Projet court

Projet:

DSSP

Auteurs:

Garis Cluzeau

Master 2 – Biologie informatique

2023-2024

Encadrée par: Professeur Jean Christophe Gelly

Introduction

Une protéine est un assemblage tridimensionnel d'acides aminés, organisé selon quatre niveaux structuraux hiérarchiques.

La structure primaire correspond à la séquence linéaire des acides aminés, reliés par des liaisons peptidiques.

La structure secondaire représente un premier niveau de repliement de la chaîne polypeptidique, maintenu par des liaisons hydrogène entre résidus proches. Elle se manifeste principalement sous forme d'hélices α et de feuillets β.

La structure tertiaire reflète l'agencement tridimensionnel complet d'une seule chaîne polypeptidique. Elle résulte d'interactions hydrophobes, de liaisons hydrogène, de forces de Van der Waals, ainsi que de ponts disulfure qui stabilisent la conformation.

La structure quaternaire, enfin, correspond à l'association de plusieurs chaînes polypeptidiques (ou sous-unités) en un complexe protéique fonctionnel.

La connaissance de la structure d'une protéine est essentielle pour comprendre ses fonctions biologiques. Toutefois, la prédiction ou la détermination expérimentale de la structure tertiaire reste complexe, coûteuse en temps et en ressources. Une alternative consiste à se focaliser sur la structure secondaire, plus accessible et souvent suffisante pour des analyses préliminaires.

La structure secondaire peut être soit prédite par des méthodes computationnelles, soit directement assignée à partir de structures expérimentales. Plusieurs algorithmes permettent cette assignation, parmi lesquels STRIDE, DEFINE, ou DSSP (Dictionary of Secondary Structure of Proteins). Nous nous intéressons ici à ce dernier.

L'algorithme DSSP, largement utilisé dans la communauté scientifique, attribue une structure secondaire à chaque acide aminé d'une protéine en se basant sur les liaisons hydrogène détectées dans sa structure tridimensionnelle. Il repose sur les critères initiaux définis par Corey et Pauling en 1951, et distingue huit types de structures secondaires (hélice α, brin β, coude, coil, etc.). Chaque type est associé à une lettre, facilitant ainsi leur identification automatique.

L'objectif de ce projet était de réimplémenter la méthode DSSP en exploitant les liaisons hydrogène d'une protéine afin d'assigner sa structure secondaire. L'ambition était également de restituer les résultats selon un format proche de celui proposé par l'algorithme DSSP, dans un souci de compatibilité et d'interprétation claire pour les utilisateurs.

Matériel et Méthode

Pour implémenter la méthode DSSP, l'article de Kabsch et Sander [3] a servi de base au code. En effet celui-ci défini clairement la façon de reconnaître les éléments composants les structures secondaires. Tout d'abord, le n-turn qui définit une liaison entre un résidu en position (i) et un autre en position (i+n). L'écart entre ces deux résidus est forcément de 3, 4 ou 5. Si l'on détecte des turn successifs, alors on peut dire que l'on a une hélice. Cette hélice est donc la première structure secondaire que l'on reconnaît dans l'article. Pour pouvoir détecter les feuillets bêta, il faut d'abord définir les bridges puis les ladders, qui sont des bridges successifs. L'article expliquant toute la méthode pour reconnaître les différentes structures, il sert donc à notre méthode de programmation.

Pour débuter nous devons choisir un fichier pdb d'une protéine issue de la Protein Data Bank (PDB).Ce fichier PDB peut être utilisé sans aucun changement avec l'algorithme DSSP ce qui est également notre cas car notre première fonction est la suivante :

Lecture et parsing du fichier PDB parse_pdb_residues(pdb_path)

Cette fonction lit un fichier PDB et extrait, pour chaque résidu, les coordonnées des atomes essentiels à l'analyse structurale (N, CA, C, O, H, H1). Les résidus sont stockés dans un dictionnaire dont la clé est un tuple (chaîne, numéro de résidu).

Dans un second temps, après avoir extrait les information importante du pdb nous allons utiliser la fonction get_hydrogen_atom_position() qui est nécessaire afin de palier à l'absence des atomes d'hydrogène ce qui est fréquent en fonction de la résolution des fichier pdb choisi.

Estimation des atomes H manquants get_hydrogen_atom_position()

Lorsque l'atome d'hydrogène n'est pas fourni dans le fichier PDB, cette fonction l'estime géométriquement à partir de la position du carbone précédent (C), de l'azote (N) et de l'alpha-carbone (CA) du résidu courant, via des vecteurs directionnels normalisés.

Apres cette nouvelle étapes, nous pouvons calculer la présence potentiel de liaison hydrogène entre chaque paire de résidu. Pour réaliser cela nous utilisons la fonction is_hydrogen_boud.

Détection des ponts hydrogène is_hydrogen_bond()

Cette fonction calcule si un pont hydrogène est présent entre deux résidus en utilisant l'énergie électrostatique de liaison E_{elec}, approximée par :

$$Eelec = q1 * q2 * f\left(\frac{1}{dON} + \frac{1}{dCH} - \frac{1}{dOH} - \frac{1}{dCN}\right)$$

Où q1=0.42 q2=0.20, f=332, et d sont les distances entre les atomes correspondants. Un pont hydrogène est validé si E_{elec}<−0.5 kcal/mol.

Maintenant que les potentiels liaison hydrogène sont détecté nous pouvons passer à la détection des structures secondaires. Pour réaliser cette étape cruciale nous utilisons plusieurs fonctions. Chacune permet de détecter un type de structure secondaire ou d'écrire le résultat.

Détection des brins β find_beta_bridges()

Deux types d'arrangements de ponts hydrogène sont recherchés :

- **Antiparallèles** : lorsque les ponts sont croisés entre deux résidus distants.
- **Parallèles** : lorsqu'un résidu i forme un pont avec j+1, et j avec i+1.

Les ponts détectés sont stockés dans un dictionnaire avec leur type.

Annotation des brins β sur la séquence annotate_beta_strands_on_sequence()

Une séquence secondaire simplifiée est générée où :

- 'E' marque un résidu impliqué dans un brin β.
- '-' indique l'absence de structure secondaire.

Des règles de continuité sont appliquées : un résidu situé entre deux brins est marqué comme brin, et les résidus isolés sont supprimés pour réduire les faux positifs.

Détection des hélices α, 3_{10} et π assign()

Cette fonction construit une **carte de ponts hydrogène** (hbmap) entre tous les résidus. Elle recherche des motifs diagonaux à distance 3, 4 ou 5 dans la matrice, caractéristiques des hélices :

- distance 3 → hélice 3_{10},
- distance 4 → hélice α,
- distance 5 → hélice π.

Ces motifs sont ensuite regroupés pour annoter les résidus hélicoïdaux. Les résidus restants sont classés comme **boucles (loops)**.

La sortie est une **matrice one-hot** représentant :

- boucle (L) = [1, 0, 0],

- hélice (H) = [0, 1, 0],
- brin β (E) = [0, 0, 1].

La dernière étapes est de combiner les resultat des fonction de recherche

Combinaison et format final combine_secondary_structure()

Cette fonction combine les deux types de détections :

- hélices détectées via les motifs de ponts hydrogène courts,
- brins β détectés via les ponts hydrogène inter-résidus.
-

Comparaison avec DSSP via MDAnalysis

Afin de valider notre prédicteur de structure secondaire, nous avons comparé ses résultats à ceux obtenus via **DSSP** à l'aide de la bibliothèque **MDAnalysis**.

Fonctionnement :

1. **Extraction DSSP**
   La fonction get_dssp_secondary_structure utilise MDAnalysis.analysis.dssp.DSSP pour attribuer à chaque résidu une structure secondaire :
   - 'H' pour hélice
   - 'E' pour brin β
   - 'L' pour boucle ou autre.
2. **Formatage des prédictions**
   La fonction get_prediction_dict convertit la sortie de notre prédicteur en dictionnaire pour permettre la comparaison.
3. **Évaluation de la précision**
   La fonction compare_structures calcule le pourcentage de correspondance entre la prédiction et l'annotation DSSP, résidu par résidu.

Résultat

Dans notre code, nous détectons l'ensemble des types d'hélices (3_{10}, α, π) ainsi que les feuillets β. Toutefois, nous regroupons ces différentes formes sous des catégories simplifiées : hélice ou brin. Ce choix a été fait dans un souci de clarté et de facilité de comparaison.

En effet, pour évaluer notre modèle, nous utilisons la fonction DSSP du module MDAnalysis. Or, cette fonction ne distingue pas les sous-types d'hélices ou de brins et renvoie uniquement trois états de structure secondaire : hélice, brin, ou désordonné.

En plus de l'annotation de la structure secondaire, notre code fournit également le nombre total de résidus ainsi que le nombre de résidus correctement prédits, ce qui permet de calculer le taux de précision de la prédiction.

Pour évaluer la robustesse de notre approche, nous avons choisi de tester une protéine représentative de chaque grande famille fonctionnelle. Les protéines sélectionnées sont les suivantes :

Tableau 1Proteine représentative des différentes familles protéiques

Famille	**Protéine**	**Code PDB**	**Commentaires**
Transmembranaire	Rhodopsine bovine	**1F88**	Rhodopsine en conformation inactive (7 hélices TM typiques).
Signal peptide	Insuline humaine	**4EY1**	Structure du dimère d'insuline.
Domaines globulaires	CDK2	**1FIN**	Domaine catalytique d'une kinase typique. (Cyclin-dependent kinase 2)
Désordonnée	p53	**2OCJ**	Domaine de liaison à l'ADN, partie centrale structurée de p53. (avec région structurée)
Facteur de transcription	NF-κB	**1NFK**	Domaine de liaison à l'ADN du complexe NF-κB. (p50/p65 dimère)
Enzyme hydrolase	Trypsine bovine	**2PTN**	Serine protéase classique avec site actif exposé.
Canal ionique	KcsA	**1BL8**	Structure tétramérique du canal potassique. (canal potassique bactérien)
GPCR	Récepteur β2-adrénergique	**2RH1**	Première structure cristallisée d'un GPCR humain.

Famille	Protéine	Code PDB	Commentaires
Liaison ADN/ARN	Hélicase DDX3X	**5E7I**	Domaine catalytique de DDX3X lié à l'ARN. (domaine hélicase)

Avec ce groupe de protéine nous avons obtenu les résultats présentés dans le tableau suivant

Tableau 2Résultat de comparaison avec DSSP MDanalisys

	1bl8	1fin	1nfk	2ocj	2ptn	2rh1	4ey1	5e7i	1f88
Résidus comparés	388	116	624	776	220	442	102	1272	643
Correspondances exactes	388	1055	265	718	192	429	66	1076	629
Précision de la prédiction (%)	100	94.53	42.47	92.53	87.27	97.06	64.71	84.59	97.82

Nous pouvons observer que dans l'ensemble les notre nouvel algorithme détecte correctement la structure secondaire des protéines. En effet la moyenne de précision de prédiction tourne autour de 90% à l'exception de la protéine 1nfk et 4ey1 avec respectivement 42.47% et 64.71% de bonne prédiction.

Ces deux mauvais résultats peuvent s'expliqué en partie par la mauvaise qualité du pdb de la protéine 1nfk dont les métriques sont mauvais d'autre parts cette protéine se lie à l'ADN ce qui peut perturber la détection de liaison hydrogène et ainsi l'assignation des structures secondaire.

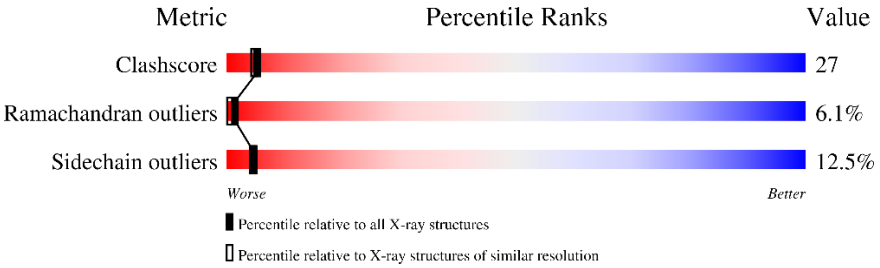

Figure 1 Métriques de la protéine 1nfk

Le second mauvais résultat pour la protéine 4ey1 pourrai s'expliquer par un nombre important de petite hélice. Cette caractéristique met en difficulté notre algorithme qui détecte parfois avec difficulté le début et la fin d'une hélice. D'autre part cette protéine est une répétition d'un même domaine donc une petite erreur est amenée à se répéter plusieurs fois dans cette protéine.

Figure 2 structure de la protéine 4ey1coloré par chaine

Discussions

Dans notre projet, nous avons fait le choix de ne représenter que trois types de structures secondaires. Cette approximation est justifiée par le peu d'intérêt que nous portons aux différents sous-types de structure secondaire. En effet, DSSP est utilisé en première intention afin d'explorer les protéines.

Parmi les packages contenant DSSP, nous retrouvons MDAnalysis, bien évidemment, mais aussi Biopython, MDTraj et PyDSSP. Parmi ces quatre packages, deux proposent une sortie directement regroupée en trois types de structures secondaires. Ce qui illustre la nécessité d'avoir 3 types de structure.

Bibliographie

Minami, S. (n.d.). *PyDSSP* [Python software]. GitHub. https://github.com/ShintaroMinami/PyDSSP

Gorelov, S., Titov, A., Tolicheva, O., Konevega, A., & Shvetsov, A. (2024). *DSSP in GROMACS: Tool for Defining Secondary Structures of Proteins in Trajectories*. **Journal of Chemical Information and Modeling**. https://doi.org/10.1021/acs.jcim.3c01344

Kabsch, W., & Sander, C. (1983). *Dictionary of protein secondary structure: Pattern recognition of hydrogen-bonded and geometrical features*. **Biopolymers, 22**(12), 2577–2637. https://doi.org/10.1002/bip.360221211