

Übungen zum Sequenzanalyse-Praktikum

Universität Bielefeld, SoSe 2016

Dr. Roland Wittler · M.Sc. Linda Sundermann

<http://wiki.techfak.uni-bielefeld.de/gi/Teaching/2016summer/SequaPrak>

praktikum-seqan@CeBiTec.Uni-Bielefeld.DE

Übungsblatt 2 vom 19./20.04.2016

Abgabe bis Sonntag bzw. Montag, 24:00 Uhr.

Aufgabe 1 (Kompression mit bzip2)

Erstelle fünf Textdateien mit je 100.000 Zeichen, die folgendermaßen aufgebaut sein sollen:

1. Zufallszeichen aus einem 4-Buchstabenalphabet,
2. DNA-Sequenz,
3. Protein-Sequenzen,
4. natürlichsprachlicher deutscher Text,
5. natürlichsprachlicher Text einer anderen Sprache deiner Wahl.

Komprimiere diese Dateien mit `bzip2`. Beschreibe in deinem Protokoll kurz, wie du die Dateien erstellt hast und gib an, welche Sequenzen bzw. Texte als Grundlage gedient haben. Gib ggf. Quellen an. Kontrolliere, ob alle fünf Dateien vor der Komprimierung die gleiche Größe haben. Vergleiche dann deine Beobachtungen in einer Tabelle und versuche sie zu erklären.

Aufgabe 2 (Bowtie 2)

Im Folgenden verwenden wir den Read-Aligner Bowtie 2. Das Tool ist auf dem CeBiTec-System unter `/vol/biotools/lib/bowtie2-2.2.7` installiert – hier findest du auch ein Manual und einen Ordner `examples`. Wie alle rechenintensive Befehle, sollte auch Bowtie nur auf einem Compute Cluster, also von einem `qxterm` aus aufgerufen werden.

1. Verschaffe dir im Manual einen Überblick über das Tool. Du musst nicht alles im Detail lesen.
2. Im Ordner `examples` findest du einige Beispieldateien. Da du in diesem Ordner keine Schreibrechte hast, solltest du ihn dir in dein Home-Verzeichnis kopieren. Führe die nachfolgenden Schritte mit der Referenzsequenz `lambda_virus.fa` aus.
 - (a) Erstelle den Index für die Referenz mit dem Programm `bowtie2-build`.
 - (b) Aligniere die Reads in der Datei `reads_1.fq` als ungepaarte Reads zur Referenzsequenz mit dem Programm `bowtie2` und speichere die Ergebnisse im SAM-Format ab (du musst hier nicht wissen, wie genau dieses Format aufgebaut ist). Schaue dir das Ergebnis an. Bowtie gibt zusätzlich eine kleine Statistik auf der Konsole aus. Wie viele Reads konnten erfolgreich aligniert werden?
 - (c) Als nächstes kommt ein Beispiel für das paired-end Alignment. Die Readpaare sind aufgeteilt in die Dateien `reads_1.fq` und `reads_2.fq`. Rufe `bowtie2` mit beiden Dateien auf und speichere das Ergebnis wieder im SAM-Format. Vergleiche kurz die Statistik zum vorherigen Aufruf. Was bedeuten die Begriffe “concordant” und “discordant”?
 - (d) Als letztes wollen wir Variationen in den Reads im Vergleich zur Referenzsequenz finden (SNPs, kurze Insertionen und Deletionen). Dazu verwenden wir `samtools` und `bcftools`. Beide Tools sind im Ce-BiTec installiert und sollten ebenfalls nur auf dem Compute Cluster aufgerufen werden. Beachte, dass `samtools` nur für Linux installiert ist und dementsprechend nur von einem `lqxterm` aus aufgerufen werden kann.

Nutze `samtools view` und `samtools sort`, um das Ergebnis deines paired-end Alignments in ein sortiertes BAM-file zu konvertieren (BAM ist die komprimierte Version von SAM).

Um die Varianten zu finden, führe `samtools mpileup -uf <referenz>.fa <sortiertes bam> | bcftools call -vc -Ob > <ausgabename>.raw.bcf` aus.

Du kannst dir das Ergebnis mit `bcftools view` anschauen. Wie viele Varianten enthält deine Ausgabe?

Beschreibe in deinem Protokoll die einzelnen Programmaufrufe und was sie bewirken. Du musst deine Ergebnisdateien nicht ins Protokoll schreiben, solltest aber alle Fragen beantworten.