Resvidex: An R package for molecular classification of Respiratory Syncytial Virus (HRSV) sequences

06 august 2024

# Summary

Resvidex aims to facilitate the classification of human respiratory syncytial virus (HRSV) sequences at the lower levels of resolution. It can handle both whole genome and partial sequences (three classification models). Resvidex comes with its own shiny app for an user-friendly option.

# Statement of need

The HRSV is one of the leading causes of acute lower respiratory tract infection in children, elderly and immunocompromised individuals. Below species level, there are two antigenic groups: HRSV subgroup A (HRSV-A) and B (HRSV-B). Within each subgroup, genotypes are defined based on statistically supported phylogenetic clades that can be inferred with the second hypervariable region (2HR) of the G gene, which encodes the attachment glycoprotein and exhibits the highest genetic and antigenic variability.

Clade classification typically involves analyzing gene sequences from current strains alongside a set of reference sequences using phylogenetic analysis. This process is usually time-consuming and requires specialized training and equipment. Alternatively, advanced machine learning methodologies have demonstrated their ability to provide accurate predictions by employing algorithms capable of uncovering intricate patterns within relevant viral datasets (Cacciabue and Marcone 2023; Humayun et al. 2021; Wang Y 2020).

Here we introduce resvidex, an open-source R package (R Core Team 2023), dedicated to aid researchers in classifying HRSV sequences (full genome, G gene or G+F region) at the lower levels of resolution in an easy, fast and reproducible way. Resvidex is a tool based on alignment‐free machine learning for HRSV classification into subtypes and clades. It is sensitive, specific, and ready to implement, as it is available to run locally for R users. It also includes a web application (Chang et al. 2023) that has a user‐friendly interface. Additionally, it can be tested on an internet connection without any installation (only for small datasets).

The overall classification algorithm that Resvidex uses is divided into three majors steps. In the initial phase, the user data is loaded in a multifasta format, and the k-mer counting operation is executed utilizing the k-mer package (Wilkinson 2018). Each count of k-mers undergoes normalization based on both the k-mer size (k = 6) and the length of the sequence. Alternatively, the user can copy and paste the query sequence directly to the app. In the second step, the predict function from the ranger package (Wright and Ziegler 2015) is invoked using a pre-trained random forest model. It calculates a probability score through a majority vote rule. Using this score, the application determines the classification score for each query sequence. Additionally,the app also calculates the proportion of N bases in the genome and the genome length. These values are important as divergencies from the expected values can impact notably over the classification results. On the final step, sequences are separated in two tables, one showing the sequences that passed all the quality checks and another with sequences that did not pass at least one of the filter steps. These filters ensure that each sequence achieves a probability score of 0.4 or higher, that the sequence length aligns closely with the expected length for the classification model (with a tolerance of up to 50%), and that the proportion of ambiguous bases (N) in the sequence does not exceed 2% of the genome length. Sequences that do not meet the necessary criteria should be analyzed manually with other methodologies (i.e. alignment-dependent tools) that may shield a more robust result. Although not recommended, the app allows the user to manually tweak these filters. Additionally, a concise report can be generated, incorporating the results table, date of analysis, and model information.

Resvidex was designed to be used by researchers who want to classify their samples of HRSV according to the Goya et. al. proposal (Goya et al. 2024). It comes with three classification models: one for whole genome sequences (“FULL\_GENOME”, 15000 nt), one for sequences that cover the G coding region (“G”, 900 nt) and one for sequences that cover the G+F coding region (“G\_F”, 2800 nt). The HRSV classification comprises 41 clades or genetic groups: 25 for subgroup A and 16 for subgroup B.

# Examples

The main functions of resvidex are the following:

* kcounter() : count and normalize the k-mers present in each sequence.
* prediction\_caller() : perform the classification based on the pretrained classification model.
* quality\_control() and quality\_filter() : add the corresponding quality FLAGs.

Additionally, classify() acts like a wrapper function, enabling the handling of all the above functions in one simple step, for example:

#load the library  
library(resvidex)  
  
# In this example, we use a test file provided with the package.   
  
file\_path<-system.file("extdata","test\_dataset.fasta",package="resvidex")  
  
# Use the wrapper function. You can change the classification model and pass other arguments as needed.  
classify(inputFile=file\_path,model=FULL\_GENOME)

Alternatively, the user can fire up the resvidex shiny app using the run\_shiny\_app() function.

Other examples are available as vignettes: How to use the shiny app [vignette](https://marcocacciabue.github.io/resvidex/articles/01_resvidex_vignette.html), step-by-step explanation of a in-built [example](https://marcocacciabue.github.io/resvidex/articles/02_resvidex_vignette_R.html), and another example with a [larger dataset](https://marcocacciabue.github.io/resvidex/articles/04_an_example.html).

# Acknowledgements

# References

Cacciabue, Marco, and Débora N. Marcone. 2023. “INFINITy: A Fast Machine Learning‐based Application for Human Influenza A and B Virus Subtyping.” *Influenza and Other Respiratory Viruses* 17 (1): e13096. <https://doi.org/10.1111/irv.13096>.

Chang, Winston, Joe Cheng, JJ Allaire, Carson Sievert, Barret Schloerke, Yihui Xie, Jeff Allen, Jonathan McPherson, Alan Dipert, and Barbara Borges. 2023. *Shiny: Web Application Framework for r*. <https://CRAN.R-project.org/package=shiny>.

Goya, Stephanie, Christopher Ruis, Richard A. Neher, Adam Meijer, Ammar Aziz, Angie S. Hinrichs, Anne Von Gottberg, et al. 2024. “The Unified Proposal for Classification of Human Respiratory Syncytial Virus Below the Subgroup Level.” Preprint. Infectious Diseases (except HIV/AIDS). <https://doi.org/10.1101/2024.02.13.24302237>.

Humayun, Fahad, Fatima Khan, Nasim Fawad, Shazia Shamas, Sahar Fazal, Abbas Khan, Arif Ali, Ali Farhan, and Dong-Qing Wei. 2021. “Computational Method for Classification of Avian Influenza A Virus Using DNA Sequence Information and Physicochemical Properties.” *Frontiers in Genetics* 12 (January): 599321. <https://doi.org/10.3389/fgene.2021.599321>.

R Core Team. 2023. “R: A Language and Environment for Statistical Computing.” Vienna, Austria: R Foundation for Statistical Computing. <https://www.r-project.org/>.

Wang Y, Du J, Bao J. 2020. “Rapid Detection and Prediction of Influenza a Subtype Using Deep Convolutional Neural Network Based Ensemble Learning.” In *Proceedings of the 2024 14th International Conference on Bioscience, Biochemistry and Bioinformatics*. Kyoto Japan: ACM.

Wilkinson, Shaun. 2018. “Kmer: An r Package for Fast Alignment-Free Clustering of Biological Sequences.” *GitHub Repository*. <https://doi.org/10.5281/zenodo.1227690>.

Wright, Marvin N., and Andreas Ziegler. 2015. “Ranger: A Fast Implementation of Random Forests for High Dimensional Data in C++ and R.” <https://doi.org/10.48550/ARXIV.1508.04409>.