BBS741 Project #3

Deadline: December 21, 2018 @ 11:59 pm
Email to BBS741.2018@gmail.com
with LastName_FirstName_Project1 as subject line

Part 0 (0 points)

We will be using two existing packages for this project:

1. A package for training a random forest model for predicting the cell type specificity of transcription factor peaks as described in Wang et. al. (paper https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5780765/; code https://github.com/VCCRI/motif-discovery-pipeline)
2. ~~A package for training a deep learning model for identifying transcription factor motifs written by Greg Andrews (GitHub link will be provided before the end of the week)~~

All resources are available on Dropbox:
https://www.dropbox.com/sh/d87m24izmjc981m/AACAeDmA9P4k9zTHTioJNsNia?dl=0

Overview of data:

As described in the deep learning lecture, accurate prediction of transcription factor binding profiles is an open problem in bioinformatics. There is ongoing research seeking to apply non-linear learning techniques to predict the affinity of transcription factors for particular DNA sequences, and also to integrate known epigenetic features and sequence features to predict transcription factor binding profiles *in vivo* in individual cell types. Such predictions lend insight into the activity of regulatory elements and the effect of variants on gene regulation when experimental data are not available. Given the time and expense of assaying all possible combinations of cell types and transcription factors, accurate prediction methods are invaluable.

The ENCODE Project has lists of known transcription factor binding sites identified by more than 1,000 ChIP-seq experiments for more than 300 transcription factors in nearly 150 cell types. We will be using a subset of these lists for some of the most widely-profiled transcription factors to train the random forest and deep learning models mentioned above. We also have lists of candidate regulatory elements (ccREs) predicted in more than 800 cell types using chromatin accessibility (DNase-seq), histone modification (H3K4me3 and H3K27ac), and ChIP-seq (CTCF) data. We will use these prediction lists from select cell types as a way of assessing the cell type specificity of the trained models.

Part I - Pre-processing data for Random Forest classifier (15 points)

Our first goal is to train a Random Forest classifier to predict the cell type specificity of individual transcription factor binding sites. Cell type specific binding profiles of individual transcription factors is known to play a critical role in cell type differentiation, cell cycle control, and response to environmental factors, among other processes. A leading hypothesis is that binding of

individual cell type specific factors is driven by co-binding with different combinations of master regulator TFs, and thus that a cell type specific “motif grammar” or co-localized TF binding motifs should be detectable in individual cell types.

Wang et. al. published a random forest classifier earlier this year which counts the occurrences of known transcription factor motifs within genomic regions and predicts the cell type in which the region is most likely to be bound by a transcription factor. They trained their model using transcription factor ChIP-seq data from ENCODE for the factors TCF7L2 and MAX in six and five cell types, respectively. For each factor, the authors considered the top 500 unique transcription factor binding sites in each cell type, sorted by p-value.

ENCODE has several other transcription factors which have been assayed in at least 5 cell types. We will train the authors’ random forest classifier on ChIP-seq data for FOXA1 in four cell types: HEK293T, HepG2, MCF-7, and K562. The peak files are available in Dropbox (rawpeaks.tar.gz). In order to train the model, these peak lists need to be pre-processed to match the input format the authors use. Write a script to identify the TF peaks for each cell type which are *unique* (do not intersect a peak from any of the other four cell types), then sort the peaks for each cell type by signal value (column 7; some peaks do not have p-values given) and take the top 500. Finally, the peaks in each file must be resized to exactly 240 basepairs each, centered on the original peak summit. You may use any language(s) and tool(s) you need and feel comfortable with to complete this task.

Part II - Training a Random Forest classifier (35 points)

The authors next identify all occurrences of known motifs within these top 500 unique peaks. This step is compute intense and is best run on a cloud computing platform or a compute cluster. The results of this step, run against the correct lists of top 500 unique peaks, are available on Dropbox (FOXA1_motif_freq.tar.gz). This is the output for Step 3 as described in the README (see the instructions at the bottom of this page on the authors’ GitHub repository https://github.com/VCCRI/motif-discovery-pipeline/tree/master/RF_implementation).

You will need to make some minor adjustments to the given scripts, such as adjusting cell type labels and file paths (hint: look for places where they hardcoded arrays with six items - we only have four cell types. The first script requires minor changes to five lines). You should obtain figures matching some of the main and supplementary figures from the original paper, showing AUROC values for individual cell types, plots of the p-values for the unique peaks and the out-of-bag evaluation of the random forest model. Submit your modified R scripts along with the PDF figures you get.

If the shell script in Step 5 is not compatible with your laptop, try the Python script on Dropbox (https://www.dropbox.com/s/y44k7ddhf88z734/AutoMergeTable_auto_5-300.py?dl=0).

- For which of the four cell types is the model most and least accurate?

- What does the OOB curve tell you about the RF model?

Part III - Extracting important features from the RF (35 points)

An important aspect of the authors' work is the ability to translate the Random Forest, which in some cases can be somewhat of a "black box" classifier, into biological insights such as relative motif importance in driving the binding of particular transcription factors. Make necessary modifications to the script described in Part 3 of the authors' instructions, then run it against the output from **Part II**. You should obtain figures similar to Fig. 3a and Fig. 3c from the paper, showing the most important motifs for FOXA1 binding, along with a list of "grammar rules" influencing FOXA1 binding in each of the four cell types.

- What does the MDA curve tell you about how many motifs contribute significantly to the "motif grammar" driving FOXA1 binding?
- Which other transcription factors seem to be most important in driving FOXA1 binding in each of the four cell types?

Part IV - Testing the RF on mystery ccREs (15 points +5 bonus)

The random forest classifier trained on the original input should be applicable to regions the model hasn't seen before. Lists of enhancer-like ccREs active in three of the four cell types are available on Dropbox (mystery1.bed, mystery2.bed, and mystery3.bed), but without cell type labels. All of the ccRE sets have been filtered to contain ccREs unique to the three cell types, and to remove transcription factor binding sites in the top 500 peaks for each cell type. All of the ccREs are bound by FOXA1 in the cell type in which they are active, and have been filtered to have as little overlap with FOXA1 binding in the other cell types as possible.

Write a script which uses the model trained in **part II** to classify the ccREs from each list according to the cell type in which they are most likely to be bound by a transcription factor. As input, you may use the pre-generated motif occurrence matrices produced on the lists from Dropbox (https://www.dropbox.com/home/BBS741.2015/bbs741.2018/Homework/Project-3/mystery_motif_freq).

- Do you feel confident guessing which list is which based on your results? Why or why not? If not, what might this imply about the model/classifier?
- If you're confident (or not): which list is which?

~~Part V - Deep learning TF motifs (30 points +5 bonus)~~

~~DeepBind and FactorNet are very compute intense, and must be trained using advanced hardware such as graphical processing units (GPUs) or Google's ultra-specialized TensorFlow processing units (TPUs). Greg Andrews has developed a deep learning method which can be trained on standard CPUs within a reasonable timeframe on an entire set of transcription factor~~

~~peaks from a ChIP-seq experiment. This method uses a convolutional layer to learn binding motifs for both high- and low-affinity transcription factor binding sites.~~

~~Choose at least one set of transcription factor peaks and one set of ccREs, and, following the instructions in the provided GitHub repository, train Greg's deep learning model and visualize the results. You should obtain a PDF with a list of motif logos representing the kernels learned by Greg's convolutional layer. Compare the motifs learned by the Deep Learning model on the known transcription factor peaks and the unknown ccREs with the most important motifs identified by the Random Forest.~~

~~By default, Greg's script will automatically generate synthetic negative sequences based on the input. Each list of ccREs has an associated list of inactive ccREs in the same cell type. For bonus points, you may alter Greg's script to use this negative set of negative ccREs in place of synthetically-generated negative sequences and see how this alters the learned motifs.~~

- ~~Do any of the deep learned motifs from the known transcription factor peaks match the most important motifs learned by the Random Forest classifier?~~
- ~~Do any of the deep learned motifs from the unknown ccREs match the most important motifs learned by the Random Forest classifier? If so, do they seem to match the most important motifs for a particular cell type?~~
- ~~Bonus: what sort of impact does the use of true negatives in place of synthetic negatives have on the motif kernels learned by the model?~~

Grading Rubric

Submissions received after **December 21, 2018 @ 11:59 pm** will earn ½ credit

Part I (15 pts)	Script correctly identifies top 500 unique peaks for each cell type	8 pts
	Script properly resizes peaks to 240 basepairs centered on original peak center	7 pts
Part II (35 pts)	Script trains the model, with appropriate adjustments to the authors' scripts	15 pts
	Script outputs AUROC values for four cell types and OOB curves	10 pts
	Correct interpretation of results	10 pts
Part III (35 pts)	Script extracts features from the model and generates figures	15 pts
	Correct interpretation of MDA curve	10 pts

	Correct interpretation of cell type heatmap	10 pts
Part IV (15 pts)	Script correctly applies a Random Forest classifier trained in **Part II** to classify ccREs	15 pts
	All three ccRE lists are correctly matched to their cell types	5 bonus pts
~~**Part V (30 pts)**~~	~~Script trains model and outputs motif kernels for one set of ccREs and one set of TF peaks~~	~~20 pts~~
	~~Appropriate comparison of output kernels with motifs learned by Random Forest~~	~~10 pts~~
	~~Appropriate interpretation of effect of true negatives on motif learning~~	~~5 bonus pts~~
Total		**100 pts** **+5 possible bonus**