Université de Bordeaux Année 2017 - 2018
Master 2 Bio-Informatique

Visualisation d'Information
Projet – Analyse visuelle de données d'expression génique –

Objectif : L'objectif de ce projet est de réaliser une analyse d'un jeu de données qui vous est fournit. Pour cela, vous trouverez ci-dessous plusieurs sous-objectifs à atteindre (le barème est donné à titre indicatifs). Le résultat sera remis sous la forme d'un fichier contenant le code python (compatible avec l'IDE Python de Tulip) permettant à partir d'un fichier au format Tulip (cf ci-après) d'atteindre ces différents sous-objectifs. Vous devrez aussi fournir un rapport d'une dizaine de pages expliquant les algorithmes, choix que vous aurez faits et informations tirées de l'analyse que vous ferez des données.

Modalités : Projet à réaliser en binôme

Date de remise : 25 janvier minuit et soutenance le 26 janvier

Fichiers fournis : Vous trouverez à l'adresse :
www.labri.fr/perso/bourqui/downloads/cours/Master/2017/Projet/
un fichier contenant un réseau de régulation de gènes d'Escherichia coli K12 qui vous servira dans ce projet.

Partie 1 : Pré-traitement & première visualisation (/5 points)

Comme vous pouvez le voir après le chargement du réseau fourni, le graphe tel qu'il est affiché n'apporte que peu d'information. Écrire le code python permettant d'affecter

- des étiquettes aux sommets du réseaux (les locus seront utilisés)
- une taille (non-nulle) à chaque sommet afin de pouvoir visualiser correctement les étiquettes
- des couleurs/formes différentes pour les différents types de régulation (les arêtes de ce réseau)
- des positions aux sommets du graphe (le choix et le paramétrage de l'algorithme de dessin est libre)

Partie 2 : Partitionnement des gènes (/5 points)

Question 2.1 : Écrire l'algorithme permettant de construire un graphe complet dont les sommets sont les sommets du réseau initial et les poids associés aux arêtes correspondent à une distance entre les gènes (basée sur leurs niveaux d'expression).

Question 2.2 : Le graphe généré par l'algorithme contient beaucoup trop d'arêtes (il est alors difficile de manipuler et/ou calculer quoi que ce soit sur ce graphe). Proposer une méthode de

filtrage pour réduire le nombre des arêtes (ce choix devra être justifier dans le rapport).

Question 2.3 : Écrire un algorithme permettant de partitionner les gènes en fonction de leurs niveaux d'expression (pensez à utiliser les plugins proposés par Tulip).

Partie 3 : Construction d'un carte de chaleur (/5 points)

Question 3.1 : Écrire un algorithme permettant de construire un graphe dont chaque sommet correspond au niveau d'expression d'un gène du réseau initial et colorer les sommets de ce graphe en fonction du niveau d'expression.

Question 3.2 : Positionner les sommets du graphe précédent sur une grille de telle sorte que chaque ligne corresponde à un gène et chaque colonne à un pas de temps.

Question 3.3 : Écrire un nouvel algorithme de dessin permettant de positionner tous les sommets de chaque groupe de gènes (issu de la question 2.3) de manière contiguë.

Partie 4 : Analyse des données (/5 points)

Cette partie est laissée relativement libre, l'objectif est de permettre de comprendre l'évolution des niveaux d'expression de gène et dans quels processus biologiques ces gènes sont impliqués.

Vous pouvez utiliser n'importe quelle source de données externe dans la mesure où cela est fait de manière programmatique (vous pourrez aussi utiliser vos propres connaissances). Vous pourrez aussi programmer n'importe quelle autre représentation, partitionnement, extraction de sous-parties sur/sous exprimées, *etc*...

Les résultats de cette partie devront être décrits dans le rapport.

Partie 5 : Pour aller plus loin [Bonus]

Question 5.1 : Le partitionnement calculer dans la question 2.3 ne permet de générer qu'un niveau de partitionnement. Écrire un nouvel algorithme permettant de générer plusieurs niveaux.

Question 5.2 : Adapter l'algorithme de la question 3.3 pour prendre en compte ce partitionnement multi-échelle.

Question 5.3 : Ajouter à la carte de chaleur un dessin d'arbre représentant le partitionnement multi-échelle.

Description du format de graphe :

Dans le fichier fourni, un certain nombre de propriétés existent (en plus des propriétés visuelles, commençant par « view* ») :

Nom	Type	Description

Locus	StringProperty	Spécifie pour chaque sommet un locus.
tp* s	DoubleProperty	Spécifie pour chaque sommet le niveau d'expression du gène correspondant à ce « time point ».
Positive	BooleanProperty	Spécifie pour chaque arête si elle représente une régulation positive.
Negative	BooleanProperty	Spécifie pour chaque arête si elle représente une régulation négative.

Voici un extrait vous donnant quelques indications sur le jeu de données utilisé :
« The dataset contains gene expression data from 17 time points, during which E. coli is grown on a mixture of glucose and lactose. The bacterium grows preferentially on glucose until that energy source is depleted, resulting in growth arrest while the cells adjust to growth on lactose. This shift, called the diauxic shift, takes places at about time point 6. At time point 14, the stationary phase is entered in which the organism stops growing due to the lack of nutrients. During this phase, many processes are shut down by the bacterial cell in order to save energy. »