Evolinc using Atmosphere

Author(s): Dr. Upendra Kumar Devisetty, CyVerse/University of Arizona and Dr. Andrew D. L. Nelson, School of Plant Sciences, University of Arizona

Introduction

Evolinc is a two-part pipeline to identify lincRNAs from an assembled transcriptome file (.gtf output from cufflinks) and then determine the extent to which those lincRNAs are conserved in the genome and transcriptome of other species.

The first part of the pipeline is the lincRNA identification. The second part is the comparative genomics and transcriptomics analysis. You feed the output from first part to second part. The pipelines were kept separate in case users did not want to perform an evolutionary analysis on the identified lincRNAs. The process is highly dependent on quality of the genomes, transcriptomes, and overall annotation datasets being used. Even a couple of SNPs could lead to a transcript being miss-identified as either a lincRNA or a protein-coding gene.

All of the necessary Python modules are already installed on this instance, so you can get started analyzing right away!

Note, currently Evolinc only identifies *intergenic* non-coding RNAs. We will incorporate identification of all lincRNAs (including natural antisense, overlapping, and those of *intra-genic*/*intronic* origins) in a later version. This is a tutorial for first part of the pipeline as a distinct Atmosphere image.

Accessing Evolinc

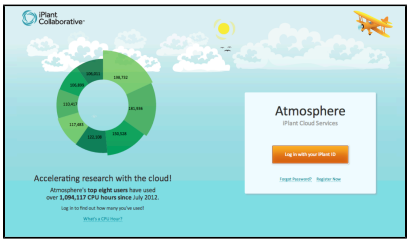
This tutorial will take users through steps of:

1. Launching the Evolinc Atmosphere image
2. Running Evolinc on an test data

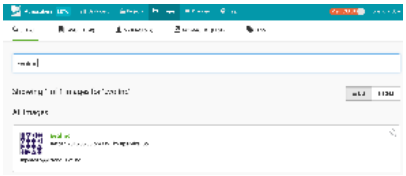
Learn about allocations
Learn about CyVerse's allocation policies here.

Part 1: Connect to an instance of an Atmosphere Image (Virtual Machine)

Step 1. Go to https://atmo.iplantcollaborative.org and log in with your cyverse credentials.



Step 2. Click on the **Launch New Instance** button and search for **Evolinc**.



Step 3. Select the image **Evolinc** and click **Launch Instance**. It will take 10-15 minutes for the cloud instance to be launched.

Note: *Instances can be configured for different amounts of CPU, memory, and storage depending on user needs. This tutorial can be*

accomplished with the small instance size, ***medium1 (4 CPUs, 8 GB memory, 80 GB root)***

Part 2: Set up a Evolinc run using the Terminal window

Step 1. Open the **Terminal**. Enter the ssh, username along with your IP address to connect the instance through the terminal

```
$ ssh <username>@Ipaddress
```

Step 2. All the dependencies and scripts for running Evolinc are located in "/opt/Evolinc" folder. You can run the command line options for Evolinc by executing

```
$ ./evolinc-part-I.sh -h

Usage : sh evolinc-part-I.sh -c cuffcompare -g genome -r CDS [-b TE_RNA] [-t CAGE_RNA]
[-x Known_lincRNA]
 -c </path/to/cuffcompare output file>
 -g </path/to/reference genome file>
 -r </path/to/cDNA reference file>
 -b </path/to/Transposable Elements file>
 -t </path/to/TSS file>
 -x </path/to/Known lincRNA file>
 -h Show this usage information
```

Explanation of the code line

1. -c: Cuffcompare output file in gtf format
2. -g: Reference genome file in fasta format
3. -r: Reference cDNA file in fasta format
4. -b: Transposable elements file in fasta format
5. -t: TSS site file in gff format
6. -x: Known Long non coding RNA in gff format

Part 3: Running sample data

The staged example data can be found in 2 folders - "Evolinc/sample.data.arabi" and "Evolinc/sample.data.brapa" within "Evolinc" folder. List its contents with the **ls** command:

```
$ cd /opt/Evolinc
$ ls sample.data.arabi/
AnnotatedPEATPeaks.gff      AthalianaslutteandluiN30merged.gtf
TAIR10_chr.fasta
Atha_known_lncRNAs.mod.gff  TAIR10_cdna_20110103_representative_gene_model_updated.fa
TE_RNA_transcripts.fa
$ ls sample.data.brapa/
Brassica_rapa_v1.2_cds.fa  Brassica_rapa_v1.2_genome.fa
cuffcompare_out_annot_no_annot.combined.gtf  TE_RNA_transcripts.fa
```

Executing the code with the provided test data

A) Arabidopsis test data

```
$ ./sh evolinc-part-I.sh -c sample.data.arabi/AthalianaslutteandluiN30merged.gtf -g
sample.data.arabi/TAIR10_chr.fasta -r
sample.data.arabi/TAIR10_cdna_20110103_representative_gene_model_updated.fa -b
sample.data.arabi/TE_RNA_transcripts.fa -t sample.data.arabi/AnnotatedPEATPeaks.gff -x
sample.data.arabi/Atha_known_lncRNAs.mod.gff
```

B) Brassica test data

```
$ ./sh evolinc-part-I.sh -c sample.data.brapa/AthalianaslutteandluiN30merged.gtf -g
sample.data.brapa/TAIR10_chr.fasta -r
sample.data.arabi/TAIR10_cdna_20110103_representative_gene_model_updated.fa -b
sample.data.brapa/TE_RNA_transcripts.fa
```

This will produce a folder named as "output". The Evolinc pipeline generates 7 different output files

1. lincRNA_final_transcripts.fa - Final Long intergenic ncRNA transcripts in fasta format
2. lincRNA_final_transcripts.bed - Final Long intergenic ncRNA transcripts in bed format
3. lincRNA_final_transcripts.promoters.fa - Promoter sequences of the final Long intergenic ncRNA transcripts in fasta format
4. lincRNA_final_transcripts_counts.txt - File showing the number of transcripts left at every step of the pipeline
5. lincRNA_final_transcripts_demographics.txt - Final Long intergenic ncRNA transcripts demographics
6. lincRNA_CAGE_final_transcripts.fa - Final Long intergenic ncRNA transcripts that have overlapping with the TSS transcripts (generated only when you have TSS file)
7. lincRNA_overlapping_known_final_transcripts.fa - Final Long intergenic ncRNA transcripts that have overlapping with the known lincRNA (generated only when you have known lincRNA file)
8. lincRNA_final_transcripts_updated.gtf - Final updated cuffcompare output with the final Long intergenic ncRNA transcripts

Part 4: Trying out your data

Make sure that you make a folder within the Evolinc folder and upload your files in to that folder and run the above script. Either Cuffcompare or Cuffmerge output files are acceptable. Genome.fasta file should be the same to which you are aligning your transcriptomic data. The transposable element data set can be either from your species of interest or from a family of closely related species. For example, there is a maintained data set of Brassicaceae transposable elements that can be used to compare *A. thaliana* lncRNAs against. If you have not generated TSS data yourself, there are publicly available data sets of transcription start sites that may be useful, but only for a limited number of species. If there are multiple public data sets of known lncRNAs for your species that you would like to compare your set against, merge them into one gff document.