

# Praktikum Data-Mining und Sequenzanalyse

## Aufgaben zu Alignments

Marvin Meusel, Bertram Vogel

01.12.2014  
Abgabe 12.01.2015

Im Unterverzeichnis `Matrices` findet Ihr Scoring-Matrizen für Proteine. Schreibt Eure Evaluatorklasse so, dass sie diese Matrizen einlesen kann. Die Matrizen enthalten keine Insertions/Deletionskosten, diese müssen zusätzlich festgelegt werden. Diese Kosten bzw. Ähnlichkeiten sollen für alle Aminosäuren gleich sein. (Was ist dabei üblich?)

Aufgaben:

- Erstellt ein Klassendiagramm (UML) für das Projekt
- Erläutert eure Implementierung. Gab es Schwierigkeiten? Gibt es noch Fehler?
- Gebt das Testsystem an

Projektbezogene Aufgaben:

1. Vergleicht die naiven Algorithmen anhand selbst gewählter, kleiner Beispiele mit den effizienten Algorithmen. Wie grosse Instanzen könnt Ihr in vertretbarer Zeit (z.B. einige Minuten) mit den verschiedenen Algorithmen behandeln?
2. Vergleicht die Proteine im Unterverzeichnis `RiboProtS1` per globalem und lokalem Alignment. Benutzt zwei unterschiedliche Matrizen. Welche Proteine sind am ähnlichsten? Welche Schlussfolgerungen zieht ihr aus dem Vergleich?
3. Vergleicht die Laufzeit und den Speicherplatzverbrauch des Hirschberg- und des Needleman-Wunsch-Algorithmus beim globalen Alignment dieser Proteine mit und ohne Traceback. Wie könnt Ihr Angaben zum Speicherplatzbedarf machen, d.h. diesen messen oder berechnen? Falls die Proteine zu kurz für einen aussagekräftigen Vergleich sind, verlängert sie, indem ihr sie (mehrfach) an sich selber anhängt. Ab welcher Sequenzlänge lässt sich das Alignment aufgrund von Speicherproblemen nicht mehr berechnen?
4. Inwieweit eignen sich die Algorithmen für die inexakte Suche in grossen Genomen? Gibt es Alternativen?

Hinweis: Gebt im Protokoll bitte nur die Bewertung (Score) der Alignments an. Mit dem Argument `-Xmx<Zahl>m` weist Ihr der Virtual Machine `<Zahl> MB` Speicher zu. Das Argument kann in der Konsole dem `java` Command übergeben werden, oder in eurer IDE in der `run configuration` o.ä. eingetragen werden.