compcodeR - an R package for benchmarking differential expression methods for RNA-seq data

Charlotte Soneson

1Bioinformatics Core Facility, SIB Swiss Institute of Bioinformatics, Quartier Sorge, CH-1015 Lausanne, Switzerland.

1 INTRODUCTION

Transcriptome profiling studies using RNA-seq with the goal of finding genes that are differentially expressed (DE) between conditions are abundant in the current scientific literature and can be expected to become even more so as next generation sequencing technology becomes cheaper and more accessible. RNA-seq experiments generate millions of short reads, which are aligned to a reference sequence to yield a quantitative measure of the expression levels of a collection of genes or other features. Typically, the processed data is represented as a count matrix, which constitute the input for many differential expression methods.

In the last few years, many novel differential expression methods applicable to count matrices obtained from RNA-seq experiments have been presented (for example, Robinson et al. (2010); Anders and Huber (2010); Hardcastle and Kelly (2010); Tarazona et al. (2011)). At this point, both users and developers of such methods would thus benefit greatly from objective and standardized benchmarking and characterization of new and existing approaches. A few comparison studies have been published (e.g., Robles et al. (2012); Soneson and Delorenzi (2013)). However, the fast pace at which new and updated methods are being presented and the varying objectives of the users create a need for a tool that makes it easy to evaluate and compare a collection of methods from many different aspects in a standardized way. In this application note we present such a tool, compcodeR (COMParison of COunt-based Differential Expression analysis methods with R) is a benchmarking R package that in a few steps lets the user evaluate and compare differential expression methods using an approach similar to the one used by Soneson and Delorenzi (2013). Accompanying the package is a large collection of simulated and real-world benchmarking data sets, together with differential expression results obtained by more than 20 different approaches (available from http://bcf.isb-sib.ch/data/compcodeR). The exact R code used to run each differential expression analysis is included and can be rerun to reproduce the results. Taken together, the package provides users with a pedagogical interface to understand and compare differential expression methods, and gives developers an accessible tool for standardized benchmarking of newly developed approaches.

2 EXAMPLE

This section outlines the three major functionalities of compcodeR. First, the package contains a function for generating synthetic RNA-seq count matrices, using the approach described by Soneson and Delorenzi (2013) and Robles et al. (2012). The user defines the properties of the data set, such as the number of genes and samples, the fraction of truly DE genes and their effect size distribution, several parameters governing the inclusion of outlier counts, and filter thresholds. The following code simulates a data set consisting of Negative Binomially distributed data for 5 samples from each of two conditions and 12,500 genes, 10% of which are truly DE.

```r
> dat <- generateSyntheticData(dataset = "mydata", 
n.vars = 12500, samples.per.cond = 5, 
n.diffexp = 1250, fraction.upregulated = 0.5, 
repl.id = 1, output.file = "mydata_5spc.rds")
```

The code generates an object of the class compData, which is saved to a file named mydata_5spc.rds. By rerunning the data simulation with repl.id set to different values it is possible to generate multiple replicate data sets for a given simulation setting, which can make method comparisons more robust and informative.

Second, the package provides an interface to several of the most commonly used methods for differential expression analysis of RNA-seq data. The intention is not to cover all available methods or exploit all of their possibilities, and new methods can easily be included by the user. The code below applies the differential expression test implemented in edgeR (Robinson et al. (2010)) to the data simulated above, and saves a new compData object containing also the test results. A list of the differential expression methods to which compcodeR provides an interface can be obtained using the function listcreateRmd. The runDiffExp function automatically includes the executed code as well as the

---

*to whom correspondence should be addressed*
**REFERENCES**


