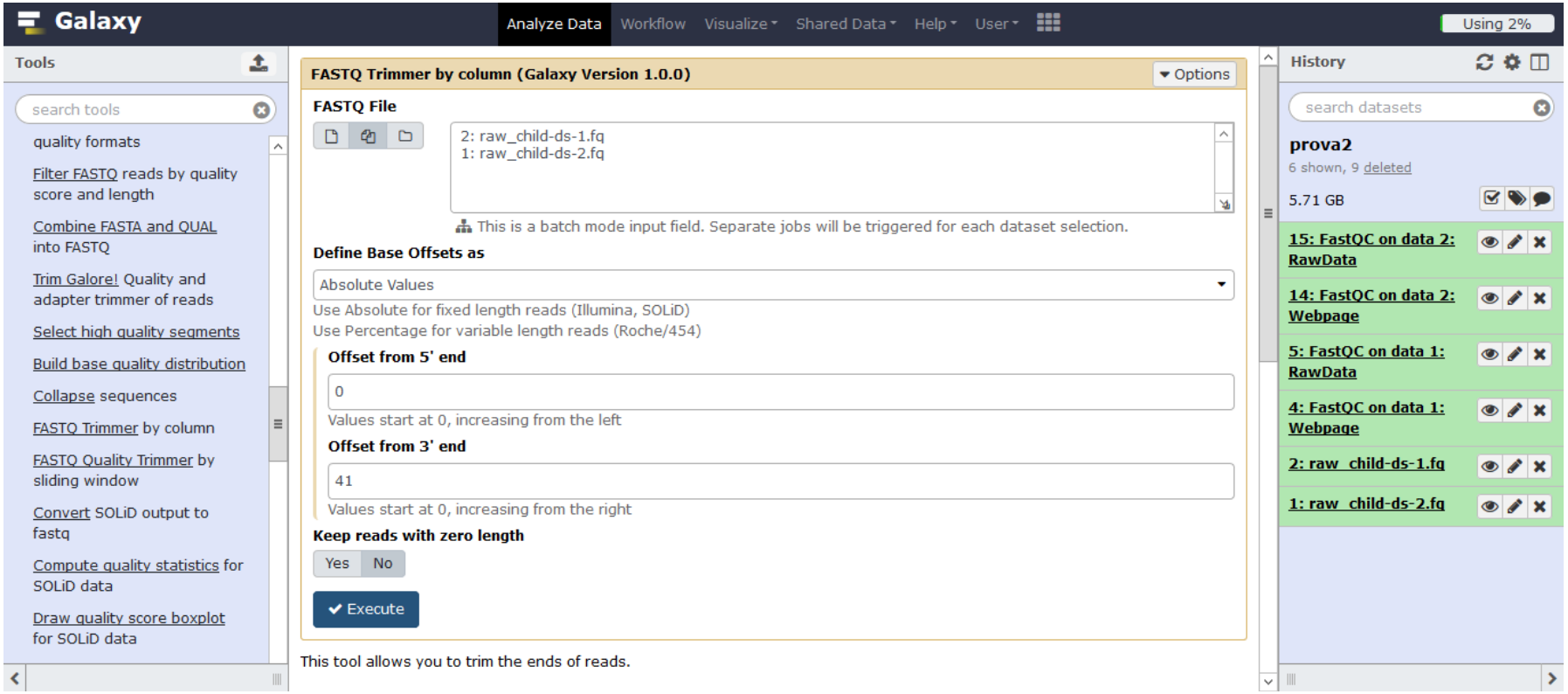
7. Re-run FastQC on the quality-controlled data, and inspect the new FastQC report

- Has the sequence quality been improved?

8. Convert your analysis history into a workflow

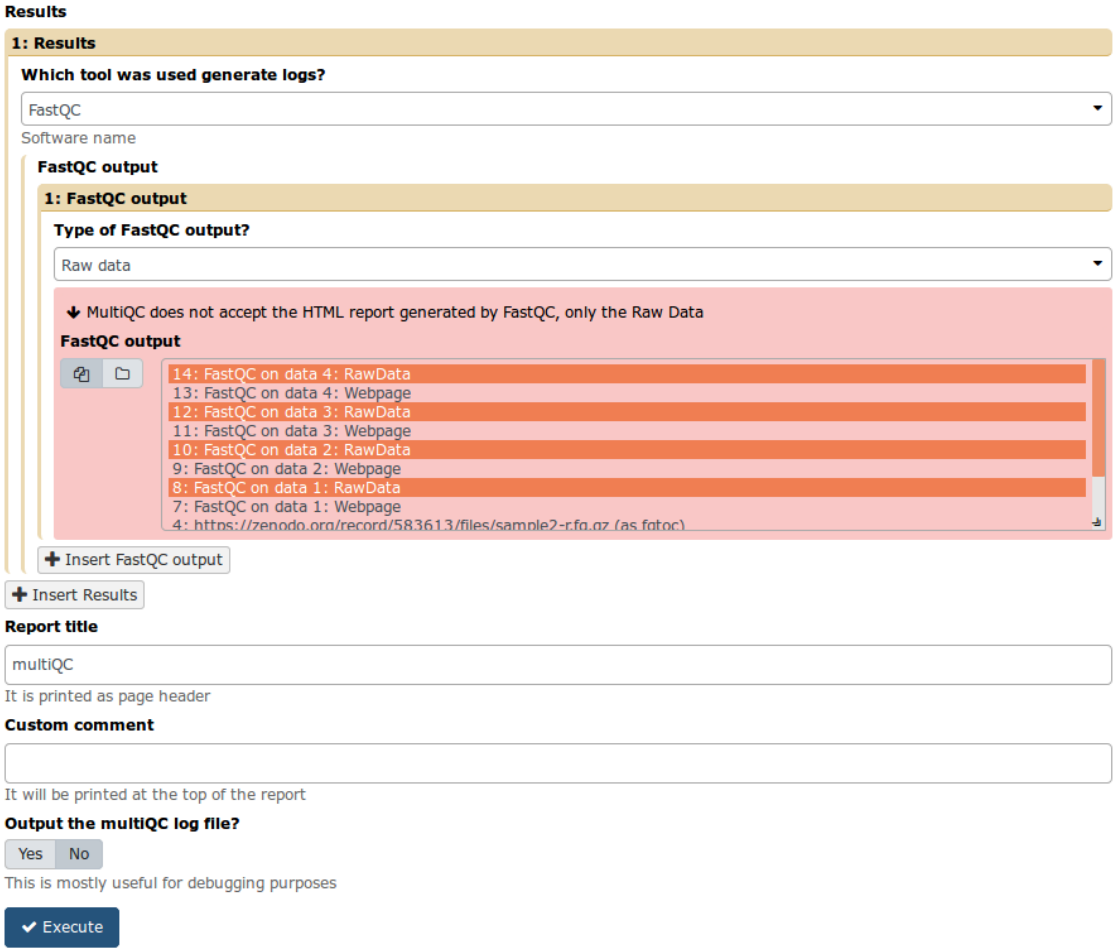
Practicum

Solution



MultiQC tool

- MultiQC tool allows to summarize multiple QC reports at once, though FastQC needs to be run on each dataset first.



Practicum

MultiQC tool

General Statistics

Copy table | Configure Columns | Plot | Showing 4/4 rows and 4/5 columns.

Sample Name	% Dups	% GC	Length	M Seqs
sample1-f	53.2%	44%	233 bp	0.0
sample1-r	55.1%	44%	233 bp	0.0
sample2-f	66.3%	44%	251 bp	0.1
sample2-r	65.7%	44%	251 bp	0.1

FastQC

FastQC is a quality control tool for high throughput sequence data, written by Simon Andrews at the Babraham Institute in Cambridge.

Sequence Quality Histograms

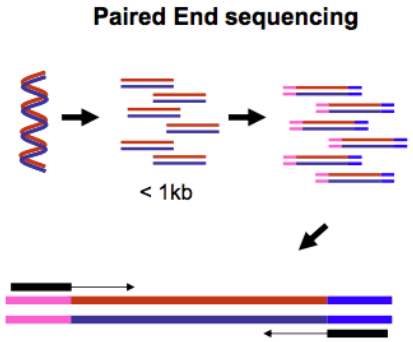
1 1 2

The mean quality value across each base position in the read. See the FastQC help.

Y-Limits: on

Processing paired-end data

Paired-end data: a single physical piece of DNA/RNA is sequenced from two ends and so generates two reads. These can be represented as separate files (two fastq files with first and second reads) or a single file were reads for each end are interleaved.



Two single files

File 1

```
@M02286:19:000000000-AA549:1:1101:12677:1273 1:N:0:23
CCTACGGGTGGCAGCAGTGAGGAATATTGGTCAATGGACGGAAGTCT
+
ABC8C,:@F:CE8,B-,C,-6-9-C,CE9-CC--C-<-C++,,+;CE
@M02286:19:000000000-AA549:1:1101:15048:1299 1:N:0:23
CCTACGGGTGGCTGCAGTGAGGAATATTGGACAATGGTCGGAAGACT
+
ABC@CC77CFCEG;F9<F89<9--C,CE,--C-6C-,CE:++7:,CF
```

File 2

```
@M02286:19:000000000-AA549:1:1101:12677:1273 2:N:0:23
CACTACCCGTGTATCTAATCCTGTTTGATACCCGCACCTTCGAGCTTA
+
--8A,CCE+,,:,<CC,,<CE@,CFD,,C,CFF+@+@CCEF,,,B+C,
@M02286:19:000000000-AA549:1:1101:15048:1299 2:N:0:23
CACTACCGGGGTATCTAATCCTGTTCGCTCCCCACGCTTTCGTCCATC
+
-6AC,EE@::CF7CFF<<FFGGDFFF,@FGGGG?F7FEGGGDEFF>FF
```

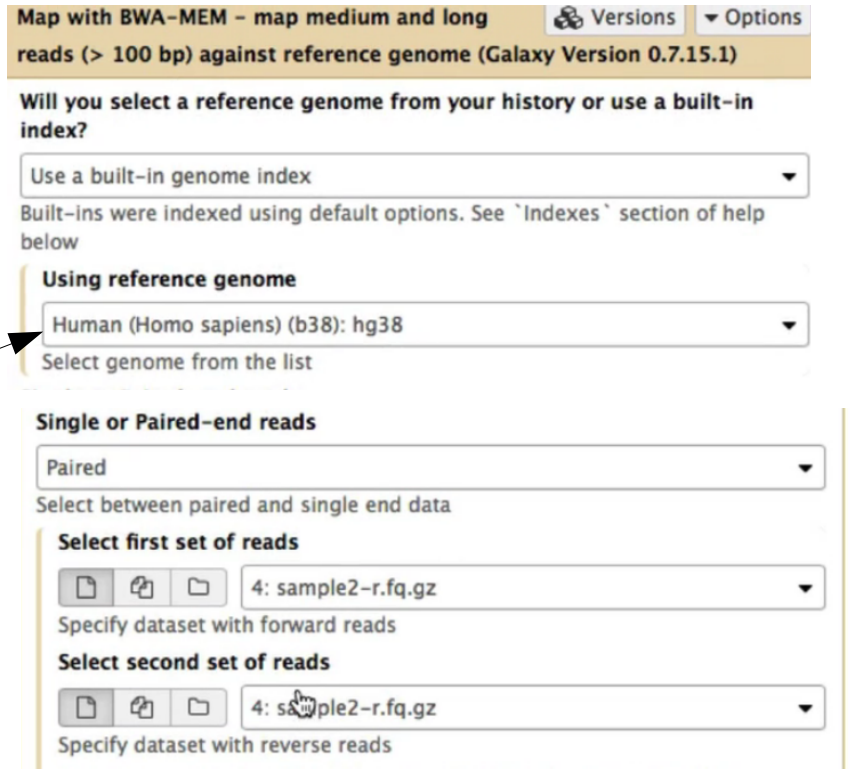
Interleaved file

```
@1/1
AGGGATGTGTTAGGGTTAGGGTTAGGGTTAGGGTTAGGGTTAGGGTTA
+
EGGEGGGDFGEEEAEECGDEGGFEEGEFGBEEDDECFEFDD@CDD<ED
@1/2
CCTAACCCTAACCCTAACCCTAACCCTAACCCTAACCCTAACCCTAAC
+
GHHHDFDFGFGEGFBGEGGEGEGGGHGFGHFHFHHHHHHHHEF?EFEFF
@2/1
AGGGATGTGTTAGGGTTAGGGTTAGGGTTAGGGTTAGGGTTAGGGTTA
+
HHHHHHEGFHEEFEEHEEHHGGEGGGGEFGFGGGGHHHHFBEEEEEFG
@2/2
CCTAACCCTAACCCTAACCCTAACCCTAACCCTAACCCTAACCCTAAC
+
HHHHHHHHHHHHHGHHHHHHGHHHHHHHHHHHHFHHHFHHHHHHHHHH
```

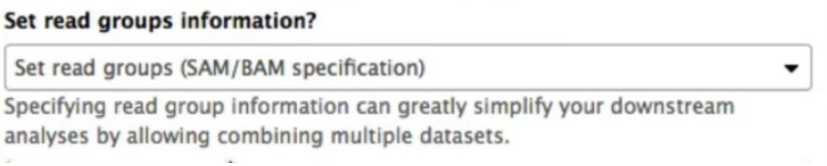
Alignment/Mapping of paired-end data

- Mapping of NGS reads against reference sequences is one of the key steps of the analysis.
- Map datasets uploaded before using “NGS: Mapping > Map with BWA-MEM” tool against hg38 version of the human genome. Set paired data (child1 (f) / child2 (r))

The hg38 contains all chromosomes as well as all unplaced contigs. The hg38 canonical does not contain unplaced sequences and only consists of chromosomes 1 through 22, X, Y, and mitochondria. The hg38 canonical female contains everything from the canonical set with the exception of chromosome Y.



- As output we get a unique BAM file



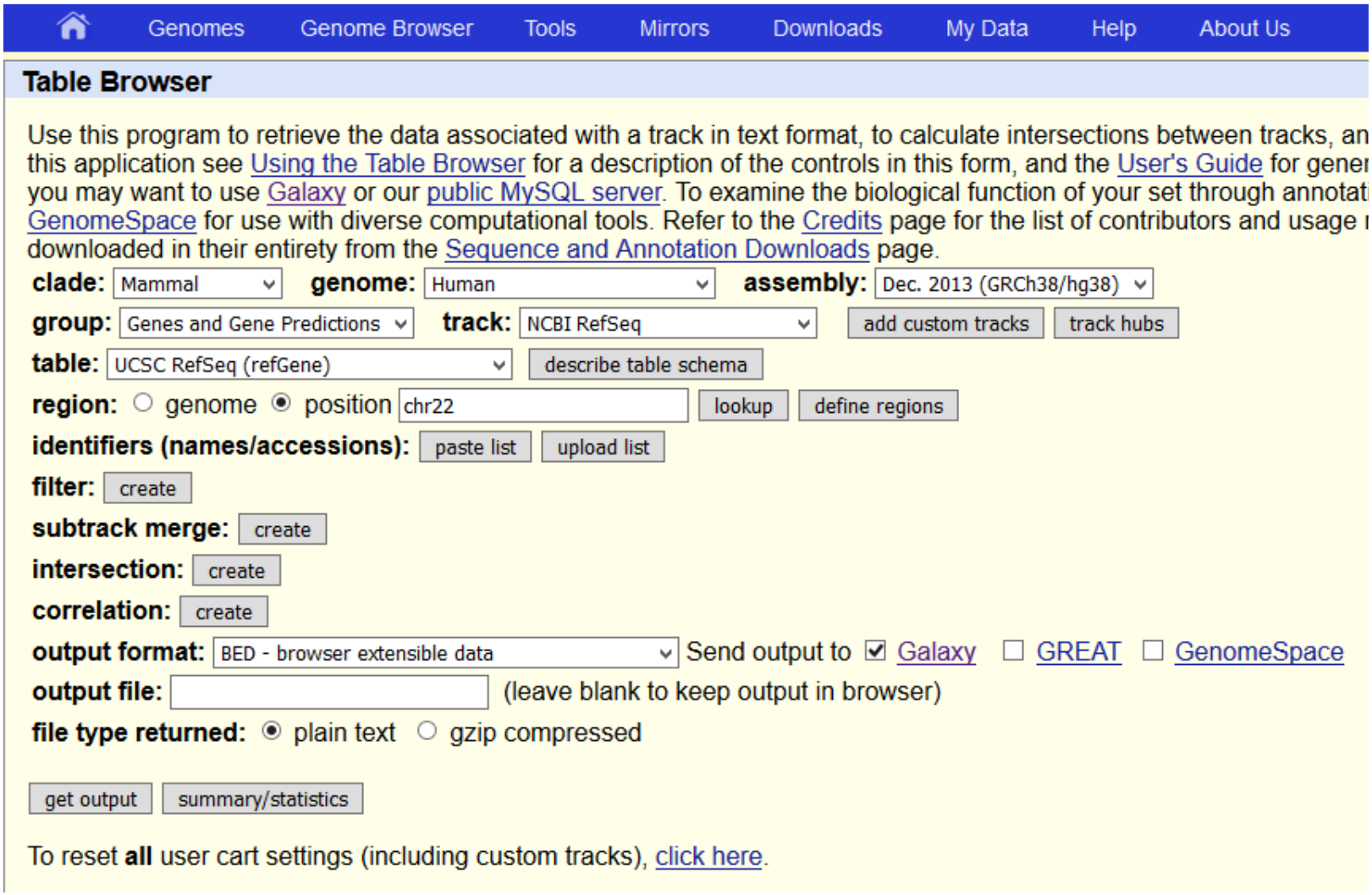
Practicum

Which coding exon has the highest number of single nucleotide polymorphisms (SNPs) on human chromosome 22?

First retrieve all coding exons and SNPs in chromosome 22

- Create a new history. Call it “SNPs in coding exons”.
- Next retrieve all coding exons in chromosome 22 from UCSC – Table browser:
    1. Use Get Data → UCSC Main tool.
    2. Change the position to search to chr22.
    3. Click the get Output button.
    4. On the output format screen, change it to one BED record per coding exon.
    5. Click Send query to Galaxy.
    6. Rename this exon dataset to “Exons_22”.

- Now retrieve the SNP data, again from UCSC.
    1. Use Get Data → UCSC Main tool.
    2. This time, change the group to Variation and Repeats.
    3. Change the position to chr22 again.
    4. On the output format page, the one BED record per gene should again be selected. In this case, “gene” is actually “SNP” (or “feature”).
    5. Click the Send query to Galaxy button.
    6. Rename this dataset to “SNPs_22”.


Genomes
Genome Browser
Tools
Mirrors
Downloads
My Data
Help
About Us
Table Browser
Use this program to retrieve the data associated with a track in text format, to calculate intersections between tracks, an
this application see Using the Table Browser for a description of the controls in this form, and the User's Guide for gener
you may want to use Galaxy or our public MySQL server. To examine the biological function of your set through annotat
GenomeSpace for use with diverse computational tools. Refer to the Credits page for the list of contributors and usage
downloaded in their entirety from the Sequence and Annotation Downloads page.
clade: Mammal
genome: Human
assembly: Dec. 2013 (GRCh38/hg38)
group: Genes and Gene Predictions
track: NCBI RefSeq
add custom tracks
track hubs
table: UCSC RefSeq (refGene)
describe table schema
region: genome position chr22
lookup
define regions
identifiers (names/accessions):
paste list
upload list
filter: create
subtrack merge: create
intersection: create
correlation: create
output format: BED - browser extensible data
Send output to Galaxy GREAT GenomeSpace
output file:
(leave blank to keep output in browser)
file type returned: plain text gzip compressed
get output
summary/statistics
To reset all user cart settings (including custom tracks), click here.

Practicum

Which coding exon has the highest number of single nucleotide polymorphisms (SNPs) on human chromosome 22?

- Now we need to associate the SNPs with their exons. Use the database join command to associate with an exon any SNP whose given genomic interval overlaps that of any exon by at least one base:
    1. Open the Operate on Genomics Intervals → Join tool.
    2. Use the exons dataset as the first dataset and the SNPs dataset as the second dataset.
    3. Click Execute.
    4, View the results

Note that the first 6 columns correspond to exons and the next 6 to SNPs positions

Practicum

Which coding exon has the highest number of single nucleotide polymorphisms (SNPs) on human chromosome 22?

- Next, we want to count the number of SNPs associated with each exon:
    1. Open the Join, Subtract, and Group → Group tool.
    2. Change the Group by Column to column 4 (i.e., the exon name).
    3. Add a new operation.
    4. Change the type of the new operation to Count and the column to col 4. This groups the dataset on the exon name, and count how many exons of the same name are in each unique group.

The resulting dataset will have two columns: one is a list of all the distinct exon names, and the other is the number of times that exon name was associated with a SNP.

Practicum

Which coding exon has the highest number of single nucleotide polymorphisms (SNPs) on human chromosome 22?

- We can now rearrange this dataset such that the exons with the highest number of SNPs are at the top.
    1. Open the Filter and Sort → Sort tool.
    2. Sort the dataset by column 2 (SNP count), in descending order (i.e., highest first).

Which exon has the highest number of SNPs?

- Finally, we want to select the five exons with the highest numbers of SNPs.
    1. Open the Text Manipulation → Select First tool.
    2. Make sure the sorted dataset is selected.
    3. Change the number of lines to select to 5.
    4. Click Execute.

1	2
NM_001136213_cds_0_0_chr22_15690078_f	40
NM_006071_cds_0_0_chr22_46256561_r	31
NM_001005239_cds_0_0_chr22_15528159_f	24
NM_138433_cds_0_0_chr22_50546244_f	24
NM_173566_cds_5_0_chr22_31712083_r	15