

Índice

Índice.....	2
Introdução	3
Materiais e Métodos	4
Resultados/Discussão.....	7
Bibliografia.....	9

Introdução

No âmbito da unidade curricular de Análise de Sequências Biológicas, foi-nos proposto pelo docente a realização de uma tarefa/trabalho com o objetivo de aperfeiçoarmos as nossas habilidades e práticas tendo como base e princípios alguns métodos aprendidos nas aulas durante o semestre onde inclusive utilizamos alguns dos programas empregues no Assignment 1.

O principal objetivo deste trabalho é esclarecer algumas dúvidas associadas a um tipo de planta amazónica. Alguns exploradores encontraram este tipo de planta e repararam que de entre todas as suas características, a que mais se sobressaía era o facto de que alguns indivíduos possuíam flores particularmente malcheirosas (Pungents) enquanto a maioria deste tipo de planta não possuía nenhum odor perceptível (Bland). Alguns destes exploradores afirmavam que este acontecimento se devia ao facto de que alguns indivíduos Pungent pertenciam a uma espécie diferente, enquanto outros diziam que se devia à variabilidade intraespecífica. Como tal, tendo em conta que o método de Sanger foi inconclusivo, os indivíduos Pungent e Bland foram sequenciados utilizando métodos RAD-Seq.

Neste trabalho, de modo a resolver o problema descrito, o grupo criou um plano de trabalho para uma abordagem mais elaborada e organizada. Decidimos então utilizar o método de High Throughput Sequencing (HTS).

Este método de High Throughput Sequencing é utilizado na maioria das vezes para estudos de expressões génicas. Esta técnica de HTS, como era antigamente chamada, era utilizada para sequenciar cópias de DNA expressos em pequenas amostras. Nos dias de hoje, o mesmo é feito, embora em muito maior escala, e agora chamado de RNA-seq que mostra os genes que são ativos numa determinada amostra biológica.

Posto isto, na realização deste trabalho, o conjunto de dados que utilizamos incluíam as espécies Pungent e Bland.

Materiais e Métodos

Neste ponto do relatório vamos fazer uma descrição dos processos e do procedimento que realizamos. Inicialmente, para dar início à metodologia do trabalho, fizemos o download de um “pre-built High Throughput Sequencing (HTS) dataset” que nos foi fornecido pelo docente. Nestes ficheiros podemos encontrar todo o datatype necessário com que iríamos trabalhar.

Posteriormente descompactamos ambos os ficheiros e todos os documentos necessários para preparação do datatype.

```
(base) prbio@prbio-VirtualBox:~/ipyrad-assembly/small_test_dataset-master$ cd Cornales/  
(base) prbio@prbio-VirtualBox:~/ipyrad-assembly/small_test_dataset-master/Cornales$ ls  
Cornales2.barcodes  Cornales3.barcodes  Cornales.barcodes  Cornales_info.md  Cornales_R1.fastq  Cornales_R2.fastq
```

Fig. 1 - Ficheiros descompactados e presentes na VM

De seguida, pelo facto de possuímos os parâmetros padrões, tivemos de realizar algumas alterações e ajustes. Houve uma necessidade de mexer nos parâmetros de modo a permitir que conseguíssemos fazer correr o ipyrad. (Figura 2)

```
----- ipyrad params file (v.0.9.77)-----  
data1      ## [0] [assembly_name]: Assembly name. Used to name output directories for assembly steps  
/home/prbio/ipyrad-assembly/small_test_dataset-master/Cornales ## [1] [project_dir]: Project dir (made in curdir if not present)  
/home/prbio/ipyrad-assembly/small_test_dataset-master/Cornales/*.fastq ## [2] [raw_fastq_path]: Location of raw non-demultiplexed fastq files  
/home/prbio/ipyrad-assembly/small_test_dataset-master/Cornales/Cornales2.barcodes ## [3] [barcodes_path]: Location of barcodes file  
denovo      ## [4] [sorted_fastq_path]: Location of demultiplexed/sorted fastq files  
## [5] [assembly_method]: Assembly method (denovo, reference)  
## [6] [reference_sequence]: Location of reference sequence file  
rad         ## [7] [datatype]: Datatype (see docs): rad, gbs, ddrad, etc.  
TGCAG,     ## [8] [restriction_overhang]: Restriction overhang (cut1,) or (cut1, cut2)  
5          ## [9] [max_low_qual_bases]: Max low quality base calls (Q<20) in a read  
33         ## [10] [phred_Qscore_offset]: phred Q score offset (33 is default and very standard)  
6          ## [11] [mindepth_statistical]: Min depth for statistical base calling  
6          ## [12] [mindepth_majrule]: Min depth for majority-rule base calling  
10000      ## [13] [maxdepth]: Max cluster depth within samples  
0.85       ## [14] [clust_threshold]: Clustering threshold for de novo assembly  
0          ## [15] [max_barcode_mismatch]: Max number of allowable mismatches in barcodes  
2          ## [16] [filter_adapters]: Filter for adapters/primers (1 or 2=strict)  
35         ## [17] [filter_min_trim_len]: Min length of reads after adapter trim  
2          ## [18] [max_alleles_consens]: Max alleles per site in consensus sequences  
0.05       ## [19] [max_Ns_consens]: Max N's (uncalled bases) in consensus  
0.05       ## [20] [max_Hs_consens]: Max Hs (heterozygotes) in consensus  
4          ## [21] [min_samples_locus]: Min # samples per locus for output  
0.2        ## [22] [max_SNPs_locus]: Max # SNPs per locus  
8          ## [23] [max_Indels_locus]: Max # of indels per locus  
0.5        ## [24] [max_shared_Hs_locus]: Max # heterozygous sites per locus  
0, 90, 0, 0 ## [25] [trim_reads]: Trim raw read edges (R1>, <R1, R2>, <R2) (see docs)  
0, 0, 0, 0  ## [26] [trim_loci]: Trim locus edges (see docs) (R1>, <R1, R2>, <R2)  
*          ## [27] [output_formats]: Output formats (see docs)  
           ## [28] [pop_assign_file]: Path to population assignment file  
           ## [29] [reference_as_filter]: Reads mapped to this reference are removed in step 3
```

Fig. 2 – Ajustes nos parâmetros

Nesta situação, por possuímos dois ficheiros fastq tivemos de usar o “*.fastq” para utilizar os dois ficheiros como demonstra a figura 3.

```
/home/prbio/ipyrad-assembly/small test dataset-master/Cornales/*.fastq ## [2] [raw fastq path]: Location of raw non-demultiplexed fastq files
```

Fig. 3 – Passo para utilizar os dois ficheiros (*.fastq)

Posteriormente, procedemos à execução do ipyrad desde o passo dois até ao passo 7. Neste último fomos impedidos de continuar devido a um erro que nos informava que não havia taxa suficiente para os barcodes mencionados. Assim, de modo a resolver este problema, utilizamos um script de Python para ultrapassar a barreira.

```
def openbarcodes():

    bland = ['Bland-22', 'Bland-28', 'Bland-51', 'Bland-70', 'Bland-90', 'Bland-82']

    f = open('Cornales.barcodes', 'r')
    lines = f.readlines()
    f2 = open("Cornales2.barcodes", "w")

    for line in lines:
        if line.startswith(tuple(bland)):
            line = ''
            f2.write(line)
        else:
            f2.write(line.strip())
            f2.write('\n')

def main():
    openbarcodes()

if __name__ == "__main__":
    main()
```

Fig. 4 – Script de Python para resolver o problema de taxa

Com este script de Python, pudemos criar um novo ficheiro de barcodes que apenas contém os barcodes necessários para a realização da tarefa.

De seguida, utilizamos a ferramenta Raxml para criar uma árvore e mais tarde utilizaremos o Figtree para visualizar o resultado obtido.

```
(base) prbio@prbio-VirtualBox:~/ipyrad-assembly/small_test_dataset-master/Cornal  
es/datal_outfiles$ raxmlHPC-PTHREADS-AVX -T 4 -m GTRCAT -s /home/prbio/ipyrad-as  
sembly/ipyrad_output.fasta -n rax
```

Fig. 5 – Processo de utilização do Raxml

Neste ponto, como apenas tínhamos utilizado um método, decidimos dar uso à ferramenta Mrbayes para obter uma outra árvore. Aqui utilizamos o ficheiro Nexus que obtivemos do ipyrad e de seguida, para simplificar o processo, usamos o bloco do Mrbayes para podermos corrê-lo.

Abrimos o terminal na pasta onde tínhamos o ficheiro Nexus que usamos na execução do processo do Mrbayes, colocando lá o resultado obtido.

```
(base) prbio@prbio-VirtualBox:~/ipyrad-assembly/small_test_dataset-master/Cornales$ mb  
  
MrBayes v3.2.6 x64  
(Bayesian Analysis of Phylogeny)  
Distributed under the GNU General Public License  
  
Type "help" or "help <command>" for information  
on the commands that are available.  
  
Type "about" for authorship and general  
information about the program.  
  
MrBayes > execute datal.nex
```

Fig. 6 – Processo de execução do Mrbayes

Para finalizar, depois de todo este processo descrito, obtivemos com output um ficheiro com uma árvore, onde utilizamos a ferramenta Figtree para observar a mesma.

Resultados/Discussão

Como não foi possível visualizarmos a nossa árvore no RAxML, decidimos utilizar a ferramenta Figtree para nos auxiliar na nossa visualização.

Nesta árvore é difícil distinguir, com facilidade, quais os ramos pertencentes à espécie Pungent dos ramos pertencentes à espécie Bland.

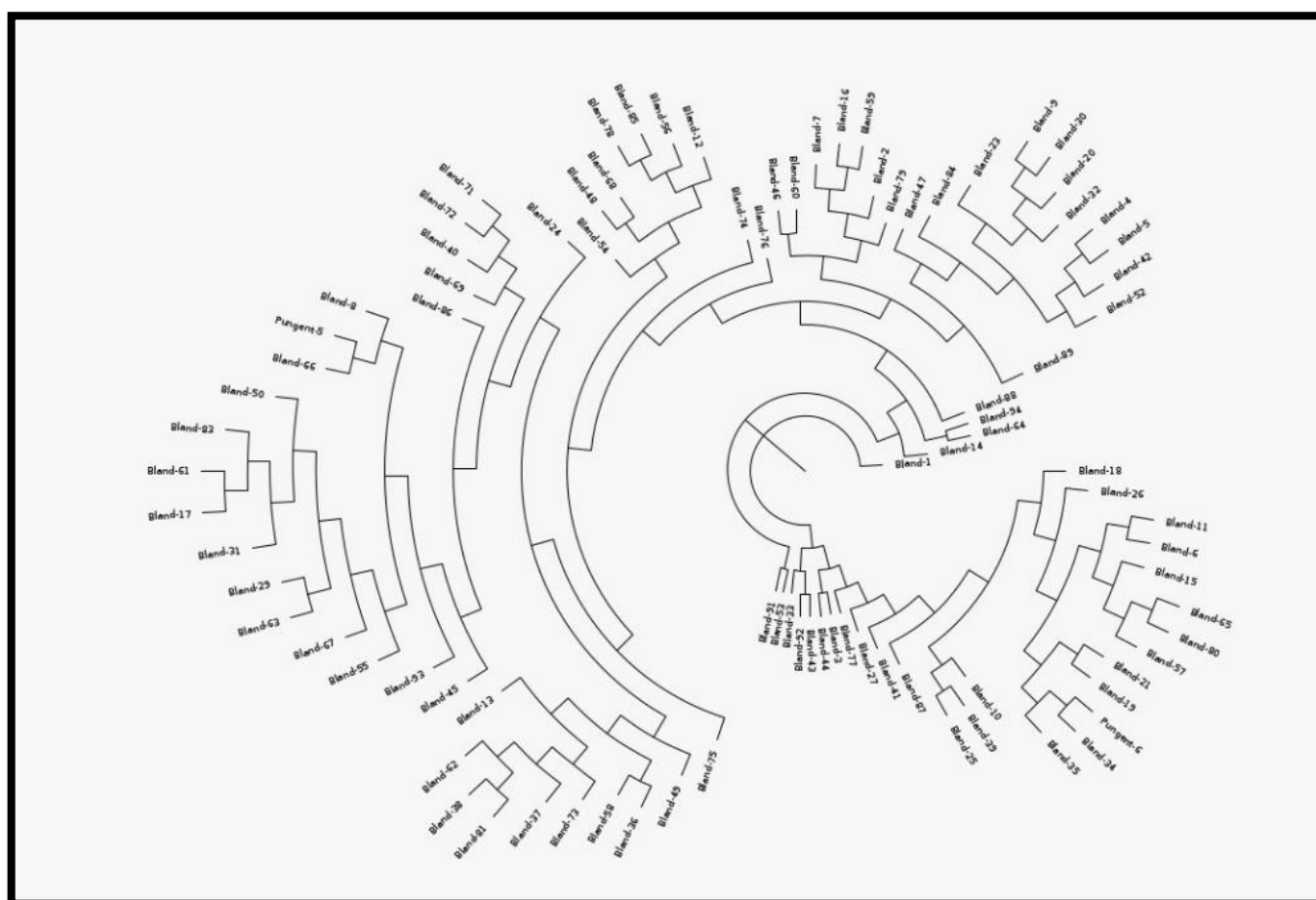


Fig. 7 - Árvore obtida através do figtree

Devido a esta dificuldade de distinção decidimos então criar uma árvore do tipo Bayesian mas desta vez aplicamos o método MrBayes.

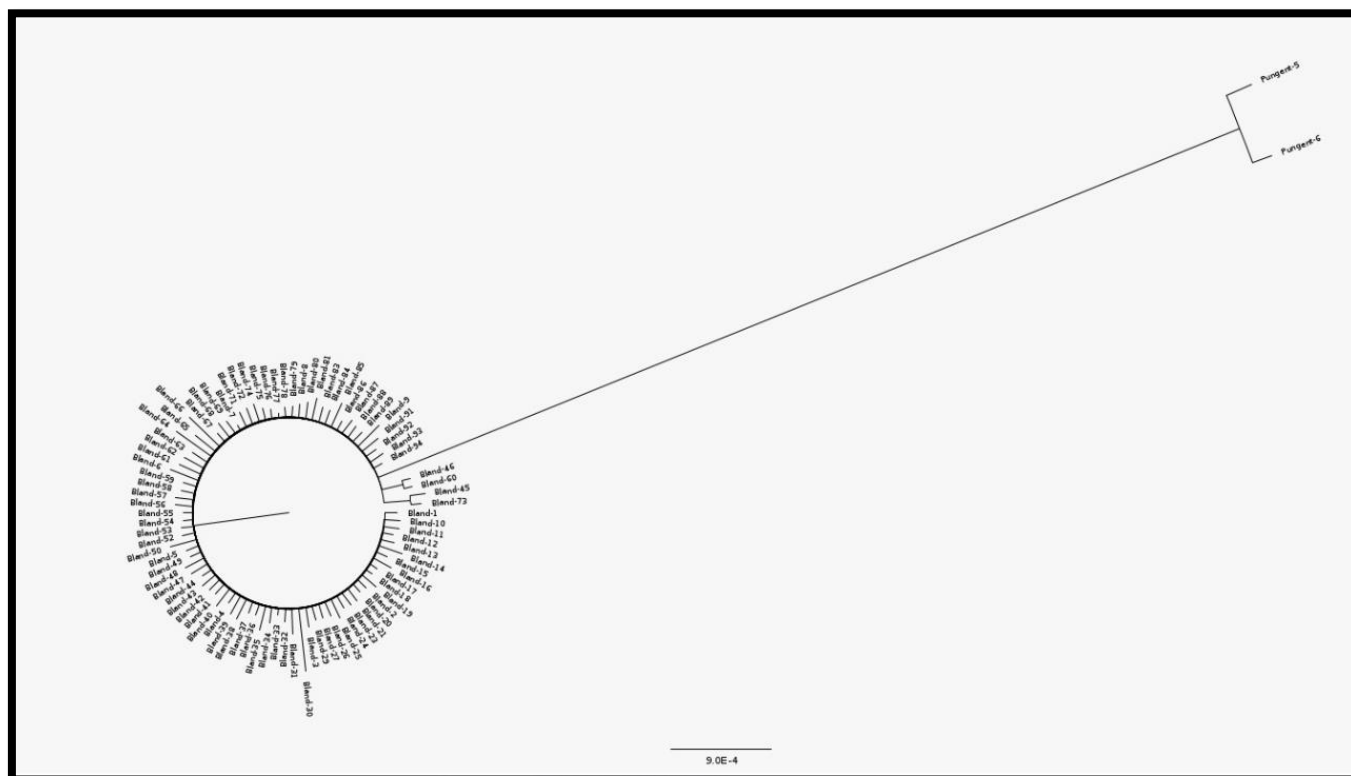


Fig. 8 - Árvore Bayesian com o método MrBayes

Nesta árvore, é possível verificar uma melhor visualização e distinção de quais os ramos pertencentes à espécie Pungent e quais os pertencentes à espécie Bland.

Como conclusão em resposta ao problema biológico, podemos verificar que após as dificuldades em distinguir quais os ramos que pertenciam à Bland e Pungent, conseguimos concluir que a espécie Bland obteve uma variedade genética muito mais elevada que a espécie Punget, pois como se pode observar na figura 2, a espécie Bland tem uma ramificação muito mais elevada em comparação com a espécie Pungent que tem uma ramificação muito reduzida.

Bibliografia

- <https://stuntspt.gitlab.io/asb2021/assignments/Assignment02.html>
- <https://cme.h-its.org/exelixis/resource/download/NewManual.pdf>
- https://github.com/NBISweden/MrBayes/blob/develop/doc/manual/Manual_MrBayes_v3.2.pdf
- <https://www.biostars.org/p/287346/>
- http://www.icsantosresearch.org/Class_2014_Spring_Comparative/pdf/week_6/Feb_12_2015_Chronograms.pdf
-