PECA 2.0 manual

Zhana Duren
zduren@clemson.edu
Jan 21, 2022

Contents

1. Getting started.

1.1 About PECA

The rapid increase of genome-wide data sets on gene expression, chromatin states and transcription factor (TF) binding locations offers an exciting opportunity to interpret the information encoded in genomes and epigenomes. This task can be challenging as it requires joint modeling of context specific activation of cis-regulatory elements (RE) and the effects on transcription of associated regulatory factors. To meet this challenge, we propose a statistical approach based on paired expression and chromatin accessibility (PECA) data across diverse cellular contexts. In our approach, we model 1) the localization to REs of chromatin regulators (CR) based on their interaction with sequence-specific TF, 2) the activation of REs due to CRs that are localized to them, 3) the effect of TFs bound to activated REs on the transcription of target genes (TG). The transcriptional regulatory network inferred by PECA provides a detailed view of how trans- and cis-regulatory elements work together to affect gene expression in a context specific manner.

PECA is a statistical tool for gene regulatory network inference from paired gene expression and chromatin accessibility data. If you use PECA software, please cite PECA and PECA2 papers:

Duren, Zhana, et al. "Modeling gene regulation from paired expression and chromatin accessibility data." Proceedings of the National Academy of Sciences 114.25 (2017): E4914-E4923.

Duren, Zhana, et al. "Time course regulatory analysis based on paired expression and chromatin accessibility data." Genome research 30.4 (2020): 622-634.

1.2 Installation

PECA is a software for inferring context specific gene regulatory network from paired gene expression and chromatin accessibility data. PECA software source code can be downloaded from Github: https://github.com/SUwonglab/PECA.

To run PECA, you need to install following:

Matlab, Macs2, Homer, Samtools and *Bedtools.*

Download and install PECA on Linux:

wget https://github.com/SUwonglab/PECA/archive/master.zip

unzip master.zip

cd PECA-master/

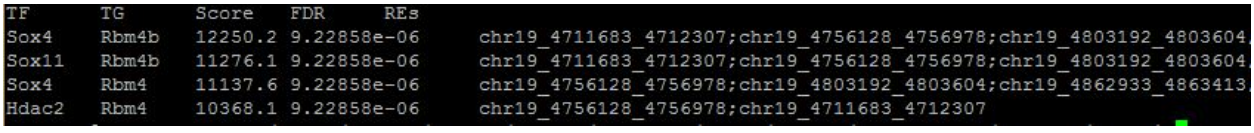
bash install.sh

1.3 Test PECA

To test successful installation of PECA, you can run PECA network inference on the example data:

bash PECA.sh RAd4 mm9

If the installed correctly, it will output a gene regulatory network in following path: *./Results/RAd4/RAd4_network.txt*



TF	TG	Score	FDR	REs
Sox4	Rbm4b	12250.2	9.22858e-06	chr19_4711683_4712307;chr19_4756128_4756978;chr19_4803192_4803604
Sox11	Rbm4b	11276.1	9.22858e-06	chr19_4711683_4712307;chr19_4756128_4756978;chr19_4803192_4803604
Sox4	Rbm4	11137.6	9.22858e-06	chr19_4756128_4756978;chr19_4803192_4803604;chr19_4862933_4863413
Hdac2	Rbm4	10368.1	9.22858e-06	chr19_4756128_4756978;chr19_4711683_4712307

If you didn't see this network file, it means there is something wrong with it. Please check the *PATH* of the dependence tools (*Matlab, Macs2, Homer, Samtools* and *Bedtools*) are added to the *.bashrc* or not. If not, please add those *PATH* to *.bashrc*.

1.4 PECA software

There are three tools included in PECA 2.0 software: network inference, comparison of two networks and comparison of two groups of networks. To run network comparison, you need to run network inference first.

2.Network inference

To run PECA network inference tool, you need to do following two steps: i) prepare input data and ii) run PECA network inference.

2.1 Prepare input data

Put the input files into folder named *./Input*. PECA network require following three input files: *${SampleName}.txt*, *${SampleName}.bam*, *${SampleName}.bam.bai*.

${SampleName}.txt* is gene expression file containing two columns (tab delimited), gene Symbol and FPKM (or TPM). *${SampleName}.bam is chromatin accessibility data, DNase-seq or ATAC-seq. *${SampleName}.bam.bai* is the index file of bam file.

Note that all the three files should have same before-dot-file-name *${SampleName},* only difference is after dot *".txt", ".bam"* or *".bam.bai".* Please see the example of RAd4 in the *./Input* directory (*RAd4.txt, RAd4.bam,* and *RAd4.bam.bai*).

2.2 Run PECA network inference

After the input data is prepared, please run the following script to do network inference.

sh PECA.sh $sampleName $genome

Example: *sh PECA.sh RAd4 mm9*

The results will *be ./Results/${SampleName}/* . Please see the description of the output files:

${SampleName}_network.txt is the tissue-specific network. Each row represents one regulation. The first two columns are TF and TG. The third column is the regulation score. A higher value represents a higher possibility of regulation. Rows are ranked by regulation score. The Forth column is FDR. The Fifth column is the list of regulatory elements (REs, including promoter, enhancers,…) which regulate TG and contain accessible motif binding sites of the TF.

TFTG_score.txt is regulation strength for all TF to TG. Each row represents one TF and each column represents one target gene. A higher value represents a higher possibility of regulation.

CRB_pval.txt is the Chromatin regulators' (CR) binding site matrix, each column represents one CR, each row represents one region, the values are p-values.

3. Comparison of two networks

If you have two samples and want to compare the two samples at the network level, please run the network comparison tool by following steps:

3.1 Prepare input data

Prepare two networks: Run PECA network inference tool on two samples one by one. (Script: *sh PECA.sh $sampleName $genome)*

3.2 Run PECA network comparison

2, Run: *sh PECA_compare_dif.sh $Sample1 $Sample2 $Organism*

Examples: *sh PECA_compare_dif.sh K562 GM12878 human*

sh PECA_compare_dif.sh mESC RAd4 mouse

Note that $Sample1 and $Sample2 **must be consistent with** the file names in the Input directory.

The results will be *./Results/Compare_${Sample1}_${Sample2}/*. Please see the description of the output files:

Specific network of two samples: *${Sample1}_specific_network.txt* and *${Sample2}_specific_network.txt*

Common network of two samples: *${Sample1}_${Sample2}_common_network.txt*

Specific modules (dense subnetwork) of two conditions: *${Sample1}_specific_module.txt* and *${Sample2}_specific_module.txt*

Common module of two samples: *${Sample1}_${Sample2}_common_module.txt*

Files *PooledNetwork.txt* or *PooledModuole.txt* can be used to visualize the networks by Cytoscape, and the node label is given in the file *Node_label.txt*. "1" and "-1" in *PooledNetwork.txt* or *PooledModuole.txt* represent "Activation" and "Repression" respectively. "1" and "2" in *Node_label.txt* represent the gene is Sample1 specific or Sample2 specific.

Corr: Pearson correlation of TF and TG expression across ENCODE data.
-log10P_TF: -log10(P_TF) P_TF is p-value of TF expression.
-log10P_TG: -log10(P_TG) P_TG is p-value of TG expression.
-log10P_Regulation: -log10(P_Regulation) P_TG is p-value of TF_TG regulation.
Fold: fold change of TF-TG regulation score in two conditions.
Activity: normalized TF-TG regulation score, ranging from 0 to 1.
Score: TF-TG regulation score.

4. Comparison of two groups of networks

If you have two conditions (multiple samples in each conditions) and want to compare the two conditions at network level, please do it by following steps:

4.1 Prepare input data

1, Run PECA network inference tool on all the samples from two conditions one by one. (Script: *sh PECA.sh $sampleName $genome)*

2, Construct labels: Write the sample names of Group1 and Group2 into text files named *$Group1* and *$Group2*, respectively. (eg. create one text file named "*Control*" and put the sample names of one condition to this file, create other text file named "*Case*" and put the names of the other condition to this file. Node that *sample names* **must be consistent with** the file names in Input directory. Note that the sample name files contain one sample name per line.)

4.2 Run PECA multiple network comparison

3, Run: *sh PECA_compare_dif_multiple.sh $Group1 $Group2 $Organism*

Example： *sh PECA_compare_dif_multiple.sh Control Case human*

The results will be *./Results/CompareGroup_${Group1}_${Group2}* . Please see the description of the output files:

Specific network of two conditions: *${Group1}_specific_network.txt* and *${Group2}_specific_network.txt*

The common network of two conditions: *${Group1}_${Group2}_common_network.txt*

The specific module of two conditions: *${Group1}_specific_module.txt* and *${Group2}_specific_module.txt*

The common module of two conditions: *${Group1}_${Group2}_common_module.txt*

Files *PooledNetwork.txt* or *PooledModuole.txt* can be used to visualize the network by Cytoscape, and the node label is given in the file *Node_label.txt*. "1" and "-1" in *PooledNetwork.txt* or *PooledModuole.txt* represent "Activation" and "Repression" respectively. "1" and "2" in *Node_ label.txt* represent the gene is Group1 specific or Group2 specific.

Corr: Pearson correlation of TF and TG expression across ENCODE data.
-log10P_TF: -log10(P_TF) P_TF is p-value of TF expression.
-log10P_TG: -log10(P_TG) P_TG is p-value of TG expression.
-log10P_Regulation: -log10(P_Regulation) P_TG is p-value of TF_TG regulation.
Fold: fold change of TF-TG regulation score in two conditions.
Activity: normalized TF-TG regulation score, ranging from 0 to 1.
Score: TF-TG regulation score.