Rechnerübungen zur Vorlesung "Algorithmen der Bioinformatik I"

Veranstalter: Peter Meinicke, Heiner Klingenberg, Burkhard Morgenstern

Hidden Markov Modell (HMM) zum unehrlichen Casino

Implementieren Sie den Viterbi-Algorithmus für das HMM zu dem Beispiel des unehrlichen Casinos, wie in der Vorlesung vorgestellt (siehe auch im Buch von Durbin et al. "Biological sequence analysis" Seite 54-57). Stimmt der von Ihnen berechnete Viterbi-Pfad zu der Beispielsequenz über 300 Würfe aus der Vorlesung (Abbildung 3.5 aus dem Durbin-Buch, Datei: `Casino.txt`) mit dem Viterbi-Pfad der Abbildung überein? Welche Möglichkeiten gäbe es, die Unsicherheit des Modells an bestimmten Stellen der Sequenz weiter zu untersuchen? Skizzieren Sie einen allgemeinen Algorithmus zur Markierung von Sequenzpositionen, an denen das Modell bezüglich der Rekonstruktion ("Vorhersage") der Zustände möglicherweise unzuverlässig ist. Was müssten Sie bei der Implementation insbesondere beachten, um numerisch sinnvolle Ergebnisse zu erhalten?

Profil-HMM zur Erkennung ribosomaler RNA-Sequenzen

Schätzen der Modellparameter

Zum Einlesen des multiplen Sequenzalignments (MSA) der ribosomalen RNA (rRNA) Trainingsequenzen (Datei: `LSU_train.fasta`) können Sie sich an dem C-Beispiel (im gleichen `Daten` Verzeichnis) orientieren oder dieses direkt in ihrem Programm verwenden.
Beim Bestimmen der Modellstruktur (Anzahl der Match-Zustände) wenden Sie die in der Vorlesung besprochene 50% Regel an: wenn in einer Spalte des MSA weniger als 50% der Sequenzen Gaps aufweisen, dann wird diese Spalte einem Match-Zustand zugeordnet. Danach können Sie die Emissionswahrscheinlichkeiten und Übergangswahrscheinlichkeiten schätzen. Verwenden Sie zum Schätzen aller Wahrscheinlichkeiten einen globalen Pseudocount-Parameter $r = 1$ (Laplace-Regel). Lassen sie sich für die Match-Zustände der Trainingssequenzen die Emissionswahrscheinlichkeiten ausgeben. Gibt es Regionen auf der 23S rRNA mit relativ geringer Entropie, also Regionen in denen bestimmte Buchstaben (Basen) deutlich häufiger vorkommen als andere? Welche Bedeutung haben solche Regionen aus biologischer und statistischer Sicht?

Anwendung auf Testsequenzen

Implementieren Sie den Viterbi-Algorithmus, um eine neue Sequenz beliebiger Länge mit dem Modell abzugleichen. Der logarithmische Score zur maximalen Verbundwahrscheinlichkeit aus Sequenz und dem maximierenden Viterbi-Pfad soll dabei als Zugehörigkeitsmaß dienen, um zu entscheiden, ob es sich um eine rRNA-Sequenz handelt.
Unter den Testsequenzen (Datei: `LSU_full_test.fasta`) befinden sich sowohl rRNA-Beispiele mit voller Länge als auch Negativ-Beispiele ("Non-rRNA"). Zusätzlich gibt es noch Fragment-Beispiele von rRNA und Non-rRNA Sequenzen (Datei: `LSU_short_test.fasta`).
Bestimmen Sie wie bei den PWMs (Blatt 1) die Score-Schwellwerte für die Erkennung von rRNA so, dass ungefähr 10%,20%,30%,...,90% der rRNA-Testsequenzen erkannt werden. Wie hoch ist jeweils der Anteil der Non-rRNA Beispiele unter den Sequenzen über dem Schwellwert ("False Discovery Rate")? Stellen Sie die Anteile in einer Tabelle oder Grafik dar.
Spielt die Sequenzlänge der Testsequenzen eine Rolle für das logarithmische Scoring? Skizzieren sie ein mögliches längenunabhängiges Scoring-Schema! Worin besteht darüberhinaus die besondere Problematik der kürzeren Testsequenzen? Schauen Sie sich an, wie der Viterbi-Algorithmus kurze Testsequenzen mit dem Modell abgleicht ("Alignment"). Versuchen Sie eine Spezialversion des Modells oder der Methode zu implementieren, die ein besseres Sequenz-Modell-Alignment für die Sequenzfragmente liefert.

Bitte den Sourcecode und die Dokumentation zu den Lösungen als `zip`,`tar.gz` oder `tar.bz2` komprimierte Ordner bis zum **31.01.18** abgeben. Es besteht die Möglichkeit, 2er-Teams zu bilden. Der Name des Verzeichnisses hat Ihren Namen und evtl. den Namen Ihres Team-Partners zu enthalten. Dabei ist folgendes Format für den Ordnernamen einzuhalten:
`name1_vorname1-name2_vorname2`

Fertigen Sie ein schriftliches Protokoll an, das Sie als separate Datei beilegen und in dem Sie die Fragen aus den Übungsaufgaben ausführlich beantworten. Insbesondere halten Sie dort auch technische Probleme fest, die Sie bei der Implementation oder Evaluation hatten, weswegen die Lösung möglicherweise noch nicht stimmig ist.
Im Abgabeordner sollte sich neben dem Quelltext und dem Protokoll auch eine Installations-/Bedienungsanleitung befinden. Graphiken können entweder im Protokoll eingebettet oder als einzelne Dateien vorliegen. Zulässige Formate für das Protokoll sind `pdf`,`html`. Markdown und plain-text sind ebenfalls möglich.
Zur Abgabe den komprimierten Ordner in das entsprechende Verzeichnis [Abgabe - Übungsaufgaben (Meinicke)] im Stud.IP hochladen.
Beachten Sie, dass die Abgabe nach Fristende nicht mehr möglich ist. Auch kann die hochgeladene Lösung nachträglich nicht mehr bearbeitet werden.
Zulässige Programmiersprachen bei der Implementierung sind C, Java, Python, Perl, R oder Matlab/Octave. Verwenden Sie keinen Code und keine Funktionen

aus existierenden Bioinformatik-Bibliotheken/Toolboxen und geben Sie die verwendete Programmiersprache im Protokoll an! Achten Sie darauf, dass Ihr Code für andere lesbar und verständlich ist, d.h. verwenden Sie sprechende Namen für die Variablendefinitionen anstelle von kryptischen Bezeichnungen, verzichten Sie auf implizite Variablen (hartkodierte Werte ohne Variable) und machen Sie Gebrauch von Kommentaren!
Informationen zum biologischen Hintergrund der rRNA-Erkennung und der Bedeutung für die sogenannte Metatranskriptomanalyse finden Sie z.B. in der Publikation: Tripp et al. "Misannotations of rRNA can now generate 90% false positive protein matches in metatranscriptomic studies" Nucleic Acids Res. 2011; 39(20): 8792-8802; `http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3203614/`.