Prática 1 - Processamento de cromatogramas

Alessandro S. Nascimento

10 de Agosto de 2016

1 Introdução

São fornecidos dois pares de cromatogramas:
121107H07R e 121107H07F;
121107G02F e 121107G02R.

Cada um deste pares representa sequenciamentos de uma mesmo molécula de DNA, entretanto cada uma das sequências foi gerada a partir de uma extremidade diferente da molécula de acordo com a figura 1.

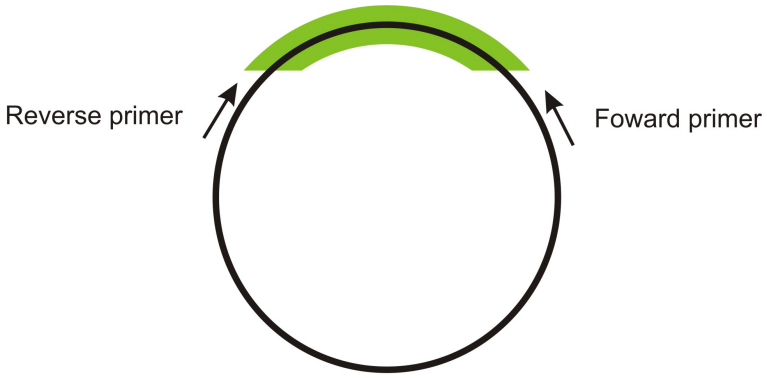

Figura 1: O círculo representa um vetor circular de DNA dupla fita e a porção verde corresponde ao inserto da molécula de DNA a ser sequenciada. As setas indicam os diferentes *primers* utilizados no seqüenciamento. Estes primers são complementares às diferentes fitas do vetor.

É possível realizar o sequenciamento de cada uma das extremidades porque diferentes *primers* que são complementares a sequência do vetor em que se encontra este fragmento foram utilizados. A terminação do nome da seqüência F ou R (*foward* ou *reverse*) se refere aos diferentes primers utilizados.

2 Atividade 1 (Sistema Operacional *Windows*)

1. Baixe os cromatogramas utilizando os links abaixo:
   - 121107H07R

- 121107H07F
- 121107G02F
- 121107G02R

2. Abra o programa bioedit;
3. Vá em *file->open* e selecione o cromatograma 121107H07F;
4. Duas janelas devem se abrir. Uma contendo o cromatograma e outra apenas a sequência, selecione a janela contendo o cromatograma;
5. Inspecione o cromatograma em toda sua extensão. Tente definir uma base a partir da qual você não considere a atribuição de bases totalmente confiável;
6. Repita o mesmo procedimento para o cromatograma da mesma molécula sequenciado com o outro primer (121107H07R);
7. Vá até a janela contendo apenas a sequência do cromatograma 121107H07F, clique em cima do nome da sequencia (este deve ficar escuro);
8. Vá em *Edit->copy sequence*;
9. Selecione a janela contendo apenas a sequência do cromatograma 121107H07R e clique em *Edit->paste sequence*.
10. Clique em cima do nome da sequência *reverse* para selecioná-la;
11. Vá em *Sequence->Nucleic acid->Reverse complement*. Isto permite obter o complemento reverso de uma das sequencias;
12. Selecione as duas sequências nesta janela clicando nos nomes das duas sequencias enquanto segura a tecla *shift* pressionada;
13. Clique em *Acessory application-> CAP assembly program*;
14. Na janela que se abrir clique em *Run application*.
15. Uma janela de comando se abrirá. Feche-a;
16. Uma nova janela se abrirá contendo uma linha adicional chamada *contig-0*. Esta sequência consenso representaria a sequência inteira da molecula obtida a partir da junção das duas sequencias;
17. Inspecione esta nova sequência. As duas linhas acima representam as sequências originais. Todas as bases das sequências originais e da sequência consenso são coincidentes? Como você poderia decidir qual é a base correta? (**Q1**).
18. Feche todas as janelas e repita o procedimento acima para as sequências 121107G02R e 121107G02F. O que ocorre quando você tenta criar a sequência consenso? Por que você acha que isso ocorre? (**Q2**).

3 Atividade 2 (Sistema Operacional Linux)

Nesta atividade utilizaremos um pacote de programas para analisar os nossos cromatogramas. Este pacote inclui os programas *Phred*, *Phrap*, *Cross/match* e *Consed*.

3.1 Phred

Realiza a leitura de cromatogramas, atribui as bases e dá um escore para a atribuição de cada base. Este escore de qualidade é relacionado à probabilidade de erro de leitura da base seguindo a seguinte formula:

$$Q = -10 \log_{10}(Pe) \tag{1}$$

onde Q é o escore de qualidade, Pe é a probabilidade de erro.

Phred quality score	**Probability that the base is called wrong**	**Accuracy**
10	1 in 10	90%
20	1 in 100	99%
30	1 in 1000	99.9%
40	1 in 10000	99.99%
50	1 in 100000	99.999%

Este escore de erro é comumente utilizado como parâmetro para seleção de trechos de sequências de alta qualidade para deposição em bancos públicos. Normalmente bases com qualidade acima de 20 são consideradas de alta qualidade.

3.2 Phrap e cross/match

O *Phrap* é um programa para montagem de sequências de DNA. Ele foi criado primariamente para analise de dados de DNA genômico, mas pode ser utilizado em sequências de cDNA também. Ele lê as informações de qualidade geradas pelo *phred* e utiliza esta informação para atribuição da sequência consenso a partir das bases de melhor qualidade.

Cross-match é uma ferramenta de comparação de duas sequências. É utilizada para detecção de seqüências proveniente do vetor.

3.3 Consed

O *Consed* é uma ferramenta para visualização e edição das montagens de seqüências geradas pelo Phrap.

A documentação do phred/phrap/consed pode ser encontrada nos seguintes sites:

http://www.phrap.org/phredphrapconsed.html
http://www.phrap.com/phred

3.4 Atividade

1. Crie na sua pasta *home* uma pasta chamada *"teste"*;

2. Dentro desta pasta, crie duas novas pastada chamadas *"chromat_dir"* e *"edit_dir"*;
3. coloque todos os cromatogramas dentro do diretorio *chromat_dir*
4. Entre no terminal de comando e digite:

   ```
   cd teste/edit_dir
   ```

5. Digite o comando:

   ```
   phredPhrap
   ```

6. Digite o comando:

   ```
   consed_linux32bit
   ```

7. Clique no arquivo *ace* indicado e aperte o botão *open*;
8. Clique nos *contigs* indicados e inspecione-os;
9. Verifique a qualidade das bases de todas as sequências. Veja onde a qualidade de bases começa a ser menor do que 20 (ao colocar o cursor em cima de cada base a sua qualidade aparece na parte de baixo da janela). Esta base é próxima ao inicio trecho que você havia indicado como de baixa confiança na atividade anterior ?
10. Na base com qualidade 20 ou inferior, clique com o botão direito e selecione a opção *display traces for all contigs*. Avalie a qualidade do cromatograma neste trecho e compare-a com um trecho de alta qualidade.
11. Repare que na bordas das seqüências existem bases mascaradas com X, baseado no desenho do experimento você saberia dizer o que estas bases representam? (**Q3**)

QUESTÃO (Q4): Faça uma análise comparativa dos dois programas utilizados nas atividades.

4 Atividade Extra: Conversão de Formatos de Sequências

Usando o script em *python* mostrado abaixo, vamos converter a sequência de uma proteína de um arquivo PDB (contendo a estrutura da proteína) para o formato FASTA. Para esta finalidade, vamos precisar também converter a sequência de um código de 3 letras para o código de 1 letra utilizado no formato FASTA.

O código abaixo deve ser escrito em um editor de texto e salvo com o nome *pdb2seq.py*.

```
#!/usr/bin/env python
import sys, time

t1=time.time()
```

```
print "#########################################################################"
print "##                        PDB2SEQ.PY                                   ##"
print "##  Exporta a sequencia de aminoacidos dada no identificador  ##"
print "##        SEQRES do arquivo PDB em formato FASTA)           ##"
print "#########################################################################"

# Abre o arquivo no PDB

try :
    pdb = open(sys.argv[1], 'r')
except:
    pdbin=raw_input('Please, provide the name of a PDB file:')
    pdb=open( pdbin , 'r')

# Dicionario de AA: 3 letras para 1 letra

aas = {'ALA':'A', 'ARG':'R', 'ASN':'N', 'ASP':'D', 'CYS':'C', 'GLU':'E', 'GLN':
'ILE':'I', 'LEU':'L', 'LYS':'K', 'MET':'M', 'PHE':'F', 'PRO':'P', 'SER':'S', 'T
'VAL':'V'}


# Vamos localizar o identificador de sequencia da proteina (SEQRES) e ler a seq

seq = []
nres=0

for line in pdb:
        line2 = line.split()
        if line2[0] == 'TITLE':
                title = line
        if line2[0] == 'SEQRES':
                nres = line2[3]
                for i in range (4, len(line2)):
                        seq.append(line2[i])

print "Esta proteina contem ", nres, " residuos:"
print
print "Checando integridade da sequencia..."
if ((len(seq)) == int(nres)):
        print "Integridade confirmada!"
else:
        print "*** PERIGO: Ha um problema com o numero de residuos. Cheque a ex

print
print "Mostrando o output em formato FASTA"
print

print "> ", title,
for i in range(0, len(seq)-1):
```

```
        sys.stdout.write(aas[seq[i]])

print
print

print "3-letter code sequence"



for i in range(0, len(seq)-1):
        print '%s' % (seq[i]),


print

t2 = time.time()
print 'Elapsed time: %0.3f ms' % ((t2-t1)*1000.0)
```

1. No ambiente LINUX, vá até o terminal e digite:

   chmod +x pdb2seq.py

   Este comando tornará o script executável.

2. Vá até o PDB (http://www.rcsb.org/pdb/explore/explore.do?structureId=3exd e baixe a estrutura da lisozima da clara do ovo da galinha, código 3EXD no PDB.

3. Execute o comando:

   pdb2seq.py 3EXD.pdb

   O script deve mostrar a sequência em formato FASTA, bem como a sequência da proteína com código de 3 letras.