Apprentissage de fonctions de tri pour la prédiction d'interactions protéine-ARN

Adrien Guilhot-Gaudeffroy[(1),(2),(3),♯], Jérôme Azé[(1),(3),(4)], Julie Bernauer[(2),(3)], Christine Froidevaux[(1),(3)]

[(1)]LRI, CNRS UMR 8623, Université Paris-Sud, 91405 Orsay
[(2)]LIX, CNRS UMR 7161, École Polytechnique, 91120 Palaiseau
[(3)]Projet AMIB, INRIA Saclay-Île de France, 91120 Palaiseau
[(4)]LIRMM, CNRS UMR 5506, UM2, CC 477, 34095 Montpellier
[♯]adrienguilhot@lri.fr

Résumé. Les fonctions biologiques dans la cellule mettent en jeu des interactions 3D entre protéines et ARN. Les avancées des techniques exérimentales restent insuffisantes pour de nombreuse applications. Il faut alors pouvoir prédire *in silico* les interactions protéine-ARN. Dans ce contexte, nos travaux sont focalisés sur la construction de fonctions de score permettant d'ordonner les solutions générées par le programme d'amarrage protéine-ARN RosettaDock. La méthodologie d'évaluation utilisée par RosettaDock impose de trouver une fonction de score s'exprimant comme une combinaison linéaire de mesures physico-chimiques. Avec une approche d'apprentissage supervisé par algorithme génétique, nous avons appris différentes fonctions de score en imposant des contraintes sur la nature des poids recherchés. Les résultats obtenus montrent l'importance de la signification des poids à apprendre et de l'espace de recherche associé.

1 Introduction

La plupart des mécanismes cellulaires mettent en jeu des complexes protéine-ARN. La compréhension de leurs fonctions dans un but thérapeutique ne peut se faire que par une connaissance fine des mécanismes moléculaires. Même si plus d'un millier de structures 3D de complexes protéine-ARN sont disponibles dans la *Protein Data Bank* [1], base de données de référence des structures 3D, la résolution expérimentale reste longue et coûteuse, parfois même impossible. Les travaux présentés dans cet article sont focalisés sur l'amélioration d'une des approches de référence dans le domaine de la prédiction de l'amarrage (docking) de structures 3D *in silico* : RosettaDock (Gray et al. (2003)). L'objectif de ces approches est de modéliser la protéine et l'ARN et d'en prédire les assemblages 3D les plus probables. De nombreuses méthodes, dont RosettaDock, fonctionnent en deux phases imbricables l'une dans l'autre : (1) génération d'un large ensemble de candidats [2] et (2) évaluation de ces candidats pour ne retenir que les plus plausibles. La "qualité" des candidats est évaluée avec une fonction de score

1. `http://www.wwpdb.org/`
2. assemblage de la protéine et de l'ARN

dédiée à la problématique de l'amarrage (Lensink et Wodak (2010)). La complexité des objets étudiés rend quasiment impossible l'obtention d'un candidat en tous points identique à la solution obtenue expérimentalement. Afin de déterminer pour chaque candidat s'il est acceptable, la mesure RMSD (Root Mean Square Deviation) est calculée entre ce candidat et la solution. Tous les candidats ayant un RMSD $\leq$ 5 Å sont considérés comme des solutions acceptables (des **presque natifs**), les autres candidats étant appelés **leurres**.

Les travaux présentés dans cet article concernent la construction d'une fonction de score permettant de trier les candidats générés. Il a déjà été montré que les techniques d'apprentissage se prêtaient bien à ce genre de problème Bernauer et al. (2007); Bourquard et al. (2011) . Nous avons choisi d'adapter l'outil RosettaDock pour l'amarrage protéine-ARN en créant une fonction de score spécifique. Les performances obtenues par RosettaDock, dans le cadre de la compétition internationale d'amarrage CAPRI [3], font de cet outil un logiciel très performant pour l'amarrage protéine-protéine.

La problématique de l'amarrage protéine-ARN est assez récente dans CAPRI (Lensink et Wodak (2010); Pons et al. (2010)) et il n'existe pas encore de consensus sur la nature des fonctions de score à utiliser. Dans sa version actuelle, RosettaDock ne dispose pas d'une fonction de score dédiée. Les fonctions de score de RosettaDock sont de la forme suivante : $f(X) = \sum_i w_i x_i$ où X représente le candidat à évaluer, w_i les poids de chaque attribut et x_i les attributs physico-chimiques. Les attributs sont tels que des poids à valeurs positives sont biologiquement interprétables mais la restriction des poids aux valeurs positives impose une contrainte forte sur l'espace de recherche des poids optimaux. Nous avons donc relâché cette contrainte en autorisant les poids à évoluer dans les trois intervalles suivants : $[-1\ ;\ 1]$, $[-1\ ;\ 0]$ et $[0\ ;\ 1]$. La recherche de ces poids est effectuée par un algorithme génétique présenté dans la section 3.

2 Données utilisées

Pour ce travail, nous avons utilisé un jeu de données de 120 complexes binaires (une protéine et un ARN) de référence issus de la PRIDB [4] (Lewis et al. (2011)). Une étape de nettoyage manuelle a été effectuée. Les 120 structures biologiques de référence (**natives**) sont utilisées pour générer des candidats **presque natifs** et **leurres**.

Le procédé de *perturbation* de RosettaDock est utilisé pour générer les candidats. Pour chacune des 120 structures natives, on génère 10 000 candidats par *perturbation* des coordonnées de l'ARN autour de sa position dans la structure native. Pour avoir au moins 30 presque-natifs et 30 leurres sur les 10 000 candidats générés pour chaque structure native, nous avons utilisé trois gammes de translations et rotations différentes : standard, restreinte et étendue (translations moyennes respectives : 3 Å, 1 Å et 9 Å et rotations moyennes resp. : 8 °, 4 °, 27 °). Les 1 200 000 candidats ainsi obtenus forment deux catégories : les **presque natifs** (590 707 candidats de RMSD $\leq$ 5 Å) et les **leurres** (609 293 candidats de RMSD > 5 Å).

Les paramètres physico-chimiques utilisés comme descripteurs sont à valeurs numériques et sont de 6 types : paramètres d'attraction et de répulsion universelle, affinité entre chaque type d'atomes, solvatation, électrostatique, liaisons non covalentes des hydrogènes et des conformations des acides aminés et nucléiques (voir Gray et al. (2003)).

3. CAPRI : Critical Assessment of PRedicted Interactions, `http://www.ebi.ac.uk/msd-srv/capri/`
4. Protein-RNA Interface DataBase : `http://pridb.gdcb.iastate.edu/`

3 Protocole expérimental

L'utilisation de RosettaDock impose que les fonctions de score recherchées sont des combinaisons linéaires des attributs physico-chimiques présentés dans la section 2. Plusieurs formes d'apprentissage ont été testées pour optimiser les poids des différents attributs : régression linéaire, régression logistique et SVM. Les meilleurs résultats ont été obtenus avec une approche par régression logistique dont les poids sont optimisés par l'algorithme génétique (ROGER (Sebag et al. (2003)) adapté à la régression logistique). La fonction à optimiser est l'aire sous la courbe ROC (ROC-AUC). $100\,000$ itérations avec $\mu = 10$ (nombre de parents) et $\lambda = 80$ (nombre d'enfants) sont effectuées.

Afin d'évaluer les performances de notre approche, nous avons mis en place une variante du cadre classique d'évaluation Leave-one-out. Au niveau de la phase d'apprentissage, nous pouvons parfaitement mélanger les candidats issus de plusieurs couples protéine-ARN différents car nous cherchons à apprendre des poids valides pour tous les couples protéine-ARN. Par contre, pour tester l'efficacité d'une fonction de score, il est impératif de travailler sur un ensemble de candidats issus d'un seul et unique couple protéine-ARN. Nous avons donc mis en place une validation de type Leave-one-**pdb**-out, où **pdb** se réfère à la structure native issue de la PRIDB.

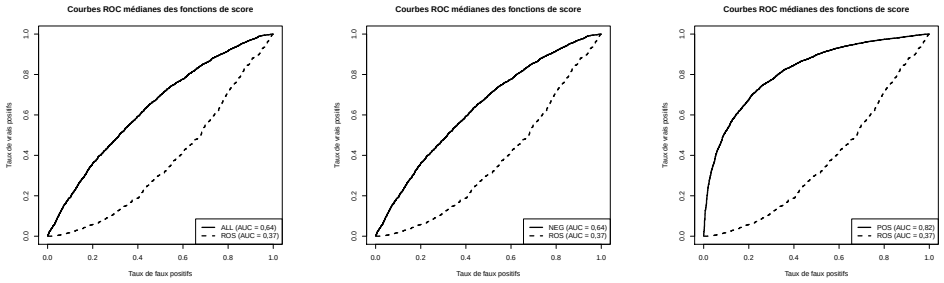
Sachant que le jeu de données de référence contient 120 structures natives, nous avons effectué 120 apprentissages à partir de $119 \times 10\,000$ candidats. Pour ne pas biaiser la phase d'apprentissage nous avons échantillonné les jeux d'apprentissage qui contiennent 30 presque natifs et 30 leurres par complexe, soit $3\,570$ candidats de chaque classe par jeu. Puis, chaque fonction apprise a été évaluée sur les $10\,000$ candidats associés à la structure native écartée pour le test.

Les résultats obtenus pour les trois fonctions de scores apprises, avec des contraintes différentes sur l'espace de recherche des poids, sont présentés sous deux angles : l'analyse "classique" des performances en ROC-AUC et l'analyse plus "biologique" des résultats en nous focalisant sur le gain de performance par rapport au cadre d'évaluation CAPRI.

4 Résultats [5]

Nous présentons les résultats obtenus pour les quatre fonctions de score suivantes : **POS**, la fonction de score à poids dans $[0\ ;\ 1]$; **NEG** à poids dans $[-1\ ;\ 0]$; **ALL** dans $[-1\ ;\ 1]$ et **ROS**, la fonction par défaut de RosettaDock. La figure 1 montre les résultats obtenus pour les 4 fonctions de score évaluées. On remarque que ROS ne permet pas de trier correctement des candidats ($AUC < 0,5$) alors que POS le permet. C'est la fonction de score la plus performante ($AUC = 0,80 \pm 0,02$). POS présente aussi une courbe ROC de forte pente à l'origine, indiquant que l'enrichissement en presque natifs dans les premiers résultats est importante. Les autres fonctions de scores ont des AUC nettement inférieures. La connaissance de l'interprétation physico-chimique des descripteurs qui incite à ne chercher que des poids positifs fait augmenter drastiquement la performance. Avec ALL, les minimima locaux de l'intervalle $[-1\ ;\ 0]$ sont si importants qu'ils restreignent les solutions à ce dernier intervalle, ne permettant pas d'obtenir les meilleures solutions dans $[0\ ;\ 1]$.

5. Les résultats complémentaires et annexes sont disponibles à la page suivante : `https://www.lri.fr/~adrienguilhot/EGC2014/`



(a) (b) (c)

FIG. 1 – *Courbes ROC médianes des 4 fonctions de score. En pointillés, la fonction de score ROS ; a) En trait plein, ALL ; b) En trait plein, NEG ; c) En trait plein, POS.*

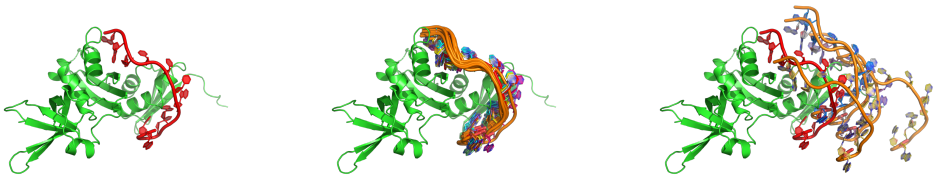
Le score d'enrichissement défini par Tsai et al. (2003) et noté SE, représente la proportion des candidats se trouvant à la fois dans les 10 % premiers candidats en score et en RMSD. C'est un critère d'évaluation courant pour les expériences à grande échelle de prédiction de structures 3D. Sa valeur varie de de 0 à 10 avec 10 pour une fonction de score extrayant parfaitement les 10 % meilleurs candidats en RMSD, notés $top10\%_{RMSD}$ et 1 pour une fonction de score triant aléatoirement. On considère habituellement qu'un score supérieur à 5 est preuve de performances très intéressantes dans ce domaine.

On observe des scores d'enrichissement supérieurs à 6 pour 27 structures natives avec POS, aucune avec les autres fonctions. Ils sont aussi supérieurs à 4 pour 54 structures natives avec POS et pour 4 avec les autres fonctions. Cela confirme la performance plus importante de POS. De façon similaire, lorsque la performance est dégradée, elle l'est moins avec POS qu'avec les autres. En effet, il y a 6 structures natives qui ont un score d'enrichissement inférieur à 1 avec POS, alors qu'il y en a 69 pour les autres fonctions. La comparaison des scores d'enrichissement montre qu'il n'y a pas de structure native pour laquelle POS a un score d'enrichissement inférieur de plus de 1 aux deux autres fonctions de score dédiées. Il y en a 6 pour lesquelles la fonction de score par défaut est meilleure de plus de 1. L'évaluation des fonctions de score sur le critère du score d'enrichissement corrobore les résultats obtenus en AUC et permet de parvenir à la même conclusion.

Le score d'enrichissement représente donc bien la capacité à obtenir des structures plausibles en premier. Cela est particulièrement visible sur les structures 3D. La figure 2 présente un exemple caractéristique. Les résultats obtenus grâce à POS sont de très bonne qualité et très resserrés dans l'espace par rapport aux leurres. L'épitope (zone d'interaction) est bien (mieux) caractérisé pour les deux partenaires après sélection des meilleurs candidats.

Pour avoir un critère d´évaluation proche de celui des expérimentalistes, on utilise le nombre de presque natifs du top$N$. C'est le nombre de presque natifs obtenus dans les N premiers candidats après tri. Pour nous placer dans l'objectif CAPRI, N est fixé à 10 candidats.

Dans 92 cas sur 120, POS prédit au moins un presque natif de plus que la valeur attendue sous l'hypothèse d'une distribution uniforme des candidats. Par contre, les fonctions apprises échouent en proposant une structure de moins dans 11 cas, contrairement aux 21 et 22 cas d'échec avec ALL et NEG. À nouveau, les fonctions ALL et NEG donnent des résultats simi-



(a) (b) (c)

FIG. 2 – *Structure 3D du complexe nusA-RNA de* Mycobacterium tuberculosis *(code PDB 2asb), avec la protéine en vert : (a) Structure native, avec l'ARN en rouge. (b) Structure native et top10 des candidats avec POS, avec l'ARN en orange. (c) Structure native et top5 des leurres, avec l'ARN en orange.*

laires. Elles ne permettent de dépasser la valeur attendue sous l'hypothèse d'une distribution uniforme que dans la moitié des cas. Il reste 9 structures natives pour lesquelles POS a au moins 1 presque natif de moins dans le top10 que ALL et NEG. Ces cas doivent être étudiés attentivement, de façon à valider les candidats et à interpréter cette différence d'un point de vue biologique. De façon intéressante, ces cas ne sont pas systématiquement plus mauvais avec ALL en terme de score d'enrichissement. Cela peut être dû à une qualité des candidats générés moins importante bien que correcte en terme de RMSD, ce dernier étant dépendant de la taille des molécules (et donc discutable).

5 Conclusion et perspectives

Dans cette étude, nous avons montré qu'une technique d'apprentissage classique, la régression logistique, réalisée à l'aide d'un algorithme génétique pour l'apprentissage des poids, se prête bien à un problème d'optimisation de poids pour les fonctions de score pour l'amarrage. Nous avons aussi mis en évidence que la connaissance des données physico-chimiques qui impose *a priori* des contraintes sur l'intervalle de recherche des poids, est ici essentielle car l'intervalle de recherche ne peut être déterminé de façon automatique. Par rapport aux trois autres, la fonction de score à poids positifs permet d'obtenir de très bonnes performances. La nature chimique fine des interactions pour chaque exemple pour lequel les performances de la fonction de score sont dégradées devra encore être analysée. Il semble aussi clair, étant donnés les résultats de cette étude et au vu des précédentes études réalisées sur ce même type de données, qu'une fonction de score linéaire des paramètres, bien que conforme au modèle physico-chimique sous-jacent, ne permet pas une classification optimale. Une approche par filtres collaboratifs pourrait, par exemple, donner de bien meilleurs résultats. En se plaçant dans un contexte réaliste d'évaluation CAPRI, la comparaison entre les structures natives du nombre attendu de presque natifs dans le top10 a montré que la nature des données ne permet pas de travailler sur des jeux de données équilibrés. Utiliser le critère ROC-AUC plutôt que le top10 pour définir la fonction à optimiser a permis d'obtenir des résultats comparables entre structures natives malgré la différence d'équilibre des jeux de données entre structures natives. Pour conclure, ces premiers travaux nous permettent d'envisager d'intégrer la fonction

de score apprise dans une nouvelle version de RosettaDock.

Ces travaux ont bénéficié d'un accès aux moyens de calcul du TGCC au travers de l'allocation de ressources t2013077065 attribuée par GENCI.

Références

Bernauer, J., J. Azé, J. Janin, et A. Poupon (2007). A new protein-protein docking scoring function based on interface residue properties. *Bioinformatics 23*(5), 555–562.

Bourquard, T., J. Bernauer, J. Azé, et A. Poupon (2011). A collaborative filtering approach for protein-protein docking scoring functions. *PLoS One 6*(4), e18541.

Gray, J. J., S. Moughon, C. Wang, O. Schueler-Furman, B. Kuhlman, C. A. Rohl, et D. Baker (2003). Protein-protein docking with simultaneous optimization of rigid-body displacement and side-chain conformations. *J Mol Biol 331*(1), 281–299.

Lensink, M. F. et S. J. Wodak (2010). Docking and scoring protein interactions: Capri 2009. *Proteins 78*(15), 3073–3084.

Lewis, B. A., R. R. Walia, M. Terribilini, J. Ferguson, C. Zheng, V. Honavar, et D. Dobbs (2011). Pridb: a protein-rna interface database. *Nucleic Acids Res 39*(Database issue), D277–D282.

Pons, C., A. Solernou, L. Perez-Cano, S. Grosdidier, et J. Fernandez-Recio (2010). Optimization of pydock for the new capri challenges: Docking of homology-based models, domain-domain assembly and protein-rna binding. *Proteins 78*(15), 3182–3188.

Sebag, M., J. Azé, et N. Lucas (2003). Impact studies and sensitivity analysis in medical data mining with roc-based genetic learning. In *Proceedings of the Third IEEE International Conference on Data Mining, ICDM 2003*, pp. 637–40.

Tsai, J., R. Bonneau, A. V. Morozov, B. Kuhlman, C. A. Rohl, et D. Baker (2003). An improved protein decoy set for testing energy functions for protein structure prediction. *Proteins 53*(1), 76–87.

Summary

Most of biological functions in the cell involve 3D interactions between proteins and RNA. Despite great progress in experimental structure solving, these techniques remain unsuitable for large-scale applications. *In silico* 3D protein-RNA interaction prediction is thus essential. The study presented here addresses this specific problem by focusing on the building of scoring functions so as to rank efficiently docking solutions generated by a protein-RNA docking program. The objective is to obtain a scoring function for a reference software suite initially designed for protein-protein docking: RosettaDock. RosettaDock requires an evaluation function that is a linear combination of physico-chemical measures. We thus set up a supervised machine learning approach using a genetic algorithm. Several scoring functions were built with different constraints on the output weights. The results obtained on the reference data set show the influence of the weight definition and the search interval.