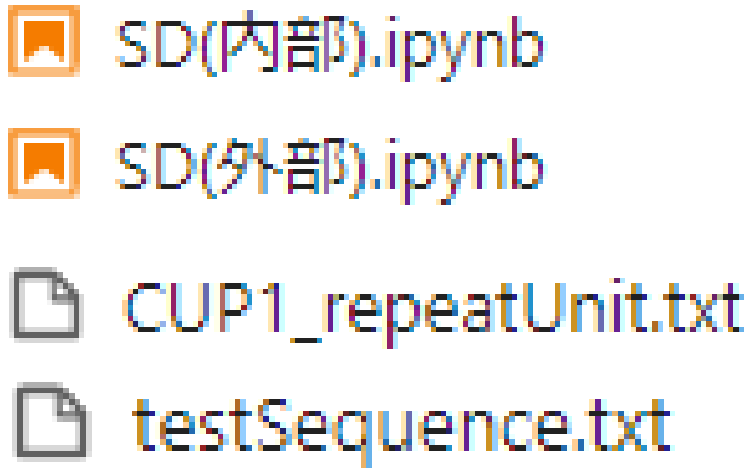
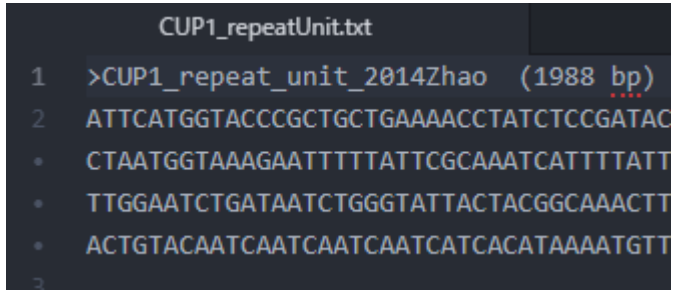
DNA_Sequence_Detector 使用法

Shintaro Miyazaki



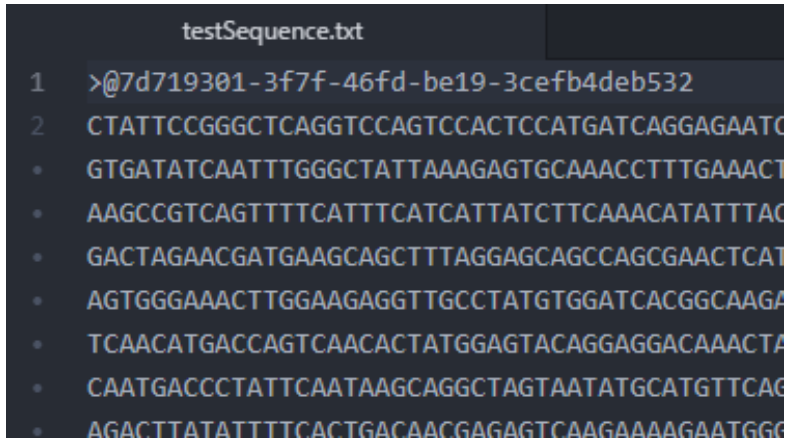
①4つのファイルを同じフォルダ内に置きます。
・SD(内部)
・SD(外部)
・リピートユニットなどの既知の配列のテキストファイル
・既知の配列の存在を知りたい配列のテキストファイル



リピートユニットなどの既知の配列のテキストファイルの中身は

>配列名(改行)
配列(改行)

となるようにしてください。同様に、既知の配列の存在を知りたい配列のテキストファイルの中身は



>配列名1(改行)
配列1(改行)
>配列名2(改行)
配列2(改行)
・・・

となるようにしてください。

インポート

```
import numpy as np
from tqdm import tqdm
import matplotlib.pyplot as plt
import gc
from scipy import signal
import math
from numba import jit, i2
import datetime
```

②SD(外部)をjupyter notebookで開いてください

③このセルを実行し、必要なものがimportできる環境か確認してください。importできないものがあった場合、importできなかったものをインストールしておいてください

テキストファイルの読み込み

```
repeatUnitFileName='CUP1_repeatUnit.txt'
f = open(repeatUnitFileName, 'r')
repeatUnitList = f.readlines()
f.close()
```

④このセルのこの部分に、リピートユニットなどの既知の配列のテキストファイルの名前を入力してください

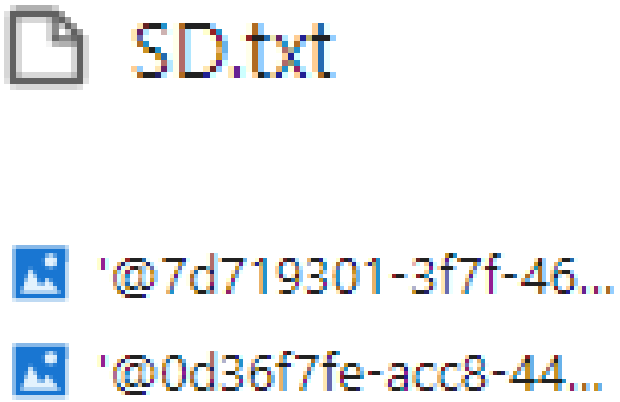
```
sequenceFileName='testSequence.txt'
f = open(sequenceFileName, 'r')
sequenceList = f.readlines()
f.close()
```

⑤このセルのこの部分に、既知の配列の存在を知りたい配列のテキストファイルの名前を入力してください

```
#パラメータ決定
NF1=0
NF2=10
L1 = np.round(len(repeatUnit)/20).astype(np.int)
density = (2*NF2+1+1+1)/L1
L1_R = np.round(len(repeatUnit)/4).astype(np.int)
L2 = np.round(len(repeatUnit)/4).astype(np.int)
minLength = np.round(len(repeatUnit)/2).astype(np.int)
Ratio=1.5
N_fourier=10
```

⑥parameter変更ができるセルです。この画像が初期値になっています。

⑦jupyter notebookのRun All Cellsを実行してください。一番下のセルの下に、進行状況が表示されます。



⑧出力には2種類あり、
テキストファイル1つ
と
既知の配列の存在を知りたい配列のテキストファイル中に含まれていた配列の個数の画像です。

```
DNA_Sequence_Detector

開始時刻　2020-05-04 06:54:15

既知の配列のファイルの名前　CUP1_repeatUnit.txt
既知の配列の名前　CUP1_repeat_unit_2014Zhao　(1988 bp)
既知の配列の長さ　1988

Sequenceファイルの名前　testSequence.txt
Sequenceファイル中のSequenceの個数　2

NF1=0
NF2=10
L1=99
density=0.23232323232323232
L1_R=497
L2=497
minLength=994
Ratio=1.5
N_fourier=10

元の配列の名前»配列の長さ——»既知配列開始位置——»既知配列終了位置——»既知配列の個数(推定)»既知配列の個数(フーリエ変換)»第n候補——»既知配
列の長さ(フーリエ変換)»領域判定値
'@7d719301-3f7f-46fd-be19-3cefb4deb532»36955——»3329——»7110——»2——»2——»1——»1890——»2.6168065683201966
'@0d36f7fe-acc8-4496-aae7-8a91b49d6c39»48330——»33781——»39557——»3——»3——»1——»1925——»2.701132457097973

終了時刻　2020-05-04 06:55:04
```

テキストファイルはこのようになっています。この部分をExcelに張り付けると良いでしょう。

元の配列の名前	配列の長さ	既知配列開始位置	既知配列終了位置	既知配列の個数(推定)	既知配列の個数(フーリエ変換)	第n候補	既知配列の長さ(フーリエ変換)	領域判定値
'@7d719301-3f7f-4	36955	3329	7110	2	2	1	1890	2.61680657
'@0d36f7fe-acc8-4	48330	33781	39557	3	3	1	1925	2.70113246

既知配列の個数(推定)：
検出された領域の長さを単に既知配列の長さで割って算出しています。検出された領域中の既知配列の個数の目安となります。

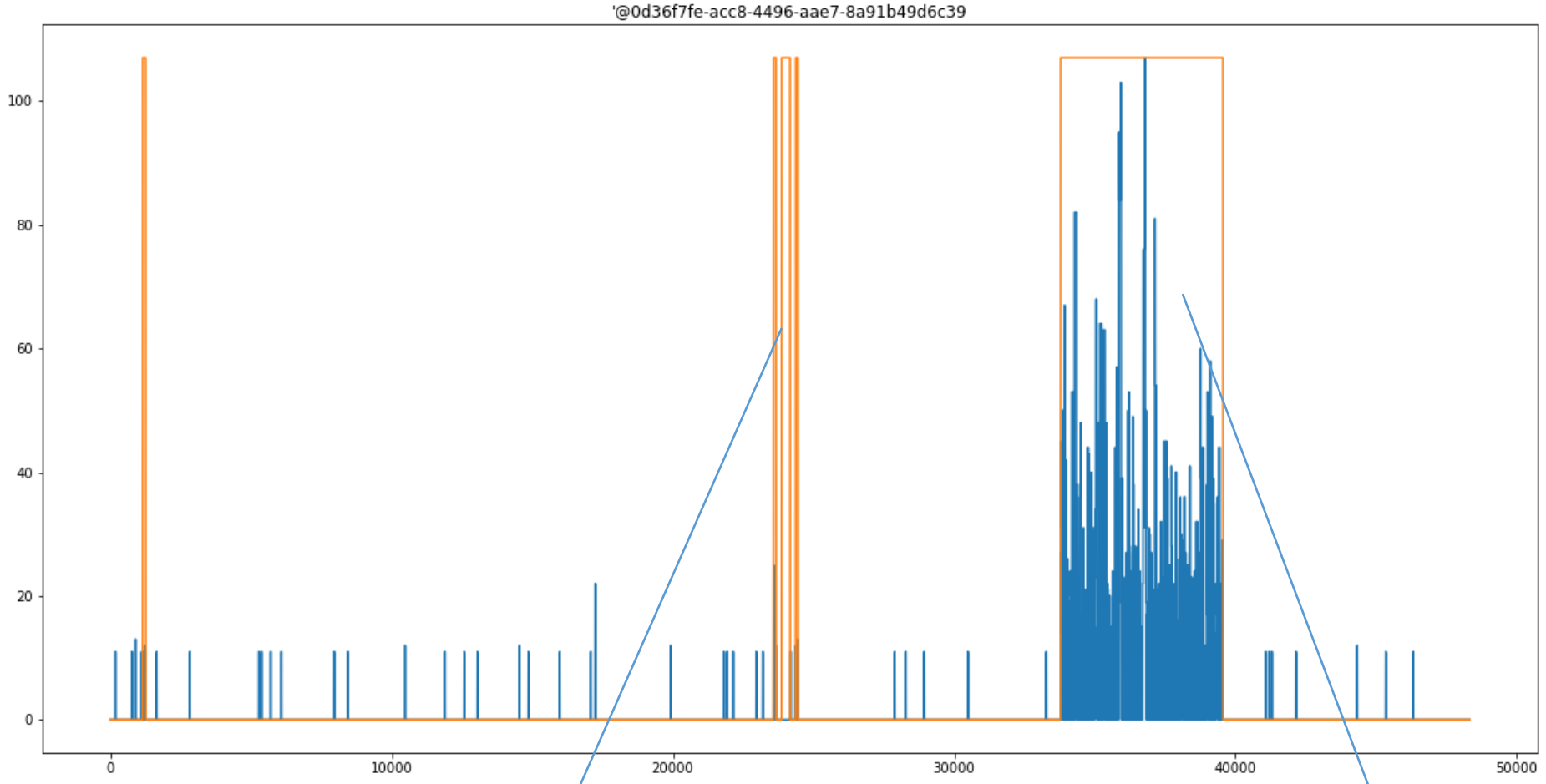
既知配列の個数(フーリエ変換)：
検出された領域中の既知配列の個数を、離散フーリエ変換を用いて求めています。

第n候補：
既知配列の個数(フーリエ変換)がいくつめの候補であるかを表します。数字が大きくなると、フーリエ変換での既知配列の個数のカウントが困難であったことを表します。

既知配列の長さ(フーリエ変換)：
検出された領域の長さ/既知配列の個数(フーリエ変換) の整数値です。

領域判定値：
領域が、既知配列を含む領域であるかを判定するための値です。大きいほど領域が既知配列を含む領域である可能性が高いです。

高い値が密集している領域が、既知配列が含まれる領域です



既知配列を含む領域以外が、多少検出されていても、後で判定できるので問題ない

既知配列を含む領域が検出されていることが分かる

既知配列の個数を数える手順を説明します

元の配列の名前	配列の長さ	既知配列開始位置	既知配列終了位置	既知配列の個数(推定)	既知配列の個数(フーリエ変換)	第n候補	既知配列の長さ(フーリエ変換)	領域判定値
'@0d36f7fe-acc8-4	48330	33781	39557	3	3	1	1925	2.70113246
'@7d719301-3f7f-4	36955	3329	7110	2	2	1	1890	2.61680657

①配列判定値で降順に並べます。分布から外れて明らかに低い値の領域は既知配列を含む領域ではありません。判定に迷う場合は出力された図を見ると判定できます。

元の配列の名前	配列の長さ	既知配列開始位置	既知配列終了位置	既知配列の個数(推定)	既知配列の個数(フーリエ変換)	第n候補	既知配列の長さ(フーリエ変換)	領域判定値
'@0d36f7fe-acc8-4	48330	33781	39557	3	3	1	1925	2.70113246
'@7d719301-3f7f-4	36955	3329	7110	2	2	1	1890	2.61680657

②第n候補が大きい値の場合(2以上)、フーリエ変換での既知配列の個数のカウントが困難であったことを表します。その場合、プログラムに入力した配列のテキストデータが荒れている可能性があります。既知配列の個数のカウントは、既知配列の個数(推定)　または　既知配列の個数(フーリエ変換)を用いてください

元の配列の名前	配列の長さ	既知配列開始位置	既知配列終了位置	既知配列の個数(推定)	既知配列の個数(フーリエ変換)	第n候補	既知配列の長さ(フーリエ変換)	領域判定値
'@0d36f7fe-acc8-4	48330	33781	39557	3	3	1	1925	2.70113246
'@7d719301-3f7f-4	36955	3329	7110	2	2	1	1890	2.61680657

③第n候補が小さい値(1)の場合、既知配列の個数のカウントは、既知配列の個数(フーリエ変換)を用いてください。既知配列の個数(推定)　と　既知配列の個数(フーリエ変換)が異なる場合がありますが、既知配列の個数(フーリエ変換)を用いてください。領域中に含まれる繰り返し単位が既知配列より少し短い(あるいは長い)場合があるからです