EMBL Advanced Course RNA-Seq and ChiP-Seq Data

Nicolas Delhomme, June 20th-22nd 2011, Heidelberg
Outline

• Sequence alignment

• Aligners

• Recent development

• Aligners’ usage

• Alignment pitfall

• Bioconductor
Who are we?

• **Me:**
  - Staff member of the Functional Genomic Center
  - Genome Biology Unit, EMBL, Heidelberg
  - co-directed by Eileen Furlong and Lars Steinmetz

• Position 50% service, 50% research
  - service: establishment of a LIMS and pre-processing system for NGS data
  - research: analyses of NGS data of various kinds: RNAseq, TagSeq, ChIPseq (TF and Histones) and *de-novo* assembly, mainly using R

• **You:**
  - your aligner’s knowledge?
Sequence alignment

• Two main approaches:
  • based on hash table
    • spaced seeds
  • based on suffix/prefix tries
    • Burrows-Wheeler transform (BWT)

• Reviewed in Li and Homer: A survey of sequence alignment algorithms for next-generation sequencing. Briefings in Bioinformatics (2010)
a) Spaced seeds

Reference genome (> 3 gigabases)

Chr1
Chr2
Chr3
Chr4

Extract seeds

Position N

Position 2
CTGC CGTA AACT AATG

Position 1

ACTG CCGT AAAC TAAT
ACTG **** AAAC ****
**** CCGT **** TAAT
**** **** AAAC TAAT
**** **** AAAC TAAT
ACTG CCGT **** ****
**** CCGT AAAC ****

Index seed pairs

Seed index (tens of gigabytes)

ACTG **** AAAC ****

ACTG **** **** TAAT

**** CCGT **** TAAT

Confirm hits by checking "****" positions

Six seed pairs per read/fragment

1 2 3 4 5 6

Look up each pair of seeds in index

Hits identify positions in genome where spaced seed pair is found

b) Burrows-Wheeler

Reference genome (> 3 gigabases)

Chr1
Chr2
Chr3
Chr4

Short read
ACTCCCGTACTCTAAT

Concatenate into single string

Burrows-Wheeler transform and indexing

Bowtie index (~2 gigabytes)

Look up 'suffixes' of read

Hits identify positions in genome where read is found

Convert each hit back to genome location

Report alignment to user

Trapnell and Salzberg, 2009
Suffix/Prefix Tries

Li and Homer, 2010
### Aligners

#### Table I: Popular short-read alignment software

<table>
<thead>
<tr>
<th>Program</th>
<th>Algorithm</th>
<th>SOLiD</th>
<th>Long&lt;sup&gt;a&lt;/sup&gt;</th>
<th>Gapped</th>
<th>PE&lt;sup&gt;b&lt;/sup&gt;</th>
<th>Q&lt;sup&gt;c&lt;/sup&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bfast</td>
<td>hashing ref.</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Bowtie</td>
<td>FM-index</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>BWA</td>
<td>FM-index</td>
<td>Yes&lt;sup&gt;d&lt;/sup&gt;</td>
<td>Yes&lt;sup&gt;e&lt;/sup&gt;</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>MAQ</td>
<td>hashing reads</td>
<td>Yes</td>
<td>No</td>
<td>Yes&lt;sup&gt;f&lt;/sup&gt;</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Mosaik</td>
<td>hashing ref.</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Novoalign&lt;sup&gt;g&lt;/sup&gt;</td>
<td>hashing ref.</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

<sup>a</sup>Work well for Sanger and 454 reads, allowing gaps and clipping.  
<sup>b</sup>Paired end mapping.  
<sup>c</sup>Make use of base quality in alignment.  
<sup>d</sup>BWA trims the primer base and the first color for a color read.  
<sup>e</sup>Long-read alignment implemented in the BWA-SW module.  
<sup>f</sup>MAQ only does gapped alignment for Illumina paired-end reads.  
<sup>g</sup>Free executable for non-profit projects only.
Aligners c’ed

- 20 aligners published in the last 2 years
- Most deal with short reads
- Some of those with ABI specific “color-space”
- A large scale study comparing them is underway:
  - GSNAP: http://research-pub.gene.com/gmap/ is the most efficient so far (personal communication, Paul Bertone, EBI)
Recent developments

• gapped alignment
  • Recent aligners are able to perform gapped alignments
    • small indels
    • no splicing events with large introns
  • BWA, Novoalign

• bisulfite sequencing
  • unmethylated C are converted to T (G complement converted to A)
  • 2 references
    • one with all C converted to T
    • one with all G converted to A
    • C-T mismatch or G-A mismatch are ignored
    • results from both alignments are combined
What aligner for my data?

• The choice of aligner depends on the data at hands (too late!)

• “Early”: it should be decided when planning the experiment

• What criteria?
  • do you always need paired end reads?
  • do you need gap alignments?
Using read quality

- lower penalty for base with lower qualities
- quality recalibration helps
Alignment usage summary

• gapped alignment for very short reads (25-36bp) is computationally challenging
  • gapped align. have a better sensitivity, same error rate
  • important for indels and SNPs
  • impact not analyzed for ChIP-Seq or RNA-Seq

• paired end alignment always outperform single end alignment

• Next tools to come:
  • multi-genome alignment (1000 genomes project, Drosophila population genomics project, 1001 genomes project...)
Aligner’s usage, an example

• What is the impact of unique alignments?

• Approach:
  • MAQ policy: keep one alignment per read
  • strict policy: keep only reads with a single alignment

• How to assess the differences?
  • comparing MAQ, strict and (MAQ - strict)

• Data
  • ChIP-Seq of an histone mark: K27Ac
Most are harmless: repetitive region small
or wide
Few result in loss of information
Most of these are very repeated elements: Histone cluster
Protein kinase involved in spermatogenesis
or unknownm...
Extremely few are not clusters.
Unique alignment summary

• Always important to assess the aligner’s effect as every aligner introduces technical biases!

• In that example, using the strict policy should
  • simplify the peak calling
  • reduces the false positives in downstream analyses
  • has only a few side-effects (redo with a gene mark?)

• Additional information to be extracted and used downstream
  • For visualization, use a mappability track
  • Filter the annotation not to introduce false negatives in the analyzes
Another caveat: what reference?

• How close is your sample’s genome to the published available reference one?

• Specific kind of data, such as RNA-Seq:
  • genome or transcriptome?
  • what about novel exon-exon junctions?
Reference modification

Unmapped reads (170M, 15% of total)

- Multi-hits in genome, 6.5M, 4%
- Unknown, 21M, 12%
- Assembly to contigs, 29M, 17%
- SNP+IndelInjected_Human.GRC37, 6.1M, 4%
- Human.GRC37_contigs, 2.7M, 2%
- All bacteria and viruses, 3.4K, 0%
- Bad quality, 104.5M, 61%
Personalized reference

- Identify SNPs and indels
- Inject them into the “reference” genome
- A “personalized” genome that rescues “only” ~4% of unmapped reads
- but significantly reduces false positive SNPs
Technical artifact or amazing new biology?

• A recent paper that spills a lot of taint:
  • Li et al. Widespread RNA and DNA Sequence Differences in the Human Transcriptome. Science (2011)

• Major critics (Joe Pickrell):
  • http://www.genomesunzipped.org/2011/05/notes-on-the-evidence-for-extensive-rna-editing-in-humans.php
What they did

• Compare RNA and DNA from matched samples
  • observe numerous events where RNA != DNA
  • process known as RNA editing
  • known in human:
    • an enzyme convert A into I (Inosine) recognized as a G during translation
    • another less frequently observed event from another enzyme:
      • C -> U
  • BUT they observe all possible conversions!
What might be

• They use reads aligning uniquely to the genome.

• The main point can be summarized like this: RNA editing involves the production of two different RNA and/or protein sequences from a single DNA sequence. To infer RNA editing from the presence of two different RNA and/or protein sequences, then, one must be very sure that they derive from the same DNA sequence, rather than from two different copies of the DNA (due to, for example, paralogs or copy number variants).
Table 1. Selected examples of sites that show RNA-DNA Differences in B-cells and EST clones.

<table>
<thead>
<tr>
<th>Gene</th>
<th>Chr</th>
<th>Position (bp)*</th>
<th>Type</th>
<th>No. of informative individuals*</th>
<th>No. of individuals with RDD*</th>
<th>Average level† [range]</th>
<th>EST</th>
</tr>
</thead>
<tbody>
<tr>
<td>HSP90AB1</td>
<td>6</td>
<td>44,926,023</td>
<td>A-to-G</td>
<td>11</td>
<td>6</td>
<td>0.30 [0.15, 0.70]</td>
<td>DQ355103 (head neck), BX119000 (B cell)</td>
</tr>
<tr>
<td>AZIN1</td>
<td>8</td>
<td>103,910,812</td>
<td>A-to-G</td>
<td>17</td>
<td>10</td>
<td>0.22 [0.12, 0.37]</td>
<td>CD359333 (testis), BF479570 (prostate)</td>
</tr>
<tr>
<td>CNNP</td>
<td>3</td>
<td>180,372,842</td>
<td>A-to-T</td>
<td>16</td>
<td>16</td>
<td>0.18 [0.10, 0.24]</td>
<td>EL055100 (oeye), BJ005100 (hepatoblastoma)</td>
</tr>
<tr>
<td>MYL6</td>
<td>12</td>
<td>54,841,626</td>
<td>G-to-A</td>
<td>16</td>
<td>16</td>
<td>0.35 [0.12, 0.60]</td>
<td>EC406428 (prostate), BG030232 (breast adenocarcinoma)</td>
</tr>
<tr>
<td>RBM23</td>
<td>14</td>
<td>22,440,217</td>
<td>C-to-G</td>
<td>11</td>
<td>5</td>
<td>0.18 [0.11, 0.35]</td>
<td>BG232763 (testis, embryonic)</td>
</tr>
<tr>
<td>RPL23</td>
<td>17</td>
<td>34,363,515</td>
<td>C-to-T</td>
<td>12</td>
<td>8</td>
<td>0.16 [0.10, 0.22]</td>
<td>BP206252 (smooth muscle), CK128701 (embryonic stem cell)</td>
</tr>
<tr>
<td>BLNK</td>
<td>10</td>
<td>97,937,643</td>
<td>G-to-A</td>
<td>14</td>
<td>7</td>
<td>0.14 [0.11, 0.17]</td>
<td>BR 972964 (leiomyosarcoma), BL081159 (lung carcinoma)</td>
</tr>
<tr>
<td>G17orf70</td>
<td>17</td>
<td>77,117,503</td>
<td>G-to-C</td>
<td>2</td>
<td>2</td>
<td>0.20 [0.24, 0.20]</td>
<td>AA625546 (melanocyte), AA661870 (prostate)</td>
</tr>
<tr>
<td>HMGN2</td>
<td>1</td>
<td>26,674,348</td>
<td>G-to-T</td>
<td>7</td>
<td>4</td>
<td>0.22 [0.14, 0.43]</td>
<td>BX383385 (neuroblastoma), BE001308 (breast)</td>
</tr>
<tr>
<td>OANX</td>
<td>5</td>
<td>170,000,533</td>
<td>T-to-A</td>
<td>0</td>
<td>8</td>
<td>0.20 [0.13, 0.30]</td>
<td>EL050952, DB558108</td>
</tr>
<tr>
<td>EIF3K</td>
<td>13</td>
<td>40,019,436</td>
<td>T-to-G</td>
<td>13</td>
<td>14</td>
<td>0.16 [0.10, 0.27]</td>
<td>AI250281 (ovarian carcinoma), AI345303 (lung carcinoma)</td>
</tr>
<tr>
<td>RPL57</td>
<td>3</td>
<td>40,071,072</td>
<td>T-to-G</td>
<td>6</td>
<td>6</td>
<td>0.27 [0.10, 0.45]</td>
<td>CI 124792 (T cell), DW459223 (liver)</td>
</tr>
</tbody>
</table>

* hg18 build of the human genome
† B-cells
∥ RNA-Seq ≥ 10 reads, DNA-Seq ≥ 4 reads
‡ Calculated by tallying RNA-Seq reads that contain RDD and those that do not.
More pleasant news

- Bioconductor offers many new possibilities including:
  - pattern matching,
  - pairwise alignment,
  - SNPs injection
  - ...

The Biostrings package

- All the classes in that package derives from the `XString` class

```r
> library(Biostrings)
> getClass("XString")
Virtual Class "XString" [package "Biostrings"]

Slots:
Name: shared offset length elementMetadata elementType metadata
Class: SharedRaw integer integer ANY character list

Extends:
Class "XRaw", directly
Class "XVector", by class "XRaw", distance 2
Class "Sequence", by class "XRaw", distance 3
Class "Annotated", by class "XRaw", distance 4

Known Subclasses: "BString", "DNAString", "RNAString", "AAString"
```  

- There are 4 subclasses:
  - `BString`: store strings without alphabet
  - `DNAString`: store strings with an DNA alphabet
  - `RNAString`: store strings with an RNA alphabet
  - `AAString`: store strings with an Amino Acid alphabet
XString Methods

• Basic utilities
  • subsequence selection
    • subseq, Views, narrow (XStringSet, IRanges package)
  • letter frequencies
    • alphabetFrequency, dinucleotideFrequency (tri..., oligo...), uniqueLetters
  • letter consensus
    • consensusMatrix, consensusString
  • letter transformation
    • reverse, complement, reverseComplement, translate, chartr
• Input / Output
  • read.DNAStringSet (...B..., ...RNA..., ...AA...)
  • write.XStringSet, save.XStringSet
XString Methods (c’ed)

• Advanced:
  • alignment utilities
    • pairwiseAlignment, stringDist
  • string matching
    • matchPDict (on a reference or a reference set (v))
      • (v)matchPDict, (v)countPDict, (v)whichPDict
    • matchPattern
      • (v)matchPattern, (v)countPattern, neditStartingAt, neditEndingAt, (which.)isMatchingStartingAt, (which.)isMatchingEndingAt
    • matchPWM
      • matchPWM, countPWM
  • others
    • matchLRPatterns, trimLRPatterns, matchProbePair, findPalindromes, findComplementedPalindromes
Example 4: String Matching

• **Match counting**

```r
> data(phiX174Phage)
> phiX174Phage
A DNAStringSet instance of length 6
  width  seq
[1] 5386 GAGTTTTATCAGCTTCCGAGACAGGTTATACCTTTGGATATTCTGTATGAGTACGAAAAATTATCTTGA
[2] 5386 GAGTTTTATCAGCTTCCGAGACAGGTTATACCTTTGGATATTCTGTATGAGTACGAAAAATTATCTTGA
[3] 5386 GAGTTTTATCAGCTTCCGAGACAGGTTATACCTTTGGATATTCTGTATGAGTACGAAAAATTATCTTGA
[4] 5386 GAGTTTTATCAGCTTCCGAGACAGGTTATACCTTTGGATATTCTGTATGAGTACGAAAAATTATCTTGA
[5] 5386 GAGTTTTATCAGCTTCCGAGACAGGTTATACCTTTGGATATTCTGTATGAGTACGAAAAATTATCTTGA
[6] 5386 GAGTTTTATCAGCTTCCGAGACAGGTTATACCTTTGGATATTCTGTATGAGTACGAAAAATTATCTTGA
> genome <- phiX174Phage[["NEB03"]]
> negPhiX174 <- reverseComplement(phiX174)
> posCounts <- countPDict(PDict(phiX174), genome)
> negCounts <- countPDict(PDict(negPhiX174), genome)
> table(posCounts, negCounts)
  negCounts
posCounts  0
   1 1030
   1  83
>
```

• So we have 1030 reads that do not align either way to the genome and only 83 aligning (and don’t ask me why...).

• The match locations can be found using:

```r
> matchPDict(PDict(phiX174[posCounts > 0]), genome)
MIndex object of length 83
```
Example 5: Pairwise alignment

- alignment scores

```r
> posScore <- pairwiseAlignment(srPhiX174, genome,
  + type = "global-local", scoreOnly = TRUE)
> negScore <- pairwiseAlignment(negPhiX174, genome,
  + type = "global-local", scoreOnly = TRUE)
> which(pmin(posScore) < pmin(negScore))
[1] 932
```

- alignment

```r
> pairwiseAlignment(srPhiX174[932], genome, type = "global-local")
Global-Local PairwiseAlignedFixedSubject (1 of 1)
pattern: [1] GCAATAACCTTGGAGTCTATTTCTTTGATTTGTC
subject: [2804] GCAATAATGTTTATGTTGGTTCTGATGG-TTTGGTC
score: 33.3176
> pairwiseAlignment(negPhiX174[932], genome, type = "global-local")
Global-Local PairwiseAlignedFixedSubject (1 of 1)
pattern: [1] GACCAAATCAAAGAAATGACTGCAAGGTTATTC
subject: [3666] GACCAAATCAAAGAAATGACTGCAAGGTTATTC
score: 61.4804
```
The next level

• Biostrings offers tools to deal with biologically meaningful intervals and objects.

• Many organism have been sequenced and their genome is known.

• An interface in R to easily access and manipulate such information: the BSgenome package.
BSgenome

- It is not just a data package; it leverages the functionalities introduced in **Biostrings**.
BSgenome methods

• Sequence selection
  • [[, $

• Subsequence selection
  • getSeq

• Accessors
  • length, names/seqnames, mseqnames, seqlengths, masknames, sourceUrl

• Matching
  • all Biostrings methods

• SNPs
  • injectSNPs, SNPlocs_pkgname, SNPcount, SNPlocs
Extending Biostrings: example 1

- Applying the Biostrings matching functions:

```r
> exclude <- setdiff(seqnames(Hsapiens), c("chr1", "chr2"))
> vcountPattern("ACYTANCAGT", Hsapiens,
+ fixed = c(pattern = FALSE, subject = TRUE),
+ exclude = exclude)

  seqname strand count
  1  chr1     +    1546
  2  chr1     -    1545
  3  chr2     +    1722
  4  chr2     -    1684

> vmatchPattern("ACYTANCAGT", Hsapiens,
+ fixed = c(pattern = FALSE, subject = TRUE),
+ exclude = exclude, asRangedData = FALSE)

GRanges with 6497 ranges and 0 elementMetadata values

  seqnames ranges strand  |
  <Rle> <IRanges> <Rle>  |
  [1] chr1 [361581, 361590]  +  
  [2] chr1 [1738000, 1738009]  +  
  [3] chr1 [1814381, 1814390]  +  
  [4] chr1 [1876408, 1876417]  +  
  [5] chr1 [1878327, 1878336]  +  
  [6] chr1 [2084437, 2084446]  +  
  [7] chr1 [2976788, 2976797]  +  
```
Example 2

• Using a Pattern Dictionary, e.g. a library of microarray probes

```r
> library(hgu95av2probe)
> probes <- DNAStringSet(hgu95av2probe$sequence[1:100])
> probes[1:10]

A DNAStringSet instance of length 10

width  seq
[1] 25 TGGCTCCTGCTGAGGTCCCCCTTTCC
[2] 25 GGCTGTGAAATTCCTGTACATATTT
[3] 25 GCCTCAATTTCCATTATGTTTTAATG
[4] 25 GCCGTTTGCAGACGCCATGCTCTTGCC
[5] 25 TGACAGACGATGCTCTGGTGTTGG
[6] 25 CCTCCTGGTGGTTTCCCCAAGCT
[7] 25 GTTCTCAGGTCATCTTCCTTCACCA
[8] 25 TCCTCGGCTTACATCGAGGGTAT
[9] 25 CTCACATGCGACAGGAGTTAAAC
[10] 25 TCCCTGTCATCAGTCTGGCTGTC

> counts <- vcountPDict(probes, Hsapiens, exclude=exclude)

> counts

Dataframe with 400 rows and 4 columns

<table>
<thead>
<tr>
<th>seqname</th>
<th>strand</th>
<th>index</th>
<th>count</th>
</tr>
</thead>
<tbody>
<tr>
<td>chr1</td>
<td>+</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>chr1</td>
<td>+</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>chr1</td>
<td>+</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>chr1</td>
<td>+</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>chr1</td>
<td>+</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>chr1</td>
<td>+</td>
<td>6</td>
<td>0</td>
</tr>
<tr>
<td>chr1</td>
<td>+</td>
<td>7</td>
<td>0</td>
</tr>
<tr>
<td>chr1</td>
<td>+</td>
<td>8</td>
<td>0</td>
</tr>
<tr>
<td>chr1</td>
<td>+</td>
<td>9</td>
<td>0</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>chr2</td>
<td>-</td>
<td>92</td>
<td>0</td>
</tr>
<tr>
<td>chr2</td>
<td>-</td>
<td>93</td>
<td>0</td>
</tr>
<tr>
<td>chr2</td>
<td>-</td>
<td>94</td>
<td>0</td>
</tr>
<tr>
<td>chr2</td>
<td>-</td>
<td>95</td>
<td>0</td>
</tr>
<tr>
<td>chr2</td>
<td>-</td>
<td>96</td>
<td>0</td>
</tr>
<tr>
<td>chr2</td>
<td>-</td>
<td>97</td>
<td>0</td>
</tr>
<tr>
<td>chr2</td>
<td>-</td>
<td>98</td>
<td>0</td>
</tr>
<tr>
<td>chr2</td>
<td>-</td>
<td>99</td>
<td>0</td>
</tr>
<tr>
<td>chr2</td>
<td>-</td>
<td>100</td>
<td>0</td>
</tr>
</tbody>
</table>

> whichMatch <- seqselect(counts$index, counts$count>0)
> whichMatch

[1] 1 2 3 4 5 6 7 8 9 10 11 12 13 14 16

> matchedProbes <- probes[whichMatch]
> matchedProbes

A DNAStringSet instance of length 15

width  seq
[1] 25 TGGCTCCTGCTGAGGTCCCCCTTTCC
[2] 25 GGCTGTGAAATTCCTGTACATATTT
[3] 25 GCCTCAATTTCCATTATGTTTTAATG
[4] 25 GCCGTTTGCAGACGCCATGCTCTTGCC
[5] 25 TGACAGACGATGCTCTGGTGTTGG
[6] 25 CCTCCTGGTGGTTTCCCCAAGCT
[7] 25 GTTCTCAGGTCATCTTCCTTCACCA
[8] 25 TCCTCGGCTTACATCGAGGGTAT
[9] 25 CTCACATGCGACAGGAGTTAAAC
[10] 25 TCCCTGTCATCAGTCTGGCTGTC
[12] 25 GGCTGTGAAATTCCTGTACATATTT
[13] 25 GCCTCAATTTCCATTATGTTTTAATG
[14] 25 GCCGTTTGCAGACGCCATGCTCTTGCC
[15] 25 TGACAGACGATGCTCTGGTGTTGG

> matchLocs <- matchPDict(PDict(matchedProbes), Hsapiens$chr2)
> matchLocs

MIndex object of length 15

Views on a 243199373-letter DNAString subject

subject: NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
Example 3

- A new interesting feature is the possibility to inject SNPs!
Acknowledgments

• Lars Steinmetz and his lab, especially:
  • Jonathan Landry
  • Julien Gagneur
  • Xing Xiaobin

• Eileen Furlong and her lab, especially:
  • Charles Girardot

• Wolfang Huber and his lab, especially:
  • Simon Anders

• Vladimir Benes and his Gene Core facility:
  • Tobias Rausch
  • Jonathon Blake