Bioinformatics
doi.10.1093/bioinformatics/xxxxxx
Advance Access Publication Date: Day Month Year
category

Subject

Investigation of RNA metabolism using pulseR

Uvarovskii Alexey [1,2]*, Christoph Dieterich [1,2]

[1] Section of Bioinformatics and Systems Cardiology Klaus Tschira Institute for Integrative Computational Cardiology Department of Internal Medicine III University Hospital Heidelberg, Im Neuenheimer Feld 669 69120 Heidelberg, and [2] German Center for Cardiovascular Research (DZHK), Im Neuenheimer Feld 669 69120 Heidelberg

*To whom correspondence should be addressed.

Associate Editor: XXXXXXX

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Abstract

Summary: The pulseR R package is designed to simplify processing of the data generated by RNA-seq pulse-chase experiments. The package allows to work with various experimental schemes and to estimate kinetic parameters such as degradation and synthesis rates from read count data.
Availability: The pulseR package is freely available at https://github.com/dieterich-lab/pulseR under the GPLv3.0 licence
Contact: alexey.mipt@gmail.com
Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

Gene expression level is defined by the rates of RNA synthesis and degradation. Understanding how certain gene levels are regulated in a response to condition changes might help to uncover the underlying control mechanisms.

The pulse-chase experimental approach allows to measure such kinetics. In this method, tracing molecules are introduced to the medium, which results in their incorporation into nascent RNA molecules. 4sU labelling, developed by (Dölken *et al.*, 2008), is used to estimate kinetic rates of RNA metabolism in a number of studies up to date, see (Wachutka and Gagneur, 2016) for the review. The RNA-seq data generated in such experiments have a discrete nature. However, there is no software available for parameter estimation in kinetic models of gene expression, which is specifically designed to handle count data. Here we present the `pulseR` package, which allows to process RNA-seq data from 4sU-labelling experiments.

2 Implementation

2.1 Parameter definition

The aim of the package is to simplify the procedure of the parameter estimating. RNA dynamics can be described by ordinary differential equations, which have simple analytic solution if the degradation and synthesis rates are assumed to be constant. In the `pulseR` package, a user need to specify the expressions for the mean RNA abundances, Alternatively, formulas can be generated using package functions for the most frequent cases.

Although the most interest is focused on the gene-specific parameters, pulseR allows to introduce shared parameters. This can be useful for taking into account the difference in the uridine content, since it can introduce a bias in the estimations, (Miller *et al.*, 2011; Schwalb *et al.*, 2012). In this case, the RNA abundances are multiplied by a probability that at least one uridine in the molecule is substituted by 4sU. The shared parameter then is the probability for a single base to be substituted by a 4sU.

2.2 Normalisation

Besides the parameters of the interest, one must estimate how different fractions and samples relate to each other, because the sequencing depth may vary. In addition, amounts of labelled and unlabelled RNA in fractions is changed due to the pull-out procedure. For example, if the labelled fraction consists of the labelled RNA L_{ij} and the unlabelled RNA U_{ij} molecules, for a sample j and gene i we have

$$[\text{labelled fraction}]_{ij} = \alpha_j L_{ij} + \beta_j U_{ij} \tag{1}$$

If spike-ins present in the probes, the normalisation coefficients are estimated directly as in the DESeq package, Anders and Huber (2010). The user must provide lists of spike-ins which are specific for different types of RNA, i.e. in order to estimate α_j and β_j separately in our example.

In case of spike-ins-free experiments, it can be possible to derive the normalisation factors, because the system is overdetermined (given a high number of genes). The user need to specify how to split samples into the groups. Inside one group (e.g. labelled fraction after 2hr of pulse) samples are normalised for sequencing depth d_j by the DESeq procedure. Normalisation between the groups is performed during the fitting procedure, and this coefficients are shared between the samples from the same group:

$$[\text{labelled fraction}]_{ij} = d_j(\alpha L_{ij} + \beta U_{ij}) \qquad (2)$$

Parameter estimation

We use the maximum likelihood method (MLE) to obtain parameter values. In RNA-seq experiments, expression level is represented by read number. To model such data, we assume them to follow the negative binomial (NB) distribution, because the NB distribution is shown to successfully describe over-dispersed RNA-seq data (Robinson and Smyth, 2007). The NB distribution has two parameters, the mean m and the dispersion parameter α. Hence, a read number of a gene i in a sample j reads

$$K_{ij} \sim \text{NB}(m_{ij}, \alpha). \qquad (3)$$

Here we treat the dispersion parameters α as being shared between all samples and genes. Otherwise it would not be possible to infer all parameters from a small number of replicates (usually, only 2 or 3 points are available).

We separated the fitting procedure into several simpler steps:

1. fitting of gene-specific parameters (e.g. degradation rate)
2. fitting of shared parameters
3. fitting of the normalisation factors (for a spike-in-free design)
4. estimation of the dispersion parameter

We repeat the steps 1-4 until user-specified convergence criteria is not met. Since gene-gene interactions are not considered by the model, it is possible to fit this parameters independently in parallel.

To optimise the likelihood functions, we use implementation of the L-BFGS-U method (Byrd *et al.*, 1995) available in the `stats` R package (R Core Team, 2017).

3 Discussion

3.1 Comparison with existing approaches

The published approaches are different in terms of data normalisation, statistical model and underlying mathematical model of the RNA metabolism. Here we analyse the following software: DRiLL (Rabani *et al.*, 2014), INSPEcT (De Pretis *et al.*, 2015), DTA (Schwalb *et al.*, 2012), HALO (Friedel *et al.*, 2010).

In most cases, samples are normalised by utilising overdetermination of the system. The normalisation coefficients are estimated via regression (DTA, HALO) or during the MLE procedure together with other parameters (INSPEcT, DRiLL). Besides the normalisation during parameter fitting, pulseR allows to use spike-ins counts as an alternative.

HALO and DTA estimate degradation rates from a ratio of labelled and total RNA fractions without any assumptions on the statistical model. In DRiLL, expression levels are fitted to the binomial distribution. However, the kinetic rates are estimated via optimisation of residual sum of squares in both, DRiLL and INSPEcT, which implies the normal distribution. In contrast, pulseR assumes the NB distribution for MLE of all parameters, which allows to work directly on the count data.

Experiments may vary in scheme and time points number, and it is important how flexible to it a package is. DTA and HALO are designed to work only on a single time point. DRiLL and INSPEcT can infer rates on the basis of several time points. Moreover, the mentioned packages can work only with the pulse-experiments. pulseR package can handle different designs including chase- and combined experiments with various number of data points, having formulas for mean read number estimation provided.

	pulseR	DRiLL	INSPEcT	DTA	HALO
statistical model	NB	N, BIN	N	-	-
spike-ins	+	-	-	-	-
several time points	+	+	+	-	-
variable design	+	-	-	-	-
non-constant rates	-	+	+	-	-
uridine bias	+	-	-	+	+
RNA processing	$*$	+	+	-	-
gene isoforms	-	+	-	-	-
language	R	MATLAB	R	R	Java

Table 1. Comparison of available software for parameter estimation in pulse-chase experiments. N: normal, NB: negative binomial, BIN: binomial. $*$ - must be defined by a user.

The DRiLL and INSPEcT packages can model time-dependent rates out of the box. Additionally, they can perform testing to select between constant rate and variable rate models. Noteworthy, the DRiLL software is able to handle data about multiple mRNA isoforms.

3.2 Application

We evaluated `pulseR` performance using simulated data. The software is able to reproduce gene-specific parameters and sample normalisation factors. For the detailed description of the workflow please refer to the supplementary material.

Acknowledgements

thanks thanks thanks thank thanks thanks thanks thank thanks thanks thanks thank thanks thanks thanks thank thanks thanks thanks thank

Funding

money money

References

Anders, S. and Huber, W. (2010). Differential expression analysis for sequence count data. *Genome biology*, **11**(10), R106.

Byrd, R. H., Lu, P., *et al.* (1995). A limited memory algorithm for bound constrained optimization. *SIAM Journal on Scientific Computing*, **16**(5), 1190–1208.

De Pretis, S., Kress, T., *et al.* (2015). INSPEcT: a computational tool to infer mrna synthesis, processing and degradation dynamics from rna-and 4su-seq time course experiments. *Bioinformatics*, **31**(17), 2829–2835.

Dölken, L., Ruzsics, Z., *et al.* (2008). High-resolution gene expression profiling for simultaneous kinetic parameter analysis of rna synthesis and decay. *Rna*, **14**(9), 1959–1972.

Friedel, C. C., Kaufmann, S., *et al.* (2010). HALO – a java framework for precise transcript half-life determination. *Bioinformatics*, **26**(9), 1264–1266.

Miller, C., Schwalb, B., *et al.* (2011). Dynamic transcriptome analysis measures rates of mrna synthesis and decay in yeast. *Molecular systems biology*, **7**(1), 458.

R Core Team (2017). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Rabani, M., Raychowdhury, R., *et al.* (2014). High-resolution sequencing and modeling identifies distinct dynamic rna regulatory strategies. *Cell*, **159**(7), 1698–1710.

Robinson, M. D. and Smyth, G. K. (2007). Moderated statistical tests for assessing differences in tag abundance. *Bioinformatics*, **23**(21), 2881–2887.

Schwalb, B., Schulz, D., *et al.* (2012). Measurement of genome-wide rna synthesis and decay rates with dynamic transcriptome analysis (dta). *Bioinformatics*, **28**(6), 884–885.

Wachutka, L. and Gagneur, J. (2016). Measures of rna metabolism rates: Toward a definition at the level of single bonds. *Transcription*, (just-accepted), 00–00.