################################

#ANS566/EEOB561X tutorial

#February 5, 2014

#Iowa State University

#Shell code by: James Koltes

#################################

OVERVIEW of goals for demo

1)Run PCIT

2)Get differential hubbing with the script PCIT.process.sh

3) OPTIONAL: Retrieve and break down target transcript correlates for use in gene ontology enrichment testing with the script correlationFetcherV2.sh

Test enrichment at DAVID online: http://david.abcc.ncifcrf.gov/

4)Add annotation to differential hubbing output files

5)View networks with Biolayout

############ INPUT FILES for the tutorial ############

#Note: the input data is mouse Affymetrix gene expression data

#For PCIT

null.txt

wt.txt

annotation.txt

DIFFERENTIAL EXPRESSION RESULTS

DElist.xls

INPUT DATA FORMAT AND DATA SIZE DESCRIPTIONS FOR PCIT

note: PCIT input files do NOT have headers!!!

nrows = total number of genes. There is one row for each unique gene.

column1 = gene IDs

column2 to column N = the normalized gene expression values with each column representing an individual microarray or sample analyzed for gene expression.

(Note: Here gene expression data was normalized by RMA)

PCIT specific file size description

nrows = total number of genes/ rows.

nconditions: refers to the number of columns, excluding the first ID column. i.e. if nconditions = 10, the file has 11 columns of data with the first column representing geneIDs and the other 10 columns representing normalized gene expression data.

##########################################################################

#1) RUNNING PCIT

#General information on the HPC-class cluster

http://hpcgroup.public.iastate.edu/HPC/hpc-class/

########## Getting started #########

1) Open a terminal window on your machine

Mac- open "Terminal" window

PC - open putty

2) Login to the ISU student cluster

ssh hpc-class.its.iastate.edu -l netID

where your netID should be the same as your email username

You may be asked to verity the authenticity of the connection- type yes and hit enter.

Then you will be prompted to provide a password (the same as you use for campus email).

3) Once you login, issue the following commands in the terminal:

bash

export PATH=${PATH}:/ptmp/jekoltes/public/bin/PCIT/

You will need to copy a directory from this location to your home directory (i.e. /home/netID/ )

cp -avr /ptmp/jekoltes/public/bin/PCIT/shellScripts /home/netID/

NOTE: if you get the message "cp: omitting directory `shellScripts/'" this is OK.

The export statement will give you access to my directory that contains the PCIT software and sample data for this tutorial. Please note, you are welcome to keep a copy of the PCIT code and sample data:

code = pcit-cd8292560a67.tar.bz2

sample data = wt.txt & null.txt (pcit.in is also included with the code for a larger example)

4) You will need to create a job script (or borrow one) to run your job. You can find two job scripts here for our demo:

shellScripts/pcit.null.job

shellScripts/pcit.wt.job

\*\*\*You will need to modify both of these 2 job files to work in your account

nano pcit.null.job

Within the file change these two paths: /home/jekoltes/shellScripts/wt.txt /home/jekoltes/shellScripts/wt.PCIT.out.txt

You should change the "jekoltes" to whatever your netID is!

To save the file, you will type the following keys (simultaneously)

control + x

y

enter (return)

This series of commands saves your file and closes nano (you will use this again below)

5) To submit a job use the command:

First- make sure you are within your shellScripts directory:

cd /home/username/shellScripts/

Then, issue the command to submit the jobs to the queue

qsub jobname

for example:

qsub pcit.null.job

To check the status of your job, type the command

qstat -a

#For more information on creating a job script on the HPC-class cluster

http://hpcgroup.public.iastate.edu/HPC/hpc-class/hpc-class\_script\_writer.html

##########################################################################

SUMMARIZING THE PCIT RESULTS

2) DIFFERENTIAL HUBBING WITH PCIT.process.sh

This script creates a number of filtered results file based on direct and partial correlation thresholds.

##when logged in on the cluster, move to the shellScripts directory (if you are not already there) and modify the job script named process.job

cd /home/your.netID/shellScripts/

nano process.job

change the three paths:

export PATH=${PATH}:/home/jekoltes/shellScripts/ &

/home/jekoltes/shellScripts/null.PCIT.out.txt & /home/jekoltes/shellScripts/wt.PCIT.out.txt

to reflect your netID where jekoltes is used in the current script

To save the file, you will type the following keys (simultaneously)

control + x

y

enter (return)

Then launch the job process.job

qsub process.job

##### IF you were to run these scripts from your own computer- NOT logged in on the cluster ###

To run the PCIT.process.sh script, issue the following command:

./PCIT.process.sh PCIT.file1 PCIT.file2 PC.correlation.threshold direct.correlation.Threshold

An example with actual file names might be:

./PCIT.process.sh null.PCIT wt.PCIT 0.5 0.9

In this case, you would filter based on partial correlations greater than the absolute value of 0.5 and direct correlations greater than the absolute value of 0.9

Why filter the data this way? The strength of the correlation between regulators may indicate target genes that are changing with the candidate regulator. These filtered correlates could be tesed with gene set enrichment analysis to determine if they have biologically meaningful results in the context of the gene expression experiment.

Take a look at the differential hubbing results in the file: differential.hubbing.txt

Can you glean information that was not evident from the differential expression (DE) results?

##########################################################################

3) RETRIEVE AND BREAKDOWN CORRELATES FOR USE IN GENE ONTOLOGY ENRICHMENT TESTING with: correlationFetcher.sh (OPTIONAL)

First, create the file selected.PCIT.regulators.txt: This is a list of regulators that you will search for in the PCIT output files.

#to create the file selected.PCIT.regulators.txt

head -6 differential.hubbing.ANNOTATED.txt | tail -5 > top5

tail -5 differential.hubbing.ANNOTATED.txt > bottom5

cat top5 bottom5 > top10.differential.hubbing.PCIT.txt

awk '{print $1}' top10.differential.hubbing.PCIT.txt > selected.PCIT.regulators.txt

rm top5 bottom5

##when logged in on the cluster, move to the shellScripts directory (if you are not already there) and modify the job script named corrFilter.job

cd /home/your.netID/shellScripts/

nano corrFilter.job

change the three paths:

export PATH=${PATH}:/home/jekoltes/shellScripts/ &

/home/jekoltes/shellScripts/wt.PCIT.out.txt.0.5.pc.filtered /home/jekoltes/shellScripts/null.PCIT.out.txt.0.5.pc.filtered

to reflect your netID where jekoltes is used in the current script

To save the file, you will type the following keys (simultaneously)

control + x

y

enter (return)

Then launch the job process.job

qsub corrFilter.job

##### IF you were to run these scripts from your own computer- NOT logged in on the cluster ###

To run the script, issue:

./correlateFetcher.sh wt.PCIT.out.txt.0.5.pc.filtered null.PCIT.out.txt.0.5.pc.filtered

where wt.PCIT.out.txt.0.5.pc.filtered and null.PCIT.out.txt.0.5.pc.filtered are filtered output files (created by PCIT.process.sh). The filtered files are used to search for targets of differentially hubbed regulators. There are a number of files that could be used depending on what your stringency of selecting correlates is.

Correlates can be tested for biological enrichment using a software such as DAVID. For more details, see: http://david.abcc.ncifcrf.gov/

##########################################################################

4)ADD ANNOTATIONS FOR DATA INTERPRETATION AND VIZUALIZATION (TIPS)

At this step there are several ways to annotate your results by converting Affy probe IDs to gene IDs. First, you could write a script to join this data together. Second, you could go to DAVID and enter your probe names to return the most up-to-date gene annotation information. For more details, see: http://david.abcc.ncifcrf.gov/

##########################################################################

5)VIEW NETWORKS WITH BIOLAYOUT (or other viewers)

Get your data from the cluster:

On a mac, I would recommend using fetch. You can download Fetch here:

https://fetchsoftworks.com/fetch/download/Fetch\_5.7.3.dmg

On a PC, I would recommend using filezilla. You can download Filezilla here:

https://filezilla-project.org/

How to input data into biolayout express

Once you have your data ready to visualize (a sub network or list of correlated genes), you could process them in this way to prepare them for Biolayout express.

These scripts will help to format the results for biolayout

awk '{print $1"\t"$2}' 1453063\_at.High.all.correlates.txt > geneNames.only.forBiolayout.txt

awk '{print $1"\t"$2"\t"$3}' 1453063\_at.LOW.all.correlates.txt > correlations.and.geneNames.forBiolayout.txt

TO RUN BIOLAYOUT:

1)load data: go to file in the menu, open

2)Select 3D view or 2D view (left side tool bar)

3)To look specific genes within subnetworks, select edit in the menu, selection, select all

Then select class viewer to see all entities and connections (left side tool bar)

Biolayout has clustering & classifier tools that are handy and many ways to view the data, depending on how you import it. Play around with these tools to create co-expression networks and sub-networks.

#############################################################################

######################## EXTRA TOPICS ########################

############ HOW TO EXTRACT YOUR RESULTS AFTER RUNNING PCIT if using pbzip2 ############

#To decompress PCIT results

bzip2 -d null.PCIT.bz2

bzip2 -d wt.PCIT.bz2

#GETTING SETUP

############ SETTING UP PCIT ############

PCIT Installation: This will depend on your computing system. The current versions of System requirements: PCIT are configured to run on either: A) linux cluster using PBS scheduler, B) A single CPU linux system, C) linux cluster with Xeon Phi technology (MIC), D) Macbook Pro using gfortran

There should be 2 makefiles in the software directory (Makefile.intel and Makefile.gfortran). To use, issue:

compile and run with:

make

make run

If your system does not have Fortran 2003 support, then you will need to compile the software by issuing:

make -f Makefile.intel

OR

ln -sf Makefile.intel Makefile

make

Software Requirements: Fortran and the appropriate compilers, A) openMP, A-C) Intel Fortran compiler, D) gfortran

Instructions on compiling PCIT are included in the read me file associated with the .tar.gz distribution of the code.

PCIT references:

1) Reverter, A. & Chan, E.K.F., 2008. Combining partial correlation and an information theory approach to the reversed engineering of gene co-expression networks. Bioinformatics, 24(21), 2491-2497.

2) Koesterke et al., 2013. Optimizing PCIT algorithm on Stampede's Xeon and Xeon Phi processors for faster discovery of biological netowrks. XSEDE '13, July 22-25 2013, San Diego, CA, USA. Copyright 2013 ACM 978-1-4503-2170-9/13/07.

3) Koesterke et al., 2014. Discovery of biological networks using an optimized PCIT algorithm on Stampede's Xeon and Xeon Phi processors. Concurrency and Computation: Practice and Experience. (Accepted)

############ SETTING UP BIOLAYOUT EXPRESS ############

To download, see: http://www.biolayout.org/

Software Requirements: Java

System Requirements: Biolayout express requires more RAM and additional CPUs as the data gets larger. For more on system requirements, see: http://www.biolayout.org/support/requirements/

Biolayout Reference:

http://www.biolayout.org/wp-content/uploads/2013/01/Theocharidis\_BioLayoutExpress3D\_Protocol\_Nature\_Protocolols\_2009.pdf

############ SETTING UP SHELL FUNCTIONS FOR DATA SUMMARIZATION ############

To compile shell scrips, issue from the command line the following code on the command line, where filename.sh is the name of the shell script you'd like to use.

chmod u+x filename.sh

To run a shell script, issue the following code on the command line:

./filename.sh (arguments if required)

\*\*To check for arguments, look at the file to see examples at the top of each .sh file, for example:

more filename.sh

############ CREATING A JOB SCRIPT FROM SCRATCH ############

#save this script from here down to the note "end of job script" as jobfilename.job

#!/bin/bash

# Instructions for new hpc-class users:

# To use this script:

# 1) Save this script as a file named myjob on hpc-class

# 2) On hpc-class, Issue

# qsub myjob to submit the job

# Use qstat -a to see job status,

# Use qdel jobname to delete one of your jobs

# jobnames are of the form 1234.hpc-class

# This script has cd as the first command.

# qsub command was executed. This is what most users want. Change

# that command if you want something else.

###########################################

# Output goes to file BATCH\_OUTPUT.

# Error output goes to file BATCH\_ERRORS.

# If you want the output to go to another file, change BATCH\_OUTPUT

# or BATCH\_ERRORS in the following lines to the full path of that file.

#PBS -o BATCH\_OUTPUT

#PBS -e BATCH\_ERRORS

#PBS -lnodes=1:ppn=16:compute,walltime=1:00:00

# Change to directory from which qsub was executed

cd $PBS\_O\_WORKDIR

pcit.eo -nconditions 5 -ngenes 1600 -method auto -threads 16 -gb 64.0 -i /ptmp/jekoltes/public/bin/PCIT/wt.txt > /home/jekoltes/wt.PCIT.out.txt

############ END JOB SCRIT ############

#Note: in this case, there are 1600 genes (rows) and N-1 columns = 5 (6 total columns with the first = gene IDs and the next 5 = normalized microarray gene expression data)

SETTING THE NROWS & NCONDITIONS ARGUMENTS IN PCIT

To get nconditions, you need to determine the number of rows and the number of columns - 1

To do so, issue:

awk 'END {print NR, NF-1}' fileName

NR is # rows (nrows)

NF-1 is number of columns - 1

#To execute the script send the job script to the queue by issuing:

qsub jobfilename.job

#MONITORING THE PCIT RUN STATUS

qstat -a

################# TROUBLESHOOTING PCIT #################

If your job fails check:

1) Is your data formatted as space or tab delimited and in unix format? This can be checked with text wrangler software, using vi, or by other means.

2) Does your file have a header? If it does, this will force PCIT to fail. Remove the header and try again. To quickly remove the header with shell, use:

sed 1d PCIT.inputFile.txt > revised.PCIT.inputFile.txt

3) Do your gene IDs or any columns in the data contain extra whitespace (spaces or tabs)? If yes, this will cause PCIT to fail. Remove these extra spaces and try again.

4) Do you have the correct number of rows and nconditions? Does nconditions = number of total columns? If you have the wrong number of rows, or nconditions, PCIT will be forced to fail. nconditions should be set to the number of columns - 1 (i.e. if 11 total columns in your PCIT input file, nconditions = 10).

5) Do you have "/" in any of your gene names? If so, PCIT may struggle with this. To overcome this problem, you will need to substitute the "/" for some other character (i.e. "-"). To do so, use a modification of the following command:

tr '/' '-' < 13.5null.txt | ./pcit.eo

6) Are all expression values for a gene 0 within a treatment? If so, this will cause the denominator of the correlation function to be set to 0 resulting in an undefined result, causing fortran to crash. A quick solution is to remove the 0's and pipe your results to pcit. First, you will need to count the number of rows when excluding 0's:

awk '$2 != 0 || $3 != 0 || $4 != 0 || $5 != 0' /data001/nextgen/jekoltes/McKay\_data/SI.lo.both.RPKM.txt\_final.txt | awk 'END {print NR, NF-1}'

Then you can submit the job to remove 0's and run PCIT:

awk '$2 != 0 || $3 != 0 || $4 != 0 || $5 != 0' /data001/nextgen/jekoltes/McKay\_data/LVR.lo.RNA.RPKM.txt\_final.txt | pcit.eo -nconditions 4 -ngenes 187121 -method v7fom -threads 32 -gb 256.0 > /data001/nextgen/jekoltes/McKay\_data/LVR.lo.RNA.RPKM.txt\_PCIT.out.txt

7) Are you running a big job? (i.e. >=100,000 rows?). If so, your cluster may not be able to move this large of data even with the memory free function in the software. You may need to increase the stacksize for the cluster and the node (required on ISU's lightning3). To do this, include the following code in the job script inbetween the PBS and PCIT code, i.e. :

ulimit -s unlimited

export OMP\_STACKSIZE=1G

#!/bin/sh

#PBS -lvmem=256Gb,pmem=8Gb,mem=256Gb,nodes=1:ppn=32:ib,walltime=48:00:00

ulimit -s unlimited

export OMP\_STACKSIZE=1G

export OMP\_NUM\_THREADS=32

awk '$2 != 0 || $3 != 0 || $4 != 0 || $5 != 0' /data001/nextgen/jekoltes/McKay\_data/SI.lo.both.RNA.RPKM.txt\_final.txt | pcit.eo -nconditions 4 -ngenes 255008 -method v7fom -threads 32 -gb 256.0 > /data001/nextgen/jekoltes/McKay\_data/SI.lo.both.RPKM.txt\_PCIT.out.txt

Other ideas for troubleshooting:

# Make sure that characters are included only in your first column for gene IDs. Missing data represented as N/A or . may cause PCIT to fail.

# Make sure that you have enough RAM and CPUs for your data. Running a small trial dataset first is recommended if RAM and CPUs are limited. Using the v7 version of PCIT will help with this problem.

#Make sure that you have put the appropriate CPU, RAM, run time information into the scheduler script if running a PBS or Slurm based scheduler script