
# Grundlagen der Bioinformatik
# Übung 6:
# Microarray Analysis

Yvonne Lichtblau

# Vorstellung Lösungen Übung 4/Übung 5

# Lösungen vorstellen - Übung 4

(1) Global Alignment (5P)
Vorstellung durch eine Gruppe

(2) Finding sequences (3P)
Vorstellung durch eine Person

(3) Hierarchical Clustering (12P)
Vorstellung durch eine Gruppe

# Lösungen vorstellen - Übung 5

(1) Aufgaben 1-6

Vorstellung durch eine Gruppe

(2) Aufgaben 7-11

Vorstellung durch eine Gruppe

# Übung 5 - Voraussetzungen (1)

**Voraussetzungen:**

- Funktionsfähige R Installation (http://cran.r-project.org/)
- Packages: affy, AnnotationsDbi, hgu133plus2cdf
- Download Packages at Bioconductor (https://www.bioconductor.org/packages/devel/bioc/html/affy.html)
- Auf gruenau[1-6] existiert eine R Installation inkl. benötigter Packages

# Übung 5 – Voraussetzungen (2)

## Bioconductor

Bioconductor provides tools for the analysis and comprehension of high-throughput genomic data.


https://www.bioconductor.org/

## Getting Bioconductor

Start R and enter the commands:

```
source("http://bioconductor.org/biocLite.R")
biocLite()
```

## Installing Bioconductor Packages

```
biocLite(c("GenomicFeatures", "AnnotationDbi"))
```

# Aufgabe 1

**Theorie zu k-Means Clustering (4P)**

1. Das Ergebnis des k-Means Clusterings hängt häufig von der Initialisierung der Clustermittelpunkte ab. Entwerfe ein Beispiel in welchem k-Means unterschiedliche Ergebnisse abhängig von der Initialisierung der Clustermittelpunkte liefert. (2P)

2. Oft ist die richtige Anzahl der Cluster in biologischen Szenarien unbekannt. Wie könnte man die optimale Clusteranzahl (vielleicht graphisch) für einen gegebenen Datensatz schätzen? (2P)

# Aufgabe 2

## Start von R

Eine R Konsole erhält man durch Eingabe des Befehls ***R*** auf der Linux Konsole. Unter Windows sollte nach der Installation ein passender Menüeintrag existieren. Die Anzeige des Promptes **>** markiert die Bereitschaft des ***R***-Interpreters. Man kann nun die vorgestellten Kommandos in der ***R***-Konsole eingeben.

Beispielsweise liefert

```
?mean
```

eine detaillierte Beschreibung zu dem Befehl ***mean***. Scrollen der Beschreibung ist mit den Kursortasten möglich. Die Hilfe wird mit der Taste ***q*** beendet.

# Aufgabe 2.1

**Einladen von Microarrays (2P)**

- Unter http://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE3678 werden Affymetrix Microarray Daten (CEL-Files) zur Verfügung gestellt, welche im Rahmen dieser Übung analysiert werden. In diesem Versuch wird ein Subtyp von Schilddrüsenkrebs mit gesundem Schilddrüsengewebe verglichen. Für beide Samples wurden je 7 Chips hybridisert.

- Zu Beginn müssen die Daten in R geladen werden. Hierfür wird die Bibliothek *affy* benötigt. Bibliotheken können wie in der Übung beschrieben mit dem Befehl *library("name")* geladen werden. Zum Laden der Daten soll der Befehl *ReadAffy* verwendet werden.

- Erzeuge einen Boxplot der Expressionswerte nachdem die Expressionswerte $\log_2$-transformiert wurden. Tipp: Zugriff auf die tatsächlichen Expressionswerte der eingeladenen Arrays ist mit einem Funktionsaufruf möglich. Finde heraus welcher Objekttyp von ReadAffy erzeugt wird und suche die passende Methode in der Bioconductor Beschreibung

# Aufgabe 2.2

**Normalisieren der Microarrays (3P)**

Um die Vergleichbarkeit der Microarrays für weitere Analysen zu gewährleisten, müssen die Microarrays erst normalisiert werden.

Eine für Affymetrix Daten häufig verwendete Methode nennt sich ***rma***. Diese baut auf der vorgestellten quantil-Normalisierung auf. Suche die benötigte Funktion und führe eine rma Normalisierung auf den Arrays durch.

Erzeuge nun einen Boxplot der normalisierten Daten.

Wichtig: Die $\log_2$ Transformation wird von ***rma*** selbst durchgeführt.

Des weiteren ist zu beachten, dass ***rma*** mehrere Probes (Spots) zu einem Probeset zusammengefasst hat.

# Aufgabe 2.3

**Suche nach differentiell exprimierten Genen (5P)**

Bestimme für jedes Probeset den p-value (two-sided t test) und den Foldchange. Lösen mittels apply oder for-Schleife (3P).

Fragen:

- Wie viele Probesets sind differentiell exprimiert ($\alpha < 0.01$)? (0.5P)
- Wie viele Probesets haben einen |Foldchange| > 1? (0.5P)
- Wie viele Probesets sind differentiell exprimiert und haben |Foldchange| > 1? (1P)

# Aufgabe 2.4

**Multiple testing correction (3P)**

Führe eine p-value Korrektur nach Benjamini Hochberg durch (1.5P).

Fragen:

- Wie viele Probesets sind jetzt noch differentiell exprimiert? (0.5P)
- Wie viele Probesets sind differentiell exprimiert und haben |Foldchange| > 1?

# Aufgabe 2.5

**Volcano Plot (3P)**

Zeichne einen Volcano Plot für die eben ermittelten p-values und Foldchanges.

Hebe Punkte mit einem nach Benjamini Hochberg signifkanten p-value und einem |Foldchange| > 1 farbig hervor

Tipp: Dafür kann man den Parameter col der Funktion plot nutzen oder die Funktion points verwenden.

# Aufgabe 2.6

**Heatmap (optional)**

Zeichne eine Heatmap für die 50 Probesets mit dem kleinsten p-value.
Erzeuge eine weitere Heatmap für 50 zufällig gezogene Probesets.

# Abgabe

- Abgabe bis Mittwoch den 13.07.2016 um 23:59 Uhr
- Abgabe per Email an: yvonne.lichtblau@informatik.hu-berlin.de (gerne auch Fragen zur Übung per Email)
  - R Skript
  - PDF Datei (mit Bildern und Erklärungen, alle x- und y-Achsen beschriften)