MATH0462 - Discrete Optimization
Project

1 Module analysis in cancer diagnosis

Large-scale gene expression profiling is a popular approach for the identification of molecular biomarkers for disease diagnosis, prognosis, and response to treatment. However, identifying single genes is often too limited as there is a large correlation between genes and sometimes a large variation from one patient to the other. Some authors have proposed to study groups of genes that are highly correlated in a disease as a fundamental marker of such a disease. In this project, your task is to recover groups of genes that can be considered as a **module**, i.e. a group of genes sharing co-expression patterns in a disease.

The first task, consisting in recovering the co-expression of two genes has already been performed in several databases. You will therefore work with pretreated data indicating the co-expressions of genes. In other words, the data file indicates a list of pairs of genes, together with a value, indicating their coexpression (as computed with Spearman's correlation). You may assume that pairs of genes that are not present in the file do not share expressions in the disease. You may also want to require a stricter threshold, i.e. pairs of genes with a too low value of the correlation may be considered as not sharing expression. The problem is to identify modules, i.e. large groups of genes that have co-expression altogether. You must therefore define what you consider to be a module. Pay attention that, since this is real data, you may allow for some missing links between the genes and still consider the subset as a module.

In the dox repository, a series of txt files are given representing co-expression of genes in a given disease. The initial study was made for identifying biological markers for breast cancer [1]. Four files related to breast cancer are indicated with a filename starting with B. You are however free to study any file in the list (see the file meaning.txt for a description of the disease they aim to study). It is also advised to cross check the results obtained on one file on the data of another file with a similar disease.

Questions

For all first three questions, you are asked to write a mixed-integer formulation and a heuristic. The fourth question may combine MIP models or heuristics in a unified program.

1. Write a model and an algorithm that finds the largest module where the module is defined as a set of genes in which all pairwise co-expressions are present.

2. Write a model and an algorithm that finds the largest module where the module is defined as a set of genes in which all pairwise co-expressions are present, except a predefined number of them.
3. Write a model and an algorithm that finds the largest module where the module is defined as a set of genes in which most pairwise co-expressions are present and the number of allowed missing co-expressions is bounded by an increasing function of the size of the module.
4. Write a program that finds a covering of the different genes in modules. You may choose any definition of a module from the first three questions. The union of all modules should cover the full list of genes and should not overlap too much. You may take any definition that you like of what *"not overlapping too much"* means.

Instructions

All projects will be done by groups of 2. You must write a very short report (3 pages maximum) including a human-friendly short version of your models and your heuristics. Everything (code+report) should be sent by e-mail to `q.louveaux@uliege.be` and `mathias.berger@uliege.be`. The deadline for submitting your project is May 10. The presentation of the project should be done on Wednesday May 12. No formal presentation is needed but you should possibly be able to discuss the various tests that you have performed.

Références

[1] Zhiao Shi, Catherine K Derow, Bing Zhang. Co-expression module analysis reveals biological processes, genomic gain, and regulatory mechanisms associated with breast cancer progression. *BMC Systems Biology*, 2010, 4 :74.