HUMBOLDT-UNIVERSITÄT ZU BERLIN

Exposé

zur Bachelorarbeit

Eine kritische, komparative Analyse von Methoden zur Untersuchung differentieller Genexpression

Autor: Jan-Niklas Rössler
jan-niklas.roessler@hu-berlin.de

Betreuer: Prof. Dr. Ulf Leser
Raik Otto

29. März 2016

1 Einführung und Motivation

Bei der Erforschung von Krebserkrankungen spielt die Identifizierung und Charakterisierung von differentiell exprimierten Genen eine bedeutende Rolle. Eine Möglichkeit, um solche Gene ausfindig zu machen, ist die Verwendung von Microarrays. Dieses Verfahren nahm seine Anfänge mit dem Aufkommen der Kolonie-Hybridisierung nach Grunstein und Hogness im Jahr 1975 [1], welche 1979 von Gergen et al. [2] verwendet wurde, um erste geordnete DNA-Arrays herzustellen. Über die Jahre hat sich die Technologie rasant entwickelt und heutzutage gibt es eine Vielzahl unterschiedlicher Arten von Microarrays, die eine breite Anwendung in der biomedizinischen Forschung finden.
Ein entscheidender Schritt, um aus Microarray-Experimenten biologisch interpretierbare Erkenntnisse zu erhalten, ist die differentielle Expressionsanalyse. Hierbei wird getestet, ob Gene aus Samples, die aus Organismen mit den zu untersuchenden Phänotypen stammen, signifikante Unterschiede in ihren jeweiligen Expresionswerten aufweisen. Um diese Experimente auszuwerten existieren viele Methoden und statistische Modelle, welche sich zum Teil stark in ihrem Ansatz und Hintergrund unterscheiden. Die Wahl des Analyseverfahrens beeinflusst die resultierenden Ergebnisse und möglichen Interpretationen. Es ist also wichtig, genau zu überlegen und abzuwägen, welches Verfahren oder welche Kombination man wählt, wenn auf Microarray-Daten eine differentielle Expressionsanalyse durchgeführt wird, um den größtmöglichen Erkenntnisgewinn aus dem Experiment zu ziehen. [3]
Eine sehr verbreitete und beliebte Methode zur differentiellen Genexpressionsanalyse ist *Linear Models for Microarray and RNA-Seq Data* (limma) [4]. Diese basiert auf einem linearen Modell, das an die Expressionswerte jedes untersuchten Gens angenähert wird. Um die Berechnung zu stabilisieren, wird eine empirische Bayes-Methode genutzt, die Bestimmung der differentiellen Expression basiert auf einem t-Test. Am Ende der Berechnungen steht eine Liste mit Genen, welche zwischen zwei Phänotypen signifikante Unterschiede in ihren Expressionswerten aufweisen. *Gene Set Enrichment Analysis* (GSEA), nach Subramanian et al. [5], verfolgt einen anderen Ansatz. Hier werden die Berechnungen nicht auf Gen-Ebene durchgeführt, sondern auf vorher definierten Gen-Sets, welche Gene enthalten, die zum Beispiel alle mit einem bestimmten Pathway assoziiert sind. Auch GSEA sortiert die Gene nach ihrer differentiellen Expression, basierend auf einer *signal-to-noise-ratio*. Anschließend wird für jedes Gen-Set auf Basis dieser Sortierung ein sogenannter *Enrichment Score* bestimmt. Am Ende der Berechnungen stehen die zwischen den beiden Phänotypen über- oder unterrepräsentierten Gen-Sets, respektive Pathways. Das Grundprinzip von *Gene Set Enrichment Analysis* wurde mehrfach aufgegriffen, weiterentwickelt oder abgewandelt, so dass mittlerweile viele Derivate existieren. [6]
Vor diesem Hintergrund möchte ich eine kritische, komparative Analyse der beiden Algorithmen durchführen.

2 Ziel der Bachelorarbeit

In meiner Bachelorarbeit möchte ich anhand eines typischen mRNA Analyse-Workflows untersuchen, wie die Wahl unterschiedlicher Methoden zur differentiellen Expressionsanalyse die Ergebnisse beeinflusst. Genauer möchte ich das Verfahren limma, nach Smyth (2004) [4] und GSEA, nach Subramanian et al. (2005) [5], im Hinblick auf ihre Ergebnisse bei der Analyse differentieller Genexpression untersuchen. Der Einfluss von Parameteroptimierung und Bootstrapping auf den Output soll untersucht werden und es soll bewertet werden, inwiefern sich die Ergebnisse der beiden Agorithmen, angewendet auf die selben Eingabedaten, ergänzen oder widersprechen. Gegebenenfalls soll ein Verfahren entwickelt werden, welches den Output der beiden Methoden kombiniert.

3 Methodik

Grundgerüst der Analyse ist eine bereits bestehende Pipeline, programmiert in R [7], welche einen mRNA Analyse-Workflow implementiert. Diese Pipeline besitzt bereits eine Funktionalität für limma und GSEA und kann diese beiden Algorithmen unabhängig voneinander auf Expressionsdaten anwenden. Die Pipeline muss an ein paar Stellen angepasst werden, damit sie für die hier vorgestellten Analysen genutzt werden kann.
Im Folgenden sind die durchzuführenden Arbeitsschritte näher beschrieben, eine anschauliche Darstellung findet sich in Abbildung 1.

3.1 Rohdaten und *Preprocessing*

Der erste kritische Punkt dieser Arbeit ist die Wahl geeigneter Testdatensätze. Da die Methoden auf ihre Genauigkeit und Störanfälligkeit beim Auffinden differentiell exprimierter Gene und Pathways untersucht werden sollen, ist es wichtig, Rohdaten zu wählen bei denen im Voraus bekannt ist, welche Gene wirklich in dem verwendeten Sample über- oder unterrepräsentiert sind. Bei den meisten öffentlich verfügbaren Rohdaten trifft dies nicht zu und sie sind daher ungeeignet für Untersuchungen zur Methoden-Validierung [8]. Doch gibt es einige Datensätze für welche die Ergebnisse im Vorfeld bekannt sind oder im Nachhinein hinreichend belegt wurden. Einer ist der *Platinum Golden Spike* Datensatz (GSE21344) [9]. Es handelt sich dabei um einen sogenannten *spike-in* Datensatz, bei dem die genutzten mRNA-Samples „konstruiert“ sind und dadurch die Mengenverhältnisse genau bekannt sind. Ein zweiter geeigneter Datensatz kommt aus einer Brustkrebs-Studie (GSE20437) [10]. Differentiell exprimierte Gene wurden durch verschiedene Analyseverfahren bestimmt und die Ergebnisse wurden anschließend durch qPCR validiert. Die Liste der gefundenen Gene wurde mit der Studie veröffentlicht.
Die Rohdaten werden von der Pipeline eingelesen. Anschließend wird eine Qualitätskontrolle durchgeführt, um sicher zu stellen, dass die Ergebnisse nicht durch strukturelle oder experimentelle Fehler verfälscht werden. Danach werden die Daten normalisiert und darauf folgend mit HGNC Gen-Symbolen annotiert. Nun steht ein Expressions-Datensatz in Form einer Matrix bereit, der für jedes Sample und jedes *Probe Set* (Feature) die je-

weiligen normalisierten, log-transformierten Expressionswerte enthält. Dieser Datensatz dient sowohl limma als auch GSEA als Grundlage für ihre Berechnungen.

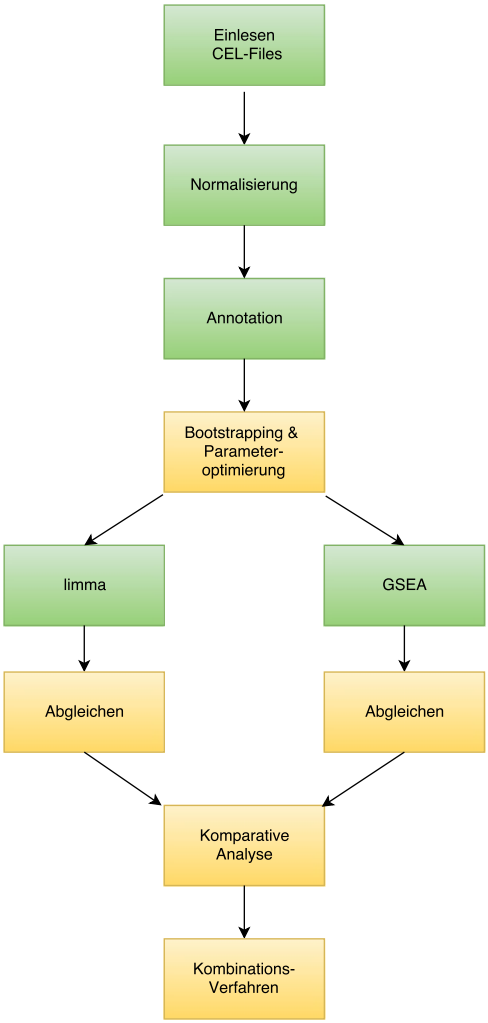

Abb. 1: Schematischer Analyse-Ablauf. Schritte in grün bestehen bereits in der Pipeline, gelb gekennzeichnete Abschnitte werden im Rahmen dieser Bachelorarbeit entwickelt.

3.2 Parameteroptimierung, Bootrstrapping und komparative Analyse von limma und GSEA

Nachdem limma und GSEA auf die Daten angewandt wurden, soll der Output jeweils mit den Referenz-Ergebnissen abgeglichen werden, um die Sensitivität und Spezifität festzustellen. Da die beiden Algorithmen auf unterschiedlichen Ebenen arbeiten, muss eine Anpassung vorgenommen werden, damit ein sinnvoller Vergleich möglich ist. Die Ergebnisse von limma und die Referenzen sind in Form einzelner Gen-Symbole angegeben,

die Resultate von GSEA liegen aggregiert, als Gen-Sets bzw. Pathways, vor. Im Rahmen dieser Bachelorarbeit soll nur die Pathway-Ebene betrachtet werden. Daher wird mit dem Output von limma und den Referenz-Ergebnissen ebenfalls eine Enrichment-Analyse durchgeführt. Als Grundlage dient hierbei die Gen-Set-Datenbank von GSEA (MSigDB [11]). Durch diese Anpassung ist ein einheitlicher Abgleich zwischen limma, GSEA und *gold standard* möglich.
Kernstück der Arbeit ist es, den Einfluss von Parameteroptimierung und Bootstrapping auf die Ergebnisse zu untersuchen. Dafür werden die Berechnungen der beiden Algorithmen mehrfach wiederholt und bei jedem Durchlauf wird eine Variation der Eingabeparameter vorgenommen. Limma und GSEA besitzen viele Stellschrauben von denen Ablauf und Output beeinflusst werden. Mögliche Variationen wären beispielsweise:

- Unterschiedliche Normalisierungsmethoden
- Verschiedene Algorithmus-immanente statistische Modelle (eBayes bei limma)

Über diese Parameter sollen die Resultate evaluiert werden. Zusätzlich soll Bootstrapping angewendet werden, um die Robustheit der Ergebnisse zu testen. Hierfür soll der Expressions-Datensatz gesamplet werden. Dies kann auf Feature-Ebene geschehen oder auf Sample-Ebene. Nachdem die Berechnungen von limma und GSEA mit veränderten Parametern durchgeführt wurden, wird der Output wieder mit der Referenz abgeglichen. Abschließend soll für diesen Teil der Analyse herausgearbeitet werden, inwiefern sich die Erkenntnisse von limma und GSEA gleichen, ob es wichtige Unterschiede gibt oder gar Widersprüche. Vor diesem Hintergrund soll abgewogen werden, ob es sinnvoll ist ein Verfahren zu entwickeln, welches die Ergebnisse der beiden Algorithmen kombiniert.

3.3 Optimale Kombination der Ergebnisse durch *machine learning*

Wenn sich durch die vorangegangen Analysen zeigt, dass der Intersect zwischen den Ergebnissen von limma und GSEA nicht groß ist, aber beide Methoden trotzdem gute und richtige Resultate liefern, soll ein Verfahren zum Vereinigen des Outputs entwickelt werden. In diesem Fall soll damit eine bessere Abdeckung der Referenz-Ergebnisse erreicht werden. Dafür soll auf Grundlage des R-Pakets caret ein simples *machine learning* Verfahren implementiert werden, welches die Ergebnis-Datensätze aus der komparativen Analyse von limma und GSEA als Trainingsdatensätze verwendet.

4 Vorläufige Gliederung

Zuerst soll eine Einführung in das Thema, mit Motivation und abschließender Zielsetzung, gegeben werden. Danach sollen die beiden Algorithmen limma und GSEA vorgestellt werden. Dabei soll ihr theoretisches Konzept und ihre Vorgehensweise herausgearbeitet werden. Danach soll die Methodik vorgestellt werden die angewandt wurde, um die komparative Analysen durchzuühren. Dann werden die Ergebnisse vorgestellt und zum Schluss interpretiert, ausgewertet und die Erkenntnisse werden diskutiert.
Somit ergibt sich folgende vorläufige Gliederung:

1. Einführung, Motivation und Zielsetzung
2. Lineare Modelle zur Analyse differentieller Genexpression: limma
3. *Gen Set Enrichment Analysis*
4. Parameteroptimierung und der Einfluss auf den Output
5. Komparative Analyse von limma und GSEA
6. Ergebnisse
7. Interpretation und Diskussion

5 Zeitlicher Ablauf

Für das Erstellen der Bachelorarbeit sind laut Prüfungsordnung 12 Wochen Bearbeitungszeitraum vorgesehen. Der erste Teil der Zeit, ungefähr 4 Wochen, soll genutzt werden, um die Implementierung umzusetzen. Das umfasst das Anpassen der Pipeline; automatische Methoden zum Validieren der Ergebnisse; Parameteroptimierung und Bootstrapping; sowie eventuell das Verfahren zur Kombination durch *machine learning*. Wenn dieser Workflow bereitsteht, sollen in den nächsten 5 Wochen die Experimente durchgeführt werden. Hierfür ist die meiste Bearbeitungszeit veranschlagt, da die Berechnungen, die GSEA macht, sehr zeitintensiv sind und mehrfach wiederholt werden müssen. Da in dieser Arbeitsphase öfter Wartezeit anfallen wird, kann diese dazu genutzt werden schon einmal den theoretischen Teil der Bachelorarbeit zu verfassen. In den letzten 3 Wochen sollen dann die Ergebnisse des praktischen Teils schriftlich festgehalten werden und der Arbeit der letzte Schliff verliehen werden.

Tabelle 1: Geplante zeitliche Gliederung der Bachelorarbeit

Woche	Vorhaben
1 - 4	Implementierung
5 - 9	Experimente & Verfassen Theorieteil
10 - 12	Verfassen Praxisteil

6 Referenzen

[1] Grunstein M, Hogness DS. Colony hybridization: a method for the isolation of cloned DNAs that contain a specific gene. Proceedings of the National Academy of Sciences of the United States of America. 1975;72(10):3961-3965. [PubMed]

[2] Gergen JP, Stern RH, Wensink PC. Filter replicas and permanent collections of recombinant DNA plasmids. Nucleic acids research. 1979;7:2115-2136. [PubMed]

[3] Ding Y, Xu L, Jovanovic BD, et al. The Methodology Used to Measure Differential Gene Expression Affects the Outcome. *Journal of Biomolecular Techniques*: JBT. 2007;18(5):321-330. [PubMed]

[4] Smyth, G. K. (2004). Linear models and empirical Bayes methods for assessing differential expression in microarray experiments. *Statistical Applications in Genetics and Molecular Biology*, Vol. 3, No. 1, Article 3. [PubMed]

[5] Subramanian A, Tamayo P, Mootha VK, et al. Gene set enrichment analysis: A knowledge-based approach for interpreting genome-wide expression profiles. Proceedings of the National Academy of Sciences of the United States of America. 2005;102(43):15545-15550. [PubMed]

[6] Lee S, Kim J, Lee S. A comparative study on gene-set analysis methods for assessing differential expression associated with the survival phenotype. *BMC Bioinformatics*. 2011;12:377. doi:10.1186/1471-2105-12-377. [PubMed]

[7] R Core Team (2015). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria.
`https://www.R-project.org/.`

[8] Pearson RD. A comprehensive re-analysis of the Golden Spike data: Towards a benchmark for differential expression methods. *BMC Bioinformatics*. 2008;9:164. doi:10.1186/1471-2105-9-164. [PubMed]

[9] Zhu Q, Miecznikowski JC, Halfon MS. Preferred analysis methods for Affymetrix GeneChips. II. An expanded, balanced, wholly-defined spike-in dataset. BMC Bioinformatics. 2010;11:285. doi:10.1186/1471-2105-11-285. [PubMed]

[10] Graham K, de las Morenas A, Tripathi A, et al. Gene expression in histologically normal epithelium from breast cancer patients and from cancer-free prophylactic mastectomy patients shares a similar profile. British Journal of Cancer. 2010;102(8):1284-1293. doi:10.1038/sj.bjc.6605576. [PubMed]

[11] We acknowledge our use of the gene set enrichment analysis, GSEA software, and Molecular Signature Database (MSigDB) (Subramanian, Tamayo, et al. (2005), PNAS 102, 15545-15550, `http://www.broad.mit.edu/gsea/`)