Condensed paragraph version

We used the auc function from the MESS [1] package as auc(x, y, from = 0, type = ”spline”, absolutearea = True) if a subdivision error occued then we used auc(x, y, from = 0, type = ”spline”, absolutearea = True, subdivision = 1000) and increased the subdivision value until the error was resolved. We exported all the auc values into .csv files before importing that data into GraphPad PRISM for visualization.

Expanded list version (supplemental?)

1. Install the required packages: {MESS} [**1**], {readr} [**2**], {ggplot2} [**3**], {tidyr} [**4**], and {dplyr} [**5**]. The user may either use the Packages tab or the install.packages function. The packages will remain installed in the environment using R. We use the if require template for convenience as this only installs the packages if the environment does not have them installed.
2. Load the installed packages by using the library(“package-name”) function.

Text

Description automatically generated

1. Format data for compilation. The format in our case is a column of distance in microns values followed by columns of fluorescence values with headers that will be used as identifiers.

Table, Excel

Description automatically generated

1. Convert the raw fluorescence values into R dataframes.
   1. Load the main .csv file which contains all the fluorescence values in columns with a header that labels the columns with unique identifiers, these will be the row 1 values.
   2. Use the grepl function to create dataframes of each condition by using the identifiers.

Text, chat or text message

Description automatically generated

1. Use the MESS AUC function to calculate the AUC with a spline.

The MESS AUC function uses the splinefun function and an integration function to calculate an integral for the area under the curve of a spline that fits the curve that is a function of x and y.

* 1. The x in this function are the distances in microns along the line scan.
  2. The y in this function are the fluorescence values at those distances. In this case we use the brackets to grab a specific column and the the drop = TRUE argument to make our wide, or multiple columns, dataframe into a long, or multiple rows, dataframe.



* 1. Use the type = “spline” argument.
  2. Choose an appropriate number of subdivisions.
  3. Use absolutearea = TRUE if dealing with negative Y values.
  4. Use from = minimum and to = maximum to only calculate AUC values from a minimum x value to a maximum x value.



1. Create a ggplot2 multiplot that is ordered by decreasing AUC to visualize the resulting AUC data.

Text

Description automatically generated

The x = reorder(Identified, -AUC\_Values) is the argument that orders by decreasing AUC value.

1. Save the dotplot using the ggsave function.



1. Save the dataframe used to create the dotplot using the write.csv function.



Citations

1. Claus Thorn Ekstrøm (2022). MESS: Miscellaneous Esoteric Statistical Scripts. R package

version 0.5.9. <https://CRAN.R-project.org/package=MESS>

2. Hadley Wickham, Jim Hester and Jennifer Bryan (2022). readr: Read Rectangular Text Data.

R package version 2.1.2. <https://CRAN.R-project.org/package=readr>

3. H. Wickham. ggplot2: Elegant Graphics for Data Analysis. Springer-Verlag New York, 2016.

4. Hadley Wickham and Maximilian Girlich (2022). tidyr: Tidy Messy Data. R package version

1.2.0. <https://CRAN.R-project.org/package=tidyr>

5. Hadley Wickham, Romain François, Lionel Henry and Kirill Müller (2022). dplyr: A Grammar

of Data Manipulation. R package version 1.0.10. <https://CRAN.R-project.org/package=dplyr>