

Species Selection for Phylogeny-Based Motif Detection

Computational Genomics Project Report

Narges Razavian, Selen Uguroglu, Andreas Zollmann

INTRODUCTION

Detecting conserved regions in multiple species alignment is crucial when modeling orthologous entities. However, in phylogenetic analysis of entities other than genes, for instance transcription factor binding sites (TFBS), this proves to be non-trivial due to the high functional turnover and incomplete orthology even within close species, such as *Drosophila* clade. Having more species does not necessarily contribute to the alignment, especially when the noise that it brings to the alignment is more than the information that it adds. Picking the correct set of species might considerably improve the accurateness of the analysis; however, in the current literature there has not been much work in identifying the most informative set of species for multiple sites.

In this paper, we tackle this problem with two approaches: The greedy search, and Weighted Voting. In greedy search selection method, we'll start with the smallest subset (with two species), and iteratively add more species to the highest scoring subset. We score the results based on its overlap to a known motif instance location.

The weighted voting algorithm is based on the widely used Boosting approach. Our method tries to take advantage of the fact that each subset of species leads to a different conserved region, and thus has a better result on a subset of predicted motifs. The idea of weighted voting is thus to combine the results of different species subsets, and build a more robust whole-sequence motif predictor.

METHODS

We performed our experiments on 12 species belonging to *Drosophila* clade, namely *D. melanogaster*, *D. simulans*, *D. sechellia*, *D. yakuba*, *D. erecta*, *D. persimilis*, *D.*

pseudoobscura, *D. ananasae*, *D. willistoni*, *D. grimshawi*, *D. mojavensis*, and *D. virilis*. For each of the species, sequences of upstream regions belonging to 21 developmental genes are obtained from UCSC Genome Browser [9]. These sequences make up the dataset on which we ran our experiments.

Sequence alignment

Sequences are re-aligned using Multi LAGAN [6] before running experiments. Multi LAGAN (MLAGAN) is a multiple alignment tool based on progressive alignment [6]. Given N sequences, alignment is constructed in $N-1$ steps, using LAGAN as the pair wise alignment subroutine. LAGAN aligns two sequences by first generating local alignments between two sequences, and constructing global map by connecting ordered subset of local alignments and then finding the best alignment within a limited area around the global map. Details can be found in the LAGAN and Multi-LAGAN paper [6].

Phylogenetic Analysis

For each of the subset of species we recalculated branch lengths using PAML [7]. PAML is a package for phylogenetic analysis using maximum likelihood. One of its uses is estimating branch lengths in a phylogenetic tree. In our experiments we used Felsenstein 84 (F84) model assuming no molecular clock [8].

To guide the process of estimating branch lengths, we used the accepted *Drosophila* phylogeny in Figure [1]. For each of the subsets, we adjust the tree by removing the unused species and then feed it into PAML, which then outputs a re-estimated tree specific to the species subset.

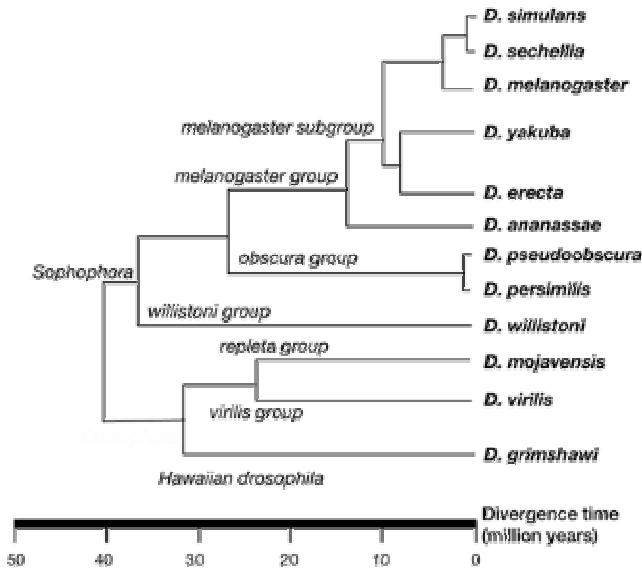


Figure 1: *Drosophila* phylogeny¹

Motif detection

EMnEM [2] is a probabilistic model for evolutionary extension of the MEME motif detection. In this model, observed sequences are assumed to have been generated from ancestral sequences that are two component mixtures of motif and background, each with their own evolutionary model. Given a set of sequences of length N , and a motif length w , and an ancestral tree topology, the likelihood of the data is defined as

$$L = \prod_{i=0}^{N-w} \sum_{m_i} p(m_i) \prod_{k=i}^{i+w-1} \sum_{t=0}^3 p(tree|A_{kb}) p(A_{kb}|m_i)$$

where m_i is an indicator function specifying the mixture component is either motif or background, and $P(m_i)$ is the prior probability of each component. If m is a motif, A_{kb} is the unobserved ancestral residue and $P(A_{kb}|m_i)$ is the relative frequency of that particular residue (A,T,C,G) in position k for that motif. For a background component, this probability is defined as the average base frequencies for the alignment, independent of position.

Given this model, with sequences and alignment as observed variables, and component indicator functions m_i s, motif priors $P(m_i)$ s, and probabilities $P(A_{kb}|m_i)$, and the evolutionary substitution matrices (which is based on Jukes-Cantor model) as hidden variables, the EM algorithm iteratively estimates the expectation for m_i s and A_{kb} s, and from that, the new model parameters $P(m_i)$, $P(A_{kb}|m_i)$, and substitution rates α_{mk} .

We used this tool for the motif detection, and implemented a Hadoop MapReduce wrapper allowing parallelized execution of EMnEM runs on a cluster, to decrease the runtime.

CONTRIBUTION

For the task of optimal subset selection for the 12 drosophila species, 2^{11} possible subsets would have to be considered in principle (one of the species is *D. melanogaster*, which should always be part of the set because evaluation takes place on the labeled TFBSs of this species). Detecting motifs on all of these 2^{11} subsets for each of the datasets (for regulatory regions of different genes) would be intractable.

Greedy Search

We propose the following solution: We start with pairwise alignments for species making up a total of 11 two-element sets (each including *D. melanogaster*). Starting with these sets, we iteratively add the rest of the species that are not in the current set one by one to the highest scoring subset, re-align, obtain a subset-specific phylogenetic tree from PAML, and re-run EMnEM to obtain motif candidates. The algorithm terminates when all species have been added to the set. The highest-scoring subset amongst all iterations is considered the most informative set. The score of a given subset of species is defined as the maximum F1 score (as a function of posterior motif probability) for the motifs returned by EMnEM on a reserved tuning set of genes. A pseudo-code specification of the Greedy Selection algorithm is given in Figure 2.

¹ <http://rana.lbl.gov/drosophila/wiki/index.php/Phylogeny>

FindMostInformativeSpecies()

```
begin
  S = {melanogaster, simulans, sechellia,
        yakuba, erecta, ananasae, pseudooscura, persimilis,
        willistoni, grimshawi, virilis, mojavensis}
  InitialSet ← {melanogaster}
  Result_array ← []
  while S is not ∅
    fmeasures ← []
    for all species in S
      NewSet ← InitialSet ∪ species
      alignment ← MLAGAN(NewSet)
      tree ← PAML(alignment)
      motifs ← EMnEM (alignment, tree)
      fl ← compare(motifs, baseline)
      flss ← [fl, species]
      NewSet ← ∅
    end
    {maxfl, bestspec} ← pickHighestF1Tuple(flss)
    InitialSet ← InitialSet ∪ bestspec
    Result_array add <InitialSet, maxfl>
    S ← Remove bestspec from S
  end
  {subset, fl} ← pickHighestF1Tuple(resultset)
  return [subset, fl]
end
```

Figure 2: Pseudo code of greedy algorithm

Weighted Voting

Our Weighted Voting method is inspired by the boosting algorithm, in which a number of “weak” classifiers are combined to create a stronger classifier. AdaBoost[10] is a popular boosting algorithm in which the final prediction of the classes is an interpolation of a classifier trained on different (weighted) subsets of the training data. This particular algorithm has been previously applied to different classification problems in the computational genomics area. Lausser et. al. [11], used AdaBoost and its high dimensional extensions, MultiBoost, MadaBoost, and AdaBoost-VC to combine single threshold classifiers for binary classification of disease based on DNA microarray gene expression data.

In another recent study, Vu and Braga-Neto [12] performed a detailed empirical analysis to see whether the ensemble scheme will improve performance of those classifiers sufficiently to

beat the performance of single stable, non overfitting classifiers, in the case of small-sample genomic and proteomic data sets. Based on their result, under t-test and RELIEF filter-based feature selection, bagging generally does a good job of improving the performance of unstable, overfitting classifiers, such as CART decision trees and neural networks.

In a different application, Niu et. al. [13] showed that AdaBoost algorithm outperforms Discriminant Function, neural networks, and SVM classifiers for predicting sub-cellular location of prokaryotic and eukaryotic proteins.

Based on these promising results, we decided to use boosting for combining the motif prediction results on multiple subsets. Our method differs from AdaBoost in that we do *not* retrain our classifier on the weighted input in iterations. This decision was made primarily because we would like to decouple the *selection procedure* from the *motif prediction algorithm*. If we had wanted to rerun the EMnEM prediction model on the weighted input, we would have needed to modify EMnEM directly, because currently it does not accept input sequences with different weight. To be able to make our algorithm useful for different motif detection models instead, we proposed a similar variation of boosting which we call weighted voting which runs in a single iteration.

The weighted voting method is as follows. Let C_1, C_2, \dots, C_n each be a subset of the species. For each C_i , the motif detection engine will predict motif instances m with associated posterior probability $p_i(m)$. As in the Greedy Selection, we use a tuning set of genes to compute the resulting F1 measure $F1_i(p)$ over all motifs m suggested by the detection engine whose posterior $p_i(m)$ exceeds a threshold p as a function of p . We define $F1_i^*$ as

$$F1_i^* = \max_p F1_i(p),$$

i.e., as the maximum F1 score obtained for subset C_i when tuning the posterior threshold.

During prediction, we combine our n candidate classifiers as follows: For each motif candidate

m proposed by any predictor, compute its weighted posterior

$$p_m = 1/Z * \sum_i [p_i(m) F_{1_i}^*]$$

where $Z = \sum_i F_{1_i}^*$ is a normalizing constant.

Each motif candidate m is thus assigned a voting-based posterior probability p_i , and can therefore be predicted in a threshold-based way.

RESULTS

We evaluate our models on the task of locating TFBSs for *D. melanogaster*. For the evaluation we used the F1 measure, and counted any match on the motif window with the gold standard as a valid match. EMnEM threshold was set to near zero, so that we could obtain all the predictions,

and finding the best threshold was done during the evaluation. We moved the threshold from 0 to 1 with 0.01 steps and at each threshold calculated the F1 measure on the predictions above that threshold. As a single-point measure, we also used the peak of this curve.

We performed initial experiments for both a completely unsupervised scenario, in which we run EMnEM with randomly initialized position weight matrices 10 times, with differing motif widths from 6 to 15 as input parameters, as well as a semi-supervised scenario, in which EMnEM was fed initial position weight matrices obtained from SAILING lab, one for each TFBS known to exist for *D. melanogaster*. Results of comparing unsupervised and semi-supervised experiments are shown in Figure 3.

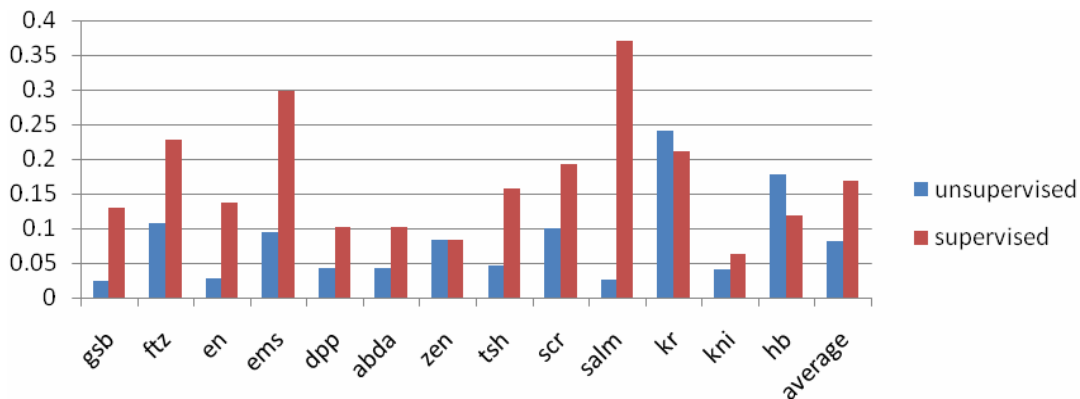


Figure 3: Initial results, showing the f measure for different datasets in unsupervised and supervised scenario

Greedy Selection Algorithm Results

We performed our experiments on 16 genes: dpp, kr, rho, scr, sna, tsh, zen, abda, ems, en, eve, ftz, gsb, h, hb and kni, under the semi-supervised scenario.

Starting with 11 pair subsets, (having melanogaster in each pair), we iteratively added the maximum-scoring species to our base set.

The best pair is found to be melanogaster-virilis, later on yakuba, sechellia, mojavensis, simulans, grimshawi, persimilis, pseudoobscura, erecta and willistoni added to base set in this order.

Relative F1 measures for each iteration and for each species are shown in Figure 3, where red bars show species that are added to the base set at that iteration. F1 measures are calculated by taking the average for each dataset.

In the figure, it shows that starting from the third iteration adding intermediate related species to the base set performs worse than adding close or distant species.

F1 scores for each iteration is shown in Figure 4. You can see that after adding virilis and yakuba, F1 scores start to decline as more species added.

In other words, the Greedy algorithm finds the (local) optimum to be the melanogaster, virilis and yakuba subset.

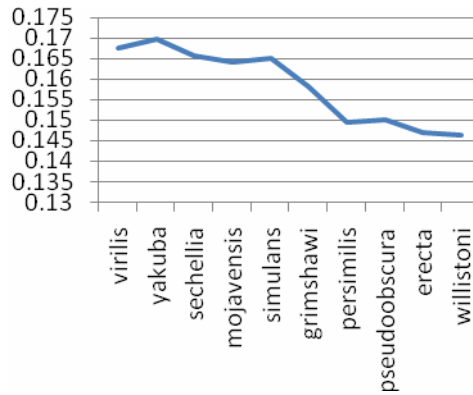


Figure 4: Maximum F1 scores at each iteration

Instead of taking the average over each dataset, when we only consider the F1 score on the eve gene, we still see the similar trend, as shown in Figure 5.

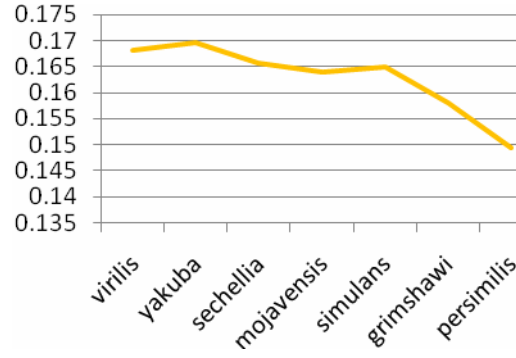


Figure 5: Maximum F1 scores on Eve

Figure 6 shows the result of the greedy selection at each iteration. Red columns indicate the species that was selected at that iteration. As can be seen from the figure, the maximum F1 decreases as the subset gets larger than a certain point.

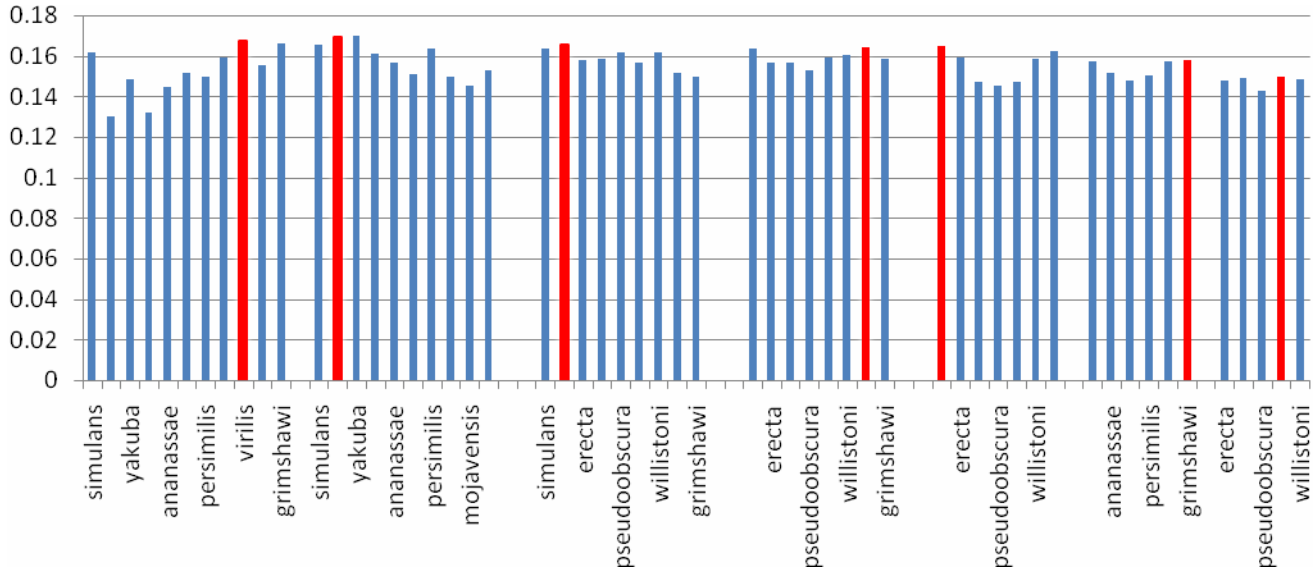


Figure 6: F1 measures for each species at each iteration

Weighted Voting Results

For this project, due to time limitations, we ran our weighted voting combination algorithm on

11 subsets of size two, pairing *D. melanogaster* with the remaining 11 species.

For each subset, we ran the MLagan, PALM, and EMnEM pipeline to get TFBS motif locations, and used the EVE gene as the tuning set. Thus each subset's max-F1 score on this dataset was used as the weight for that subset in the weighted voting process.

Figure 6 below shows the resulting F1 measure over all other datasets, as well as the average of F1 over all datasets. In the figure we compare the result of individual subsets, {melanogaster, simulans}; {melanogaster, sechellia}; {melanogaster, yakuba }; {melanogaster, erecta}; {melanogaster, ananassae}; {melanogaster, pseudoobscura}; {melanogaster, persimilis }; {melanogaster, willistoni}; {melanogaster, mojavensis }, {melanogaster, virilis }; {melanogaster, grimshawi}; all 12 species together; and finally, the weighted voting result combining all paired subsets.

The graph shows strong variance in the results, over different datasets, and it suggests that cross

validation should be used for future experiments. In our experiment we only used the EVE dataset for tuning, due to time constraints.

Additionally, Figure 7 shows the F1 averaged over all datasets for each 11 pair subsets as well as their weighted combination on Semi-supervised versus un-supervised model. As it can be seen, adding very close or very far species in the phylogenetic tree (comparatively) to melanogaster does not perform as well as adding species in the middle.

This is particularly evident in our unsupervised experiment, where the middle distance species indeed outperform the other ones. Please note that this data is tuned on EVE and in the unsupervised experiment is not directly comparable to the result of our Greedy selection.

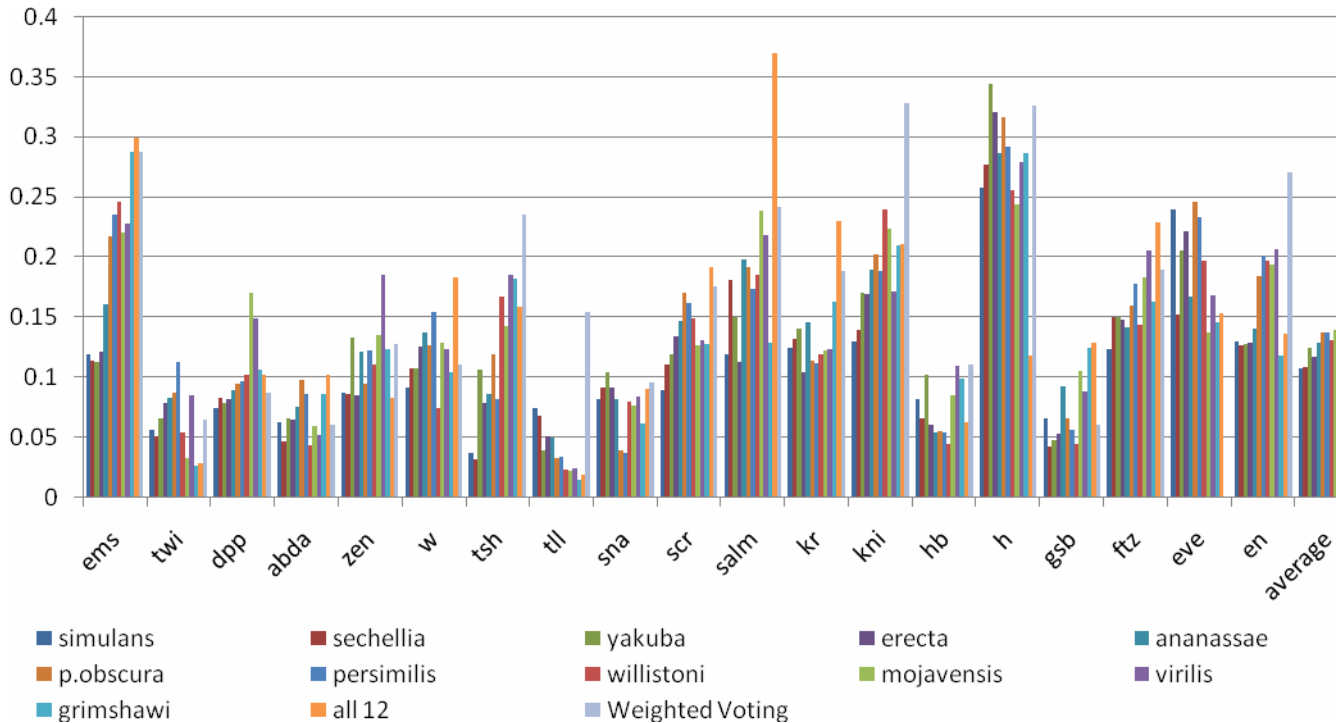


Figure 6: F1 scores of all pairs, 12 species, Weighted voting for individual datasets

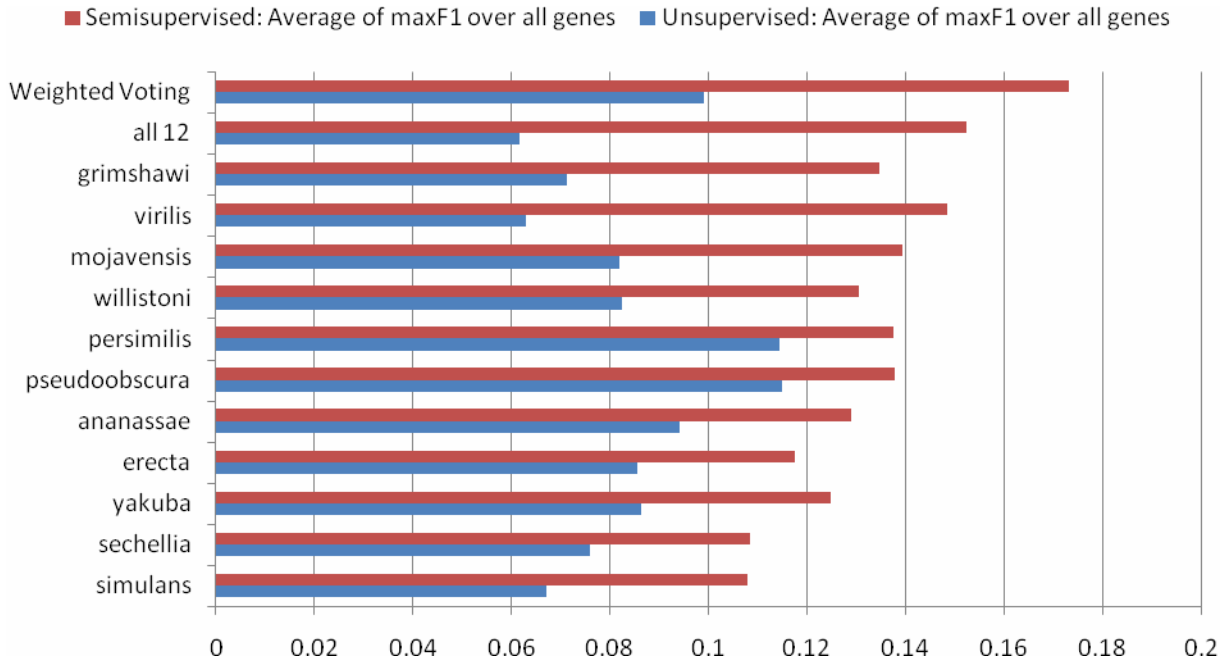


Figure 7: F1 scores of all pairs, 12 species, Weighted voting averaged over all datasets

From the averaged results, we can observe that the weighted voting combination method results in a better F1 measure overall, both in the unsupervised and the semi-supervised scenarios.

CONCLUSIONS AND FUTURE WORK

In this project we tackled the problem of species selection with two solutions. A greedy search algorithm was proposed to select the highest scoring species in iterations, thus giving us a subset with local maximum based on the F1 measure.

In another approach we used weighted voting method, to combine the results of different subsets. Given that different subsets have different conserved regions, and each different conserved region will result in better motif detection in a different part of the sequence, we decided to combine these results to have a more robust prediction for the whole sequence. Our combination method was inspired by the boosting approach, and we showed that

combining paired subsets indeed results in a better F1 score.

Since there is high variance between datasets, in the future, we will apply cross validation to achieve better results. Another direction for improvement is for us to change our evaluation scheme, and instead of taking average over all datasets, combine the datasets into a one big dataset, to obtain F1 score over that.

To overcome the shortcomings of our greedy selection method, one interesting next step is to use weighted voting combination over the results of the greedy algorithm in different iterations, thereby mitigating the local maximum problem of the greedy method.

So far we have only been experimenting with the Drosophila dataset. Extending our empirical experiments to other datasets, such as the Yeast dataset, will also be insightful and can lead to stronger claims.

REFERENCES

1. A Siepel, D Haussler, Combining Phylogenetic and Hidden Markov Models in Biosequence Analysis.
2. Am Moses, Dy Chiang, Mb Eisen. Phylogenetic motif detection by Expectation-Maximization on Evolutionary Mixtures.
3. JD McAuliffe, MI Jordan, L Pachter. Power Analysis and Species Selection for Comparative Genomics.
4. J Hu, YD Yang, D Kihara. EMD: an ensemble algorithm for discovering regulatory motifs in DNA sequences.
5. P Ray, M Kolar, EP Xing, S Shringarpure. CSMET: Comparative Genomic Motif Detection via Multi-Resolution Phylogenetic Shadowing.
6. LAGAN and Multi-LAGAN: efficient tools for large-scale multiple alignment of genomic DNA, *Genome Research* 2003 Apr;13(4):721-31.
7. Ziheng Yang. PAML 4: Phylogenetic Analysis by Maximum Likelihood. *Molecular Biology and Evolution* 2007 24(8):1586-1591.
8. Joseph Felsenstein, G A Churchill 1996 "A Hidden Markov Model approach to variation among sites in rate of evolution" *Mol Biol Evol*: 13(1): 93-104
9. Kent WJ, Sugnet CW, Furey TS, Roskin KM, Pringle TH, Zahler AM, Haussler D. The human genome browser at UCSC. *Genome Res.* 2002 Jun;12(6):996-1006.
10. Yoav Freund, "An adaptive version of the boost by majority algorithm", *Machine Learning*, 43(3):293--318, June 2001.
11. Ludwig Lausser, Malte Buchholz, Hans A. Kestler, "Boosting Threshold Classifiers for High-Dimensional Data in Functional Genomics", *Proceedings of the 3rd IAPR workshop on Artificial Neural Networks in Pattern Recognition*, 2008.
12. T. T. Vu, U. M. Braga-Neto, Is Bagging Effective in the Classification of Small-Sample Genomic and Proteomic Data?", *EURASIP J Bioinform Syst Biol*. April 2009.
13. Bing Niu · Yu-Huan Jin · Kai-Yan Feng, Wen-Cong Lu · Yu-Dong Cai · Guo-Zheng Li, "Using AdaBoost for the prediction of subcellular location of prokaryotic and eukaryotic proteins", *International Journal of Molecular Diversity*, Volume 12, Number 1, February, 2008.