Karlsruhe Institute of Technology

Integration of internal and external gene expression and drug-perturbation data to empower novel immune therapies against Parkinson's Disease

Master Thesis of

Rudolf Biczok

At the Department of Informatics
Institute of Theoretical Computer Science

Reviewer:	Prof. Dr. Alexandros Stamatakis
	Prof. Dr. Ralf Reussner
Advisor:	Dr. Alexey Kozlov

Time Period: 1st August 2018 – 31st January 2019

Statement of Authorship

I hereby declare that this document has been composed by myself and describes my own work, unless otherwise acknowledged in the text.

Heidelberg, 31st January 2019

Abstract

Bla bla

Deutsche Zusammenfassung

Bla bla

Acknowledgements

Bla Bla

Contents

1. Introduction

1.1 Motivation

2. State of the art

2.1 Gene Expression Analysis

The most recurring task in pharmaceutical research & early development is the gene expression analysis on a given date source.

2.1.1 Methods for differential gene expression inferrence

Name | Strategy | Prefered Input Data —–|————|—————— [edgeR](http://doi.org/10.1093/bioinforma
| Negative binomial distribution + Trimmed Mean of M values (TMM) Normalization | RNAseq [DEseq](https://doi.org/10.1186/s13059-014-0550-8) | Negative binomial distribution + scaling factor normalization procedure | RNAseq [limma](https://doi.org/10.1093/nar/gkv007) | Linear Modeling + voom transformation of counts (vor RNAseq) | RNAseq & microarray

The "rule of thumb" is to use limma for microarray data and edgeR for RNAseq data. DEseq is used to verify that a hypothesis based on edgeR results can also be derived from DEseq results (since edgeR is known to report more false positives).

2.1.2 Methods for batch-effect correction in meta-analysis

We use ComBat, SVA, and BioQC. Here we will probably have to compare different methods and reach a conclusion.

2.1.3 Methods for gene-set/pathway enrichment analysis

We use CAMERA, BioQC, and Fisher's exact test. Previously we found out that CAMERA and BioQC will lead to false negatives when many genes are differentially expressed, while Fisher's exact test will not (not published). We can verify this and make a meta-method to accommodate different scenarios.

2.2 ROGER

2.2.1 ROGER Database

* **Annotations** * Gene Annotation (consumes biomart) * GeneAnnotation: Ensemble & NCBI gen IDs. ROGER-internal GeneIndex, Gen meta data * Orthologs: Mapping between orthologous genes between different species * TranscriptioAnnotation: Ensemble

Transcription ID and meta data * TranscriptRefSeq: NCBI Transcription ID and meta data * Genesets (consumes mongodb/json & gmt) * DefaultGenesets: Available gen set data * DefaultGenesetCategory: For gen set categorization * DefaultGenesets2gene: Mapping of gen set data to gen annotations * **Input** * Datasets: Raw expression data * Phenodata * Designs: Relevant Feature matrix * Contrasts: Contrast matrix * **Methods & Results** * GSEmethods: Used Gen enrichment method (e.g. CAMERA) * GSEtables: Gen enrichment results * DGEmethods: Used Differential Gen Expression inference method (e.g. edgeR, limma) * DGEmodels: Used DGE model based on Desing and Cntrast information * DGEtables: Results from DEG inference

2.2.2 Annotation Problems

* Have to support both Ensembl and NCBI IDs * Ensembl has many unconsistent / deprecated data: Some Gene Symbols apper in multiple EnsembleGeneIds, * Possible fix: pick the "most accurate on" (e.g. does it have a proper chromosone? Number of Transcripts etc.)

2.2.3 RESTful APIs for scientific R pipelines

* [rplumber](https://www.rplumber.io/) * Fastest way to deploy REST services * Very low-level: No load balancing, no authentication, task management, ... * [OpenCPU](https://www.opencpu.org) * Load balancing * Lightwing WEB API basedn on JavaScript * No build-in support for [long running jobs] (https://github.com/opencpu/opencpu/issues/141) * No build-in task management * [Flask](http://flask.pocoo.org) * Python equivalent to OpenCPU * Established in the department * Requires wapper functions between python <-> R * No build-in task management

2.3 Differential Gene Expression

Lets assume we have a study consisting of a set of samples $D = \{A, B, C, D, E, F\}$. Samples A and B are from macrophage cells, C and D are from microglia cells, and E and F are from monocyte cells.

Then we have basically two ground ways to model the experiments:

	1 vs Average	**1 vs 1**
❶	`model.matrix(~CellType)`	`model.matrix(~0+CellType)`
❷	$\begin{array}{cc} & \begin{array}{ccc} \mu & \text{micro} & \text{mono} \end{array} \\ \begin{array}{c} A \\ B \\ C \\ D \\ E \\ F \end{array} & \begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \end{pmatrix} \end{array}$	$\begin{array}{cc} & \begin{array}{ccc} \text{macro} & \text{micro} & \text{mono} \end{array} \\ \begin{array}{c} A \\ B \\ C \\ D \\ E \\ F \end{array} & \begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix} \end{array}$
❸	$y \sim \mu + \beta_{\text{micro}} x_{\text{micro}} + \beta_{\text{mono}} x_{\text{mono}}$	$y \sim \beta_{\text{macro}} x_{\text{macro}} + \beta_{\text{micro}} x_{\text{micro}}$ $\beta_{\text{mono}} x_{\text{mono}}$
❹	$\begin{array}{cc} \begin{array}{c} \mu \\ \text{micro} \\ \text{mono} \end{array} & \begin{pmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{pmatrix} \end{array}$	$\begin{array}{cc} \begin{array}{c} \text{macro} \\ \text{micro} \\ \text{mono} \end{array} & \begin{pmatrix} -1 & -1 & 0 \\ 1 & 0 & -1 \\ 0 & 1 & 1 \end{pmatrix} \end{array}$

Table 2.1: Overview of example experiments

3. Conclusion and Future Work

Bibliography

Appendix