Praktikum Sequenzanalyse und Genomik WS20

1 Einführung

1.1 Grundsätzliches

Dieser Text ist als Aufgabenstellung und nicht als Anleitung gedacht. Um herauszfinden, wie die einzelnen Programme funktionieren, verwenden sie bitte die Manuals (on-line oder mit "-h" oder "–help" aufzufinden).
Bitte zögern Sie auch nicht, die anwesenden Betreuerinnen und Betreuer zu fragen, wenn etwas unklar ist oder wenn Sie Probleme haben. Bitte tun Sie das aber wirklich erst, nachdem Sie versucht haben, ihr Problem mit

- Den (on- oder off-line) Hilfen
- Google (oder einer anderen Suchmaschiene)

zu lösen.
Vor allem, wenn Sie eine Analys zum ersten Mal durchführen, **müssen** Sie ihre (Zwischen-)**Ergebnisse** sofort **überprüfen**, um etwaige Fehler nicht in weitere Analyseschritte mitzuschleiffen.

1.1.1 Deteitypen

Bei NGS Datenanalysen werden sie vor allem mit drei Typen von Dateien konfrontiert:

- Sequenzdateien
  - fasta, fastq
- Regionsdateien
  - bed, bedgraph, gff
- Alignmentfiles (enthalten meist Sequenz und Regioninformation)
  - sam, bam

Zur besseren Durchsuchbarkeit können manche dieser Files indiziert werden. Es empfiehlt sich auch, die Dateien durch eine Eindeutige Endung (.fa, .fq, .bed, .bg, .gff, .sam, .bam) zu kennzeichnen, auch wenn das bei Linuxsystemen meist nicht notwendig ist. Im Anhang befindet sich eine kurze Zusammenfassung der einzelnen Dateitypen. Bitte konsultieren Sie aber auch das Internet, um genaue Informationen z.B. zum exakten Aufbau der Dateien zu erhalten.

1.2 Wissenschaftliche Fragestellung

In dem Archaeum *Haloferax volcanii* sind einige wenige RNAs bekannt, die gespleisst werden. Zur Untersuchung diese Spleissens und zur Identifizierung von neuen Spleisskandidaten wurde ein Knock out mit Hile von CRISPR CAS-9 durchgeführt. Wir wollen die entstandenen Datensätze aber vor allem nach möglichen nicht annotierten Transkripten durchsuchen, die Kodingpotential haben, d..h translatiert werden könnten. Diese Transkripte können lang sein oder aber sehr kurze Peptide sein. Es könnte sich auch um kurze Peptide in UTR-regionen von bekannten Genen handeln.

1.3 Daten

Wir haben zwei RNAseq Datensätze mit 3 jeweils Replikaten:

- *Haloferax volcanii* Wildtyp single end sequencing
- *Haloferax volcanii* CRISPR - KO single end sequencing

1.4 Aufgabenstellungen

- Entfernen der Adaptersequenzen
- Qualitätskontrolle
- Mapping auf das Genom
- Quantifizieren der Gene
- Analysieren der differentiellen Genexpression
- Generierung von Sequenziertiefen
- Analyse des Codingpotentials von nicht annotierten exprimierten Bereichen

1.5 Installieren von Programmen: conda

Wir werden conda verwenden, um Programme zu installieren, wo auch immer das möglich ist. Die wichtigsten Befehle dazu sind:

- conda create -n IHREUMGEBUNG IHREPROGRAMME

  Generiert eine neue conda Umgebung namens IHREUMGEBUNG

  In dieser Umgebung sind IHREPROGRAMME installiert

- conda activate IHREUMGEBUNG

  Aktiviert IHREUMGEBUNG, IHREPROGRAMME können jetzt verwendet werden

2 Arbeitsaufträge

2.1 Entfernen der Adaptersequenzen

2.1.1 Hintergrund

Im Rahmen der Probenvorbereitung werden für das next generation sequencing Platformspezifische Adaptersequenzen (genauer: Terminal-, Adapter-, Index- und Primer Binding Sequenzen, siehe Abb.) an alle Nukleinsäurefragmente ligiert. Diese Adaptersequenzen sind im Genom (hoffentlich) nicht enthalten. Deswegen ist es meist nicht möglich, Reads, die Adaptersequenzen enthalten, auf das Genom zu mappen. Die Adapter am Beginn der Sequenz (5' Adapter) werden normalerweise schon vom base-caller entfernt. Die Adapter am Ende der Sequenz können so allerdings nicht entfernt werden. Daher müssen diese Adaptersequenzen vor den weiteren Analysen weggeschnitten (getrimmt) werden.

2.1.2 Benötigte Programme

- trim-galore (cutadapt)

  Installation:

  ```
  conda create -n IHREUMGEBUNG trim-galore
  ```

trim-galore erkennt automatisch, welche adaptersequenz verwendet wurde.

2.1.3 Benötigte Dateien

Input:

- originale fastq files (6x)

Output:

- gzippte getrimmte fastq files (6x)

2.1.4 Zu beantwortende Fragen

- Wieviele Reads gibt es (pro Replikat)
- Wieviele Reads enthalten adapter Sequenzen (pro Replikat)
- Wieviele Reads bleiben nach dem trimmen übrig (pro Replikat)

2.2 Qualitätskontrolle

Vor der weiteren Analyse sollte die qualität der Sequenzierung überprüft werden. Schwere Qualitätsprobleme wie Verunreinigungen, Sequenzbias und Sequenzierfehler könnten zusätzliche Schritte in der Nachfolgenen Analyse nötig machen oder sogar die Sinnhaftigkeit der weiteren Analysen in Frage Stellen. Für die Qualitaetskontrolle werden deswegen einige wichtige Parameter erhoben.

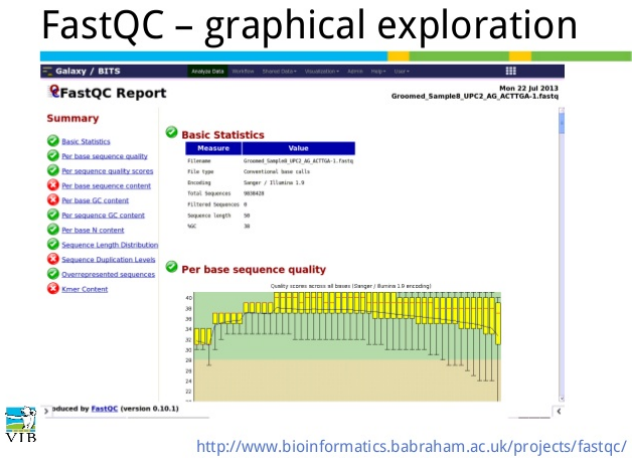

Abbildung 1: Erste Seite des fastqc Qualitätskontroll Outputs. Die Zusammenfassung links zeigt in rot und gelb zu untersuchende Auffälligkeiten an. Taken from slideshare

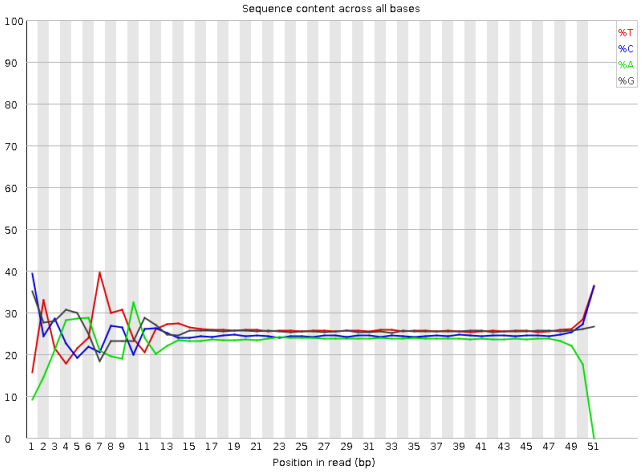

Abbildung 2: Standard per sequence quality Ergebnis einer RNAsequenzierung. Der starke Sequenzbias am 5' Ende ist normal, und wahrscheinlich auf die verwendeten Reagenzien zurückzuführen. Besorgniserregend ist es z.B., wenn sich 2 Experimente, die zusammen verwendet werden sollen, am 5' Ende stark unterscheiden.

2.2.1 Benötigte Programme

- fastqc

  Installation:

  conda create -n IHREUMGEBUNG fastqc

2.2.2 Benötigte Dateien

Input:

- gzippte getrimmte fastq files (6x)

Output:

- fastqc output files (.html, Bilder etc (6x))

2.2.3 Zu beantwortende Fragen

- Was ist die durchschnittliche Readlänge?
- Gibt es einen Sequenzbias?
- Variieren Readlängen oder Sequenzbias zwischen den Datensätzen?
- Git es angereicherte Sequenzen?
- Wo kommen diese Sequenzen her (NCBI BLAST)?

2.3 Mapping

2.3.1 Hintergrund

Als Mapping wird das Alignieren der Sequenzreads auf das Referenzgenom verstanden. In RNA Sequencing Experimenten dient es dazu, festzustellen, welche Gene expremiert werden. Da wir es in RNAseq Experimenten mit Spleissen zu tun bekommen, muss ein mappingprogramm verwendet werden, dass mit gespleissten Reads zurechtkommt. Wir verwenden dazu segemehl. Samtools ist eine Zusammenstellung von Programmen zur Bearbeitung von Alignement files (sortieren, ansehen, ausschneiden indizieren etc.)

2.3.2 Benötigte Programme

- segemehl, samtools

  Installation:

  conda create -n IHREUMGEBUNG segemehl samtools

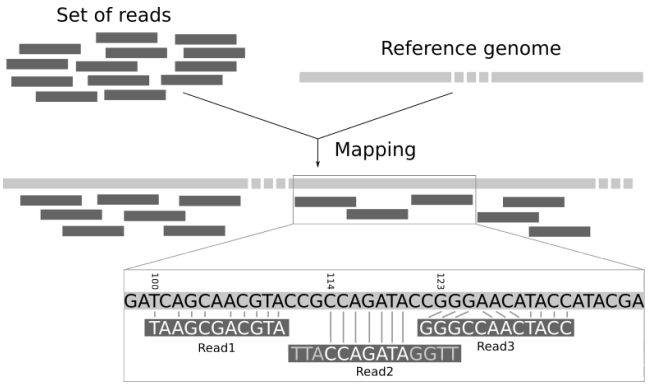

Abbildung 3: Illustration des Sequenz mappings. Taken from The Galaxy Project.

2.3.3 Benötigte Dateien

Input:

- gzippte getrimmte fastq files (6x)
- Genom fasta file
- Segemehl index file des genoms (mit segemehl erstellbar)

Output:

- (sortierte) Alignment files (bam format)
- Index files dieser Alignment files (bai format)

2.3.4 Zu beantwortende Fragen

- Wieviele Reads konnten gemappt werden?
- Wieviele Reads wurden eindeutig zugeordnet?
- Wieviele Reads sind gespleisst gemappt worden?

2.4 Genquantifizierung

2.4.1 Hintergrund

Nach dem Mappen müssen die Gene quanitfiziert werden. Dabei müssen einige Dinge beachtet/entschieden werden:

- Ist die Sequenzierung strangspezifisch? Wenn ja: welche Richtung?
- Behandlung von Multimappern

- Behandlung von gespleissten Reads
- Behandlung von etwaigen intronischen Reads

Strangspezifische Sequenzierung kann entweder immer auf dem richtigen oder immer auf dem Gegenstrang erfolgen, während nicht strangspezifische Sequenzierung auf beiden Strängen mit ca. gleicher Wahrscheinlichkeit erfolgt. Um festzustellen, welche Art von Sequenzierung man vor sich hat, kann man nach allen 3 Arten zählen und dann die Ergebnisse vergleichen.
Multimapper werden oft ignoriert, können aber auch fraktional gezählt werden. Gespleisste Reads sollten für jedes Gen, dass sie treffen einmal gezählt werden, während (rein) intronische Reads normalerweise nicht zur Genexpression gezählt werden sollten.

2.4.2 Benötigte Programme

- featureCounts

  Installation:

  conda create -n IHREUMGEBUNG subread

2.4.3 Benötigte Dateien

Input:

- bam und bai files aus dem mapping
- annotations file (gff3 format)

Output:

- featurecounts output file mit counts und normalisierten counts.

2.4.4 Zu beantwortende Fragen

- Welche art von Strangspezifizität haben wir?
- Ist diese in allen Experimenten gleich?
- Wieviele Reads konnten in den Experimenten zugeordnet werden?

2.5 Differentielle Genexpression

2.5.1 Hintergrund

Für differentielle Expressionsanalysen gibt es verschiedene Statistische herangehensweisen. Wir verwenden hier DESeq2, ein weitverbreitetes Programm, das auf negativer Binomialverteilung basiert. DESeq2 benötigt als input eine Matritze mit Read Counts, da die Normalisierung von DESeq2 durchgeführt wird.

2.5.2 Benötigte Programme

- R, DESeq2

  Installation:

  conda create -n IHREUMGEBUNG R

  In R:

  if (!requireNamespace("BiocManager", quietly = TRUE))
  install.packages("BiocManager")
  BiocManager::install("DESeq2")

2.5.3 Benötigte Dateien

Input:

- Matrix mit gene counts

Output:

- DESEQ2 output file

2.5.4 Zu beantwortende Fragen

- Welches Gen wurde mittels CRISPR ausgenockt?
- Was sind die 10/20 am stärksten differentiellen Gene?

2.6 Coverage Berechnungen

2.6.1 Hintergrund

Wir wollen sehen, welche nicht annotierten Bereiche des *Haloferax* Genoms zwar expremiert, aber (noch) nicht annotiert sind. Auch kann man mittels der coverage gut die Expremierung visualisieren. Wichtig ist eine Normierung, um die vergleichbarkeit der Ergebnisse zu gewährliesten.

2.6.2 Benötigte Programme

- bedtools (genomecov)
- wiggletools

  Installation:

  conda create -n IHREUMGEBUNG bedtools wiggletools

2.6.3 Benötigte Dateien

Input:

- originale bam files (6x)
- Genome chromosome size file

Output:

- bedgraph files (6x)
- mean bedgraph files (4x)

2.7 Coding Potential von nicht annotierten expremierten Bereichen

2.7.1 Hintergrund

In vielen auch gut annotierten Organismen findet man transkribierte Bereiche, für die es keine Annotation gibt. Diese Bereiche können UTR Bereiche oder unbekannte Isoformen eines benachbarten Gens sein. Eine andere Möglichkeit ist, dass es sich um transkriptionelles Rauschen, also um quasi sinnlose Transkription handelt. Die unbekannten transkribierten bereiche können aber auch zu einem unbekannten Gen gehören. Um zwischen Genen und Rauschen zu unterscheiden, kann man verschiedene Informationsquellen nutzen, wie zB differentielle Expression, Gewebespezifische Expression und evolutionäre Konservierung. ~~Wir werden hier Untersuchen, ob die unbekannten transkribierten Bereiche für Peptide oder Proteine codieren könnten, indem wir anhand von evolutionären Substitutionsmustern das sogenannte Coding Potential untersuchen. Dazu benötigen wir gute Kandidatensequenzen sowie Alignements dieser Kandidaten mit verwandten Sequenzen. Die Kandidaten können wir finden, indem wir zunächst unannotierte, transkribierte Bereiche identifizieren, und diese dann mit einem Genome Browser untersuchen. Für die Bewertung des Coding Potentials verwenden wir RNAcode. Um einen Überblick über die zu erwartenden Ergebnisse zu bekommen, suchen sie sich bitte auch ein Stück eines exprimierten Proteins aus und analysieren das Coding Potential. Sollte es kein vorgefertigtes sinnvolles Alignment geben, dass die zu untersuchende Sequenzregion enthält, müssen sie zunächst ein solches Alignment erstellen. Das tut man zum Beispiel, indem man mittels NCBI BLAST ähnliche Sequnzen in verwandten Organismen sucht und diese dann mittels zB Clustal alignt.~~
What John said...

2.7.2 Benötigte Programme

- ~~RNAcode (vorinstalliert, sonst git-hub)~~
- IGV (conda)
- bedtools (s. o.)

- ~~eventuell ClustalX~~

2.7.3 Benötigte Dateien

Input:

- bedgraph files (4x means)
- bam files für IGV(6x)
- ~~Alignement files (clustal format)~~
- Coding potential regions file (John)

Output:

- Report (IGV snapshots, Figures, Listen etc.)

2.7.4 Zu beantwortende Fragen

- Welche und wieviele Bereiche zeigen Transkription, die mit der vorhandenen Annotation nicht erklärbar/erklärbar ist?
- Wieviele und welche dieser Bereiche davon haben Coding Potential?

3 Dateiformate

3.1 Sequenz Dateien

3.1.1 FASTA Dateien

Fasta Dateien beinhalten eine oder mehrere Sequenzen. Jede Sequenz erhält einen Header, der durch eine ">" Zeichen am Beginn der Zeile gekennzeichnet ist und Namen (ID) und Informationen zur nachfolgenden Sequenz enthalten sollte. Alle darauffolgenden Zeilen bis zum nächsten Header gehören dann zu einer Sequenz.
Fasta Dateien enthalten meist genomische oder proteomische Sequenzen, z.B. das menschliche Genom, das Maus Transkriptom, alle Proteine, deren Struktur bekannt ist, etc. Manchmal wird durch die Endung gekennzechnet, welche art von Sequenz das fasta file beinhaltet (.faa für Aminosäuren, .fna für Nukleotide).

3.1.2 FASTQ Dateien

Fasq files haben sich als Output Dateien von NGS Maschinen etabliert. Sie enthalten neben der Sequenzinformation auch noch Qualitätsinformation für jede sequenzierte Base

3.2 Dateien die Regionen beschreiben

Alle Dateinformate unten sind im Prinzip Tab separierte Formate.

3.2.1 Bed Dateien

Minimale Bed Dateien enthalten genomische Koordinaten (chromsom start stop). Bedfiles können mit weiteren Informationen erweitert werden (zB enthält erst ein 6 spaltiges Bed file Stranginformation. Es gibt auch 12 spaltige Bed files, die auch z.B. Transkriptstrukturen kodieren können. Bed Dateien haben sehr weite Verbreitung fü unter anderem Genannotationen, Chromatin Segmentierungen und Trankriptionsfaktorbindestellen.

3.2.2 Bedgraph Dateien

Bedgraph Dateien sind 4-spaltige Bed Dateien, bei denen in der 4. Spalte eine Mengeninformation (Zahl) steht. Sie werden für z.B. Coverage Daten oder Methylierungsdaten verwendet.

3.2.3 gff files

GFF files werden meist für Genomannotationen verwendet. Sie sind vor allem in der Lage, komplizierte Genome (en Transkript, Exon, CDS etc) zu kodieren.

3.3 Alignemnt Dateien

3.3.1 SAM und BAM

SAM Dateien sind auch Tab separiert und enthalten die Ergebnisse des Mappings von NGS reads auf ein Genom. Für jeden read gibt es (mindestens) eine Zeile, in der neben Informationen zur Mapping Quality und zu etwaigen Mates auch die Region, wo sich das beste gefundene Alignment von Genom und Read im Genom befindet sowie genaue Informationen über dieses Alignment und die Originalsequenz- und -qualitätsstrings befindet. Dadurch sind sam files noch grösser als die fastq files, aus denen sie entstanden sind. BAM files sind eine binäre, komprimierte Version der SAM files.