Sara WERNIG-ZORC

LAB NOTEBOOK

2017.06.21 PROJECT PROPOSAL

Do transcription factors have preferential binding to promoters/exons/introns/3'UTR/5'UTR of long non-coding RNAs (lncRNAs)?

I plan to use known motifs of transcription factors and search for those motifs in sequences of lncRNAs, together with their promoters, to see if TFs have preferential binding to certain features. I will use available databases for TF motifs and known lncRNA sequences.

2017.06.21 PROJECT APPROVED

Lars Arvsted approved the project.

2017.07.25 PROJECT OUTLINE Part I

- Chose databases to work with.

For TF database, I will use JASPAR, which contains a curated, non-redundant set of profiles, derived from published collections of experimentally defined transcription factor binding sites for eukaryotes.

- Read: *Ghosh S., Sati S., Sengupta S. and Scaria V. 2015 Distinct patterns of epigenetic marks and transcription factor binding sites across promoters of sense-intronic long noncoding RNAs. J. Genet. 94, 17–25*

2017.07.26 PROJECT OUTLINE Part II

- Start with 1-3 TFs (for example; CTCF, p53) and manually assign motifs; so you can validate the targets (Validate with JASPAR). Than later on connect to the database and do a major search (JASPAR).
- construct the motif sequence/logo for known TFs; for example assign TF1 = ATCGGAGTTN
- LncRNA sequence (you can start with protein coding as well so you have to make it work)
- protein coding known targets
- check motif in lncrna sequence (seq. Similarity packages – check!), for each window 1-7,2-8,3-8 etc. and stop when
- for N you have to check for all combinations
- Scoring system – if motif slightly differs it will still bind, but less, thus you have to score somehow (CAN BE DONE LATER)
- JASPAR database ; use frequency matrix

PROJECT PLAN:
1. construct the consensus motif or extract from database
2. use JASPAR to get the frequency matrix
3. extract window from lncRNA; 1bp window sliding
- take X windows
4. For each window check the preference of the motif

- ATCG preference at bp resolution
5. Score single window according to the frequency matrix
- similarity do it for all the windows from the lncRNA sequence
6. Compare the scores between each window
- find the best match (based on the score)
7. compare the best matches from each motif or TFs and score for specificity
8. validate targets with JASPAR database
9. Later on when you do for all TFs; make a calculation if there is a preference for certain parts of lncRNAs to bind to.

EXTRACTING THE lncRNA sequence
- genome FASTA file of organism
- GTF file of that organism
- fastacmd command (available also in SAM tools; perhaps a python package as well?) and extract the sequences

FREQUENCY MATRIX
- use flat file from JASPAR, so it can be used with Python

CHALLENGES
- In order to confirm a certain preference of binding, I would need to also check for all the genes (cca. 60.000) and than see if this is the case only for lncRNAs or for protein coding genes as well.

- How to deal with different sizes of motifs for different TFs (some have 4bp motifs, other 12bp motifs)??

CTCF: CCGCGNGGNGGCAG

2017.07.27 PYTHON MODULES

- *numpy* for specificity calculations of the binding
- *re* for searching with regular expression. I will search the motif in the lncRNA sequence.
- *biopython* for importing FASTA files and the frequency matrix??

#Figure out how to properly import the frequency matrix

2017.07.28 CHALLENGES

- How to deal with motifs being of different length??
    > How many different lengths are there? CHECK! If it only between 10-15, than it’s not a big issue.

2017.07.31

- Import the frequency matrix as a numpy array or with biopython??
- At the end of the header of the fasta seq., add the position of the motif; example 1-10, 2-11 etc.

- There is a motif analysis module inside Biopython (Bio.motifs) and a module TAMO, specifically made for this purpose! --CHECK THE MANUALS!

> There is a way to connect directly with JASPAR database with Biopython module.

2017.08.03

I have taken a completely different approach. I've re-written the whole program and saved it as motif_preference_2.py

- I've used **nt_search** (inside of Biopython) function that finds motifs. I added a for loop that does the motif search in multiple FASTA input sequences and outputs the location of the motif for each sequence. This function saved a lot of time.

- To recap today's work: I managed to import one motif from .sites file and produce a degenerate consensus sequence from that. The degenerate consensus sequence was than used to do a motif search in all the lncRNA sequences from Gencode. Lastly, my script produced an output file with the name of the lncRNA sequence on one line and the motif + start of the motif in the sequence on the second line. Third line is the next sequence and so on throughout the file.

- What still remains to be done:

1. FOR loop (around the current script) that would go through all the motif files (import, make consensus sequence and save it in a variable) and perform the motif search with all of them.

2. Make a dictionary with all possible motif sequences per TF, with their corresponding SCORE as a key in the dictionary.

3. Than connect the dictionary with the motifs found with the current script and connect them with the score.

4. Filter motifs based on score

5. Use remaining motifs for preference calculations.

6. Connect the fasta file with sequences with the GTF file, to get locations.

7. Count number of occurrences per location and calculate if there is a preference for certain locations → calculate p-value

8. Possibly compare the p-value from all the genes vs. lncRNA genes, to se if actual preference for lncRNAs exists or it's a general preference for all the genes.

2017.08.07

- I made a separate script that generates a file with consensus sequences for all motifs, from a frequency matrix downloaded from JASPAR db.

- I used that file than in the main script to produce an output file with binding sites for all motifs on all the sequences. (2GB file) (Point 1 in to do list)

- Next I'll start with the scoring.

Sara WERNIG-ZORC

2017.10.05 Scoring motifs

- Biopython's motif module can score found motifs based on PSSM matrix (which you can download from JASPAR db).

2017.10.15

- I have reshaped the script so it includes introns as well, by using the whole genome FASTA file and GTF annotation file and extracting the lincRNA sequences from FASTA file by positions in the GTF file.
- I have also made individual modules so the script can be used with separate functions.
- I will include promoters of lincRNAs by -2500 position from lincRNA start position to the lincRNA start position

2017.10.16

- Finding all the possible TF motifs in all known lncRNAs was relatively easy to obtain. But than remapping those to the intron/exon/promoter regions for all of them would require at least one more month to do. It is possible to do but it requires a lot more programming. So I decided to do a proof of concept on a few of lncRNAs and in the future I hope to go for all of them and publish the study.

2017.10.17

- For loop over motifs and lincRNAs is only calculating score and position of motif for either one lincRNA and all TFs or one TF for all lincRNAs. → find help, why is this not working?

2017.10.18

- Two ways of finding out if there is a preference for a specific region of lincRNAs, that TFs bind to:
1. get preference towards 5' or 3' for a particular TF
- dataset: lincRNA **transcribed** transcript sequences (these do not include introns, nor 5' and 3' ends before start codon and after stop codon)
- calculate position of transcript divided by the transcript length
- calculate mode() of the above calculation for all binding sites
- define calculations separately!

*Example: A*RNT TF

mode=statistics.mode(all_positions) =	0.5
mean=statistics.mean(all_positions) =	0.4996809211528865
stdev=statistics.stdev(all_positions) =	0.2893302698763421
var=statistics.variance(all_positions)=	0.08371200506671694

2. get preference for specific exon for example exon 2
- than collect exon start/end "relative to the transcript OR change the position of the motif to a position relative to the whole genome (chr,start). In this case to get the end position you would have to extract motif length and than start of motif + motif length = stop of motif. Although it probably doesn't matter if the motif is not in between intron/exon boundery."
- is position of motif smaller than stop of exon AND bigger than start of exon, if so than write exon 1 to the dictionary?

Sara WERNIG-ZORC

2017.10.19

What is left to do for today:
- Connect the modules into one main module
- calculate number of occurrences of exon1/2/3 etc. ← just count occurrences of ‘words’ in the list
- solve the FOR loop problem ← if you cannot solve than run the script 5 times, each time for a different motif and calculate based on that.
- Change the script so it takes input filenames fro system arguments instead of filenames being hard coded to the script. But also leave the filenames in, if the user doesn’t provide any filenames (for me).