

Bioinformatisches Praktikum: Modul "Nukleinsäuren"

Montag (3.4.06)

Nach einer kurzen Einführung und Vorbesprechung bekamen wir die Gammaproteobacteria zugeteilt. Aus dieser Gruppe wählten wir 6 Bakterienarten aus:

Escherichia_coli_k12 (Esche): Das bekannte Darmbakterium E.coli

Salmonella_typhimurium_LT2 (Salmo): Ein gefährlicher Erreger für eine meist tödliche verlaufende Darminfektion.

Haemophilus_influenzae_LT2 (Haemo): Ein sehr bekannter Erreger für Sekundärsymptome bei Grippeerkrankung.

Shigella_flexneri_2a2457T (Shige): Die Shigellen, Bakterien, die auch zu schweren Erkrankungen führen können, vor allem bei Kindern und alten Menschen.

Legionella_pneumophiladelphia_1 (Legio): Der Auslöser der Legionärskrankheit.

Yersinia_pestis_KIM (Yersi): Der Erreger der Lungen- und Beulenpest.

Zu diesen Bakterien luden wir aus der NCBI die passenden .ptt-Dateien runter und wendeten das Programm cut_coding.pl darauf an, um die kodierenden Bereiche herauszuschneiden.

Dann stellten wir mit Hilfe der NCBI einen phylogenetischen Baum auf:

```
((Haemo)(Legio)(Esche)(Yersi)(Shige)(Salmo)))
```

und ließen das Programm all_bz laufen.

Dies dauerte die ganze Nacht.

Dienstag (4.4.06)

Zunächst wurden die Ergebnisse von all_bz weiterverarbeitet.

Mit dem Programm tba wurden die vielen Ergebnisse von all_bz align und zusammengefasst in eine tba.maf-Datei.

Die restliche Zeit des Vormittags verbrachten wir damit Perl zu lernen.

Nach mehreren Anläufen hatten wir dann ein Skript geschrieben, welches die tba.maf durchgeht und die einzelnen Blöcke der tba.maf-Datei in einzelne .fasta-Dateien weiterleitet. Die Namensgebung der .fasta-Dateien ist gamma_(Nummer des Blockes)_(Position der Sequenz).fasta.

Insgesamt erhielten wir 572 .fasta-Dateien.

Diese Dateien wurde mit Hilfe von Clustalw in das .aln-Format umgewandelt.

Die .aln-Dateien sollten dann mit RNAz weiterverarbeitet werden. Dies war aber auf Grund diverser Probleme nicht möglich.

Mittwoch (5.4.06)

Anstelle der vielen .aln-Dateien wendeten wir nun RNAz auf die tba.maf Datei an und experimentierten ein wenig mit den Einstellungen (mal rnazWindows vorher angewandt mal nicht) und erzeugten so mehrmals die rnaz.out Ausgabedatei.

Endgültig entschieden wir uns dann rnazWindows vorher anzuwenden und dann RNAz.

Dieses Ergebnis wurde in rnaz.out gespeichert.

Aus dieser Datei erzeugten wir dann mit Hilfe von rnazCluster eine HTML-Ansicht.

Auf Grund eines Fehlers bei rnazCluster konnte der letzte Locus nicht ausgewählt werden.

Donnerstag (6.4.06)

Wir wendeten rnazBlast auf unsere rnaz.out-Datei an und durchsuchten so die NONCODE-Datenbank. Das Ergebnis speicherten wir ab in resultsNoncode.fna. Aus dieser Datei erzeugten wir wieder mit rnazCluster eine HTML-Sicht.

Das selbe machten wir danach mit der Rfam-Datenbank.

Inzwischen wurde auch das Problem mit den letzten Locus gelöst, wenn auch nicht sehr schön. Wir erzeugten einfach einen neuen letzten "Dummy"-Locus, den wir dann manuell aus der results.html-Datei löschten.

Über die .fasta-Dateien ließen wir tRNAscan laufen und siebten mit Hilfe eines etwas umständlichen Shell-Skript alle leeren Dateien aus. Wir erhielten dadurch 58 .trna-Dateien.

Freitag (7.4.06)

Der Übersicht halber wurde noch mal eine neue rnaz.out erzeugt. Diese wurde nun zunächst mit der NONCODE-Datenbank abgeglichen und die daraus resultierende Datei mit der Rfam. So wurde eine results.fna Datei erzeugt, die die Einträge beider Datenbank enthält. Durch rnazCluster wurde eine HTML-Sicht dieses Ergebnisses generiert.

Von der NCBI luden wir die zu den .ptt-Dateien passenden .gbk-Dateien runter und wendeten das Programm getRNA auf diese Dateien an. Aus den Ergebnissen erzeugten wir wieder eine tba.maf Datei und eine rnat.out Datei. Diese Datei verglichen wir wieder mit den beiden Datenbanken (NONCODE, Rfam) und erzeugten eine HTML-Sicht.

Donnerstag (13.4.06)

Aufgrund eines Datenverlustes mussten die Ergebnisse vom Freitag noch mal erzeugt werden.

Die aln.-Dateien sind leider verloren.