



## 7. Übung

zur

### Vorlesung mit Übungen: Bioinformatik Wintersemester 2008/09

Bitte legen Sie vor der Übung ein Verzeichnis `Uebung07` an, in das sie *alle* Dateien dieser Übung speichern.

Im ersten Teil dieses Übungsblatts soll ein Problem ähnlich zur ersten Aufgabe des letzten Übungsblatts gelöst werden. Dabei war Ihnen vielleicht aufgefallen, dass die Suche in Sequenzdatenbanken in ganzen Genomen recht zeitaufwändig sein kann. Dies rührt unter anderem daher, dass den Dienst des NCBI sehr viele Menschen auf der ganzen Welt gleichzeitig nutzen möchten. Aufgrund der begrenzten Rechnerkapazität sind dadurch lange Wartezeiten vorprogrammiert, insbesondere bei Suchen auf großen Datenmengen, z.B. auf ganzen Genomen.

Abhilfe kann man hier schaffen, indem man die Suche auf seinem eigenen Rechner durchführt. Dazu sind allerdings einige vorbereitende Schritte notwendig.

Auf den Rechnern des Pools ist das Programm `blastall` vorinstalliert, mit dem Sie die Suche nach Sequenzen auf dem Rechner ausführen können, der direkt vor Ihnen steht. Der Installationspfad ist `/usr/local/bin/`. Erweitern Sie dazu bitte gegebenenfalls entsprechend die Umgebungsvariable `PATH` in der `.profile` bzw. `.bashrc` z.B. mit `export PATH="/usr/local/bin:${PATH}"` (näheres dazu in der Übungsgruppe), wodurch sie direkt `blastall` statt immer `/usr/local/bin/blastall` aufrufen können. Ferner muss jeder Benutzer BLAST noch mitteilen, in welchem Verzeichnis sich die allgemein für Berechnungen benötigten Daten (z.B. Alignment-Matrizen, Übergangsmatrizen) befinden. Diese Information müssen Sie in der Datei `.ncbirc` in Ihrem Heimatverzeichnis ablegen. Eine entsprechend vorbereitete Datei finden Sie in dem Verzeichnis

`~shenz/data/ncbirc`. Mit dem Befehl

```
cp /henz/data/ncbirc ~/.ncbirc
```

kopieren Sie diese an die gewünschte Stelle, so dass BLAST dann loslegen kann. Beachten Sie bitte den Punkt am Anfang des Dateinamens, bei der Kopie, die Sie anlegen! Dieser gehört zum Namen der Datei und wird von BLAST so erwartet. Zudem verhindert der Punkt, dass die Datei mit `ls` angezeigt wird, und sorgt so dafür, dass Ihr Heimatverzeichnis immer noch schön aufgeräumt erscheint.

#### Aufgabe 1 Vorbereitung der Sequenzdatenbank

Von NCBI können Sie die Sequenzen der auf Chromosom 1 von *Arabidopsis thaliana* kodierten Proteine als FASTA-Datei herunterladen. Sie können jeden Webbrowser benutzen, im folgenden beim Runterladen, wird dies jedoch konkret für den Firefox beschrieben. Zum Herunterladen wählen Sie aus der oberen Leiste direkt Genome aus. Dann können Sie links unter Eukaryota zu Chromosome navigieren. Klicken Sie dann auf *Arabidopsis thaliana* und folgen Sie dem `ftp download` und von da weiter zu `CHR_I`. (FTP steht für File Transfer Protocol, ein Netzwerkprotokoll zur Dateiübertragung). Sie erhalten eine Übersicht der zur Verfügung stehenden Dateien. Die Datei mit der Endung `.faa` ist die gesuchte FASTA-Datei. Speichern Sie diese als `Arabidopsis_chr1.fasta` im Verzeichnis `Uebung07`. (Dies erreichen Sie z.B., falls Sie den Firefox

benutzen, indem sie mit der rechten Maustaste auf die entsprechende Sequenz klicken und `Save Link Target As...` anklicken. Sie müssen dann nur noch das richtige Verzeichnis auswählen und den Namen `Arabidopsis_chr1.fasta` eintragen, unter dem Sie das File ablegen wollen.

Damit BLAST die Sequenzen der FASTA-Datei als Datenbank für seine Suche verwenden kann, müssen Indexdateien erzeugt werden. Das erledigt das Programm `formatdb` für Sie.

Wenn Sie `formatdb` mit der Option `--help` aufrufen, so wird eine Liste möglicher Kommandozeilenparameter angezeigt. Die für Sie relevante Option ist `-i`. Damit können Sie eine Sequenzdatei für die Vorbereitung angeben. Rufen Sie `formatdb` für Ihre FASTA-Datei auf. Das Kommando erzeugt keine Ausgabe auf der Konsole, stattdessen werden eine Reihe weiterer Dateien angelegt, die BLAST als Eingabe benötigt. Überzeugen Sie sich, dass `formatdb` wirklich etwas getan hat, indem Sie `ls` verwenden!

### **Aufgabe 2** BLAST-Suche auf dem *Arabidopsis*-Chromosom

Auf der Website zur Vorlesung finden Sie bei den Übungsblättern eine Datei mit Namen `unknown_u5_2.fasta`, die eine Proteinsequenz enthält. Für diese Sequenz sollen Sie herausfinden, ob es in *A. thaliana* auf Chromosom 1 stark homologe Sequenzen gibt. Laden Sie dazu die FASTA-Datei herunter und speichern Sie sie auf Ihrem Rechner.

Sie können nun eine BLAST-Suche durchführen, indem Sie das Programm `blastall` verwenden. Wichtige Kommandozeilenparameter werden Ihnen ebenfalls wieder mit `--help` angezeigt, oder indem Sie einfach nur `blastall` ohne Optionen und ohne Parameter aufrufen. Lassen Sie sich durch die Vielzahl an Parametern nicht verwirren! Relevant für Ihr Problem sind in erster Linie die folgenden Parameter:

- `-p` für die Auswahl des korrekten BLAST-Programms: Sie möchten eine Proteinsequenz in einer Protein-datenbank suchen!
- `-d` für die Auswahl der Datenbank, auf der Sie suchen möchten. Hier genügt der Name Ihrer FASTA-Datei.
- `-e` für die Angabe eines geeigneten E-Werts. Alignments mit schlechteren E-Werten als der angegebene Wert werden dann nicht mehr mit ausgegeben.
- `-i` für die Angabe der Sequenz, nach der Sie suchen.
- `-o` für die Datei, in die die Ausgabe von BLAST geschrieben werden soll.

Führen Sie nun BLAST mit geeigneten Parametern aus, und versuchen Sie das Ergebnis in der Ausgabedatei nachzuvollziehen. Um was für ein Protein handelt es sich bei dem unbekanntem Protein mit sehr großer Wahrscheinlichkeit?

*Hinweis:* Schauen Sie für die Wahl des korrekten BLAST-Programmes noch einmal in die Folien der letzten Vorlesung.

### **Aufgabe 3** Suche nach Nukleotidsequenzen

Die funktionelle Annotation des Reisgenoms ist noch nicht so weit fortgeschritten, wie die des *Arabidopsis*-Genoms. Die Datei `unknown_u5_3.fasta` von der Website der Vorlesung enthält die DNA-Sequenz eines Gens aus dem Reisgenom. Laden Sie die FASTA-Datei herunter, und versuchen Sie dem Gen eine Funktion zuzuordnen, indem Sie mit BLAST eine homologe Sequenz auf Chromosom 1 von *A. thaliana* suchen.

*Hinweis:* Achten Sie darauf, dass Sie das richtige BLAST-Programm verwenden.

### **Aufgabe 4** BLAST und BioPython

BioPython erlaubt es Ihnen, viele der Arbeitsschritte, die Sie in den vorhergehenden Aufgaben von Hand ausgeführt haben, insbesondere die Analyse der Ergebnisse, zu automatisieren. Das Python-Programm `annotate.py` versucht dies zu veranschaulichen.

Wenn Sie diesem Programm auf der Kommandozeile eine FASTA-Datei mitgeben, so ruft es BLAST (`blastp`) für die Datei auf und stellt fest, ob es Sequenzen gibt, die sehr stark homolog zur Anfragesequenz sind. Ist der E-Wert für die beste gefundene Sequenz kleiner als  $10^{-10}$ , so wird die Beschreibung der Sequenz ausgegeben.

Versuchen Sie die Funktionsweise des Programms nachzuvollziehen. Verwenden Sie dazu die Datei aus Aufgabe 2.

Modifizieren Sie anschließend das Programm so, dass es auch für weniger stark homologe Sequenzen (z.B. E-Wert  $< 10^{-5}$ ) eine Annotation ausgibt, den Benutzer allerdings warnt, dass diese Annotation unzuverlässiger ist, und zur Überprüfung das Alignment der beiden Sequenzen mit ausgibt. Die benötigten Informationen zur Benutzung des BLAST-Parsers in BioPython finden Sie auf

<http://biopython.org/DIST/docs/api/public/Bio.Blast-module.html>

unter Record/HSP.

### **Aufgabe 5** Hardcore-BioPython

Modifizieren Sie das Programm aus Aufgabe 4 derart, dass Sie eine FASTA-Datei mit mehreren Sequenzen automatisch annotieren können. Lesen Sie dazu die FASTA-Datei ein, schreiben Sie die Sequenzen daraus eine nach der anderen in eine neue FASTA-Datei und rufen Sie für diese jeweils `blastall` auf.

*Hinweis:* Beachten Sie, dass ein UNIX-Rechner Groß- und Kleinschreibung unterscheidet.

Vergessen Sie Ihre Unterschrift nicht, bevor Sie gehen. Danke.