Rechnerübungen zur Vorlesung "Algorithmen der Bioinformatik I"

Veranstalter: Peter Meinicke, Heiner Klingenberg, Sebastian Gross, Burkhard Morgenstern

Position Weight Matrices (PWM) zur Erkennung von Startcodons

Implementieren Sie die Identifikation von Startcodons, wie in der Vorlesung vorgestellt, mittels Positionsgewichtsmatrizen. Das PWM-Modell soll dabei eine Region von L Basen vor dem Startcodon, eine sog. Translation Initiation Site (TIS) repräsentieren. Die Sequenzen zum Schätzen der Modellparameter (Emissions-Wahrscheinlichkeiten) finden Sie in der Textdatei `TIS-Ecoli.txt`.
Jede Zeile entspricht einem Sequenzausschnitt der Länge 200 bei dem sich der Beginn des Startcodons in der Mitte an Position 101 befindet. Mögliche Startcodons sind ATG,GTG,TTG. Wie viele mögliche Kandidaten für die verschiedenen Startcodon-Varianten gibt es insgesamt? Zählen Sie dabei nicht nur die tatsächlichen "richtigen" Startcodons in der Mitte sondern auch alle "falschen" Kandidaten im kodierenden (rechte Seite) und nicht-kodierenden Bereich (linke Seite).
Verwenden Sie zunächst eine PWM-Länge von $L = 30$ und als Hintergrund-Modell die Gleichverteilung, d.h. die Hintergrund-Wahrscheinlichkeiten sind 0.25 für alle Basen an allen Positionen. Arbeiten Sie beim Schätzen der Wahrscheinlichkeiten mit einem Pseudo-Count von $r = 1$.
Versuchen Sie nun einen geeigneten Detektionsschwellwert für den logarithmischen Score zu finden. Der Score wird dabei generell nur für gültige Startcodon Kandidaten (s.o.), vor denen mindestens L Basen beobachtbar sind, berechnet. Wählen Sie die Schwelle t zunächst so, dass 50% der richtigen Kandidaten erkannt werden. Wie hoch ist der Fehler, d.h. wie viele der falschen Kandidaten detektieren Sie damit zwangsläufig auch?
Unterteilen Sie nun die Daten in eine Trainings- und eine Test-Menge. Verwenden Sie die ersten 400 Sequenzen zum Schätzen der Parameter und zum Ermitteln der Detektionsschwelle gemäß dem 50% Kriterium. Verwenden Sie den Rest der Sequenzen, um die Erkennungsrate zu ermitteln. Wie hoch ist hier der Fehler (falsche Kandidaten über der Schwelle)? Fertigen Sie eine Tabelle oder Grafik an, in der Sie die Anzahl der Fehler gegen den Anteil der richtig erkannten Kandidaten in 10% Schritten (von 10% bis 90%) abtragen.
Versuchen Sie die Erkennnungsrate auf den Testdaten durch verschiedene Maßnahmen zu verbessern und überprüfen Sie bei allen Maßnahmen auch die Fehlerquote. Erweitern Sie das PWM-Fenster um eine Position in den kodierenden Bereich d.h. inklusive der ersten Base des Startcodons. Erhöht sich damit die Trefferquote bei einem vergleichbarem Fehler? Was könnte man mit Blick auf die Startcodon-Varianten als Nachteil dieser Erweiterung sehen?

Probieren Sie auch alternative Werte für den Pseudo-Count Parameter. Führen hier höhere oder niedrigere Werte als $r = 1$ eventuell zu einer Verbesserung? Welche Möglichkeiten gibt es, das Hintergrund-Modell zu verbessern? Untersuchen Sie eine der Möglichkeiten. Fertigen Sie für das unter Umständen verbesserte Modell die obige Grafik (oder Tabelle) an. Wie könnte über die hier untersuchten Möglichkeiten hinausgehend eine generelle Verfeinerung oder Erweiterung des PWM-Modells aussehen?

Bitte den Sourcecode und die Dokumentation zu den Lösungen als `zip`,`tar.gz` oder `tar.bz2` komprimierte Ordner bis zum **30.11.17** 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 speziellen 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!