VAR-Seq project report template: Some Descriptive Title

Project ID: RNAseq_PI_Name_Organism_Jun2014
Project PI: First Last (first.last@inst.edu)
Author of Report: First Last (first.last@inst.edu)

June 29, 2014

Contents

1 Introduction

This report describes the analysis of an VAR-Seq project from Dr. First Last's lab which studies the genetic variances of ... in *Organism x*. The experimental design is as follows...

2 Sample definitions and environment settings

2.1   Environment settings and input data

Typically, the user wants to record here the sources and versions of the reference genome sequence along with the corresponding annotations. In the provided sample data set all data inputs are stored in a `data` subdirectory and all results will be written to a separate `results` directory, while the `systemPipeVARseq.Rnw` script and the `targets` file are expected be located in the parent directory. The R session is expected to run from this parent directory.

To run the this sample report, mini sample FASTQ and reference genome files can be downloaded from here. The chosen data set SRP010938 contains 18 paired-end (PE) read sets from *Arabidposis thaliana* Howard et al. (2013). To minimize processing time during testing, each FASTQ file has been subsetted to 90,000-100,000 random sampled PE reads that map to the first 100,000 nucleotides of each chromosome of the *A. thalina* genome. The corresponding reference genome sequence (FASTA) and its GFF annotation files (provided in the same download) have been truncated accordingly. This way the entire test sample data set is less than 200MB in storage space. A PE read set has been chosen for this test data set for flexibility, because it can be used for testing both types of analysis routines requiring either SE (single end) reads or PE reads.

2.2   Required packages and resources

The *systemPipeR* package needs to be loaded to perform the analysis steps shown in this report (Girke, 2014).

```
> library(systemPipeR)
```

If applicable load custom functions not provided by *systemPipeR*

```
> source("systemPipeVARseq_Fct.R")
```

2.3   Experiment definition provided by `targets` file

The `targets` file defines all FASTQ files and sample comparisons of the analysis workflow.

```
> targetspath <- paste0(system.file("extdata", package="systemPipeR"), "/targets.txt")
> targets <- read.delim(targetspath, comment.char = "#")[,1:4]
> targets
                  FileName SampleName Factor SampleLong
1   ./data/SRR446027_1.fastq       M1A     M1   Mock.1h.A
2   ./data/SRR446028_1.fastq       M1B     M1   Mock.1h.B
3   ./data/SRR446029_1.fastq       A1A     A1    Avr.1h.A
4   ./data/SRR446030_1.fastq       A1B     A1    Avr.1h.B
5   ./data/SRR446031_1.fastq       V1A     V1    Vir.1h.A
6   ./data/SRR446032_1.fastq       V1B     V1    Vir.1h.B
7   ./data/SRR446033_1.fastq       M6A     M6   Mock.6h.A
8   ./data/SRR446034_1.fastq       M6B     M6   Mock.6h.B
9   ./data/SRR446035_1.fastq       A6A     A6    Avr.6h.A
10  ./data/SRR446036_1.fastq       A6B     A6    Avr.6h.B
11  ./data/SRR446037_1.fastq       V6A     V6    Vir.6h.A
12  ./data/SRR446038_1.fastq       V6B     V6    Vir.6h.B
13  ./data/SRR446039_1.fastq      M12A    M12  Mock.12h.A
14  ./data/SRR446040_1.fastq      M12B    M12  Mock.12h.B
15  ./data/SRR446041_1.fastq      A12A    A12   Avr.12h.A
16  ./data/SRR446042_1.fastq      A12B    A12   Avr.12h.B
17  ./data/SRR446043_1.fastq      V12A    V12   Vir.12h.A
18  ./data/SRR446044_1.fastq      V12B    V12   Vir.12h.B
```

3 Read preprocessing

3.1 FASTQ quality report

The following `seeFastq` and `seeFastqPlot` functions generate and plot a series of useful quality statistics for a set of FASTQ files including per cycle quality box plots, base proportions, base-level quality trends, relative k-mer diversity, length and occurrence distribution of reads, number of reads above quality cutoffs and mean quality distribution. The results are written to a PDF file named `fastqReport.pdf`.

```
> args <- systemArgs(sysma="tophat.param", mytargets="targets.txt")
> fqlist <- seeFastq(fastq=infile1(args), batchsize=100000, klength=8)
> pdf("./results/fastqReport.pdf", height=18, width=4*length(fqlist))
> seeFastqPlot(fqlist)
> dev.off()
```
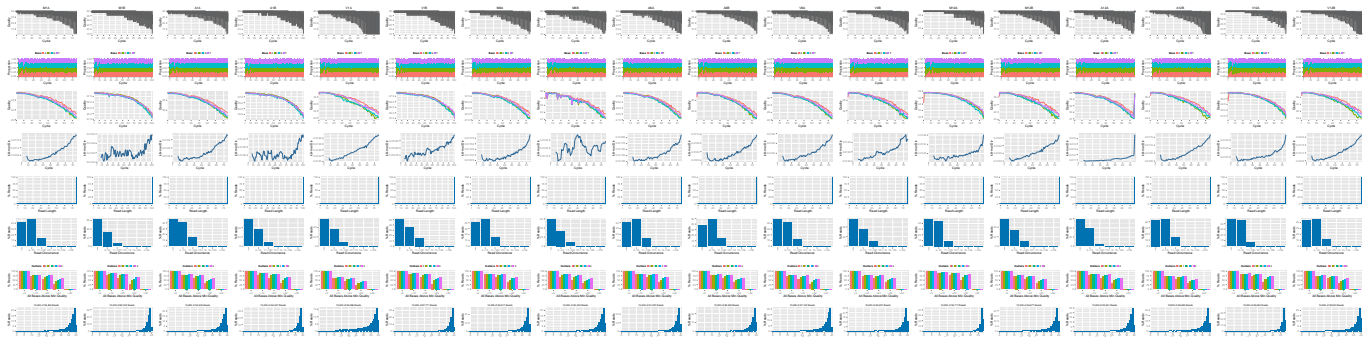

Figure 1: QC report for 18 FASTQ files.

4 Alignments

4.1 Read mapping with BWA

The NGS reads of this project will be aligned against the reference genome sequence using BWA (Li, 2013; Li and Durbin, 2009). The parameter settings of the aligner are defined in the `bwa.param` file.

```
> args <- systemArgs(sysma="bwa.param", mytargets="targets.txt")
> sysargs(args)[1] # Command-line parameters for first FASTQ file
```

Submission of alignment jobs to compute cluster, here using 72 CPU cores (18 qsub processes each with 4 CPU cores).

```
> moduleload(modules(args))
> system("bowtie2-build ./data/aedes-aegypti-liverpool_scaffolds_AaegL3.fa ./data/aedes-aegypti-liverpool_s
> qsubargs <- getQsubargs(queue="batch", cores=cores(args), memory="mem=10gb", time="walltime=20:00:00")
> (joblist <- qsubRun(args=args, qsubargs=qsubargs, Nqsubs=18, package="systemPipeR"))
```

Check whether all BAM files have been created

```
> file.exists(outpaths(args))
```

4.2 Read and alignment stats

The following provides an overview of the number of reads in each sample and how many of them aligned to the reference.

```
> read_statsDF <- alignStats(args=args, fqgz=TRUE)
> write.table(read_statsDF, "results/alignStats.xls", row.names=FALSE, quote=FALSE, sep="\t")

> read.delim("results/alignStats.xls")
```

4.3   Create symbolic links for viewing BAM files in IGV

The `symLink2bam` function creates symbolic links to view the BAM alignment files in a genome browser such as IGV.
The corresponding URLs are written to a file with a path specified under `urlfile`, here IGVurl.txt.

```
> symLink2bam(sysargs=args, htmldir=c("~/.html/", "projects/AlexRaikhel/2014/"),
+             urlbase="http://biocluster.ucr.edu/~tgirke/",
+             urlfile="./results/IGVurl.txt")
```

5   Variant calling

To be continued...

6   Annotating variants

To be continued...

7   Summary statistics of variants

To be continued...

8   Version Information

```
> toLatex(sessionInfo())
```

- R version 3.1.0 (2014-04-10), `x86_64-unknown-linux-gnu`
- Locale: `C`
- Base packages: base, datasets, grDevices, graphics, methods, parallel, stats, utils
- Other packages: AnnotationDbi 1.26.0, Biobase 2.24.0, BiocGenerics 0.10.0, DBI 0.2-7, GenomeInfoDb 1.0.2, RSQLite 0.11.4, systemPipeR 1.0.11
- Loaded via a namespace (and not attached): AnnotationForge 1.6.1, BBmisc 1.7, BSgenome 1.32.0, BatchJobs 1.2, BiocParallel 0.6.1, BiocStyle 1.2.0, Biostrings 2.32.0, Category 2.30.0, GO.db 2.14.0, GOstats 2.30.0, GSEABase 1.26.0, GenomicAlignments 1.0.1, GenomicRanges 1.16.3, IRanges 1.22.9, MASS 7.3-33, Matrix 1.1-4, RBGL 1.40.0, RColorBrewer 1.0-5, Rcpp 0.11.2, Rsamtools 1.16.1, ShortRead 1.22.0, XML 3.98-1.1, XVector 0.4.0, annotate 1.42.0, bitops 1.0-6, brew 1.0-6, checkmate 1.0, codetools 0.2-8, colorspace 1.2-4, digest 0.6.4, edgeR 3.6.2, fail 1.2, foreach 1.4.2, genefilter 1.46.1, ggplot2 1.0.0, graph 1.42.0, grid 3.1.0, gtable 0.1.2, hwriter 1.3, iterators 1.0.7, lattice 0.20-29, latticeExtra 0.6-26, limma 3.20.6, munsell 0.4.2, plyr 1.8.1, proto 0.3-10, reshape2 1.4, rjson 0.2.14, scales 0.2.4, sendmailR 1.1-2, splines 3.1.0, stats4 3.1.0, stringr 0.6.2, survival 2.37-7, tools 3.1.0, xtable 1.7-3, zlibbioc 1.10.0

9 Funding

10 References

Thomas Girke. systemPipeR: NGS workflow and report generation environment, 28 June 2014. URL https://github.com/tgirke/systemPipeR.

Brian E Howard, Qiwen Hu, Ahmet Can Babaoglu, Manan Chandra, Monica Borghi, Xiaoping Tan, Luyan He, Heike Winter-Sederoff, Walter Gassmann, Paola Veronese, and Steffen Heber. High-throughput RNA sequencing of pseudomonas-infected arabidopsis reveals hidden transcriptome complexity and novel splice variants. *PLoS One*, 8 (10):e74183, 1 October 2013. ISSN 1932-6203. doi: 10.1371/journal.pone.0074183. URL http://dx.doi.org/10.1371/journal.pone.0074183.

H Li and R Durbin. Fast and accurate short read alignment with Burrows-Wheeler transform. *Bioinformatics*, 25(14): 1754–1760, July 2009. ISSN 1367-4803. doi: 10.1093/bioinformatics/btp324. URL http://dx.doi.org/10.1093/bioinformatics/btp324.

Heng Li. Aligning sequence reads, clone sequences and assembly contigs with BWA-MEM. 03 2013. URL http://arxiv.org/abs/1303.3997.