# Projet en BIN702

Ces projets ne sont pas imposés. Ils sont présentés pour vous donner des idées pour lancer le vôtre. Au final, c'est à vous de définir la nature de votre projet, le but étant de développer une aptitude à résoudre des problématiques de niveau avancé.

Il est important de noter que la plupart des problèmes présentés ici ont été étudiés par des chercheurs — on ne s'attend pas à ce que vous amélioriez l'état actuel de la littérature en un seul semestre! Au contraire, on attend de vous que vous maîtrisiez la problématique et que vous proposiez vos idées et solutions (même si elles sont mauvaises). N'oubliez pas qu'en science en général, les mauvais résultats sont de toute façon des résultats.

## Types de projet

Pour la plupart des projets, vous pouvez étudier la problématique sous l'un des angles suivants:

- ✎ – **Théorie**. Vous pouvez tenter de résoudre un problème purement théorique. Ceci inclut, par exemple, de développer un nouvel algorithme atteignant une complexité améliorée, résoudre un problème ouvert en algorithmes pour la bio-informatique, créer de nouveaux modèles mathématiques, établir de nouvelles preuves formelles, etc.

  Ces projets sont souvent difficiles. Il est commun de restreindre la portée du problème, par exemple en étudiant des cas spécifiques, en restreignant les classes d'instances, ...

  *À mettre dans votre rapport.* Explication de la problématique, résultats partiels, solutions tentées, lesquelles sont intéressantes, pourquoi les autres ont échoué, qu'est-ce qu'on pourrait regarder avec plus de temps, ...

- </> – **Pratique**. Dans ce type de projets, vous devez programmer ou développer un algorithme exact ou une heuristique pour résoudre un certain problème et analyser les résultats. Il est également possible d'analyser un nouveau jeu de données en utilisant une chaîne d'outils existants pour les biologistes.

  *À mettre dans votre rapport.* Détaillez la problématique, votre approche haut-niveau, les défis d'implémentation principaux, votre cadre expérimental (quelles données evaluez-vous? devez-vous générer des données simulées? comment? quelle métrique d'évaluation utilisez-vous? pourquoi sont-elles pertinentes), donnez les résultats obtenus, puis discutez-les.

- ▶ – **Vulgarisation Scientifique**. L'idée de ces types de projets est de reprendre une question de recherche de pointe et de parvenir à l'expliquer à un public plus large. Par exemple, pourriez-vous expliquer l'algorithme d'Ukkonen à une personne étudiante de premier cycle?

  Ce type de projet ne demande pas nécessairement pas de créer de nouvelles connaissances, mais vous devez faire preuve de créativité dans votre format de vulgarisation (ex: produire une vidéo sur YouTube, un site Web interactif, une trace détaillée d'un exemple d'instance, etc.). Vous ne devez pas reprendre des algorithmes déjà vus en classe.

*À mettre dans votre rapport.* Expliquez la problématique et les solutions que vous vulgarisez d'un point de vue haut-niveau, détaillez votre approche de vulgarisation (choix du média, des analogies, etc.), faites une auto-critique de votre vulgarisation...

- – **Revue de littérature**. Vous pouvez résumer l'état-de-l'art sur un sujet qui vous intéresse en bioinformatique. Ceci demande de lire et surtout **comprendre** plusieurs articles scientifique, puis de les résumer de façon haut-niveau. Ceci se distingue de la vulgarisation, car on demande ici un survol substantiel état-de-l'art: il est attendu que vous parcouriez plusieurs articles et que vous les résumiez, en filtrant les éléments fondamentaux à rapporter.

  Si vous n'avez jamais lu d'article scientifique, commencez tôt, car ce n'est pas une tâche facile la première fois! Si vous en avez déjà lu, commencez tôt quand même.

  *À mettre dans votre rapport.* Détaillez d'abord la problématique. Ensuite, déterminez comment vous allez structurez votre état-de-l'art. Souvent, les résultats sont présentés en ordre chronologique, ou bien ils sont regroupés en catégories d'approche (par ex. approches combinatoires, approches machine-learning, approches X, ...). Présentez ensuite les articles les plus importants ainsi que les solutions qui y sont abordées.

## Autres étiquettes

- – **Recherche**. Le projet représente un défi considérable et est de niveau recherche. Des solutions intéressantes peuvent mener à une publication scientifique (mais vous n'avez pas à réussir). La difficulté du projet sera considérée dans l'évaluation du projet.
- – **Auto-didacte**. Le projet fait appel à beaucoup de notions qui ne sont pas vues dans le cours. Vous devrez apprendre par vous-mêmes. Cette difficulté sera considérée dans l'évaluation du projet.

# Quelques idées de projet

## 1 Evaluation de méthodes de clustering pour l'homologie

</>

- **Description:** Lorsque nous disposons d'un grand nombre de gènes provenant de plusieurs espèces, nous souhaitons répartir les gènes dans une famille d'homologues, c'est-à-dire dans des groupes de gènes qui ont un ancêtre commun et qui sont très similaires. L'idée est de calculer une distance entre chaque paire de gènes, puis d'appliquer une technique de clustering. Évaluez un certain nombre de méthodes de regroupement existantes, ou créez la vôtre.

  Pour mesurer la qualité de vos résultats, je recommande de simuler l'évolution de familles de gènes homologues et de tester les performances de chaque algorithme pour retrouver les gènes.

- **Lecture suggérée:**

  - Quelques algorithmes de clustering: `https://bmcbioinformatics.biomedcentral.com/articles/10.1186/s12859-014-0445-4`
  - Simulateurs de familles de gènes:
    - SimPhy: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4748750/`
    - Asymmetree: `https://www.mdpi.com/2674-113X/1/3/13/pdf`
    - ALF: `http://alfsim.org/`

## 2 Trouvez les meilleurs matches de gènes

</>

- **Description:** Supposons que vous avez deux génomes $G$ et $H$, chacun constitué d'un ensemble de gènes $g_1, \ldots, g_n$ et $h_1, \ldots, h_m$, respectivement. Pour chaque gène $g_i$ de $G$, on veut connaître le gène $h_j$ de $H$ dont le score d'alignement est maximum (ou, plus généralement, on voudrait les $k$ gènes de $H$ avec les $k$ meilleurs scores).

  La façon naïve est de calculer Smith-Waterman entre chaque paire $(g_i, h_j)$. C'est trop long. Essayez de trouver une façon plus rapide qui ne perd pas de précision, ou peu.

- **Lecture suggérée:**

  - Practical Construction of $k-$Nearest Neighbor Graphs in Metric Spaces: `https://link.springer.com/chapter/10.1007/11764298_8`
  - Quickly finding orthologs as reciprocal best hits with BLAT, LAST and UBLAST: how much do we miss? `https://pubmed.ncbi.nlm.nih.gov/25013894/`

# 3 Transformation d'une séquence par duplications

- **Description:** Une duplication est une opération qui copie une sous-chaîne d'une séquence et qui insère la copie à un autre endroit. Une *duplication en tandem* c'est une duplication qui insère la copie immédiatement après la sous-chaîne copiée. Par exemple $ABC\underline{CXY}XXZ \longrightarrow ABC\underline{CXYCXY}XXZ$ fait une duplication en tandem de $CXY$.

  Étant donné un ensemble de génomes et $S$ et $T$, quel est le nombre minimum de duplications en tandem pour transformer $S$ en $T$. On a récemment montré que ce problème est NP-complet. Par contre, la complexité reste ouverte si l'alphabet de $S$ et $T$ est binaire. Que se passe-t-il si la taille des mots dupliqués est limitée, par exemple si nous ne pouvons avoir que des duplications de taille 2? Pourrions-nous développer un algorithme efficace? Sinon, peut-on développer un algorithme avec un ratio d'approximation garanti?

- **Lecture suggérée:**

  - Résultat de NP-complétude pour la distance tandom dup: `https://drops.dagstuhl.de/opus/volltexte/2020/11876/pdf/LIPIcs-STACS-2020-15.pdf`.

# 4 Programmez votre propre BLAST

- **Description:** Étant donné une petite chaîne $P$ et une grande chaîne $T$, trouvez les sous-chaînes de $T$ où le score d'alignement global avec $P$ est maximal. Vous pouvez également remplacer "global" par "local" ici, c'est vous qui voyez. Vous devez développer vos propres idées d'heuristiques et comparer votre vitesse et votre précision avec les approches qui existent actuellement.

  Vous pourriez également faire une revue de littérature sur toutes les variantes de BLAST qui existent (PSI-BLAST, BLAT, ...).

- **Lecture suggérée:**

  - Tout ce que vous trouverez sur BLAST: `https://scholar.google.com/scholar?hl=en&as_sdt=0%2C5&q=blast+alignment&btnG=`

# 5 L'algorithme des quatre russes

- **Description:** Il est possible d'implémenter un alignement global en $O(n^2/\log(n))$ (si les deux séquences données sont de taille $n$ chacune). Ceci est faisable avec l'algorithme des quatre russes. L'idée est de séparer le tableau de programmation dynamique en blocs de

dimension $t \times t$ et de les résoudre indépendamment, puis de connecter leur solution. Ce projet vous demandera d'abord de comprendre les détails de l'approche.

Si vous choisissez un projet de type </> , l'idée est de le mettre en œuvre et de le comparer aux méthodes traditionnelles. Si vous choisissez un projet de type , votre vidéo doit également inclure une explication de la programmation dynamique.

- **Lecture suggérée:**
  - Algorithms on Strings, Trees, and Sequences de Dan Gusfield, Chapitre 12
  - `https://pdfs.semanticscholar.org/3e94/9581ade25f54d1b5a75935f9189d3c63c167pdf`

## 6 Aligneurs de séquences à lectures courtes

</>

- **Description:** Une tâche commune en séquençage est de prendre un ensemble de (beaucoup de) lectures, i.e. de petits segments, et de retrouver leur emplacement le plus probable sur un génome. Il existe beaucoup d'outils pour ce faire.

  Vous pouvez préparer une analyse documentaire ou une revue de littérature des outils qui ont été utilisés pour cette tâche, comme `Bowtie` ou `segehmel`. Essayez de choisir des outils qui utilisent différentes approches algorithmiques et/ou heuristiques.

  Si vous souhaitez vous attaquer à un problème plus concret, vous pouvez comparer différents outils avec des jeux de données, ou même inventer votre propre outil.

- **Lecture suggérée:**
  - Segemehl : `https://academic.oup.com/bioinformatics/article/30/13/1837/2422281#84882600`
  - Bowtie : `https://genomebiology.biomedcentral.com/articles/10.1186/gb-2009-10-3-r25`

## 7 La distance d'édition (Levenshtein) en pratique

- **Description:** La distance de Levenshtein est largement appliquée dans l'alignement de séquences. Une question intéressante concernant ce problème est de savoir dans quelles applications **biologiques** il est réaliste de l'utiliser. Cette distance ne modélise que les insertions, suppressions et mutations, alors que d'autres événements peuvent survenir (par exemple des inversions ou des insertions en bloc).

  Les questions que vous pourriez expérimenter dans ce projet concerneraient les différentes approches que les théoriciens ont utilisées pour ce problème. Par exemple : Quels sont les différents types de distances qui ont été proposés? Quelles variantes de ce problème ont

été étudiées? Est-il judicieux de considérer une entrée comprimée? Quelles structures de données ont été inventées pour effectuer une analyse rapide de ce problème?

Ce projet est exploratoire et vous demandera de poser vos propres questions.

- **Lecture suggérée:**
  - Analyse théorique des algorithmes de distance d'édition : Une perspective appliquée. `https://www.mdpi.com/1999-4893/15/7/242/pdf?version=1657632032`
  - Pour un résumé plus compréhensible et l'inspiration de questions, consultez les diapositives de l'auteur : `https://www.pangenome.eu/wp-content/uploads/2020/11/tasba-talk-11-16-2020.pdf` .

    `https://www.pangenome.eu/wp-content/uploads/2020/11/tasba-talk-11-16-2020.pdf` .
  - Edit distance with block operations. `https://drops.dagstuhl.de/opus/volltexte/2018/9496/pdf/LIPIcs-ESA-2018-33.pdf`
  - A new String Edit Distance and Operations. `https://www.mdpi.com/1999-4893/15/7/242/pdf?version=1657632032`
  - Edlib: a C and C++ library for fast, exact sequence alignment using edit distance. `https://academic.oup.com/bioinformatics/article/33/9/1394/2964763`

# 8 Reconciliation avec les duplications segmentales

- **Description:** La *réconciliation* phylogénétique est utilisée pour expliquer les différences entre l'évolution des gènes et celle des espèces. Développez un bon algorithme ( un algorithme d'approximation ou une heuristique qui donne de bons résultats) pour réconcilier plusieurs arbres de gènes avec un arbre d'espèces, en considérant qu'un événement de duplication peut inclure plusieurs gènes. Comprendre ce problème est déjà difficile, et trouver une nouvelle solution est un défi encore plus grand. Cette difficulté sera prise en compte dans votre note.
- **Lecture suggérée:**
  - Ce document explique la réconciliation de façon relativement simple: `https://www.iro.umontreal.ca/~mabrouk/Publications/chapter_Moret_2019.pdf`
  - Ensuite, le problème de réconciliation segmentale est introduit ici: `https://arxiv.org/abs/1806.03988`

# 9 Distance d'édition en temps quasi-linéaire

- **Description:** Supposons que $S_1$ soit une chaîne de bits uniformément aléatoire de longueur $n$ et que $S_2$ soit la chaîne de bits obtenue en appliquant des substitutions, des insertions et des suppressions à chaque bit de $S_1$ avec une certaine probabilité. La distance d'édition entre $S_1$ et $S_2$ peut être calculée en $O(n \ln n)$ temps avec une probabilité élevée sous certaines conditions.

  L'algorithme de base utilise la programmation dynamique classique et fournit une justification théorique pour les heuristiques d'alignement telles que BLAST, FASTA et MAFFT. Il s'agit d'un résultat récent, et aucune implémentation n'existe pour le moment. Fournissez une implémentation de cet algorithme et expliquez sa contribution théorique et son impact.

- **Lecture suggérée:**
  - https://drops.dagstuhl.de/opus/volltexte/2020/12806/pdf/LIPIcs-WABI-2020-17.pdf

# 10 Meilleur aligneur multiple

- **Description:** Un des objectifs principaux de l'alignement multiple est d'établir une correspondance entre les positions de plusieurs séquence afin de détecter des signaux de conservation.

  Faites une étude comparative de logiciels d'alignement multiple (Clustal, MUSCLE, Mafft, ProbCons, ...) et évaluez leur capacité à retrouver les positions conservées. Evaluez le temps et surtout, la précision de chaque logiciel.

  Une option est de simuler l'évolution de séquences à l'aide de logiciels connus (ex: INDELible) et trouver une façon de mesurer la précision de la sortie des logiciels sur les séquences générées. Si le temps le permet, effectuez aussi des tests sur des données réelles (voir e.g. Ensembl, GenBank, ...).

- **Lecture suggérée:**
  - Alignment Methods: Strategies, Challenges, Benchmarking and Comparative Overview. `https://link.springer.com/protocol/10.1007/978-1-61779-582-4_7`
  - A Comprehensive Benchmark Study of Multiple Sequence Alignment Methods: Current Challenges and Future Perspectives. `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3069049/`

## 11 Arbres de suffixes en temps $O(n)$

- **Description:** L'algorithme d'Ukkonen et l'algorithme de Farach-Colton construisent un arbre de suffixes en temps $O(n)$. Toutefois, il ne sont pas faciles à comprendre, ni à implémenter.

  Dans ce projet, vous pouvez soit expliquer un de ces deux algorithmes de façon compréhensible à travers une vulgarisation, ou bien en implémenter un et l'évaluer par rapport à l'approche naïve vue en classe (ou par rapport aux implémentations existantes).

- **Lecture suggérée:** voir section 3 sur la page du cours.

## 12 Le motif planté

- **Description:** Développez et implémentez un nouvel algorithme pour résoudre le problème du *motif planté*. Dans ce problème, nous avons $n$ séquences $s_1, s_2, \ldots, s_n$ et deux entiers $l$ et $d$. Nous voulons trouver une séquence $X$ telle que $|X| = l$ et telle que chaque chaîne $s_i$ contienne une sous-chaîne $X'$ qui soit proche de $X$ avec $d$ de modifications. Le défi ici est sa performance. Essayez de résoudre des instances avec $l = 26, d = 11$. Vous ne devez pas nécessairement réussir.

- **Lecture suggérée:**

  - Le wiki officiel du problème : `https://en.wikipedia.org/wiki/Planted_motif_search`
  - Algorithmes séquentiels et parallèles efficaces pour la recherche de motifs plantés `https://arxiv.org/abs/1307.0571`

## 13 Qu'est-ce qui n'est pas un motif?

- **Description:** Vous aimez peut-être l'apprentissage machine. Supposons que vous vouliez inventer un classificateur binaire qui, à partir d'une séquence donnée, puisse vous dire si elle représente un "motif" ou non. On peut définir un motif comme une séquence qui a une structure similaire à d'autres séquences connues qui ont un rôle biologique important.

  Il ne suffit pas d'entraîner un modèle avec des exemples de ce qui est un motif, il faut aussi avoir quelques mauvais exemples, c'est-à-dire ce qui n'est PAS un motif. Un des problèmes est qu'à peu près toutes les séquences ne sont pas des motifs, alors qu'on ne veut pas entraîner sur toutes les séquence. Essayez de trouver un moyen d'apprendre à un classificateur binaire à faire la distinction entre un motif et quelque chose qui n'en est pas un.

- **Lecture suggérée:**
  - A survey on deep learning in DNA or RNA motif mining. `https://academic.oup.com/bib/article/22/4/bbaa229/5916939`.
  - Efficient large-scale machine learning algorithms for genomic sequences. `https://escholarship.org/uc/item/7jz4h3rd`.

# 14 La recherche de motifs en pratique

- **Type de projet:** </>
- **Description:** Le problème du motif planté présenté ci-haut est en fait l'un des nombreux modèles que les informaticiens ont inventés pour résoudre le problème biologique réel. Il existe plusieurs modèles plus sophistiqués qui permettent de découvrir des motifs avec des fonctions différentes, par exemple des paires de motifs appelés *motifs coopératifs* qu'on retrouve souvent ensemble sur des génomes.

  Pourriez-vous proposer une formulation qui ajoute des contraintes supplémentaire à la recherche de motifs afin de la rapprocher de ce que recherchent les biologistes? Regardez quelles sont les limites actuelles. Ou, si vous optez pour une approche plus simple, pouvez-vous identifier le contexte biologique dans lequel certains des meilleurs algorithmes ne parviennent pas à identifier les motifs? Quand une approche est-elle préférable à une autre?

- **Lecture suggérée:**
  - Les limites de la découverte de novo des motifs de l'ADN. `https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0047836`.
  - Un algorithme efficace pour l'identification de motifs structurés dans les séquences promotrices d'ADN. `https://ieeexplore-ieee-org.ezproxy.usherbrooke.ca/stamp/stamp.jsp?tp=&arnumber=1631994&tag=1`.
  - Algorithmes rapides et pratiques pour la recherche de $(l, d)-$motifs plantés. `https://ieeexplore.ieee.org/document/4359890`.
  - Algorithmes d'extraction de motifs structurés à l'aide d'un arbre de suffixes avec une application à l'identification du consensus des promoteurs et des sites de régulation. `https://www.liebertpub.com/doi/epdf/10.1089/106652700750050826`.

# 15 Que pouvez-vous trouver avec des modèles de Markov cachés

- **Description:** Un des buts de l'annotation de génomes est de classifier les nucléotides. Ceux-ci peuvent appartenir à plusieurs catégories: ils peuvent faire de gènes, être dans des régions intergéniques, servir de promoteurs, faire partie d'introns, d'exons, d'ilôts CpG, et bien d'autres.

  On peut entraîner des segments de nature connue afin d'en classifier d'autres, notamment avec les HMM. Dans ce projet, vous pouvez faire une revue de littérature sur les capacités des HMM, ou encore implémenter une approche d'annotation vous-mêmes.

- **Lecture suggérée:**

  - Pour en savoir plus sur ce problème, lisez le chapitre 9 *Finding Signals in DNA Sequences* de ce livre : `https://link.springer.com/book/10.1007/978-3-540-78506-4`.
  - Pour en savoir plus sur le contexte biologique du problème, voici le wiki pour le *CG-islands* (également connu sous le nom de *CpG-sites*). `https://en.wikipedia.org/wiki/CpG_site`.
  - L'algorithme principal pour ce problème : *Algorithme de Viterbi*. `https://www.audiolabs-erlangen.de/resources/MIR/FMP/C5/C5S3_Viterbi.html`.
  - Plus sur les HMM: `https://www.ebi.ac.uk/training/online/courses/pfam-creating-protei what-are-profile-hidden-markov-models-hmms/#:~:text=Profile%20HMMs%20are% 20probabilistic%20models,the%20alignment%2C%20see%20Figure%202..`
  - Détection de souches de VIH avec les HMM. `https://bmcbioinformatics.biomedcentral.com/articles/10.1186/1471-2105-7-265`
  - Recherche de zones de recombinatison. `https://link.springer.com/chapter/10.1007/11557067_12`

# 16 Programmation dynamique automatique pour prédire le repliement de l'ARN

- **Description:** La prédiction de la structure secondaire de l'ARN est une application classique de la programmation dynamique en bioinformatique. Elle reste néanmoins une tâche difficile lorsque des pseudo-noeuds apparaissent. Dans ce projet, vous explorerez l'idée de générer automatiquement des règles de programmation dynamique pour surmonter les difficultés introduites par l'apparition de pseudo-noeuds.

  Ce projet exige que vous compreniez les notions de haut niveau de certains concepts de conception algorithmique avancés tels que *tree-width* et *fixed-parameter tractability*. Les

auteurs fournissent un pipeline qui met en oeuvre cette idée. Une fois que vous aurez compris ces concepts, essayez de penser à un scénario biologique dans lequel cet algorithme pourrait être utile.

- **Lecture suggérée:**
  - Pour savoir ce qu'est un pseudo-noeud. `https://en.wikipedia.org/wiki/Pseudoknot`.
  - Le document principal qui décrit la méthodologie. `https://drops.dagstuhl.de/opus/volltexte/2022/17041/pdf/LIPIcs-WABI-2022-7.pdf`.

## 17 Analyse de l'expression des gènes

- **Description:** Faites une revue de la littérature sur l'analyse de l'expression des gènes à partir de données RNA-seq. Vous devez d'abord apprendre de manière autodidacte les problèmes associés, puis expliquer ce que font les méthodes traditionnelles.
- **Lecture suggérée:**
  - Ceci est un bon point de départ : `https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0190152`
  - Introduction à l'ARN séquentiel. `https://chagall.med.cornell.edu/RNASEQcourse/Intro2RNAseq.pdf`

## 18 Reconstruction phylogénétique de tumeurs cancéreuses

- **Description:** Le cancer est un processus évolutif: il démarre d'une cellule maligne qui subit des mutations et se reproduit, et ces cellules se reproduisent à leur tour.

  Il y a plusieurs façons de représenter l'évolution d'une tumeur. Si on suppose que l'on a séquencé plusieurs cellules malignes, on peut reconstruire un arbre qui représente les liens de proximité entre ces cellules, avec la racine qui serait l'origine de la pathologie. Une autre façon est de construire l'arbre des mutations, qui donne un ordre de précédence d'apparition de mutations.

  Vous pouvez faire une revue de littérature des approches existantes, ou bien vulgariser un de ces approches. Vous pouvez même développer votre propre méthode (ce qui ne sera pas facile).
- **Lecture suggérée:**
  - The evolution of tumour phylogenetics: principles and practice. `https://www.nature.com/articles/nrg.2016.170`

- Parsimonious Clone Tree Integration in cancer. https://almob.biomedcentral.com/articles/10.1186/s13015-022-00209-9
- Distinguishing linear and branched evolution given single-cell DNA sequencing data of tumors. https://almob.biomedcentral.com/articles/10.1186/s13015-021-00194-5

## 19 Réarrangements massifs au sein du cancer

- **Description:** Les cellules cancéreuses subissent des événement de *chromothripsie*, un réarrangement qu'on voit rarement ailleurs. Il constitue en une mutation qui casse le génome en deux ou plusieurs morceaux, puis réunit les fragments résultants dans un ordre différent.

  Déduire le nombre minimum de différents types de ces réarrangements est bien sûr une tâche difficile sur le plan informatique. Le calcul de l'explication la plus simple pour les données devient un défi lorsque des réarrangements complexes sont impliqués.

  Dans ce projet, vous devriez apprendre les modèles actuels pour modéliser les événements qui affectent le cancer. Vous pouvez les résumer, ou bien développer vos propres approches.

- **Lecture suggérée:**
  - Modèles et algorithmes pour l'histoire de réarrangements chez les cellules cancéreuses: https://drops.dagstuhl.de/opus/volltexte/2022/17055/pdf/LIPIcs-WABI-2022-21.pdf.
  - Un point de vue appliqué sur la question: https://www.cell.com/fulltext/S0092-8674(13)00212-2.
  - Un point de vue très théorique sur la question: https://www.mdpi.com/1999-4893/14/6/169.

## 20 Prédiction de structure 3D d'ARN

- **Description:** nous avons brièvement mentionné que l'ARN est une molécule qui peut se replier de plusieurs façons dans l'espace (contrairement à l'ADN qui est en double-hélice). Il existe plusieurs approches pour prédire la structure 3D d'un ARN selon sa séquence. Faites une revue de littérature ou une vulgarisation du sujet.

- **Lecture suggérée:**
  - Une introduction détaillée sur le sujet. https://link.springer.com/content/pdf/10.1007/978-3-642-25740-7.pdf.
  - Prédiction de structure 3D d'ARN avec deep learning: https://www.biorxiv.org/content/10.1101/2021.08.30.458226v1

## Quelques articles supplémentaires qui pourraient vous inspirer.

- STAR: logiciel de mapping d'ARN sur l'ADN: `https://academic.oup.com/bioinformatics/article/29/1/15/272537?login=false`
- Un algorithme probabiliste pour l'alignement de séquences `https://drops.dagstuhl.de/opus/volltexte/2020/12806/`
- ProbCons: alignement multiple probabiliste. `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC546535/`
- SPAdes, un des assembleurs les plus reconnus pour le séquençage. `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3342519/`.
- Un méthode pour comparer des séquences sans alignement. `https://academic.oup.com/bioinformatics/article/30/14/1991/2391234?login=true` Une autre méthode du même style `https://www.liebertpub.com/doi/pdfplus/10.1089/cmb.2009.0198`
- et quelques milliers d'autres...