

Projets L3 informatique

Sabine Peres

sabine.peres@univ-lyon1.fr



Lyon 1

Inria



Biometry and Evolutionary Biology
University Claude Bernard Lyon 1 - France

2024-2025

SP1: Visualisation interactive de données médicales

Objectif:

- ▶ Développement d'une **plateforme interactive** permettant d'explorer, visualiser et analyser des données biologiques dans les cellules de différents patients.
- ▶ Comprendre les processus bioénergétiques ou identifier des anomalies métaboliques associées à certaines situations physiologiques et/ou physio-pathologiques.

SP1: Fonctionnalités principales

- ▶ Importation et gestion des données : Chargement de données physiologiques et métaboliques issues de fichiers CSV ou de bases de données publiques. **Visualisation** des réponses adaptatives et des flux métaboliques sous forme de tableaux ou de graphiques.
- ▶ **Graphiques interactifs** : visualisation des flux entre différents patients, évolution des concentrations métaboliques ou des flux sur une période donnée, visualisation des réponses adaptatives lors de variations des conditions physiologiques (exemple de l'exercice).
- ▶ Simulation et comparaison : **Comparaison** des réponses adaptatives et des flux normaux et perturbés pour chaque patient.
- ▶ Analyse avancée : Détection des anomalies métaboliques liées à des maladies, analyse prospective des interventions physiques et cliniques susceptibles d'être proposées dans la prise en charge de ces sujets.

- ▶ **Interface web conviviale** avec une navigation intuitive. Options d'exportation des graphiques et des rapports d'analyse.
- ▶ Développement backend : Utiliser Python avec Flask pour gérer les données. Manipuler les données avec pandas.
- ▶ Visualisation frontend : Créer des graphiques interactifs avec Plotly ou Dash pour des diagrammes dynamiques.

Compétences mises en œuvre:

- ▶ Science des données : pandas, NumPy, ...
- ▶ Visualisation : matplotlib, seaborn, Plotly/Dash, ...
- ▶ Développement web : Flask, HTML/CSS/JS ou React ...

SP2: Construction d'un automate cellulaire pour simuler les interactions entre cellules biologiques

Objectif

- ▶ Développement d'un simulateur basé sur un **automate cellulaire** ou **un système multi-agents** permettant de modéliser les interactions entre cellules biologiques (exemple : prolifération cellulaire, communication, et réponse aux signaux environnementaux).
- ▶ Peut être appliqué à des domaines tels que la biologie des tissus, la modélisation des tumeurs, ou la réponse immunitaire.

Langage et bibliothèques:

Python, Dash, NumPy, Pygame ...

SP2: Description générale

L'idée est de construire un environnement où les cellules sont représentées par des agents ou des cellules sur une grille. Chaque cellule suit un ensemble de règles ou de comportements pour évoluer en fonction de :

- ▶ Ses propres caractéristiques internes (état, énergie, type, etc.).
- ▶ Les interactions avec ses voisines.
- ▶ Les signaux ou stimuli provenant de l'environnement.

Par exemple :

- ▶ Une cellule saine peut se diviser si elle a suffisamment de ressources.
- ▶ Une cellule tumorale peut proliférer rapidement et émettre des signaux pour perturber l'environnement.
- ▶ Une cellule immunitaire peut détecter et attaquer des cellules anormales.

SP2: Fonctionnalités attendues

- ▶ Simulation d'un automate cellulaire ou d'un système multi-agents: - Une grille 2D ou 3D représente l'espace dans lequel les cellules interagissent. - Chaque cellule a des états possibles : sain, proliférant, apoptotique (mort), tumoral, immunitaire, etc.
- ▶ Modélisation des règles d'interaction: - Les cellules interagissent avec leurs voisines selon des règles simples : - Communication chimique (par gradients de molécules). - Transitions entre états en fonction de signaux reçus.
- ▶ Évolution dans le temps: - Visualisation dynamique de la croissance, de la mort ou de la propagation des cellules.
- ▶ Personnalisation des paramètres: - Taille de la grille ou de l'environnement. - Probabilités associées à certains événements (division, mort, etc.).
- ▶ Visualisation et export des résultats - Affichage graphique des états cellulaires au cours du temps. - Export des données (CSV, images) pour analyse ultérieure.

SP3: Visualisation interactive de voies métaboliques calculées par des méthodes d'optimisation

Objectif

- ▶ Développement d'une interface de visualisation interactive permettant d'explorer les voies métaboliques (i.e chemins dans un hypergraphe) calculés par des méthodes d'optimisation (programmation linéaire).
- ▶ L'outil doit permettre aux utilisateurs de comprendre la structure et la dynamique des réseaux métaboliques, ainsi que d'analyser les flux optimaux pour différents scénarios biologiques (par exemple, croissance cellulaire, production de biomolécules, réponse au stress).

Langage et bibliothèque (proposés):

Python, CobraPy Cytoscape, Dash, Plotly

SP3: Fonctionnalités attendues

- ▶ Visualisation de la carte métabolique : - Représenter les métabolites et les réactions sous forme de graphe (nœuds et arêtes). - Utiliser des codes de couleur et des épaisseurs de lignes pour indiquer l'intensité des flux dans chaque réaction.
- ▶ Interactivité: - Permettre de zoomer et de se déplacer dans la carte métabolique. - Cliquer sur une réaction ou un métabolite pour afficher des informations détaillées - Activer/désactiver certaines voies ou métabolites pour analyser leur impact.
- ▶ Analyse de scénarios multiples: - Générer des graphiques pour visualiser les différences entre scénarios.
- ▶ Exportation des visualisations: PNG, SVG, PDF, CSV ou XML.