The Journal of Open Source Software

mxnorm: An R Package to Normalize Multiplexed Imaging Data

Coleman Harris[1¶], **Julia Wrobel**[2*], **and Simon Vandekar**[1*]

1 Department of Biostatistics, Vanderbilt University Medical Center, Nashville, TN, USA **2** Department of Biostatistics & Informatics, Colorado School of Public Health, Aurora, CO, USA ¶ Corresponding author * These authors contributed equally.

DOI: 10.21105/joss.04180

Software

- Review
- Repository
- Archive

Editor: Øystein Sørensen

Reviewers:

- @askerdb
- @tijeco

Submitted: 15 February 2022
Published: 30 March 2022

Summary

Multiplexed imaging is an emerging single-cell assay that can be used to understand and analyze complex processes in tissue-based cancers, autoimmune disorders, and more. These imaging technologies, which include co-detection by indexing (CODEX), multiplexed ion beam imaging (MIBI), and multiplexed immunofluorescence imaging (MxIF), provide detailed information about spatial interactions between cells (Angelo et al., 2014; Gerdes et al., 2013; Goltsev et al., 2018). Multiplexed imaging experiments generate data across hundreds of slides and images, often resulting in terabytes of complex data to analyze through imaging analysis pipelines. Methods are rapidly developing to improve particular parts of the pipeline, including software packages in R and Python like `spatialTime`, `imcRtools`, `MCMICRO`, and `Squidpy` (Creed et al., 2021; Palla et al., 2021; Schapiro et al., 2021; Windhager et al., 2021). An important, but understudied component of this pipeline is the analysis of technical variation within this complex data source – intensity normalization is one way to remove this technical variability. The combination of disparate pre-processing pipelines, imaging variables, optical effects, and within-slide dependencies create batch and slide effects that can be reduced via normalization methods. Current state-of-the-art methods vary heavily across research labs and image acquisition platforms, without one singular method that is uniformly robust – optimal statistical methods seek to improve similarity across images and slides by removing this technical variability while maintaining the underlying biological signal in the data.

`mxnorm` is open-source software built with R and S3 methods that implements, evaluates, and visualizes normalization techniques for multiplexed imaging data. Extending methodology described in Harris et al. (2022), we intend to set a foundation for the evaluation of multiplexed imaging normalization methods in R. This easily allows users to extend normalization methods into the field, and provides a robust evaluation framework to measure both technical variability and the efficacy of various normalization methods. One key component of the R package is the ability to supply user-defined normalization methods and thresholding algorithms to assess normalization in multiplexed imaging data. Core features, usage details, and extensive tutorials are available in the package documentation and vignette on CRAN and the software repository.

Statement of need

Multiplexed imaging measures intensities of dozens of antibody and protein markers at the single-cell level while preserving cell spatial coordinates. This allows single-cell analyses to be performed on biological samples like tissues and tumors, much like single-cell RNA sequencing, with the added benefit of *in situ* coordinates to better capture spatial interactions between individual cells (Chen et al., 2021; McKinley et al., 2022). Current research using platforms like MxIF and MIBI demonstrate this growing field that seeks to better understand cell-cell populations in cancer, pre-cancer, and various biological research contexts (Gerdes et al., 2013; Ptacek et al., 2020).

In contrast to the field of sequencing & micro-array data and the established software, analysis, and methods therein, multiplexed imaging lacks established analysis standards, pipelines, and methods. Recent developments in multiplexed imaging seek to address the broad lack of standardized tools – the MCMICRO pipeline seeks to provide a set of open-source, reproducible analyses to transform whole-slide images into single-cell data (Schapiro et al., 2021). Researchers in the field have also developed a ground truth dataset to evaluate differences in batch effects and normalization methods (Graf et al., 2022), while other open issues in the field that may produce open-source solutions include tissue segmentation, end-to-end image processing, and removal of image artifacts. With this diversity of open issues in multiplexed imaging, our work focuses specifically on normalization methods and evaluating these results in multiplexed imaging data. Namely, standard normalization software in the sequencing field includes open-source packages in R and Python like sva, limma, and Scanorama (Hie et al., 2019; Leek et al., 2012; Smyth, 2005), but an analogue for evaluating and developing normalization methods does not exist for multiplexed imaging data.

We recently proposed and evaluated several normalization methods for multiplexed imaging data, which along with other recent work shows that normalization methods are important in reducing slide-to-slide variation (Burlingame et al., 2021; Chang et al., 2020; Harris et al., 2022). These recently developed algorithms are the beginning of contributions to normalization literature, but lack a simple, user-friendly implementation. Further, there is no software researchers can use to develop and evaluate normalization methods in their own multiplexed imaging data; multiplexed imaging software is limited mostly to Matlab, Python, and only a scattered few R packages exist. Two prominent packages, cytomapper and giotto, contain open-source implementations for analysis and visualization of highly multiplexed images (Dries et al., 2021; Eling et al., 2020), but do not explicitly address normalization of the single-cell intensity data. Hence, there is a major lack of available tools for researchers to explore, evaluate, and analyze normalization methods in multiplexed imaging data. The mxnorm package provides this framework, with easy-to-implement and customizable normalization methods along with a foundation for evaluating their utility in the multiplexed imaging field.

Functionality

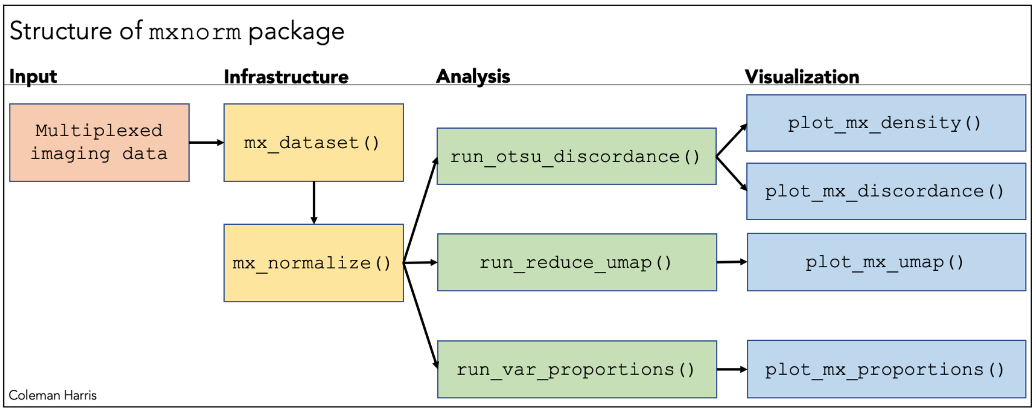

Figure 1: Basic structure of the mxnorm package and associated functions

As shown in **Figure 1**, there are three main types of functions implemented in the mxnorm package – infrastructure, analysis, and visualization. The first infrastructure function, mx_dataset(), specifies and creates the S3 object used throughout the analysis, while the mx_normalize() function provides a routine to normalize the multiplexed imaging data, which specifically allows for normalization algorithms defined by the user. Each of the three analysis functions provides methods to run specific analyses that test for slide-to-slide variation and preservation of biological signal for the normalized and unnormalized data, while the four visualization functions provide methods to generate ggplot2 plots to assess the results. We also extend the summary() generic function to the mx_dataset S3 object to provide further statistics and summaries.

The statistical methodology underlying the methods we implemented in mxnorm builds upon existing work in both R and Python. Normalization algorithms available in mx_normalize() leverage methodology derived in the ComBat paper, the fda package, and the tidyverse framework (Johnson et al., 2007; Ramsay et al., 2021; Wickham et al., 2019). The threshold discordance methods available in run_otsu_discordance() leverage methodology from Otsu's original paper and the scikit-image implementation of Otsu thresholding in Python (Otsu, 1979; Walt et al., 2014). Our implementation of the UMAP algorithm in run_reduce_umap() leverages both the UMAP paper and the uwot implementation of the UMAP algorithm in R (McInnes et al., 2018; Melville, 2021). The random effects modeling options available in run_var_proportions() leverage the lme4 R package (Bates et al., 2015). Even more information for the statistical methodology behind these normalization and analysis methods are detailed further in our package vignette and in the methods paper (Harris et al., 2022).

A minimal example

The following code is a simplified example of a normalization analysis applied to the sample dataset included in the mxnorm package, mx_sample. Here we specify the creation of the S3 object, normalize using the mean_divide method, run a set of analyses to compare our normalized data with the unnormalized data, and finally generate summary statistics and plots to understand the results.

```
## load package
library(mxnorm)

## create S3 object & normalize
mx_data = mx_dataset(mx_sample, ”slide_id”, ”image_id”,
                     c(”marker1_vals”,”marker2_vals”,”marker3_vals”),
                     c(”metadata1_vals”))
mx_data = mx_normalize(mx_data, ”mean_divide”, ”None”)

## run analyses
mx_data = run_otsu_discordance(mx_data, ”both”)
mx_data = run_reduce_umap(mx_data, ”both”,
                         c(”marker1_vals”,”marker2_vals”,”marker3_vals”))
mx_data = run_var_proportions(mx_data, ”both”)

## results and plots
summ_mx_data = summary(mx_data)
p1 = plot_mx_density(mx_data)
p2 = plot_mx_discordance(mx_data)
p3 = plot_mx_umap(mx_data, ”slide_id”)
p4 = plot_mx_proportions(mx_data)
```

Acknowledgements

We would like to extend sincere thanks to Samantha Bowell for her feedback on the mxnorm package. We would also like to thank Eliot McKinley, Joseph Roland, Qi Liu, Martha Shrubsole, Ken Lau, and Robert Coffey for their help in making this work possible. This work was supported by NIH grants U2CCA233291 and R01MH123563.

References

Angelo, M., Bendall, S. C., Finck, R., Hale, M. B., Hitzman, C., Borowsky, A. D., Levenson, R. M., Lowe, J. B., Liu, S. D., Zhao, S., & others. (2014). Multiplexed ion beam imaging of human breast tumors. *Nature Medicine*, *20*(4), 436–442. https://doi.org/10.1038/nm.3488

Bates, D., Maechler, M., & Bolker, B. (2015). Fitting linear mixed-effects models using lme4. *Journal of Statistical Software*, *67*(1), 1–48. https://doi.org/10.18637/jss.v067.i01

Burlingame, E. A., Eng, J., Thibault, G., Chin, K., Gray, J. W., & Chang, Y. H. (2021). Toward reproducible, scalable, and robust data analysis across multiplex tissue imaging platforms. *Cell Reports Methods*, *1*(4), 100053. https://doi.org/10.1016/j.crmeth.2021.100053

Chang, Y. H., Chin, K., Thibault, G., Eng, J., Burlingame, E., & Gray, J. W. (2020). RESTORE: Robust intEnSiTy nORmalization mEthod for multiplexed imaging. *Communications Biology*, *3*(1), 1–9. https://doi.org/10.1038/s42003-020-0828-1

Chen, B., Cherie'R, S., McKinley, E. T., Simmons, A. J., Ramirez-Solano, M. A., Zhu, X., Markham, N. O., Heiser, C. N., Vega, P. N., Rolong, A., & others. (2021). Differential pre-malignant programs and microenvironment chart distinct paths to malignancy in human colorectal polyps. *Cell*, *184*(26), 6262–6280. https://doi.org/10.1016/j.cell.2021.11.031

Creed, J. H., Wilson, C. M., Soupir, A. C., Colin-Leitzinger, C. M., Kimmel, G. J., Ospina, O. E., Chakiryan, N. H., Markowitz, J., Peres, L. C., Coghill, A., & others. (2021). spatialTIME and iTIME: R package and shiny application for visualization and analysis of immunofluorescence data. *Bioinformatics*, *37*(23), 4584–4586. https://doi.org/10.1093/bioinformatics/btab757

Dries, R., Zhu, Q., Dong, R., Eng, C.-H. L., Li, H., Liu, K., Fu, Y., Zhao, T., Sarkar, A., Bao, F., & others. (2021). Giotto: A toolbox for integrative analysis and visualization of spatial expression data. *Genome Biology*, *22*(1), 1–31. https://doi.org/10.1186/s13059-021-02286-2

Eling, N., Damond, N., Hoch, T., & Bodenmiller, B. (2020). Cytomapper: An R/Bioconductor package for visualization of highly multiplexed imaging data. *Bioinformatics*, *36*(24), 5706–5708. https://doi.org/10.1093/bioinformatics/btaa1061

Gerdes, M. J., Sevinsky, C. J., Sood, A., Adak, S., Bello, M. O., Bordwell, A., Can, A., Corwin, A., Dinn, S., Filkins, R. J., & others. (2013). Highly multiplexed single-cell analysis of formalin-fixed, paraffin-embedded cancer tissue. *Proceedings of the National Academy of Sciences*, *110*(29), 11982–11987. https://doi.org/10.1073/pnas.1300136110

Goltsev, Y., Samusik, N., Kennedy-Darling, J., Bhate, S., Hale, M., Vazquez, G., Black, S., & Nolan, G. P. (2018). Deep profiling of mouse splenic architecture with CODEX multiplexed imaging. *Cell*, *174*(4), 968–981. https://doi.org/10.1016/j.cell.2018.07.010

Graf, J., Cho, S., McDonough, E., Corwin, A., Sood, A., Lindner, A., Salvucci, M., Stachtea, X., Van Schaeybroeck, S., Dunne, P. D., & others. (2022). FLINO: A new method for immunofluorescence bioimage normalization. *Bioinformatics*, *38*(2), 520–526. https://doi.org/10.1093/bioinformatics/btab686

Harris, C. R., McKinley, E. T., Roland, J. T., Liu, Q., Shrubsole, M. J., Lau, K. S., Coffey, R. J., Wrobel, J., & Vandekar, S. N. (2022). Quantifying and correcting slide-to-slide variation in multiplexed immunofluorescence images. *Bioinformatics*, btab877. https://doi.org/10.1093/bioinformatics/btab877

Hie, B., Bryson, B., & Berger, B. (2019). Efficient integration of heterogeneous single-cell transcriptomes using Scanorama. *Nature Biotechnology*, *37*(6), 685–691. https://doi.org/10.1038/s41587-019-0113-3

Johnson, W. E., Li, C., & Rabinovic, A. (2007). Adjusting batch effects in microarray expression data using empirical bayes methods. *Biostatistics*, *8*(1), 118–127. https://doi.org/10.1093/biostatistics/kxj037

Leek, J. T., Johnson, W. E., Parker, H. S., Jaffe, A. E., & Storey, J. D. (2012). The sva package for removing batch effects and other unwanted variation in high-throughput experiments. *Bioinformatics*, *28*(6), 882–883. https://doi.org/10.1093/bioinformatics/bts034

McInnes, L., Healy, J., & Melville, J. (2018). Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv Preprint arXiv:1802.03426 [Stat.ML]*. https://doi.org/10.48550/arXiv.1802.03426

McKinley, E. T., Shao, J., Ellis, S. T., Heiser, C. N., Roland, J. T., Macedonia, M. C., Vega, P. N., Shin, S., Coffey, R. J., & Lau, K. S. (2022). MIRIAM: A machine and deep learning single-cell segmentation and quantification pipeline for multi-dimensional tissue images. *Cytometry Part A*. https://doi.org/10.1002/cyto.a.24541

Melville, J. (2021). *Uwot: The uniform manifold approximation and projection (UMAP) method for dimensionality reduction*. https://CRAN.R-project.org/package=uwot

Otsu, N. (1979). A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, *9*(1), 62–66. https://doi.org/10.1109/tsmc.1979.4310076

Palla, G., Spitzer, H., Klein, M., Fischer, D., Schaar, A. C., Kuemmerle, L. B., Rybakov, S., Ibarra, I. L., Holmberg, O., Virshup, I., & others. (2021). Squidpy: A scalable framework for spatial single cell analysis. *bioRxiv*. https://doi.org/10.1101/2021.02.19.431994

Ptacek, J., Locke, D., Finck, R., Cvijic, M.-E., Li, Z., Tarolli, J. G., Aksoy, M., Sigal, Y., Zhang, Y., Newgren, M., & others. (2020). Multiplexed ion beam imaging (MIBI) for characterization of the tumor microenvironment across tumor types. *Laboratory Investigation*, *100*(8), 1111–1123. https://doi.org/10.1038/s41374-020-0417-4

Ramsay, J. O., Graves, S., & Hooker, G. (2021). *fda: Functional data analysis*. https://CRAN.R-project.org/package=fda

Schapiro, D., Sokolov, A., Yapp, C., Chen, Y.-A., Muhlich, J. L., Hess, J., Creason, A. L., Nirmal, A. J., Baker, G. J., Nariya, M. K., & others. (2021). MCMICRO: A scalable, modular image-processing pipeline for multiplexed tissue imaging. *Nature Methods*, 1–5. https://doi.org/10.1101/2021.03.15.435473

Smyth, G. K. (2005). Limma: Linear models for microarray data. In *Bioinformatics and computational biology solutions using R and Bioconductor* (pp. 397–420). Springer. https://doi.org/10.1007/0-387-29362-0_23

Walt, S. van der, Schönberger, J. L., Nunez-Iglesias, J., Boulogne, F., Warner, J. D., Yager, N., Gouillart, E., Yu, T., & contributors, the scikit-image. (2014). scikit-image: Image processing in Python. *PeerJ*, *2*, e453. https://doi.org/10.7717/peerj.453

Wickham, H., Averick, M., Bryan, J., Chang, W., McGowan, L. D., François, R., Grolemund, G., Hayes, A., Henry, L., Hester, J., & others. (2019). Welcome to the tidyverse. *Journal of Open Source Software*, *4*(43), 1686. https://doi.org/10.21105/joss.01686

Windhager, J., Bodenmiller, B., & Eling, N. (2021). An end-to-end workflow for multiplexed image processing and analysis. *bioRxiv*. https://doi.org/10.1101/2021.11.12.468357