Lions User Guide

Dr. Richard Thompson

June 10, 2018

1 LIONS Pipeline Architecture

LIONS is a bioinformatic analysis pipeline which brings together a few pieces of software and some home-brewed scripts to annotate a paired-end RNAseq library against a reference TE annotation set (such as Repeat Masker).

East Lion proccesses a bam file input, re-aligns it to a genome, builds an ab initio assembly using Tophat2. This assembly is then proccessed and local read searches are done at the 5' ends to find additional transcript start sites and quality control the 5' ends of the assembly. The output is a file-type <library> .lions which annotates the intersection between the assembly, a reference gene set and repeat set.

West Lion compiles different .lions files, groups them into biological catagories (i.e. Cancer vs. Normal or Treatment vs. Control) and compares and analyzes the data to create graphs and meaningful interpretation of the data.

2 LIONS Installation Steps

1. Download LIONS core files from github
2. In ./LIONS/resources copy 'example' folder to '<genomeName>' folder
3. Populate the resource files:
   (a) In ./LIONS/resources/<genomeName>/genome/ add a <genomeName>.fa genome sequence file
   (b) In ./LIONS/resources/<genomeName>/annotation/ add the ucsc annotation file
   (c) In ./LIONS/resources/<genomeName>/repeat/ add the ucsc repeatMasker file
4. Set up the project in parameter.ctrl:
   (a) Give your project a name and fill out all the parameters as you see fit.
   (b) In the software list, refer LIONS to the paths for each software on your system. If it's in $BIN/ already then you can simply type the command

3 LIONS File Architecture

3.1 ./

base directory: LIONS is a self-contained pipeline and references needed by LIONS from the system are linked within this directory.

./lions.sh <parameter.ctrl> The master script from which the entire pipeline is ran The script reads and procceses all files in input.list and all parameters can be controlled from parameter.ctrl

3.2 ./controls

Control Files: This folder contains project and system-specific parameters for running LIONS. There are three main files which need to be set-up, LIONS run parameters, system parameters and input RNA-seq libraries.

./controls/parameter.ctrl A bash script which defines global project-specific variables such as Project Name, library input list etc... The .sysctrl and .list file are defined here as well

./controls/system.sysctrl A bash script which defines global variables for all LIONS scripts. Also defines system-specific variables such as System Name, number of CPU cores etc...

./controls/input.list A three column tab-delimited file defining: <Library_Name><Library_Path_on_system><Biological_Grouping> Group Naming Convention: 1 = control 2 = experimental 3 = other

Binary: A folder for symbolic links to binaries needed by the pipeline and script to initilize the folder. Make sure to set the correct commands for the software list in parameters.ctrl for your system

3.3 ./projects

For each <Project_Name> a single folder will be initilized in which the data will be organized.

./projects/<Project_Name> The main directory for this project. Each individual library in the input will have a folder generated here called <Library>.

./projects/<Project_Name>/logs Folder contains run-specific information such as input file at time of run and a copy of the input parameter file at time of run.

./projects/<Project_Name>/Analysis(_RUNID) Not implemented yet. This contains all data analysis for a run of LIONS. All graphs and project-wide .lions files are stored here

./projects/<Project_Name>/<Library> Library-specific data and primary analysis files.

./projects/<Project_Name>/<Library>/<Library>.lcsv Raw output file from LIONS containing all possible TE-Exon interaction data. This will include initiations, exonizations and terminations along with many calculated values about these loci from which LIONS will sort initiation events from the others. <Library>.pc.lcsv is the same file with additional information about overlapping protein coding genes.

./projects/<Project_Name>/<Library>/<Library>.lions Initiation only TE-exon data from post-sorting. This file is the complete list of transcripts initiated by TEs in this library. This data is passed on to the West Lion protocol to compare TE usage between libraries.

./projects/<Project_Name>/<Library>/alignment tophat2-generated alignment and the re-aligned .bam file which will be used for analysis. Also contains flagstats and log files. Note: Once the alignment is generated it will not be re-generated even if you change the alignment parameters in parameter.ctrl. To re-make alignments simply start the project with a new name or delete these files.

./projects/<Project_Name>/<Library>/assembly cufflinks-generated assembly in 'transcripts.gtf'

./projects/<Project_Name>/<Library>/expression The output from a series of custom scripts 'RNAseqPipeline' which will generate wig files and perform RPKM calculations on a series

./projects/<Project_Name>/<Library>/resources Charlie-foxtrot of library-specific files used to calculate a score of parameters in the pipeline.

Scripts: all scripts to run lions are held here except for the controlling lions.sh script which is in the base folder. Check initializeScripts.sh for complete list of scripts The main scripts are

./scripts/eastLion.sh Alignment, Assembly, Chimeric Detection pipeline

./scripts/westLion.sh LIONS analysis pipeline

<INDEX>: the name of the index set. To be compatible with different genome versions, species and gene sets there can be different sets of data. The <INDEX> global variable is set in parameter.ctrl file

./resources/<INDEX>/annotation <GENESET> file: A ucsc formatted file containing a gene set to be used in the analysis (i.e. look for overlapping genes to transcripts) Download from: https://genome.ucsc.edu/cgi-bin/hgTables The standard annotation set used was RefSeq 'RefGene' table.

./resources/<INDEX>/genome <INDEX>.fa: The only requisite file here for running LIONS is a fasta formatted genome. This could be a symbolic link. LIONS will generate the other files neccesary from INDEX.fa. If you have the .bt2 index files already generated you can symbolically link them in this folder to skip re-generating them.

./resources/<INDEX>/repeat <REPEATMASKER>.ucsc: a ucsc formatted file containing the repeatMasker annotation for the genome. (Download from UCSC genome browswer or format from RepeatMasker) Columns are; bin, swScore, milliDiv, milliDel, milliIns, genoName, genoStart, genoEnd, genoEnd, genoLeft, strand, repName, repClass, repFamily, repStart, repEnd, repLeft, id

Packaged with LIONS is a few bits of software which will set-up your system to run the pipeline. Namely setuptools and pysam are the most challenging things to install. I found that it's easiest to set-up the pipeline using pip and download the package pysam from there. Pysam is used to read teh bam files in the python scripts.

4 LIONS Error Codes

Error 1 internal software error. Check last-run software.

4.1 Initialization Codes

Error 2 Initializal file missing or inaccesible A file is missing. Ensure you have a complete version of LIONS and/or make the missing script readable/exectable

Error 3 A LIONS script is missing A script is missing from ./LIONS/scripts/ ; ensure your copy of LIONS is complete or redownload.

Error 4 Initialization bin missing A binary is not found on the system. Configure ./LIONS/bin/intializeBin.sh for your system

Error 5 A resource file is missing or unreadable Checking/initialization of ./LIONS/scripts

Error 6 A python requisite is missing.

Error 7 The input read file (.bam or .fastq) is non-readable or empty

7A Bam file error

7B FastQ file error. Ensure the two files are comma seperated in the input

4.2 eastLion Error Codes

Error 10 alignment not generated An attempt was made to generate an alignment but the output file was empty at the end of the script

Error 12 wig not generated An attempt was made to generate the wig file but the output file wasn't there after the script ran

4.3 westLion Error Codes

Error 15 A lions file wasn't generated In the run, one of the lions files wasn't generated which means there was an error. Don't run West Lions pipeline.

5 LIONS output definitions

5.1 Output File-types

LIONS produces several outputs from different stages of the analysis apart from the standard outputs one would expect (.bam / .gtf).

'<library>.lcsv' / '.pc.lcsv' These are LIONS CSV files; that is the raw calculations for all major numeric operations. This includes ALL TE-exon interactions types (Initiation, Exonization and Termination). As such there are usually hundreds of thousands of TEs which have read fragments joining them to some assembled exon.

The '.pc.' pre-suffix means the data has been intersected to the input set of protein coding genes.

Use this file for re-calculating "TE-Initiations" with new parameters.

'<library>.lion' This is the filtered set of TE-exon interactions which have been classified as "TE-Initiations" or TE transcription start sites. This is per-library input.

'<project>.lions' A merged file of several '.lion' files combining biological groups defined in the 'input.list'. A good example of this is merging 10 cancer libraries and 10 normal libraries and outputing only those TE-initiations which are in at least 20% of Cancer and no Normal libraries. These parameters can be changed in the 'paramter.ctrl' input.

'<project>.rslions' The 'rs' is for Recurrent and Specific TE-initiations only. That is if you compare the set of libraries 1 (Normal) vs set 2 (Cancer), this contains only those TE-initiations which occur multiple times in Cancer (recurrant) and do not occur in Normal (specific). As defined by '$cgGroupRecurrence' and '$cgSpecificity' in the 'paramter.ctrl' file.

'<project>.inv.rslions' The '.inv.' pre-suffix is simply the **inverse** of the '.rslion' file. So instead of "Cancer vs. Normal", "Normal vs. Cancer". A necessary control if one makes any conclusions based on enrichment/depletion.

5.2 Output Columns

5.2.1 '.lions'

Most columns should be self-explanatory, some are not.

transcriptID Unique identifier for the transcript (isoform). Usually taken from the assembly/reference transcriptome

exonRankInTranscript For each TE-exon interaction combination (row) which exon in the 'transcriptID' is this row referring to

repeatName The <repeat_name>:<repeat_class>:<repeat_family> taken from input set

coordinates Useful coordinates for visualizing the interaction. It starts/ends in the exon and repeat so when opening in a visualization tool you can see the reads spanning this area.

ER_Interaction: The type of relative intersection in the genome between the exon and the repeat. Definitions are relative to the exon. Can be "Up", "UpEdge", "EInside", "RInside", "Down", "DownEdge".

IsExonic ??

ExonsOverlappingWithRepeat A list of <transcriptID:exonRank> which overlap the repeat.

ER / DR / DE / DD / Total A count of the number of TE-Exon sequence fragments which join this rows TE and Exon. ER means that one end overlaps the Exon and one end overlaps the Repeat exclusively, DD means that both ends of the fragment overlap both (dual) exon and repeat ...

Chromosome / EStart / EEnd / EStrand Start, end and strand of the exon

RStart / REnd / RStrand Start, end and strand of the repeat

RepeatRank Relative exon/intron position of the repeat to the contig

UpExonStart / UpExonEnd Coordinates used for calculating expression of genome immediatly adjacent an exon boundary. Useful for quantifying read- through or spurious transcriptional events.

UpThread The number of read 'threads' going upstream of the exon. See Manuscript for a figure explaining this.

DownThread The number of read 'threads' going downstream of the exon. See Manuscript for a figure explaining this.

ExonRPKM RPKM calculation for this exon

ExonMax The maximum coverage count reached within the exon boundaries. Often more reliable measure of expression then RPKM for small exons.

UpExonRPKM / UpExonMax The expression of the exon immediatly upstream of the one this row is referring to. (i.e. Exon 1 expression if the row refers to Exon 2). Useful for quantifying the relative increase in expression when a TE is acting as an alternative promoter into a downstream exon.

RepeatRPKM / RepeatMaxCoverage Expression level within repeat boundaries.

UpstreamRepeatRPKM / UpstreamRepeatMaxCoverage The expression adjacent to the repeat, a test for background expression levels.

RefID When intersecting to a reference gene set, the gene symbol of any genes which intersect the area between the Exon-Repeat coordinates.

RefStrand Strand of the reference genes defined above

assXref The strand-relationship between the reference gene and the contig exon This accounts for anti-sense long non-coding RNA (as), or transcripts which run anti-sense to the reference gene. (s) is sense and (c) means complex, often some combination of multiple genes. (u) means it could not be determined.

Contribution An estimate of the promoter contribution of this Repeat TSS to the expression of gene in total. Calculated with ExonMax and UpExonMax.

UpCov Ratio of the coverage adjacent to an exon and the exon expression

UpExonRatio Ratio of hte expression of the exon and it's upstream exon

ThreadRatio DownThread / UpThread. Set to [10] if dividing by zero.

RepeatID A unique Identifer for each Repeat in the genome (left-most coordinate). Can repeat and thus be used for determining one repeat inititating a trancsript in different assemblies.

LIBRARY Library from which this repeat-exon interaction was calculated from.

5.2.2 '.rslions'

Normal_occ Number of times this TE-initiation was found in the "normal" set of libraries. (Usually set 1)

Cancer_occ Number of times this TE-initiation was found in the "cancer" set of libraries. (Usually set 2)

Library A semi-colon seperated list of the LIBRARY identifiers in which this TE-initiation was found in.