CARNEGIE MELLON UNIVERSITY - COMPUTATIONAL BIOLOGY DEPARTMENT

May 1, 2017

Predicting Pseudouridylation of Individual RNA Bases Using Machine Learning

Christine Baek
christib@andrew.cmu.edu

Kevin Chon
khchon@andrew.cmu.edu

Deepank Korandla
dkorandl@andrew.cmu.edu

Tianqi Tang
tianqit1@andrew.cmu.edu

TODO: CHRISTINE abstract here. CB will complete this section after all other sections are written

We have built a software that predicts the probability of each site of input RNA sequence being pseudoUracil.

1 INTRODUCTION

TODO: DEEPANK. 3 sections minimum covering 1) RNA modification - what is it, biological significance. 2) PseudoUracil - what is it, maybe include figure of pseudouracil 3) What has been done with PseudoUracil research. DO NOT FORGET TO CITE.

example for Deepank :

This is how you cite stuff [2].

2 METHODS

The goal of this project is that we have to build a per-base/position predictor of pseudouridylation probability. Main challenge were small amount of related previous work whose models which could be used as reference, and extremely limited labeled data to train with. Our data

represents both *breadth and depth* in that the data represents intra-species, intra-gene, and inter-species diversity of pseudouracil. We have built models using different combinations of data to build a final model that is generalizable to different species and genes. The final software takes in standard bioinformatics data format such as `.fasta` or `.mfa` and outputs the probability of each base being pseuouracil.

2.1 Relevant Domain Knowledge

Previous work on RNA modification suggest that such modifications are sequence-dependent. Also, the secondary RNA structure is known to be important for various biological function. We worked with all publicly available RNA pseudouridylation data and for these, both the raw sequence and secondary structure were used as features. We chose two popular machine learning approaches (random forest and SVM) for our learning model, given their popularity and known robustness when analyzing biological data.

2.2 Data

This section discusses the type of data and their source, how they were processed and the features chosen for our models.

2.2.1 Data Collection

Total of 3 data sets were used.

1. *homo sapiens* chromosome rRNA [2] - this dataset represents intra-genome, inter-gene data (breadth-focused - same species, varied genes)
2. *homo sapiens* rRNA [4] - this dataset represents intra-genome, intra-gene data (depth-focused)
3. Aligned rRNA from following organisms [3] - this dataset represents inter-genome, intra-gene data (breadth-focused - varied species, same gene)
    - *Escherichia coli*
    - *Bacillus subtilis*
    - *Clostridium acetobutylicum*
    - *Thermotoga maritima*
    - *Thermus thermophilus*
    - *Pyrococcus abyssi*
    - *Sulfolobus solfataricus*
    - *Arabidopsis thaliana*
    - *Homo sapiens*
    - *Saccharomyces cerevisiae*

Dataset 2 functions as the common ground, or the connection between dataset 1 and 3.

2.2.2 Data Processing

TODO: CHRISTINE : obtaining the actual data, modifying it

2.3 Feature Selection & Engineering

TODO: CHRISTINE : GC content, sliding window

While it is possible that both direct neighboring sequences and potentially far sequences (in the form of secondary structure/hairpin) may have an impact on pseudouridylation, that would also indicate that the sequence Why delta G was rejected.

TODO: TIANQI : discuss stuff like vienna/RNA fold

2.4 Machine Learning Algorithm

The most popular strategy in applying machine learning currently is to attempt multiple strategies, and select for the best performing models. We chose SVM and Random Forest for their established performance and popularity for medical and biological data.

2.4.1 SVM

TODO: TIANQI - talk about why you picked SVM, what SVM is, and some settings you may be using and why.

There are a numbers of different classifiers that can help solving the problem, e.g. logistic regression, Naive Bayes classifier, neural network, random forest and support vector machine(SVM). Among these classifiers SVM has a great balance between performance and computational complexity, so we are using support vector machine with radial basis kernel in this project. Support vector machine is a supervised learning algorithm whose goal is to find the hyperplane that has the largest separation between two classes.

When the training samples are not linearly separable, we can use two techniques, soft margin and kernel. Soft margin allows misclassification in training step. Kernel projects the samples into a different space, making the samples to be linear separable. For example, in a two dimensional space two sets of data points locate on two different circles, both have the origin as center, and have radius r_1 and r_2. These data points are not separable by a straight line, but after projection $\phi(\boldsymbol{x}) = x_1^2 + x_2^2$, the data points can be separated by a line. In practice people usually use the product of the projection of two points, denoted as $\boldsymbol{K}(\boldsymbol{x}, \boldsymbol{x}') = \phi(\boldsymbol{x})\phi(\boldsymbol{x}')$.

In this project we used radial basis function(RBF) kernel. The kernel function of RBF is $\boldsymbol{K}(\boldsymbol{x}, \boldsymbol{x}') = \exp(-\frac{\|\boldsymbol{x}-\boldsymbol{x}'\|_2^2}{2\sigma^2})$, σ^2 is the parameter given by user. Under Taylor series expansion it is a polynomial function with infinite order, so it can project the samples into a infinity dimension space and can fits any shape of sample distribution.

2.4.2 Random Forest

TODO: KEVIN - talk about why you picked SVM, what SVM is, and some settings you may be using and why.

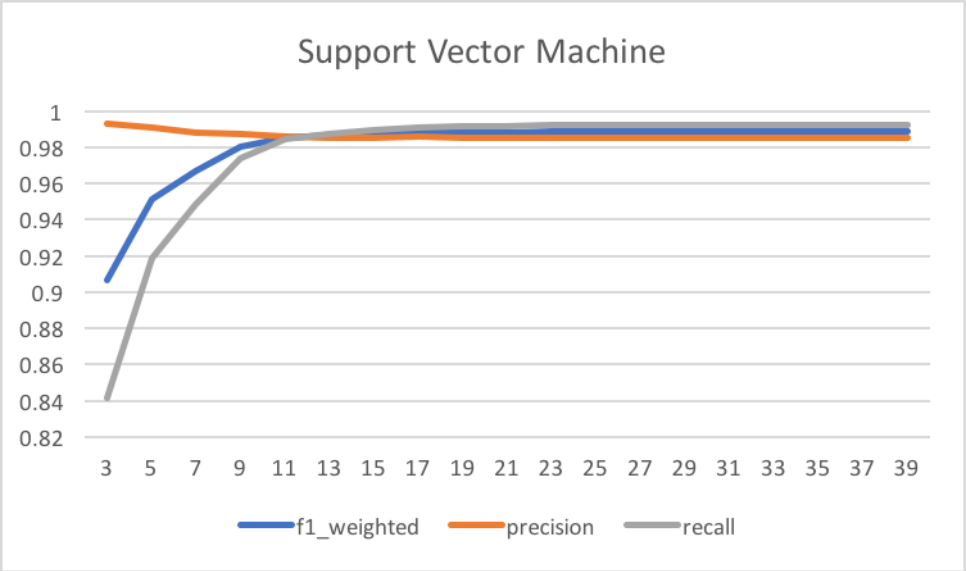
3 Results

3.1 SVM Results

TODO: TIANQI - discuss the results, and provide the table of performances where columns are input data type, and rows are various parameters used. Please use some form of heat map for the resulting table. I recommend matlab. Kevin & Tianqi - talk to each other so that the two tables look simliar (colorscheme, format, etc.) for uniformity

Table 3.1: Add caption

Window size	f1_weighted	precision	recall
3	90.64%	99.29%	84.12%
5	95.08%	99.09%	91.84%
7	96.70%	98.83%	94.84%
9	97.99%	98.75%	97.35%
11	98.51%	98.58%	98.46%
13	98.62%	98.50%	98.75%
15	98.73%	98.50%	98.96%
17	98.82%	98.59%	99.09%
19	98.82%	98.51%	99.14%
21	98.83%	98.51%	99.15%
23	98.86%	98.51%	99.22%
25	98.86%	98.51%	99.21%
27	98.87%	98.51%	99.24%
29	98.87%	98.51%	99.24%
31	98.88%	98.51%	99.25%
33	98.88%	98.51%	99.25%
35	98.88%	98.51%	99.25%
37	98.88%	98.51%	99.25%
39	98.88%	98.51%	99.25%



3.2 Random Forest Results

TODO: KEVIN - discuss the results, and provide the table of performances where columns are input data type, and rows are various parameters used. Please use some form of heat map for the resulting table. I recommend matlab. Kevin & Tianqi - talk to each other so that the two tables look simliar (colorscheme, format, etc.) for uniformity

4 Evaluation of Other Teams' Work

4.1 Team B

4.1.1 Issues during Installation

- how easy was it to install?

Would be better if support both Python 2 and 3 support, or indicate which Python 2 only packages are used. Not easy to install and use because of tons of prerequisite softwares.

4.1.2 Model Design

- does design make sense? do you understand what the design is?

The design seems to be clear in terms of how a machine learning algorithm is being trained and tested, even though there are references to several other libraries and how to use them.

4.1.3 Model Performance

- does it seem to perform well ?

Seems good

4.1.4 Potential Improvements

- clear interface/definitions ? what could be done better?

Just keep the least necessary files when distributing your software. A pipeline means user can do the whole things with one command, not download a bunch of prerequisite softwares, generate the data you need, to serve your software. People will also not going to train the model themselves, what they need is a pre-trained model. Just keep users away from those verbose information if they do not need it.

4.2 Team C

4.2.1 Issues during Installation

- how easy was it to install?

- If asking people to download stuff, either include the commands OR direct link at the minimum.

There is no context for what the input files should contain (at least in the description section). There is no warning that pysam is not supported by Windows. No sample files input provided. This means I cannot skip the alignment step and move on to the detection step. For the alignment step, I am not sure where I am supposed to get a fastq file with sequence reads, and the group expects me to download the entire human genome (no reference site is provided either). The usage section could have better step-by-step directions. It is not immediately evident that the "Alignment of Reads" section can be skipped.

4.2.2 Model Design

- does design make sense? do you understand what the design is?

Seems to align the input sequence to the model genome using BowTie2 and SamTools, which then can be put through the pre-trained classifier, outputting windows of m1a modifications.

4.2.3 Model Performance

- does it seem to perform well ?

It seems that the model is trained on only human... ? or that's my impression based on the readme.

4.2.4 Potential Improvements

- clear interface/definitions ? what could be done better?

- Include the location of m1a on the reference genome instead of just the window
- only humans?
- Seriously? You want ME to download the reference genome? How about at least including a direct link

5 Future Directions

TODO: Christine, but get input from others - more data - try more models -

6 Conclusion

TODO: CHRISTINE - done after all other sections are filled out - our model works reasonably well - RNA hairpin didn't work

References

[1] AUTHOR1, AUTHOR2, et al. *TITLE* http://LINK, DATE OF PUBLICATION

[2] Schraga Schwartz and Douglas A. Bernstein and Maxwell R. Mumbach and Marko Jovanovic and Rebecca H. Herbst and Brian X. Leon-Ricardo and Jesse M. Engreitz and Mitchell Guttman and Rahul Satija and Eric S. Lander and Gerald Fink and Aviv Regev *Transcriptome-wide Mapping Reveals Widespread Dynamic-Regulated Pseudouridylation of ncRNA and mRNA* http://doi.org/10.1016/j.cell.2014.08.028, 2014

[3] International Institute of Molecular and Cell Biology in Warsaw *RNA sequences with modifications* http://modomics.genesilico.pl/sequences/list/ accessed April 1, 2017

[4] Thomas M. Carlile, Maria F. Rojas-Duran, Boris Zinshteyn, Hakyung Shin, Kristen M. Bartoli, and Wendy V. Gilbert *Pseudouridine profiling reveals regulated mRNA pseudouridylation in yeast and human cells* https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4224642/ DATE OF PUBLICATION