016_Rosalind_MPRT

July 4, 2019

1 Finding a protein motif

1.1 Overlapping motifs and Regular Expression

See www.regular-expressions.info/lookaround.html

The default setting for re to find a motif is find the motif in the string and then to start checking for another motif *after the original motif*. So in this problem you can have over lapping motifs

NNTSA N**NTSA**

The re function will not detect the second one with a normal re query. To check for overlapping motifs you need to be aware of the *lookaround* concept using the (?=u) syntax, where u in our case is the part of the query following N.

So `N[^P][ST][^P]` will become `N(?=[^P][ST][^P])`.

[1]:
```
# import all the required modules
import re, os, requests
```

[2]:
```
# Function to load rosalind list and parse into list
def loadRosalind(filepath):
    # get file path
    print(filepath)
    ids = []
    try:
        with open(filepath) as file:
            txt = file.read()
        ids = txt.split('\n')
    except:
        print("File not found")
    #print(ids)
    ids = [i for i in ids if len(i) > 0]
    print(ids)
    return ids
```

[3]:
```
# get fasta from Uniprot
def getFastas(ids):
    faPro = {}
    for protID in ids:
        UniFastaULR = "http://www.uniprot.org/uniprot/" + protID + ".fasta"
        print(UniFastaULR)
        UniFasta = requests.get(UniFastaULR)
```

```
        faPro[protID] = "".join(UniFasta.text.split('\n')[1:])
        #print(faPro[protID])
    return faPro
```

[18]:
```
def checkMotif(fasta):
    inxLst = []
    motif = re.compile("N(?=[^P][ST][^P])")
    inxLst = [(i.start() + 1) for i in re.finditer(motif, fasta)]
   # inxLst = [i.end() for i in re.finditer(motif, fasta)]
    return inxLst
```

[8]:
```
def main(fp):
    IDs = loadRosalind(fp)
    fastaD = getFastas(IDs)
    resInx = {}
    for key in fastaD:
        tmp = checkMotif(fastaD[key])
        print(key, len(tmp))
        if len(tmp) > 0:
            resInx[key] = tmp

    print()
    print("Results")
    print()

    for k in resInx:
        print(k)
        print(*resInx[k], sep=" ")
```

[12]:
```
main("/mnt/c/Users/rwswo/Documents/Bioinformatics/git/rosalindTry/proMotifTest.
 ↪txt")
```

```
/mnt/c/Users/rwswo/Documents/Bioinformatics/git/rosalindTry/proMotifTest.txt
['A2Z669', 'B5ZC00', 'P07204_TRBM_HUMAN', 'P20840_SAG1_YEAST']
http://www.uniprot.org/uniprot/A2Z669.fasta
http://www.uniprot.org/uniprot/B5ZC00.fasta
http://www.uniprot.org/uniprot/P07204_TRBM_HUMAN.fasta
http://www.uniprot.org/uniprot/P20840_SAG1_YEAST.fasta
A2Z669 0
B5ZC00 5
P07204_TRBM_HUMAN 4
P20840_SAG1_YEAST 11

Results

B5ZC00
85 118 142 306 395
P07204_TRBM_HUMAN
47 115 382 409
```

```
P20840_SAG1_YEAST
79 109 135 248 306 348 364 402 485 501 614
```

[20]:
```
main("/mnt/c/Users/rwswo/Documents/Bioinformatics/git/rosalindTry/rosalind_mprt.
 ↪txt")
```

```
/mnt/c/Users/rwswo/Documents/Bioinformatics/git/rosalindTry/rosalind_mprt.txt
['P21809_PGS1_BOVIN', 'P02974_FMM1_NEIGO', 'P05113_IL5_HUMAN', 'A8F2D7',
'P04180_LCAT_HUMAN', 'Q4FZD7', 'Q8ER84', 'P00304_ARA3_AMBEL', 'Q1E9Q9',
'P01878_ALC_MOUSE', 'Q5PA87', 'P81428_FA10_TROCA', 'P01047_KNL2_BOVIN',
'Q8R1Y2']
http://www.uniprot.org/uniprot/P21809_PGS1_BOVIN.fasta
http://www.uniprot.org/uniprot/P02974_FMM1_NEIGO.fasta
http://www.uniprot.org/uniprot/P05113_IL5_HUMAN.fasta
http://www.uniprot.org/uniprot/A8F2D7.fasta
http://www.uniprot.org/uniprot/P04180_LCAT_HUMAN.fasta
http://www.uniprot.org/uniprot/Q4FZD7.fasta
http://www.uniprot.org/uniprot/Q8ER84.fasta
http://www.uniprot.org/uniprot/P00304_ARA3_AMBEL.fasta
http://www.uniprot.org/uniprot/Q1E9Q9.fasta
http://www.uniprot.org/uniprot/P01878_ALC_MOUSE.fasta
http://www.uniprot.org/uniprot/Q5PA87.fasta
http://www.uniprot.org/uniprot/P81428_FA10_TROCA.fasta
http://www.uniprot.org/uniprot/P01047_KNL2_BOVIN.fasta
http://www.uniprot.org/uniprot/Q8R1Y2.fasta
P21809_PGS1_BOVIN 2
P02974_FMM1_NEIGO 3
P05113_IL5_HUMAN 2
A8F2D7 0
P04180_LCAT_HUMAN 4
Q4FZD7 1
Q8ER84 1
P00304_ARA3_AMBEL 1
Q1E9Q9 5
P01878_ALC_MOUSE 4
Q5PA87 0
P81428_FA10_TROCA 1
P01047_KNL2_BOVIN 7
Q8R1Y2 0

Results

P21809_PGS1_BOVIN
271 312
P02974_FMM1_NEIGO
67 68 121
P05113_IL5_HUMAN
```

```
47 90
P04180_LCAT_HUMAN
44 108 296 408
Q4FZD7
528
Q8ER84
33
P00304_ARA3_AMBEL
41
Q1E9Q9
185 255 347 640 1326
P01878_ALC_MOUSE
38 99 314 329
P81428_FA10_TROCA
254
P01047_KNL2_BOVIN
47 87 168 169 197 204 280
```

[19]:
```
seq='ANNTTAAAANNTTAAA'
# 2 3 10 11
checkMotif(seq)
```

[19]:
```
[2, 3, 10, 11]
```

[ ]:
```

```