10-810, Computational Molecular Biology: a machine learning approach: Problem Set 2

This problem set is due on Monday, 03/06 in class.

Bipartite graphs and whole genome alignment
Given genomes from two species, A and B we discussed in class a bipartite graph based approach to whole genome alignment. In this problem we will explore some of the computational issues involved in this algorithm.

1. a. Given an adjacency matrix for an undirected bipartite graph suggest an algorithm for determining the set of connected components in that graph. What is the running time of the algorithm?

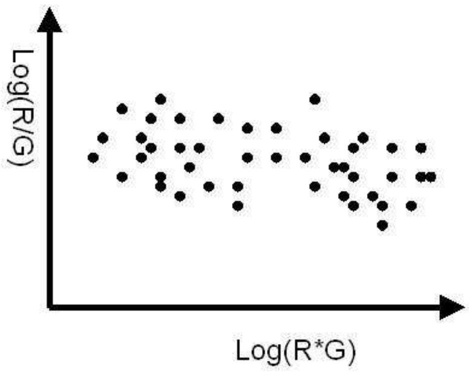
1. b. Our goal is to find a mapping for the set of genes in both genomes. Given a bipartite graph as discussed above, we denote by 'graph mapping' an algorithm that assigns to a some (or all) of the genes in genome A *at most* one gene in genome B (note that a genes $a \in A$ can be assigned a gene $b \in B$ only if the graph contains an edge between a and b). 'maximal mapping' is a graph mapping that results in the highest number of genes in A being assigned a gene in B. In other words, we are trying to find as many pairs as we can between the two sides of the graph so that no gene is assigned to more than one pair. Present an algorithm for solving this problem (Hint: it is often referred to as 'matching'). What is the running time of the algorithm you presented?

Normalization
In class,we discussed locally weighted linear regression and mentioned that we can use a Gaussian centered at x to determine the weight that should be assigned to points (genes) around x. Here we will explore issues related to this weight.

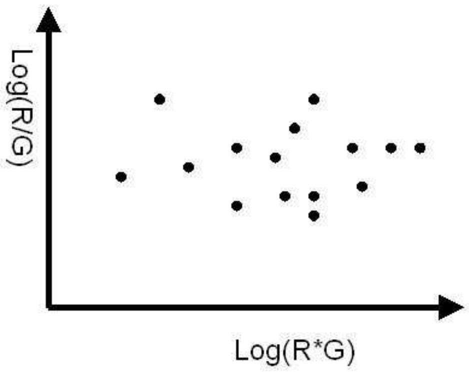
2. a. What is the effect of having a large variance for such a Gaussian? A small variance?

2. b. Which variance (large or small) would be appropriate to each of the two figures below? Explain.

2. c. In gene expression experiments we measure thousands of genes. However, most of these genes are expressed at relatively low levels and only a few are expressed at high levels (high $R * G$ values). How can we accommodate such a dataset with the Gaussian weighting method? Explain.



(a)



(b)

Bi-Clustering

In this problem you will develop and implement a bi-clustering algorithm. A Bi-cluster is a cluster containing a subset of the experiments and a subset of the genes. In this problem we will not allow overlap between the Bi-clusters, though other methods allow such overlap.

We will once again rely on bipartite graphs. By answering the questions below you will develop (and implement) a method that uses bipartite graphs for bi-clustering.

3. a. Assume you are given a time series expression dataset where rows represent genes and columns represent time points. How can these be represented using a bipartite graph ? What do edges in this graph correspond to?

Since gene expression data contains non-discrete values we will first discretize the data. Every value above (log ratio) 0.9 will be set to 1 and every value below (log ratio) -0.9 will be set to -1. Values between -0.9 and 0.9 will be set to 0.

3. b. Using an unweighted bipartite graph (that is, all edges have the same weight of 1) as you have described above, how can you represent both activation (1) and repression (-1) (remember, we would like to cluster activated genes in a different cluster than the repressed ones)?

Assume the graph has a bounded out degree on the left (that is, no node on the left side has more than d outgoing edges). Also, assume that we are looking for complete subgraphs, that is a subset of the nodes on the left ($l \subseteq A$) and a subset of the nodes on the right ($r \subseteq B$) where each node in l is connected to all nodes in r and vice versa. **3. c.** What is the largest possible size of r?

3. d. Present a $O(n2^d)$ algorithm for finding the maximal complete subgraph (where maximal means that it has the most number of nodes from B).

While the algorithm you presented in **d.** is useful, the running time may be too large. Instead, we will use heuristic search to find large subgraph (which will correspond to a bi-cluster). Below I suggest a possible (simple) heuristic. If you have a better idea you are more than welcomed to implement your own heuristic, however you will need to explain what exactly you did and why (there will be 5 points bonus for useful heuristics different from the one discussed below).

For each of the nodes in $v \in B$ we will first determine the set of nodes connected to it in A. Denote this set by l. Next we determine the set of nodes in B that are connected to nodes in l, denote this

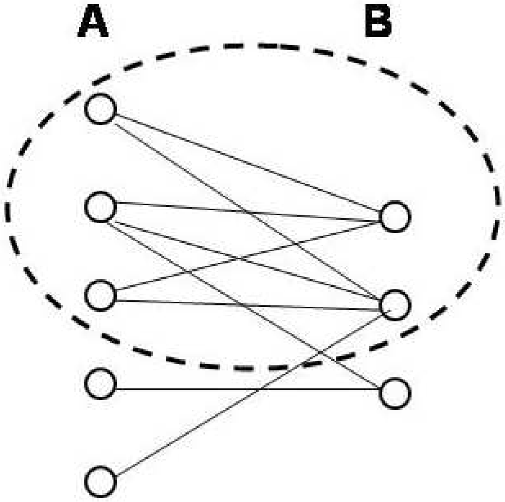


Figure 1: A bipartite graph. The dashed circle contains a complete subgraph in this graph and is a good candidate for a bi-cluster.

set by r (note that this set includes v). We next compute a score for the l, r subgraph (see below). This is process is repeated for all nodes in B and we select the highest scoring subgraph as our first bi-cluster, remove all edges in this subgraph and repeat this process to find the second bi-cluster and so on. The only problem left is to determine a score for a subgraph.

3. e. How can we use binomial distribution to compute a score for a subgraph? Present the formula and the meaning of each of the parameters you are using.

Download the time series dataset from the course website (alphaCycle.txt). This file can be uploaded directly to Matlab. Also download and the list of gene names (alphaGenes.txt). Each row in the time series file corresponds to a gene in the gene name file (that is, the first row is for the first gene, the second for the second and so forth). Each column in the time series file represents one time point.

3. f. Implement the above algorithm (or your own heuristic). Select the first five bi-clusters. For each one hand in a plot of the average expression value for genes in that cluster over all time points, the set of time points selected for this bi-cluster and the top 5 GO categories enriched for genes in that bi-cluster. For the GO enrichment, go to:

http://llama.med.harvard.edu/cgi/func/funcassociate

Paste the names on the genes in each of your top five clusters (one at a time) and select S. cerevisia as the organism. For each cluster copy the top five GO categories and their p-values.

To summarize, in addition to your answers to the questions in problem 3, you will need to hand in the following:

1. Create a directory with your program, the input files you used and a README file that explains how to perform **f** using your program. Email me (zivbj@cs.cmu.edu) a zipped version of this directory.

2. For **f** plots of the average expression for each of the top 5 bi-clusters, the time points that where selected for each and the GO terms with their p-values.