



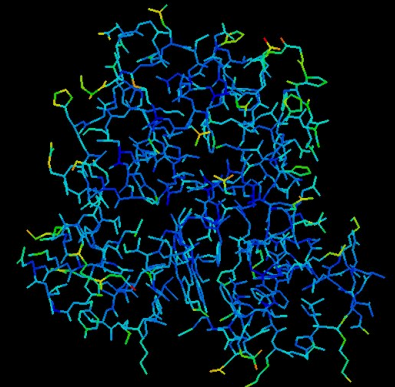
DATAMINING : CLASSIFICATION DE PROTÉINES

Détermination d'une classification de protéines à partir de grands jeux de données.

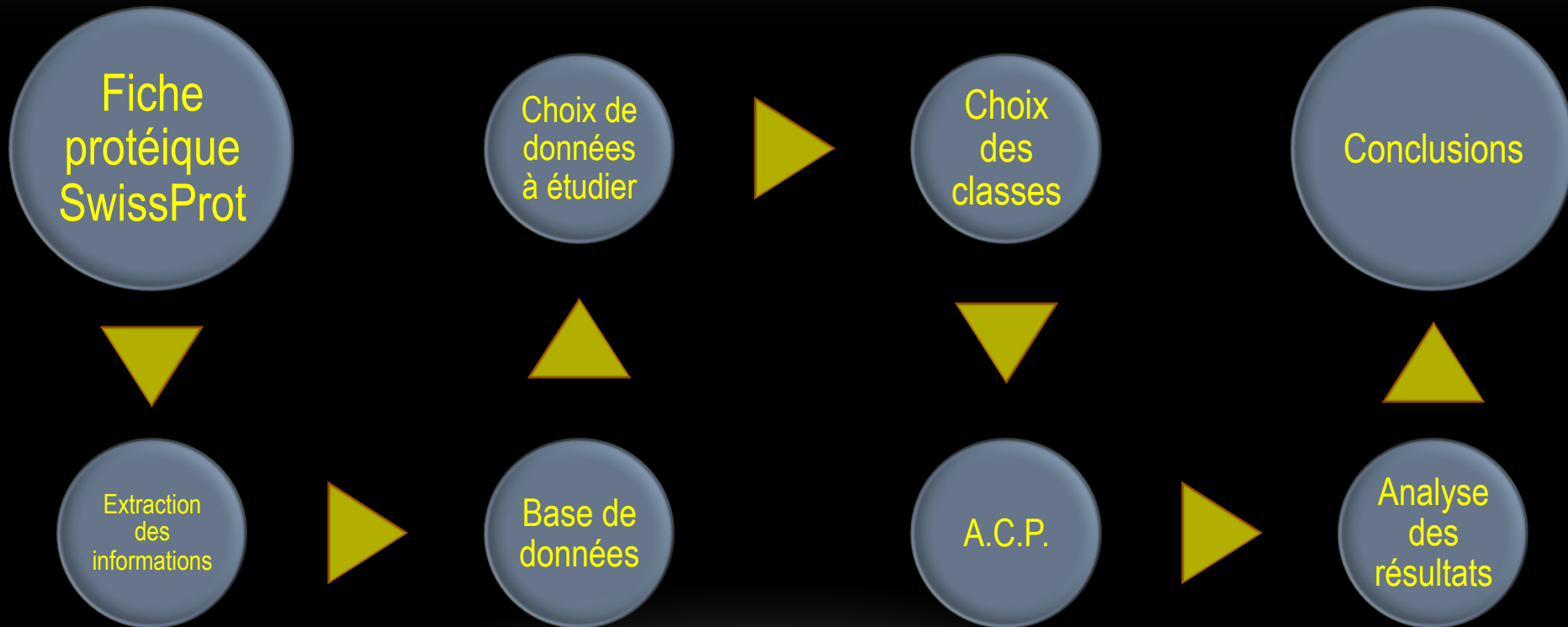
Merce Clémentine Beauquis Sébastien
Gilbert Florian
Campbell Sills Hugo Rai Ghadi

INTRODUCTION

- ✓ Déterminer différentes classes de protéines, à partir de grands jeux de données en utilisant des techniques de datamining
- ✓ Protéine : Macromolécule biologique composée d'un enchaînement d'acides aminés, ayant une activité biologique.
- ✓ Nous avons fait le choix de réaliser notre étude seulement sur les enzymes



1. PIPE-LINE D'ÉTUDE



1.1 ELABORATION DE LA BASE DE DONNÉES

1.2 FICHES SWISSPROT ET EXTRACTION DES DONNÉES

- ✓ Récupération des fiches grâce à un script Bio-Python
- ✓ Extraction des données choisies grâce à BioJava

Exemple de code BioJava :

```
SimpleNamespace namespace = new SimpleNamespace("biojava");  
RichSequenceIterator richSequenceIterator =  
RichSequence.IOTools.readGenbankProtein(bufferedReader, namespace);  
while (richSequenceIterator.hasNext()) {  
    RichSequence richSequence = richSequenceIterator.nextRichSequence();  
    nomProteine=richSequence.getName();  
    sequence = this.genBank.seqString();  
}
```

1.3 REMPLISSAGE DE LA BASE DE DONNÉES

- ✓ Utilisation d'une instance de JPA (Java Persistence API), eclipseLink
- ✓ Facilité de mise en place et rapidité à appréhender l'API

Exemple de code :

```
EntityManagerFactory entityManagerFactory =Persistence.createEntityManagerFactory("dataMining");
EntityManager entityManager = entityManagerFactory.createEntityManager();
entityManager.getTransaction().begin();
if (entityManager.find(EntryInformation.class, entryInformation.getIdProtein())!=null){
    entityManager.merge(entryInformation);
}else{
    entityManager.persist(entryInformation.getEntryInformation());
}
entityManager.getTransaction().commit();
entityManager.close();
```

1.4 CHOIX DES DONNÉES À ÉTUDIER ET DES CLASSES

- ✓ Étude basée sur la possible corrélation entre la composition en acides aminés d'une enzyme avec la présence de structure secondaire ou de domaine identifiable de cette molécule.
- ✓ Les classes sont réalisés en classant les protéines en groupe logique et de taille égale

1.5 ANALYSES EN COMPOSANTES PRINCIPALES

- ✓ Réalisée grâce à un script Java trouvé sur <http://www.classification-society.org/csna/mda-sw/>
- ✓ Produit des fichiers de sortie comportant:
 - La matrice de la somme des carrés et produits en croix
 - Les eigens Vectors
 - Les eigens values de chaque vecteur
 - Ainsi que les coordonnées de chaque point pour les représenter sur les axes

1.6 REPRÉSENTATION GRAPHIQUE DES RÉSULTATS D'ACP

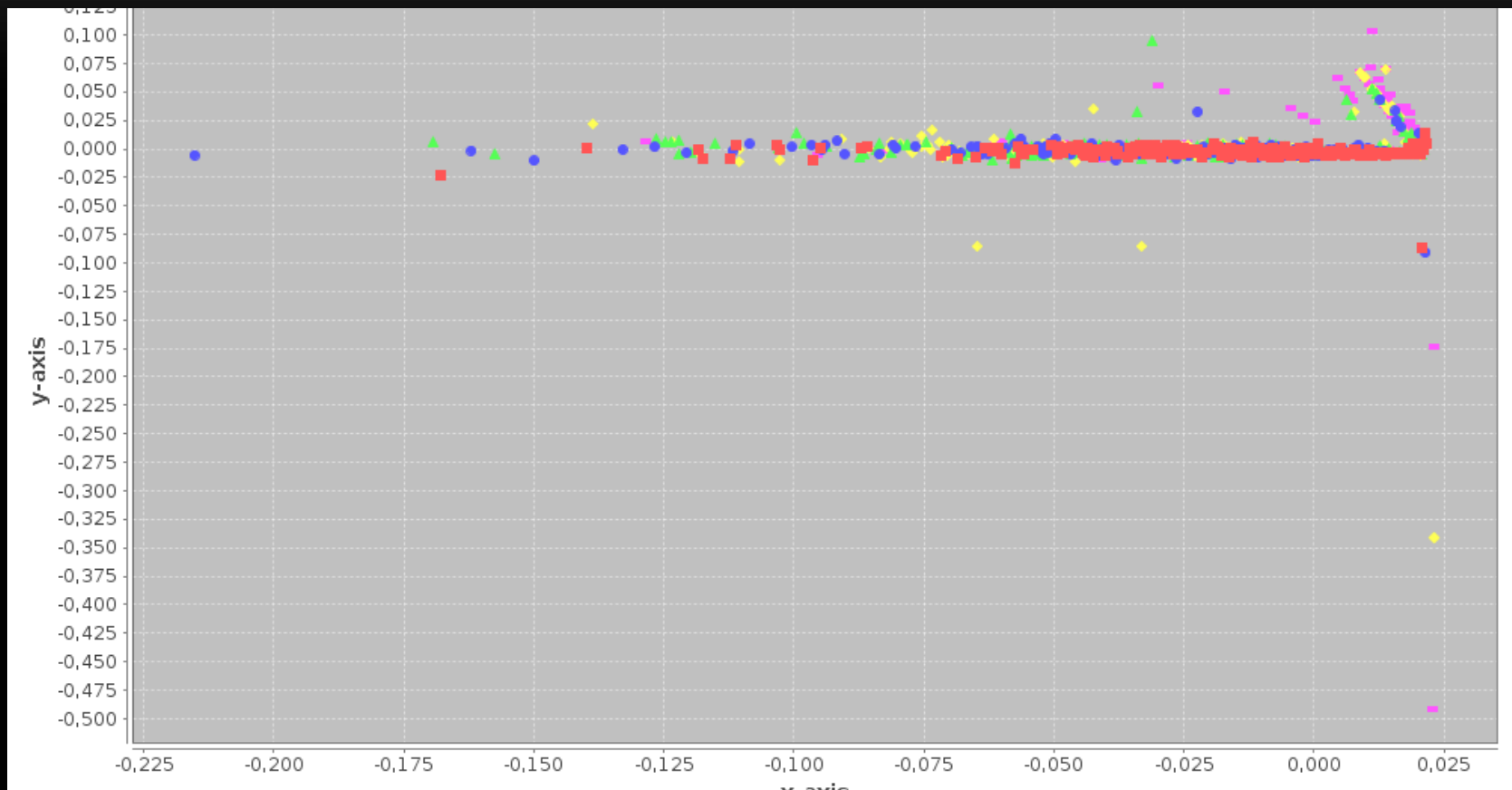
- ✓ Les résultats d'ACP sont représentés grâce à la bibliothèque JFreeChart.
- ✓ Facilité d'approche de la bibliothèque, ainsi que multiple possibilités

Exemple de code:

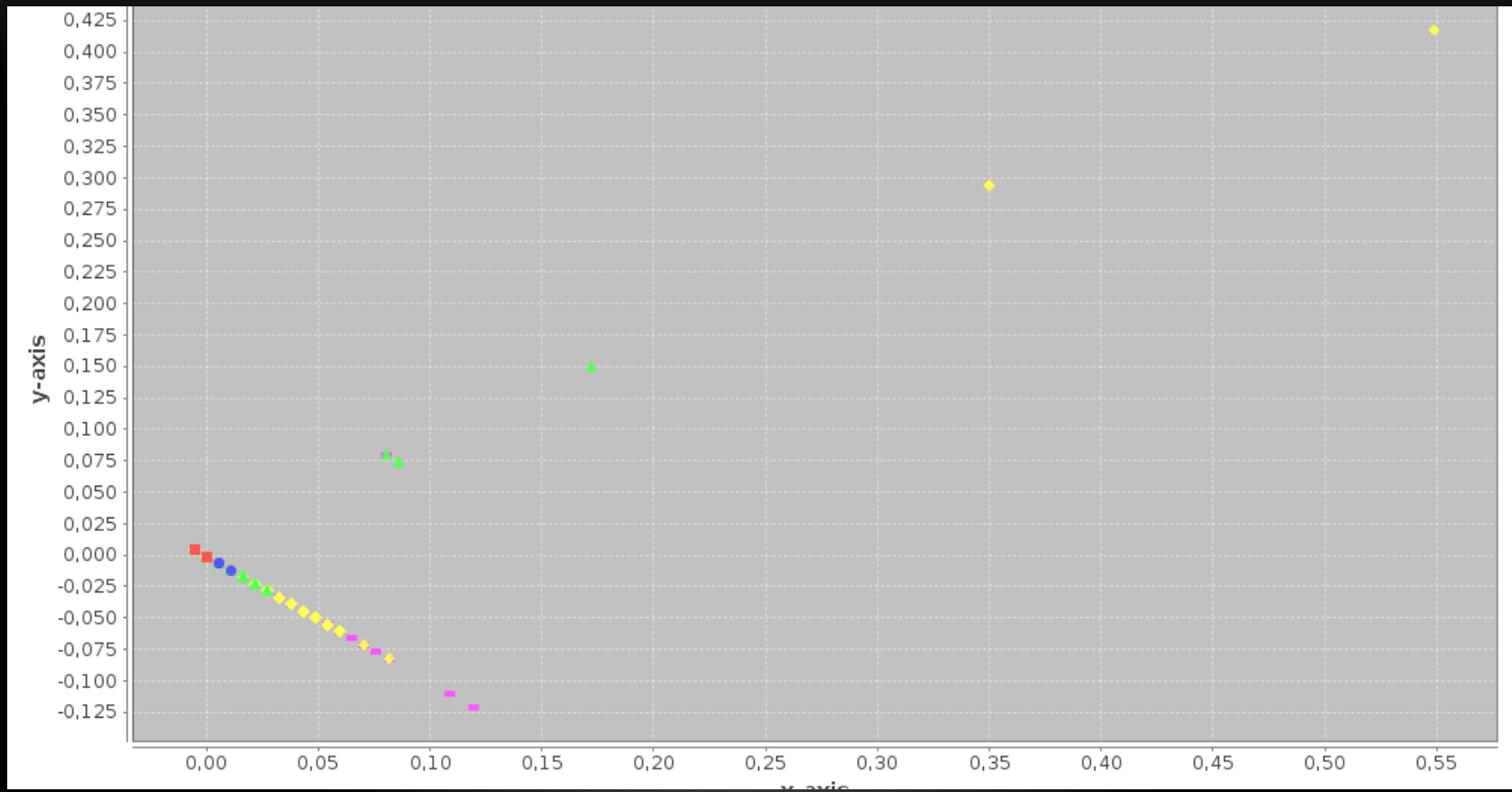
```
YSeries series = new XYSeries("Average Size");
series.add(20.0, 10.0);
series.add(40.0, 20.0);
series.add(70.0, 50.0);
XYDataset xyDataset = new XYSeriesCollection(series);
JFreeChart chart = ChartFactory.createAreaXYChart
    ("Sample XY Chart", // Title
    "Height",          // X-Axis label
    "Weight",          // Y-Axis label
    xyDataset,         // Dataset
    true               // Show legend
    );
```

2. ANALYSE DES RÉSULTATS

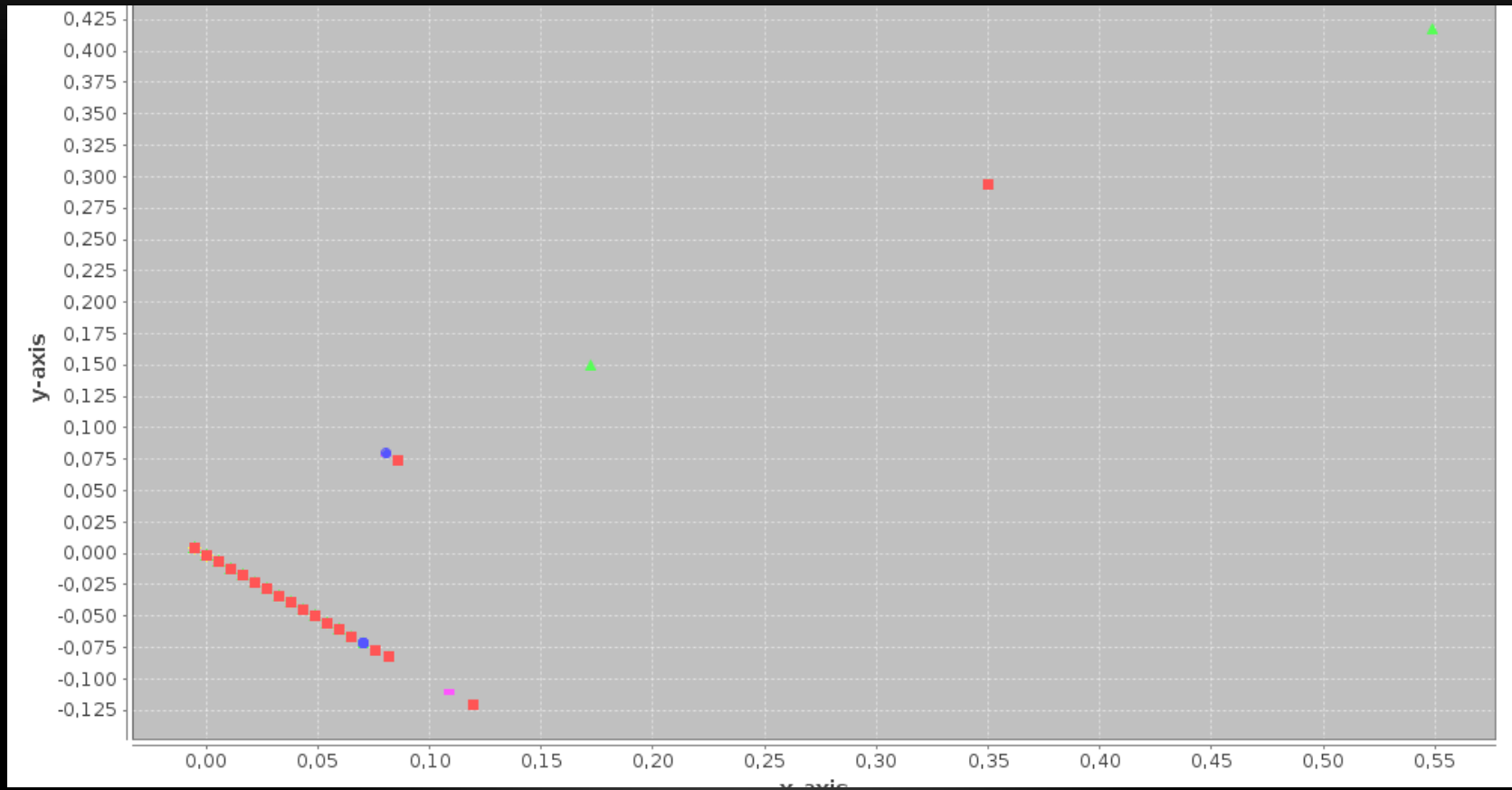
2.1 HYDROPHOBICITÉ DE L'ENZYME VS DOMAINES ET STRUCTURES SECONDAIRES



2.2 NOMBRE D'HÉLICES VS DOMAINES



2.3 NOMBRE DE FEUILLETS VS DOMAINES



CONCLUSION