The Journal of Open Source Software

cellanneal: A user-friendly deconvolution software for transcriptomics data

Lisa Buchauer [1,2¶] **and Shalev Itzkovitz** [1]

1 Department of Molecular Cell Biology, Weizmann Institute of Science, Rehovot, Israel **2** Department of Infectious Diseases and Respiratory Medicine, Charité-Universitätsmedizin Berlin, Berlin, Germany ¶ Corresponding author

DOI: 10.21105/joss.05610

Software

- Review
- Repository
- Archive

Editor: Jacob Schreiber

Reviewers:

- @ritika-giri
- @ManavalanG

Submitted: 12 June 2023
Published: 12 January 2024

Summary

Single-cell sequencing methods enable precise characterization of gene expression patterns in individual cells. However, they may provide inaccurate information about the cell type composition of samples, as required preprocessing procedures such as tissue dissociation or cell sorting affect viability of different cell types to varying extent (Erdmann-Pham et al., 2021). Further, especially in the clinical context, single-cell sequencing of patient samples is currently not routinely applied because of high cost and required expertise, while bulk sequencing is more prevalent.

For these reasons, computational deconvolution methods are gaining popularity in basic and clinical research. Computational deconvolution approaches infer the cell type proportions constituting a given bulk RNA sample based on separately obtained cell type reference data. Several computational deconvolution methods have been developed in the last decade and have contributed to our understanding of tissue composition (Cobos et al., 2020; Sturm et al., 2019). Generally, during deconvolution, the computational mixture is constructed from a set of cell type fractions and reference gene expression vectors for each of the participating cell types, most commonly derived from single-cell data. The cell type fractions are then iteratively changed until agreement between the *in silico* gene expression vector and the observed bulk sample gene expression vector is optimal by a measure of choice. Here, published methods rely almost exclusively on minimizing the sum of squared residuals between bulk and computationally mixed vectors. Algorithms for such optimization problems are readily available and include variants of least squares regression (e.g. weighted least squares regression (Racle et al., 2017), non-negative least squares regression (Jew et al., 2020; Wang et al., 2019) or least trimmed squares (Hao et al., 2019)) and support vector regression (Newman et al., 2015, 2019).

However, least squares-based optimization is faced with a particular challenge in bulk RNAseq deconvolution because of the highly skewed nature of mRNA copy number distributions, ranging from less than 1 to more than 10,000 average mRNA copies per cell (Li et al., 2016; Schwanhäusser et al., 2011). In such settings, optimization results may be strongly influenced by few highly expressed genes and are thus not robust to noise or platform effects influencing the readout of these genes. Support vector regression based models like CIBERSORT (Newman et al., 2015) perform gene feature selection out of a user-defined signature gene list, the contents of which can strongly affect the cell proportion estimates. Overall, identifying the right genes for deconvolution becomes a task in itself (Aliee & Theis, 2021). As a result, deconvolution methods may yield inferred mixed gene expression vectors that do not correlate well with measured bulk gene expression.

Here, we introduce cellanneal, a python-based software for deconvolving bulk RNA sequencing data. cellanneal relies on the optimization of Spearman's rank correlation coefficient between experimental and computational mixture gene expression vectors using simulated annealing.

Transforming gene expression values into ranks prior to optimization allows genes of different expression magnitudes to contribute similarly to deconvolution; further, `cellanneal` employs a permissive gene selection procedure that includes as many informative genes as possible. Together, these approaches limit the influence of highly expressed genes on the one hand and reduce dependency on specific gene list choices. `cellanneal` can be used as a python package or via a command line interface, but importantly also provides a simple graphical user interface which is distributed as a single executable file for user convenience.

Statement of need

Making sense of bulk RNA sequencing datasets often requires analysis of the cell type composition of the samples. This is particularly relevant in clinical samples that analyze the transcriptome of tissues or tumors which consist of epithelial, stromal and immune cell types. In parallel, publicly available single-cell data sets enable precise characterization of the expression signature of multiple individual cell types. However, software tools for computational bulk deconvolution are often slow, non-robust and not easy to use. Some existing methods address the aspect of user-friendliness by providing graphical web interfaces, but submitting sensitive medical data to an external web server is not always compatible with privacy legislation.

To address these challenges, we have developed `cellanneal`, a deconvolution approach that uses Spearman's rank correlation coefficient between synthetic and bulk gene expression vectors as the optimization procedure's objective function. Because this correlation measure is calculated from ranks rather than absolute data values, each gene influences the optimization result to a similar extent. Users are encouraged to include as many informative genes as possible in the analysis. `cellanneal` optimizes cell type fractions by simulated annealing, a flexible, rapid and robust algorithm for global optimization (Kirkpatrick et al., 1983; Virtanen et al., 2020). `cellanneal` can be used as a python package, via its command line interface or via a user-friendly graphical software which runs locally. Its typical processing time for one mixture sample is below one minute on a desktop machine (MacBook Pro 2020, 2.3 GHz Quad-Core Intel Core i7, 16 GB RAM).

Availability and Features

The python package and command line interface are available at https://github.com/LiBuchauer/cellanneal and can be installed using `pip`. The graphical software for Microsoft Windows and MacOS can be downloaded at http://shalevlab.weizmann.ac.il/resources and does not require installation. Instructions for installation and use as well as general documentation is available at https://github.com/LiBuchauer/cellanneal.

The python package provides functions for the three main steps of a deconvolution analysis with `cellanneal`: identification of a gene set for deconvolution, deconvolution using simulated annealing, and plotting the results. A quick start workflow is available as part of the documentation. For the command line interface and the graphical user interface, these three steps are combined into one call (click).

`cellanneal` runs which were started from either the command line or the graphical user interface produce a collection of result files including tabular deconvolution results (cell type fractions for each sample) and figures illustrating these cell type distributions. Further, `cellanneal` computes and stores the gene-wise fold change between the observed bulk expression and the estimated expression based on the inferred cell type composition. This enables identifying genes for which expression may be specifically induced or inhibited in the bulk sample compared to the single cell reference. Such genes may be of biological or medical interest.
Figures produced by `cellanneal` include a heatmap showing sample compositions (Figure 1), pie charts showing sample compositions (Figure 2), and scatter plots showing correlation between experimental bulk gene expression values and their `cellanneal`-derived counterparts

from the best identified computational mixture (Figure 3). The examples presented in this manuscript use data from (Massalha et al., 2020).

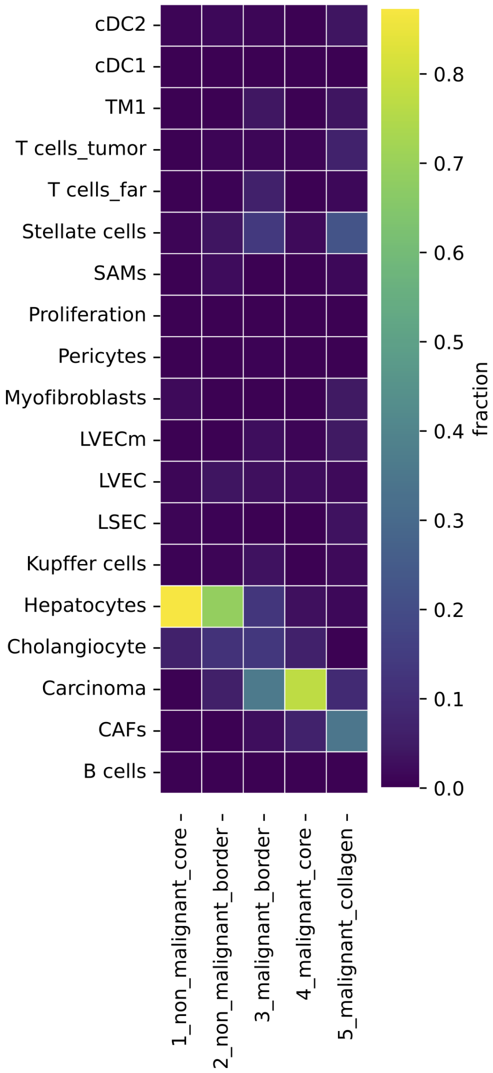

Figure 1: A heatmap produced by `cellanneal`. Constituting cell types are on the y-axis, deconvolved bulk sample names on the x-axis. The colour scale shows the fractional presence of cell type in each bulk.

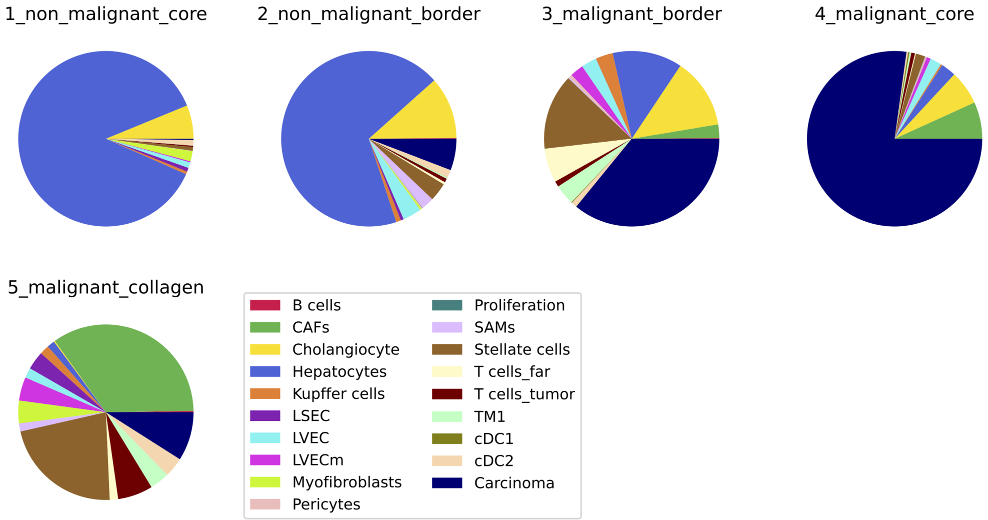

Figure 2: Pie charts produced by `cellanneal`. Each pie corresponds to one deconvolved bulk sample from the input data.

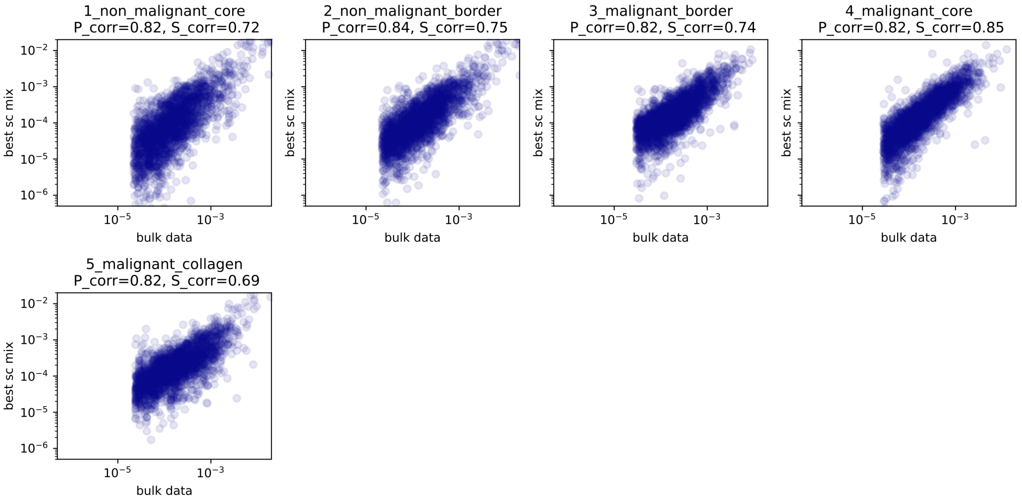

Figure 3: Gene correlation scatter plots produced by `cellanneal` Each panel corresponds to one deconvolved bulk sample from the input data. Each dot represents a gene used during deconvolution. The x-axis shows the experimentally measured expression of each gene after normalizing so that the total count sum is 1. The y-axis shows the normalized expression of each gene in the best identified synthetic bulk mixed from cell type signature data .

`cellanneal` relies on the python packages `scipy` (Virtanen et al., 2020), `numpy` (Harris et al., 2020), `pandas` (Pandas Development Team, 2020), `seaborn` (Waskom, 2021) and `matplotlib` (Hunter, 2007).

Citations

Examples of published research projects using cellanneal include (Egozi et al., 2023) and (Berková et al., 2022).

Acknowledgements

We thank all members of the Itzkovitz lab for valuable feedback and for testing the software.

L.B. was supported by the European Molecular Biology Organization under EMBO Long-Term Fellowship ALTF 724-2019. S.I. is supported by the Wolfson Family Charitable Trust, the Edmond de Rothschild Foundations, the Fannie Sherr Fund, the Helen and Martin Kimmel Institute for Stem Cell Research grant, the Minerva Stiftung grant, the Israel Science Foundation grant No. 1486/16, the Broad Institute-Israel Science Foundation grant No. 2615/18, the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program grant No. 768956, the Chan Zuckerberg Initiative grant No. CZF2019-002434, the Bert L. and N. Kuggie Vallee Foundation and the Howard Hughes Medical Institute (HHMI) international research scholar award.

References

Aliee, H., & Theis, F. J. (2021). AutoGeneS: Automatic gene selection using multi-objective optimization for RNA-seq deconvolution. *Cell Systems*, *12*(7), 706–715.e4. https://doi.org/10.1016/j.cels.2021.05.006

Berková, L., Fazilaty, H., Yang, Q., Kubovčiak, J., Stastna, M., Hrckulak, D., Vojtechova, M., Brügger, M. D., Hausmann, G., Liberali, P., & others. (2022). Terminal differentiation of villus-tip enterocytes is governed by distinct members of tgfβ superfamily. *bioRxiv*, 2022–2011. https://doi.org/10.1101/2022.11.11.516138

Cobos, F. A., Alquicira-Hernandez, J., Powell, J. E., Mestdagh, P., & De Preter, K. (2020). Benchmarking of cell type deconvolution pipelines for transcriptomics data. *Nature Communications*, *11*(1), 1–14. https://doi.org/10.1038/s41467-020-19015-1

Egozi, A., Olaloye, O., Werner, L., Silva, T., McCourt, B., Pierce, R. W., An, X., Wang, F., Chen, K., Pober, J. S., & others. (2023). Single-cell atlas of the human neonatal small intestine affected by necrotizing enterocolitis. *PloS Biology*, *21*(5), e3002124. https://doi.org/10.1371/journal.pbio.3002124

Erdmann-Pham, D. D., Fischer, J., Hong, J., & Song, Y. S. (2021). Likelihood-based deconvolution of bulk gene expression data using single-cell references. *Genome Research*, *31*(10), 1794–1806. https://doi.org/10.1101/gr.272344.120

Hao, Y., Yan, M., Heath, B. R., Lei, Y. L., & Xie, Y. (2019). Fast and robust deconvolution of tumor infiltrating lymphocyte from expression profiles using least trimmed squares. *PLoS Computational Biology*, *15*(5), e1006976. https://doi.org/10.1371/journal.pcbi.1006976

Harris, C. R., Millman, K. J., Walt, S. J. van der, Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., Kern, R., Picus, M., Hoyer, S., Kerkwijk, M. H. van, Brett, M., Haldane, A., Río, J. F. del, Wiebe, M., Peterson, P., … Oliphant, T. E. (2020). Array programming with NumPy. *Nature*, *585*(7825), 357–362. https://doi.org/10.1038/s41586-020-2649-2

Hunter, J. D. (2007). Matplotlib: A 2D graphics environment. *Computing in Science & Engineering*, *9*(3), 90–95. https://doi.org/10.1109/MCSE.2007.55

Jew, B., Alvarez, M., Rahmani, E., Miao, Z., Ko, A., Garske, K. M., Sul, J. H., Pietiläinen, K. H., Pajukanta, P., & Halperin, E. (2020). Accurate estimation of cell composition in bulk expression through robust integration of single-cell information. *Nature Communications*, *11*(1), 1–11. https://doi.org/10.1038/s41467-020-15816-6

Kirkpatrick, S., Gelatt, C. D., & Vecchi, M. P. (1983). Optimization by simulated annealing. *Science*, *220*(4598), 671–680. https://doi.org/10.1126/science.220.4598.671

Li, B., Severson, E., Pignon, J.-C., Zhao, H., Li, T., Novak, J., Jiang, P., Shen, H., Aster, J. C., Rodig, S., & others. (2016). Comprehensive analyses of tumor immunity: Implications for cancer immunotherapy. *Genome Biology*, *17*(1), 1–16. https://doi.org/10.1186/s13059-016-1028-7

Massalha, H., Bahar Halpern, K., Abu-Gazala, S., Jana, T., Massasa, E. E., Moor, A. E., Buchauer, L., Rozenberg, M., Pikarsky, E., Amit, I., Zamir, G., & Itzkovitz, S. (2020). A single cell atlas of the human liver tumor microenvironment. *Mol. Syst. Biol.*, *16*(12), e9682. https://doi.org/10.15252/msb.20209682

Newman, A. M., Liu, C. L., Green, M. R., Gentles, A. J., Feng, W., Xu, Y., Hoang, C. D., Diehn, M., & Alizadeh, A. A. (2015). Robust enumeration of cell subsets from tissue expression profiles. *Nature Methods*, *12*(5), 453–457. https://doi.org/10.1038/nmeth.3337

Newman, A. M., Steen, C. B., Liu, C. L., Gentles, A. J., Chaudhuri, A. A., Scherer, F., Khodadoust, M. S., Esfahani, M. S., Luca, B. A., Steiner, D., & others. (2019). Determining cell type abundance and expression from bulk tissues with digital cytometry. *Nature Biotechnology*, *37*(7), 773–782. https://doi.org/10.1038/s41587-019-0114-2

Pandas Development Team. (2020). *Pandas-dev/pandas: pandas* (latest). Zenodo. https://doi.org/10.5281/zenodo.3509134

Racle, J., Jonge, K. de, Baumgaertner, P., Speiser, D. E., & Gfeller, D. (2017). Simultaneous enumeration of cancer and immune cell types from bulk tumor gene expression data. *eLife*, *6*, e26476. https://doi.org/10.7554/eLife.26476

Schwanhäusser, B., Busse, D., Li, N., Dittmar, G., Schuchhardt, J., Wolf, J., Chen, W., & Selbach, M. (2011). Global quantification of mammalian gene expression control. *Nature*, *473*(7347), 337–342. https://doi.org/10.1038/nature10098

Sturm, G., Finotello, F., Petitprez, F., Zhang, J. D., Baumbach, J., Fridman, W. H., List, M., & Aneichyk, T. (2019). Comprehensive evaluation of transcriptome-based cell-type quantification methods for immuno-oncology. *Bioinformatics*, *35*(14), i436–i445. https://doi.org/10.1093/bioinformatics/btz363

Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S. J., Brett, M., Wilson, J., Millman, K. J., Mayorov, N., Nelson, A. R. J., Jones, E., Kern, R., Larson, E., … SciPy 1.0 Contributors. (2020). SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, *17*, 261–272. https://doi.org/10.1038/s41592-019-0686-2

Wang, X., Park, J., Susztak, K., Zhang, N. R., & Li, M. (2019). Bulk tissue cell type deconvolution with multi-subject single-cell expression reference. *Nature Communications*, *10*(1), 1–9. https://doi.org/10.1038/s41467-018-08023-x

Waskom, M. L. (2021). Seaborn: Statistical data visualization. *Journal of Open Source Software*, *6*(60), 3021. https://doi.org/10.21105/joss.03021