ITX Integrative Transcriptomics
Dr. K. Nieselt, Faculty of Science

EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

October 5 -7, 2015

Workshop: Visual Analytics of large-scale biological data

Practical Session: The Pan-genome

Theoretical Background

In bacteria individual strains within one species can show extensive variation in their gene content, such that either individual genes or larger clusters of genes can be lost or newly acquired by horizontal gene transfer. In particular, in pathogenic strains of a bacterial species the degree of virulence can be attributed to the absence or presence of genes. This latter observation has led to the coining of the term pan-genome, which traditionally encompasses the full repertoire of all genes of a bacterial species.
In this part of the workshop, we are going to construct, visualize and analyze the pan-genome of four different *Staphylococcus aureus* strains, based on a multiple genome alignment. First, the challenge will be to find homologous genes (often called orthologs) through all individuals to construct the pan-genome.

Practical Part

1. **Computing a whole genome alignment**

   Whole genome alignments (WGA) are often used in bioinformatics to detect (or align) conserved regions and to determine evolutionary relationships among the genomes. The construction of these alignments is computationally demanding, because evolutionary events like rearrangements as well as segmental duplication, gene gain and gene loss have to be considered.
   A common used tool for this task is *Mauve* (`http://darlinglab.org/mauve/mauve.html`), which only needs the genome sequences in a FASTA format. Here, is a quick demonstration how to use `progressiveMauve` (recommended) from the command-line:

   ```
   >progressiveMauve --output=<filename>.xmfa <genome 1> <genome 2> ... <genome n>
   ```

   Please make sure that the files for the genomes are either in FASTA or in GenBank file format and the output variable contains the desired output folder location (e.g. /home/nieselt/mauve/alignment.xmfa).

   (a) Take a look at your genome files (fasta format)
   (b) Type `progressiveMauve` in your console and get an overview of all available options
   (c) Generate the command to align all genomes with `progressiveMauve` (order of genomes is not important)
   (d) Inspect the output of `progressiveMauve` (only the xmfa file)

2. **Pan-Genome**

Based on the WGA and individual genome annotations, we are going to use the platform independent Java-based application `PanGee` for the computation of the pan-genome. Based on our SuperGenome approach, which represents a coordinate system between the individual genomes and the alignment, we can check, which annotations are aligned in the WGA and thus are supposedly orthologous. Most of these "overlapping" genes are then, with further validation and processing, be identified as orthologs. In the end, the set of all orthologous gene groups represents the pan-genome.

(a) Take a look at an annotation file (gff format) with a text editor. Which information can you find there?

(b) Start `PanGee` on your console by typing:

```
>java -Xmx1G -jar PanGee.jar
```

Note: The java-specific option -Xmx determines the size of RAM java can use.

A graphical user interface should appear, which lets you specify all needed input. Do not bother about the parameter settings, the default settings are in most cases appropriate.

(c) Set the path to the alignment, genome sequences (fasta files) and annotations (gff file), respectively. The easiest way is to put all files in one folder and set the respective path to this folder. The program should automatically recognize the order of the genomes and the respective sequence and annotation filse.

(d) After you have done that, simply press the "Run" button to start the calculation.

(e) Take a look at your results. In the `PanGenomeMap` file you will find all pan-genes, ie. all orthologous groups. The ordering of them is based on their appearance in the WGA, meaning that a group that appears at the beginning of the alignment appears also at the beginning of the file.

3. Visualization

Now that we computed the pan-genome of the four *S. aureus* strains, we want to visualize it. For this we are going to use the software `Pan-Tetris`, which visualizes the content of the PanGenomeMap in a tetris-like style and let you explore the whole pan-genome.

(a) First take a look oat a short video demonstration of how `Pan-Tetris` works (`https://www.youtube.com/watch?v=_G9fUi3PdPM`).

(b) Start `Pan-Tetris` on your console by typing:

```
>java -Xmx1G -jar PanTetris.jar
```

(c) As you could see in the video, the genes can be colored based on their respective function. The differentiation of the gene function is based on TIGRFAM annotation (`http://www.jcvi.org/cgi-bin/tigrfams/index.cgi`). We provided you with a file of the TIGRFAM annotation for the respective *S. aureus* strains.
Load the TIGRFAM annotations in addition to the PanGenomeMap.

(d) Explore the pan-genome and point out interesting orthologous groups or patterns. Can you think of a biological or technical answer for these?

(e) From a perspective of a person in the field of visualizations, which aspects of `Pan-Tetris` are questionable? Can you think of a way how to improve these?