微生物多重PCR引物设计流程_使用说明

一. 准备工作及说明

- 2~4部分为生成用于引物设计的简并序列的方法
- 使用JCVI Primer Designer设计简并引物只看第5部分就可以了。
- 检测引物相互作用，只看第6部分就可以了。

脚本程序采用Python3编写，用到了Numpy和Pandas，最好通过安装Anaconda来安装Python
除特殊说明，一般将python脚本、待处理文件放入同一个文件夹，使用cmd进入该文件夹目录下，输入：

```
python 脚本程序名称 文件名称 (其他文件名，参数名等)
```

即可运行脚本。
需安装程序及其下载地址：
Anaconda:
Windows 7 64位：https://repo.continuum.io/archive/Anaconda3-4.3.0.1-Windows-x86_64.exe
Windows XP 32位：https://repo.continuum.io/archive/Anaconda3-2.2.0-Windows-x86.exe

二. 确定候选参考序列

从NCBI等数据库中检索和下载用于生成共同序列的候选参考序列，有时需截取某段序列用于序列比对，序列较多时操作比较麻烦。编写可自动生成截取序列的脚本。
相关脚本程序：Extract_Feature_Sequence.py（位于Extract Feature Sequences文件夹中）
需安装Biopython后才能使用，**Biopython安装方法：**

- 安装Anaconda后，cmd窗口输入如下命令安装：

```
pip install biopython
```

- 或从http://biopython.org/DIST/biopython-1.68.win32-py3.5.msi下载安装

输入：候选参考序列，GenBank full sequence格式。
处理：批量从GenBank full sequence格式的序列中截取出某段序列（Features》CDS》product项），用于下一步序列比对
输出：截取的序列组成的fasta文件。
使用方法：

1. 将需截取序列的genbank文件（以'.gb'结尾）放在一个文件夹下，如 .\examples
2. 修改参数：
   - 16行，search_terms中引号部分替换成想要检索的CDS名称。
   - 17行，min_seq, max_seq部分替换成规定长度。
3. 输入命令: python + 脚本名称（Extract_Feature_Sequence.py）+ 序列存放文件夹路径（可以为相对或绝对路径）即可生成待检测基因的序列。

例子：使用从NCBI下载的艰难梭菌Genbank full length序列为输入序列，希望提取Features》CDS》product项为toxin B、tcdB或toxin b的对应序列，要求序列长度为7000~7200bp，用于下一步序列比对。
序列长度符合要求，写入输出文件output_file.fasta。
如果超出范围，则写入outrange.fasta文件。
源文件格式为：

```
FEATURES
......
CDS             8078..15178
                     /locus_tag="A4X29_RS16135"
                     /inference="COORDINATES: similar to AA
                     sequence:RefSeq:YP_001087135.1"
                     /note="Derived by automated computational analysis using
                     gene prediction method: Protein Homology."
                     /codon_start=1
                     /transl_table=11
                     /product="toxin B"
                     /protein_id="WP_021363304.1"
                     /db_xref="GI:544958414"
                     /translation="MSLVNRKQLEKMANVRFRTQEDEYVAILDALEEYHNMSENTVVE
                     KYLKLKDINSLTDIYIDTYKKSGRNKALKKFKEYLVTEVLELKNNNLTPVEKNLHFVW
                     IGGQINDTAINYINQWKDVNSDYNVNVFYDSNAFLINTLKKTVVESAINDTLESFREN
......
```

修改下列参数：

```
search_terms = {"tcdB","toxin b","toxin B"} #将引号内的tcdB等名称可修改为想要提取的CDS名称
min_seq, max_seq = (7000,7200) #将括号内7000,7200可修改为提取CDS的长度范围
```

cmd窗口输入输入命令

```
python Extract_Feature_Sequence.py  .\examples
```

生成output_file.fasta，生成有特征CDS的序列。
和outrange.fasta两个文件，包括有特征CDS，但长度不符合要求的序列。

三. 生成用于设计引物的共同序列

使用Clustal X做序列比对，将比对后的fasta文件生成consensus sequence， 设置cut-off值为0.01，即生成的consensus sequence可覆盖99%的参考序列。
相关脚本程序：Alligned_Fasta_to_Consensus.py（位于Alligned Fasta to Consensus文件夹）
输入：序列比对后的文件，fasta格式，使用Clustal X等序列比对程序生成。
处理：序列比对过程中经常会引入gap, 表示一些序列的可能存在的插入或缺失
de_gap_cut_off表示去掉gap的cut off值，脚本中设置为0.1，当gap的比例大于0.9或小于0.1时，将consensus sequence中的gap去掉。
ambiguous_cut_off表示consensus sequence中碱基出现的最低频率，出现频率低于ambigous_cut_off的碱基将被舍弃，不在最后生成的consensus sequence中。脚本中设置为0.02。
分别在第23行和24行，可根据需要调整。
输出：起始行显示输入和输出序列的信息，以及序列简并度的统计。
然后输出3个fasta格式序列 分别是：

- consensus_seq： 将所有gap去掉，以碱基表示所有位置
- consensus_seq with gaps：如果一个位置的gap比例在10%~90%之间，则将gap显示出来，尽量避免在附近区域设计引物。
- consensus_seq all gaps： 所有序列中即使只存在一个gap，也将gap显示出来。

使用方法：

1. 根据需要调整代码第29行和第30行的de_gap_cut_off，ambiguous_cut_off
2. 输入命令：python + Alligned_Fasta_to_Consensus.py + 序列比对文件.fasta

例子： 生成轮状病毒的consensus sequence
轮状病毒的序列比对后的文件example_rotavirus.fasta为输入文件；
将下列参数调整为所需的值

```
de_gap_cut_off = 0.10  #define the de gap cut off value
ambiguous_cut_off = 0.02 #define the cut off value
```

cmd窗口运行：

```
python Alligned_Fasta_to_Consensus.py example_rotavirus.fasta
```

生成输出文件example_rotavirus_consensus_seq.fasta， 包括统计信息以及consensus sequence。

四. 确定用于设计引物的目标区域

从上一步生成的consensus sequence中找到简并度最低（即保守度最高）的一段序列，用做引物设计的输入序列。
相关脚本程序：Conserved_Region_2.py（位于Conserved Region文件夹）
输入： Fasta格式的consensus sequence序列
处理：找到找到consensus sequence中**Conserved Region**(简并度最低的n bp序列)。并在这段保守序列上下游各延伸m bp，得到**Output Region**(总长(m+2n)bp的序列作为设计引物的目标区域)。n, m的值可以调整。脚本中设置为n=150, m=250.
在脚本程序的第8行和第9行。
输出：Report文件包括conserved region以及output region的序列及统计。Region文件包括output region序列
例子：由example_rotavirus_consensus_sequence.fasta找到简并度最低的序列
根据需要调整conserved_len与ajacent_len的值。

```
conserved_len = 150 #The length of the most conserved region
ajacent_len = 250 #The length next to the most conserved region
```

输入命令：

```
python Conserved_Region_2.py example_rotavirus_consensus_sequence.fasta
```

生成
example_rotavirus_consensus_sequence_conserved_region.fastaexample_rotavirus_consensus_sequence_conserved_report.fasta
两个文件

五. 使用JCVI Primer Designer生成候选简并引物

JCVI Prmer Designer下载地址：
https://sourceforge.net/projects/primerdesigner/files/primerdesigner/PrimerDesigner20101122/
安装Linux系统：
运行JCVI Primer Designer需使用Linux系统
可以参考：
http://www.linuxdiyf.com/linux/11165.html
安装Ubuntu或其他Linux系统。

- 该软件用于生成高通量测序用的候选简并引物，会将简并引物还原为ATGC序列再检测非特异扩增，发卡结构，引物退火温度，引物相互作用等。要求序列简并度最好<10%，之前使用显示序列简并度<20%也可使用。
- 需使用Linux基础的系统，测试在Ubuntu 15.04下可以使用。
- 按照说明书按照软件后仍可能无法直接使用，可能需添加环境变量，修改一下代码。

假设软件安装在/home/username/primerdesign/primer_design文件夹中。
添加环境变量， 命令行输入：

```
export PERL5LIB=/home/username/primerdesign/primer_design
export PATH="/home/username/primerdesign/primer_design:$PATH"
export EMBOSS_ACDROOT=/home/username/primerdesign/primer_design/external_software/EMBOSS/acd
export PATH="home/username/primerdesign/primer_design/external_software/blast/blast-2.2.15:$PATH"
```

应将上述的/home/username/primerdesign/primer_design文件夹替换为JCVI Primer Desinger实际安装文件夹
安装libgd2-xpm-dev， 命令行窗口输入：

```
sudo apt-get -y install libgd2-xpm-dev build-essential
```

修改primer_design/PrimerDesigner/ProjectTools/RegionParallel/Submit_Parallel_Local_Scripts.pl 文件

```
把第116行的">&" 改为"&>"
```

相关脚本程序： Primer_name_conversion_1.py（位于JCVI Primer Design文件夹）

```
#JCVI Primer Designer生成引物名称为如下格式:
>adenovirus_f_00000_0000.l /begin=11 /end=29 /orientation=1 /length=18
CTCGATGATGCCGCAATG
>adenovirus_f_00000_0000.r /begin=291 /end=312 /orientation=-1 /length=21
GCGGATGTCAAAGTAGGTGCT
>adenovirus_f_10001_0001.l /begin=30 /end=49 /orientation=1 /length=19
TCTTACATGCACATCGCCG
#Primer3_Plus 生成引物名称为如下格式:
>HumanRotavirus_A_left_F
KAATGCTTTTCAGTGGTTGHTGCT
>HumanRotavirus_A_left_R
GTHGAAGTDGCAGCRACDACYGC
>HumanRotavirus_A_left_1_F
TCAGTGGTTGHTGCTCAAGATGGAGT
```

用于下一步引物筛选不是很方便，需要将引物名称转变为：

```
组名#F/R#index
>adenovirus_f#F#0
CTCGATGATGCCGCAATG
>adenovirus_f#R#0
GCGGATGTCAAAGTAGGTGCT
>adenovirus_f#F#1
TCTTACATGCACATCGCCG
>adenovirus_f#R#1
GGGRCCACGATCCAGCAC
```

如果下一步只是检测引物或探针之间的相互作用，不需要分组，则无需转换名称，但引物名中不能包括'@'
输入：JCVI或Primer 3 Plus得到的引物序列
处理：转变引物名称
输出：以"#"分隔的引物名称

例子，将JCVI_primer_design_results.fasta转变引物名称
输入：python Primer_name_conversion_1.py + 引物文件名+ jcvi或primer3_plus

```
python Primer_name_conversion_1.py JCVI_primer_design_results.fasta jcvi
```

得到引物名称转变后文件：JCVI_primer_design_results_name_converted.fasta

六. 排除有相互作用的引物，生成用于实验的引物

使用Python脚本检测引物之间的相互作用，生成没有相互作用的引物组合。并排除扩增片段大小或扩增片段差异不合适的引物组合。
相关脚本程序：（位于Primer Selector文件夹）
PrimerCheck.py: 检测引物之间的相互作用，可显示规则规定下所有可能的相互作用, 引物名称任意，不能包括'@'。
PrimerSelector_Group.py: 要求引物名称为：组名#F/R#index， 生成不同引物组没有相互作用的组合，其他与PrimerCheck.py相同。
PrimerSelector_Amplicon.py:

- 可以检测引物与Amplicon的相互作用，输出Amplicon在规定大小的引物。
- 优化计算过程，简并引物中有一个还原为ATGC后的引物与其他引物有作用即停止计算。
- 可规定简并引物的退火温度范围，简并度范围等。

输入： Fasta格式的引物序列，序列名格式 如下：

```
>Norovirus_GI#F#3
GARAARTTYTACAGRAAGAT
>Norovirus_GI#R#4
RACCCARCCATTRTACATY
>Rotavirus_A#F#0
GGCWTTTAATGCTTTTCAGT
>Rotavirus_A#R#0
TTWACRCCWGARTCATCCAT
......
```

Fasta格式的input region, 一般选择之前的引物设计区域，用于检测这个区域是否存在引物的非特异结合，计算Amplicon大小。

```
>SapoVirus_GI_consensus_seq
RGWYGTTGACCYKCCTGGCRCRRYDGGYCCGRCCACATCCMAYGTTGTTGTKKCTAATCCRGARCAACCCAATGGGSCCGCACARCGCYTGGARHTGGCTGTTGCYAC
```

处理： 如上所述, 可修改相关参数以符合需求，但应注意比对长度>=cutoff值，如

```
rule_2_len = 8 #3' ends len(with one mismatch)
rule_2_cutoff = 7 #3' ends cutoff score(with one mismatch)
```

参数均位于文件的21行，具体意义请参考注释

输出：

引物文件名_atgc.fasta ： 将简并引物的简并碱基转变为ATGC之后的引物

引物文件名_confliction_report.fasta ： 引物之间可能存在的相互作用

引物文件名_confliction_table.csv： 引物间可能存在相互作用的excel表， 没有相互作用的引物为0，有相互作用的为1.

引物文件名_primer_combinations.txt： 没有相互作用的引物组合。

引物文件名_primers_information.txt： 引物的简并度、退火温度等信息。

例子：

输入文件为：test_primers.fasta， test_input_region.fasta（仅用于PrimerSelector_Amplicon.py检测）

修改参数，与Amplicon, Tm值计算相关的参数只有PrimerSelector_Amplicon.py中才有：

```
#Parameters to calculate delatG
mono = 50 #mono-valent ion concentration, mM
diva = 1.5 #divalent ion concentraion, mM
oligo = 50 #oligo concentraion to calculate Tm
dntp = 0.25 #dntp concentraction, mM
degenerate_max_tm = 66 #The max Tm of degenerate primers, Celsius Degree
degenerate_min_tm = 44 #The min Tm of degenerate primers, Celsius Degree
max_degenerate_number = 128 #The max degeneration number of primer

#Parameters to calculate primer interactions
rule_1_len = 4 #3' ends len
rule_1_cutoff = 4 #3' ends cutoff score
rule_2_len = 8 #3' ends len(with one mismatch)
rule_2_cutoff = 7 #3' ends cutoff score(with one mismatch)
rule_3_len = 10 #primer len
rule_3_cutoff = 10 #primer cutoff score
rule_4_len = 13 #primer len(with one mismatch)
rule_4_cutoff = 12 #primer cutoff score(with one mismatch)
rule_5_len = 14 #match percent calculation min len
rule_5_cutoff = 0.75 #match percent cutoff score
rule_6_len = 8 #alignment score len
rule_6_cutoff = 8 #alignment cutoff score
rule_7_len = 7 #delta G calculation min len
rule_7_cutoff = -9 #delta G cutoff score, kcal/mol
rule_7_cutoff = -1*rule_7_cutoff #change cut-off score to positive value


amplicon_primer_check_len = 14 #The 3' ends of primer that will be checked with amplicon sequence
amplicon_primer_cutoff = 12 #The score of primer-amplicon interaction
amplicon_len_max = 550 #The maximum length of amplicon
amplicon_len_min = 70 #The minimum length of amplicon
amplicon_lendiff_min =15 #The minimum length differences between amplicons
```

另外将MPprimer计算吉布斯自由能的模块添加到了脚本程序中，ThermodynamicsParameters.py为储存相应计算参数的文件。

该软件采用SantaLucia JR在1997~1998年发表文献中的参数。如果需要别的参数，可对ThermodynamicsParameters.py进行修改。

```
dH_full={
        'AATT' : -7.9, 'TTAA' : -7.9,
        'ATTA' : -7.2, 'TAAT' : -7.2,
        'CAGT' : -8.5, 'TGAC' : -8.5,
        'GTCA' : -8.4, 'ACTG' : -8.4,
        'CTGA' : -7.8, 'AGTC' : -7.8,
        'GACT' : -8.2, 'TCAG' : -8.2,
        'CGGC' : -10.6, 'GCCG' : -9.8,
        'GGCC' : -8.0, 'CCGG' : -8.0,
        'initC' : 0.1, 'initG' : 0.1,
        'initA' : 2.3, 'initT' : 2.3,
        # Like pair mismatches
        'AATA' : 1.2, 'ATAA' : 1.2,
        'CAGA' : -0.9, 'AGAC' : -0.9,
        'GACA' : -2.9, 'ACAG' : -2.9,
        'TAAA' : 4.7, 'AAAT' : 4.7,
......
dS_full={
        'AATT' : -22.2, 'TTAA':-22.2,
        'ATTA' : -20.4, 'TAAT':-21.3,
        'CAGT' : -22.7, 'TGAC':-22.7,
        'GTCA' : -22.4, 'ACTG':-22.4,
        'CTGA' : -21.0, 'AGTC':-21.0,
        'GACT' : -22.2, 'TCAG':-22.2,
        'CGGC' : -27.2, 'GCCG':-24.4,
        'GGCC' : -19.9, 'CCGG':-19.9,
        'initC' : -2.8, 'initG':-2.8,
        'initA' : 4.1, 'initT':4.1,
        'sym' : -1.4,
        # : Like:pair:mismatches
        'AATA' : 1.7, 'ATAA':1.7,
......
```

根据需求，在cmd窗口运行如下命令：

```
python PrimerCheck.py test_primers.fasta
或
python PrimerSelector_Group.py test_primers.fasta
或
python PrimerSelector_Amplicon.py test_primers.fasta test_input_region.fasta
```

得到test_primers_atgc.fasta， test_primers_confliction_report.fasta， test_primers_primer_information.txt，test_primers_primer_combinations.txt等文件。
查看文件，决定调整参数或订购引物进入实验阶段。