Домашнее задание №3: «Рибосома и иерархическая кластеризация»

Дедлайн 1 (20 баллов): 5 марта, 23:59
Дедлайн 2 (10 баллов): 12 марта, 23:59

Домашнее задание нужно написать на Python и сдать в виде одного файла. Правило именования файла: `name_surname_3.py`. Например, если вас зовут Иван Петров, то имя файла должно быть: `ivan_petrov_3.py`.

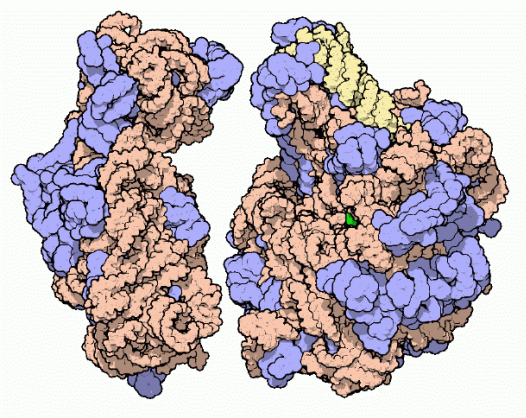

Рис. 1: Источник: `http://www.rcsb.org/pdb/101/motm.do?momID=10`

Рибосома — это белок, который участвует в синтезе всех других белков в клетках живого организма. Рибосома состоит из двух субъединиц: малой и большой. Для целей задания удобно думать о каждой субъединице как о строке (последовательности) в алфавите $\{A, C, G, U\}$. Например, большая субъединица рибосомы домашней мыши выглядит так:

```
CCUUCCCGCGAUGUUGGGAGAACCUCCGCGUAGUGUUCCCCGGCCGAGGUCCCGCCCCCCGGGACCGACGGUUUUCCGCG
CGGCGCCUCGCCUCGGCCGGGCGCCUAGCAGCCCGACUUAGAACUGGUGCGGAACCAGAGGAAUCCGACUGUUUAAUUAA
AACAAAGCAUCGCGAAGGCCCGCGGCGGGUGUUGACGCGAUGUGAUUUCUGCCCAGUGCUCUGAAUGUCAAAGUGAAGAA
AUUCAAUGAAGCGCGGGUAAACGGCGGGAGUAACUAUGACUCUCUUAAGGUAGCCAAAUGCCUCGUCAUCUAAUUAGUGA
CGCGCAUGAAUGGAUGAACGAGAUUCCCACUGUCCCUACCUACUAUCCAGCGAAACCACAGCCAAGGGAACGGGCUUGGC
GGAAUCAGCGGGGAAAGAAGACCCUGUUGAGCUUGACUCUAGUCUGGCACGGUGAAGAGACAUGAGAGGUGUAGAAUAAG
UGGGAGGCCCCCGGCGCCCGGCCCCGUCCUCGCGUCGGGGUCGGGGCACGCCGGCCUCGCGGGCCGCCGGUGAAAUACCA
CUACUCUCAUCGUUUUUUCACUGACCCGGUGAGGCGGGGGGGCGAGCCCCGAGGGGCUCUCGCUUCUGGCGCCAAGCGUC
CGUCCCGCGCGUGCGGGCGGGCGCGACCCGCUCCGGGGACAGUGCCAGGUGGGGAGUUUGACUGGGGCGGUACACCUGUC
AAACGGUAACGCAGGUGUCCUAAGGCGAGCUCAGGGAGGACAGAAACCUCCCGUGGAGCAGAAGGGCAAAAGCUCGCUUG
AUCUUGAUUUUCAGUACGAAUACAGACCGUGAAAGCGGGGCCUCACGAUCCUUCUGACCUUUUGGGUUUUAAGCAGGAGG
UGUCAGAAAAGUUACCACAGGGAUAACUGGCUUGUGGCGGCCAAGCGUUCAUAGCGACGUCGCUUUUUGAUCCUUCGAUG
UCGGCUCUUCCUAUCAUUGUGAAGCAGAAUUC
```

В этом задании необходимо воспользоваться иерархическои кластеризациеи последовательностей большои субъединицы рибосомы для восстановления эволюционного родства между биологическими видами.

По ссылке[1] находится файл в формате FASTA, содержащий последовательности большой субъединицы рибосомы для шести видов:

- человек,
- комнатная муха,
- серая крыса,

[1] `https://gist.github.com/superbobry/56c851a0b2a36861c341`

- собака,
- чернопятнистая лягушка,
- рыба «дамский чулок».

1 Формат FASTA представляет из себя последовательность записей вида

```
> идентификатор последовательности
строка и
её продолжение
> ещё один идентификатор
новая
длинная
строка
```

Реализуйте функцию `read_fasta`, которая принимает путь к FASTA файлу и возвращает список пар, где первый элемент пары — идентификатор последовательности, а второй — сама последовательность без символов переноса строки.

```
def read_fasta(path):
    # ...
    return records
```

2 В качестве метрики расстояния между последовательностями будем использовать расстояние Левенштейна[2]. Реализуйте функцию `levenshtein`, вычисляющую расстояние Левенштейна для двух строк, не обязательно равной длины.

3 Существуют и другие метрики расстояния между парой строк, например, расстояние Хемминга или расстояние Жаккарда. Расстояние Жаккарда позволяет оценить степень непохожести двух множеств. Для работы со строками его удобно обобщить на случай мультимножеств. Зафиксируем некоторое n и для каждой строки построим мультимножество всех подстрок длины n. Обозначим полученные мультимножества за A и B, тогда расстояние Жаккарда можно посчитать как:

$$J(A, B) = 1 - \frac{|A \cap B|}{|A \cup B|}$$

Реализуйте функцию `jaccard`, вычисляющую расстояние Жаккарда для двух данных строк:

```
def jaccard(s, t, n):
    # ...
```

4 Реализуйте функцию `lance_williams`, кластеризующую входные данные с помощью алгоритма Ланса-Вильямса. Функция должна принимать `X` — вектор объектов и `dist` — функцию расстояния между одноэлементными кластерами. Для вычисления расстояния между объединенными кластерами следует использовать групповое среднее.

Результатом работы является матрица `Z` размерности `(n - 1, 3)`, где `n = len(X)`. Матрица организована следующим образом.

[2] `http://en.wikipedia.org/wiki/Levenshtein_distance`

- Пусть i — номер итерации.
- В ячейках `Z[i, 0]` и `Z[i, 1]` находятся индексы кластеров, из которых был получен новый кластер.
- Расстояние между кластерами `Z[i, 0]` и `Z[i, 1]` находится в ячейке `Z[i, 2]`.
- Полученный на i-й итерации кластер имеет индекс `n + i`.

На Python функцию можно записать так:

```
def lance_williams(X, dist):
    n = len(X)
    Z = np.zeros((n - 1, 3))
    # ...
    return Z
```

Нарисовать дендрограмму по матрице `Z` можно с помощью функции `dendrogram` из пакета SciPy:

```
def show_dendrogram(Z, **kwargs):
    from scipy.cluster.hierarchy import dendrogram, from_mlab_linkage
    from matplotlib import pyplot as plt
    dendrogram(from_mlab_linkage(Z), **kwargs)
    plt.show()
```

Функция `dendrogram` принимает необязательный аргумент `labels` — список строковых меток для листьев дендрограммы.

5 Примените иерархическую кластеризацию к данным `ribosome.fasta`. Попробуйте обе реализованные метрики расстояния: расстояние Левенштейна и расстояние Жаккарда с разными параметрами `n = 1, 8, 16`. Ответьте на следующие вопросы:

1. Зависят ли результаты кластеризации от используемой метрики?
2. Насколько устойчивы результаты кластеризации при разных значениях параметра `n`?
3. Какую из метрик разумнее использовать для сравнения последовательностей?
4. К каким видам вы бы отнесли последовательности с номерами 2613, 5113 и 205334? Почему?
5. Приведите дендрограмму кластеризации, которая кажется вам наиболее правдоподобной. Какое дерево эволюционного родства ей соответсвует?