

Softwarewerkzeuge der Bioinformatik

Prof. Dr. Volkhard Helms
PD Dr. Michael Hutter, Markus Hollander,
Marcial Josef Paszkiel
Wintersemester 2022/2023

Universität des Saarlandes
Zentrum für Bioinformatik

Übungsblatt 2

Sequenzanalyse: Paarweise Alignments

***Lernziel:** In dieser Übung werden Sie praktische Erfahrungen mit dem Needleman–Wunsch Algorithmus und verschiedenen BLAST–Suchen sammeln. Zusätzlich werden Sie sich mit Jupyter Notebooks vertraut machen, die Sie in den nächsten Übungen benutzen werden, um die Programmiersprache Python zu lernen.*

Aufgabe 2.1: Dynamisches Alignment

Führen Sie ein **globales** Alignment nach **Needleman–Wunsch** der Sequenzen **ACDEFAFGHI** und **KDELAFG** durch.

		A	C	D	E	F	A	F	G	H	I
K											
D											
E											
L											
A											
F											
G											

Globales Alignment:

Aufgabe 2.2: ProteinBLAST

Führen Sie eine **Protein-Protein BLAST** Suche (<http://blast.ncbi.nlm.nih.gov/Blast.cgi>) mit dem Protein **P00042** durch. Wählen Sie die **UniProtKB/Swiss-Prot** Datenbank aus, um nur non-redundante und hochwertige Resultate zu erhalten.

Unter "Algorithm parameters" können weitere Einstellungen vorgenommen werden, die Sie hier einfach übernehmen können. Standardmäßig verwendet ProteinBLAST eine E-value Schranke von 0.05 und BLOSUM62 als Matrix. Diese Einstellungen sind geeignet, um viele ähnliche und vermutlich homologe Proteine mit einem mittelmäßigen Verwandtschaftsgrad zu finden.

- a) Lassen Sie sich die Sequenzen der 10 Proteine mit der größten Homologie zu P00042 anzeigen.
- b) Wählen Sie wieder alle Resultate aus. Um was für Proteine handelt es sich?
- c) Für welche Organismengruppen wurden Treffer gefunden?

Aufgabe 2.3: MegaBLAST

Wählen Sie **Human** als Genom auf der BLAST-Startseite. Suchen Sie nach der mRNA **NM_175054** des menschlichen Gens *HIST4H4* mit **megaBLAST** in der Datenbank **Genome (GRCh38.p13)**.

- a) Auf welchem Chromosom liegt *HIST4H4*?
- b) Gibt es ein Paralog?
- c) Finde zwei bis drei direkt benachbarten Gene zu *HIST4H4*.

Aufgabe 2.4: PSI-BLAST

Benutzen Sie **PSI-BLAST**, um **viele weit entferne Homologe** des Proteins **Q57997** in der UniProt Datenbank zu finden. Setzen Sie die E-value Schranke auf 0.02 und die "PSI-BLAST Threshold" auf 0.001 um nach homologen Proteinen zu suchen, und wählen Sie entweder die BLOSUM45 oder die PAM250 Matrix aus, um viele weit entfernte Verwandtschaften zu finden.

- a) Was ist das Besondere an der 1. PSI-BLAST Iteration?
- b) Wie verändert sich das Ergebnis, wenn weitere Iterationen durchgeführt werden?

Aufgabe 2.5: Python and Jupyter Notebooks

Um Sie mit Python vertraut zu machen, werden wir **Jupyter Notebooks** benutzen. Ein Notebook besteht aus Zellen (*Cells*), die entweder ausführbaren Code oder Notizen enthalten, z.B. für Beschreibungen und Erklärungen. Notizen können mit *Markdown* formatiert werden. Jupyter Notebooks sind damit besonders geeignet, um Code und Analysen übersichtlich darzustellen.

- a) **Vorbereitung:** Damit Sie keine Software installieren müssen, werden wir die Seite [kaggle.com](https://www.kaggle.com) verwenden, wo Sie Notebooks erstellen und in der Cloud speichern und ausführen können.
 - i. Erstellen Sie einen Benutzeraccount auf [kaggle.com](https://www.kaggle.com). Sie können sich entweder mit einem Google-Account anmelden, oder mit ihrer Email-Adresse und einem Passwort. Im letzteren Fall bekommen Sie eine Bestätigungs-Mail an Ihre Adresse geschickt. Klicken Sie auf den Link in der Mail, um den Account zu aktivieren.
 - ii. Loggen Sie sich ein und erstellen Sie Ihr erstes Notebook. Auf der linken Seite der Website finden Sie die Option + **Create**. Klicken Sie darauf, und wählen Sie **New Notebook**. Neue Notebooks verwenden bereits standardmäßig Python als Programmiersprache. Sie können auf den Namen des Notebooks oben links klicken, um den Namen zu ändern. Wählen Sie einen Namen aus, den Sie wiedererkennen, wie z.B. SWW Tutorial 2.

- iii. Ihr Notebook ist nun in Ihrem Benutzeraccount gespeichert. Wenn Sie links auf `<>` oder `<> code` klicken, können Sie alle Notebooks unter **Your Work** finden.

Alternativ Sie können Jupyter Notebook (<https://jupyter.org/>) auch lokal auf Ihrem Rechner installieren, müssen dann aber selbständig die folgenden Schritte auf die veränderte Benutzeroberfläche umdenken.

- b) Ihr Notebook enthält bereits eine Zelle. Wählen Sie diese aus, und drücken Sie auf das Papierkorb-Symbol oben links, um ihren Inhalt zu löschen, damit wir von vorne anfangen können.
- c) Schreiben Sie $2 + 2$ in eine leere Zelle, und klicken Sie den Pfeil-Knopf links von der Zelle. Das Resultat wird anschließend unter der Zelle angezeigt.
- d) Erstellen Sie eine neue Code Zelle mit dem `+` Symbol, schreiben Sie `print('Hello World')` in die Zelle, und führen Sie die Zelle aus. Die Zeichenfolge *Hello World* sollte nun unter der Zelle zu sehen sein.
- e) Sie können Werte zu Variablen zuweisen. Diese Werte sind anschließend in der Umgebung des Notebooks gespeichert und können später wieder benutzt werden. Schreiben Sie $a = 5$ in eine leere Code Zelle, und führen Sie sie aus. Öffnen Sie eine neue Zelle, und schreiben Sie nur a . Dies wird Ihnen den Wert anzeigen, der in a gespeichert ist. Wenn die letzte Zeile einen Wert enthält oder zurückgibt, dann wird dieser Wert unter der Zelle angezeigt.
- f) Erstellen, editieren, starten und löschen Sie ein paar Zellen, bis Sie den Eindruck haben, dass Sie die Grundkonzepte von Notebooks verstanden haben.

Viel Spaß!