Technical Bulletin – BIN06
RBioplot: Automated statistical analysis and data visualization
Hanane Hadj-Moussa and Jing Zhang PhD, 2016.

Overall:

RBioplot is an R package designed for highly accessible yet comprehensive statistical analysis and data visualization. It represents a fully automated and versatile data processing solution for molecular biology and biochemistry. We need to break free from the confinements of SigmaPlot 12 and Jing Zhang PhD has developed RBioplot to help us do just that. RBioplot is a black box that will do all our graphing, plotting, and statistical analyses for us with minimal data reorganization, time, and energy. This techbull is the RBioplot for dummies user's manual.

***Note:** This techbull will refer you to various websites and to the RBioplot webpage that is hosted on our lab website (http://kenstoreylab.com/?page_id=2448), where you can find all the sample input and output files used to make this techbull.

Before using RBioplot you should read the following paper that explains the core principles, functions, commands, and selected options. Also, the package is free of charge.

If using this program for data analysis and generating figures when publishing, remember to cite the paper below in your methods (or other appropriate) section:

Zhang J, Storey KB. (2016) RBioplot: an easy-to-use R pipeline for automated statistical analysis and data visualization in molecular biology and biochemistry. PeerJ 4:e2436.

I. Installing R and RStudio

To run RBioplot you must first install R and RStudio on your computer. RStudio is an IDE (interactive development environment) that allows you to interact with R in a more user friendly way. RStudio will be the main interface for R – even though you could launch R console alone, we almost never do so.

Links to both of these programs are on the lab website (www.kensotreylab.com) → Research → Research Tools → RBioplot (http://kenstoreylab.com/?page_id=2448). Both R and RStudio are open-source software, meaning they are completely free. Additionally, both programs are cross platform, ie they will run on Windows, Mac and Linux computers.

***Note: You will find that the "*Notes" in this techbull will be useful for all the applications, not just the one it is listed in.**

A. Installing "R"

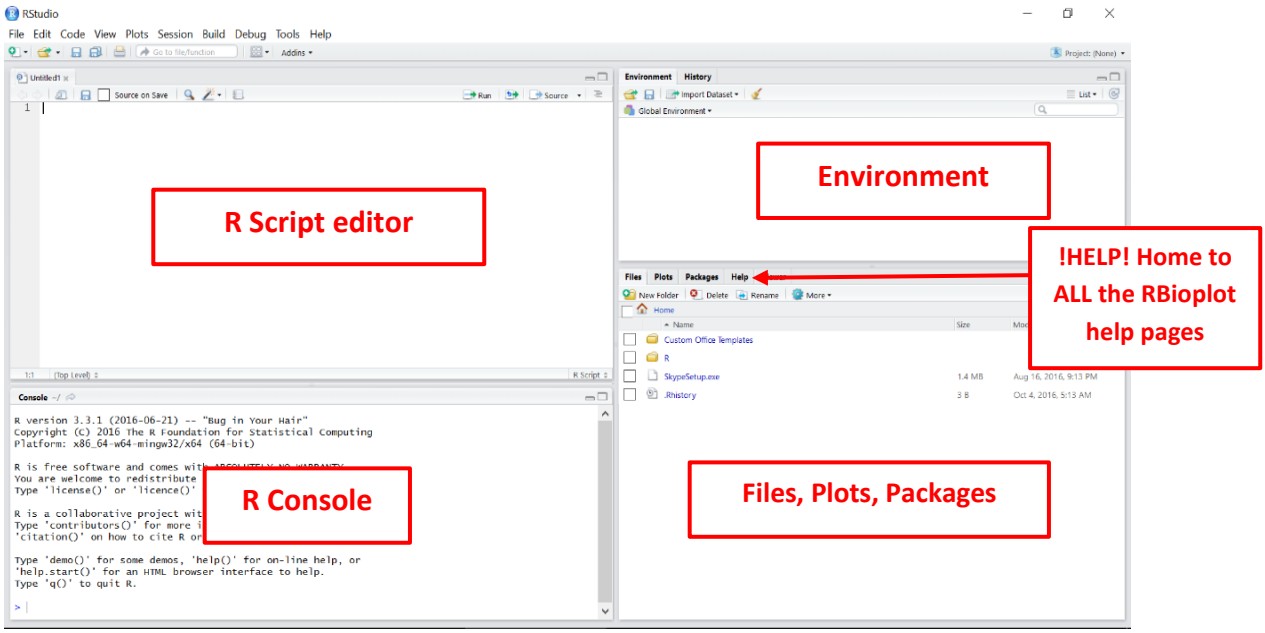
1) Install '**R**' by visiting (https://www.r-project.org) and then selecting your preferred CRAN mirror to download the installation file. We will use the University of Toronto's (http://cran.utstat.utoronto.ca/).
2) Download the version of R that corresponds with your operating system. For example, if you are working on a Windows computer click **Download R for Windows → install R for the first time → Download R 3.3.1 (or the latest version number if not 3.3.1) for Windows.**
3) Save the installation .exe file → open the file → follow the R for Windows 3.3.1 Setup Wizard's simple installation instructions.
   ***Note:** You should install all the installation components and make sure to accept the default start-up options.

B. Installing RStudio

1) To install **RStudio** visit (https://www.rstudio.com/). Click **RStudio → Desktop → Download RStudio Desktop**
2) Download the RStudio version that corresponds with your operating system. For example, if you are working on a Windows computer click **RStudio 0.99.903 (or the latest version number if not 0.99.903) - Windows Vista/7/8/10**
3) Save the installation .exe file → open the file → follow the RStudio Setup Wizard's simple installation instructions.
   ***Note:** Once the program has installed you should locate your RStudio shortcut and place it on your desktop.

II. Setting up RBioplot

Open the RStudio application, this will be the platform you will use to run RBioplot. The picture below is a breakdown of the different RStudio quadrants and panels. **EVERYTHING IN RSTUDIO IS CASE-SENSITIVE**



1) To setup the R Script text editor go to **File → New File → R Script (make sure to select R Script, otherwise you will not have the option to run your commands)**. The R Script editor is the main window in which you will interact with R. You should prepare all your R commands before running your commands in the console. Also, the R script editor allows you to run multiple commands at the same time (as long as you highlight all the commands you want to run).
2) To use the latest version of RBioplot we need to first install the package **devtools.** This package is the main interface by which RBioplot can be installed. Both devtools and RBioplot are a one-time installation, unless there are recommended updates that are released. To install 'devtools' type the command below in the R Script editor.

```
install.packages("devtools")
```

Note: **You must be connected to the internet**. You should type all the commands written in this techbull and not just copy them into the text editor. Microsoft word has altered the formatting and syntax of the code written in this techbull. So if you copy/paste this into RStudio, your command will not run, because R Script quotation marks are uniquely annotated. If you copy/paste them then make sure you rewrite your quotation marks.

3) To run your R commands, you have two options:
   a. **R Script editor**, (preferred): On Windows PC, highlight the command you want to run and press Ctrl+Enter. Or place your text cursor on the command you want to run and press Ctrl+Enter. If you separate your command into different lines then make sure all the lines are highlighted before pressing Ctrl+Enter, otherwise your full command will not run.
   Note: on Mac, replace Ctrl key with Command key.

b. **Console**: Copy your command from your text editor to the Console and press Enter.

4) Once your command has been successfully installed you should receive a message that reads something like this:

```
The downloaded binary packages are in
C:\Users\Hanane\AppData\Local\Temp\Rtmp4gUWDW\downloaded_packages
```

5) Now you are ready to install the RBioplot package. Run the following command:

```
devtools::install_github("jzhangc/git_R_STATS_KBS/package/RBioplot")
```

6) (optional) You may run into a few errors as the current version of **devtools** (1.12.0) has a bug that prevents from installing dependencies that RBioplot needs. In such case, you will need to manually install the missing packages.

*Note: the bug will likely be fixed in the next release of devtools. To update devtools, just run the command again: `install.packages("devtools")`

Here are a few examples of errors and solutions, in this first one I was missing the **scales** package:

```
Error in loadNamespace(i, c(lib.loc, .libPaths()), versionCheck = vI[[i]])
: there is no package called 'scales'
```

so I installed it manually:

```
install.packages("scales")
```

After I installed the missing package (**scales)** I re-ran the RBioplot installation command in step 5.

```
Error in loadNamespace(j <- i[[1L]], c(lib.loc, .libPaths()), versionCheck
= vI[[j]]) : there is no package called 'sandwich'
```

Now, I am missing the package (**sandwich)**, so I installed it with the command below and then re-ran the RBioplot installation command from step 5. Repeat this process until you have installed all the missing packages.

```
install.packages("sandwich")
```

Once RBioplot has successfully been installed you will get this message:

```
* DONE (RBioplot)
```

7) The next step is to setup your working directory, this is where you will place all your input data files and where all your plots, stats, and graphs will be exported to. In this example my working directory file is on my desktop and it's called "RBioplot stuffs". To setup your working directory run the following command and replace "C:\Users\Hanane\Desktop" with your folder address:

For Windows: `setwd("C:\\Users\\Hanane\\Desktop\\RBioplot stuffs")`

For Mac and Linux: `setwd("C:/Users/Hanane/Desktop/RBioplot stuffs")`

***Note:** You will be required to reset your working domain once you exit and re-enter RStudio.

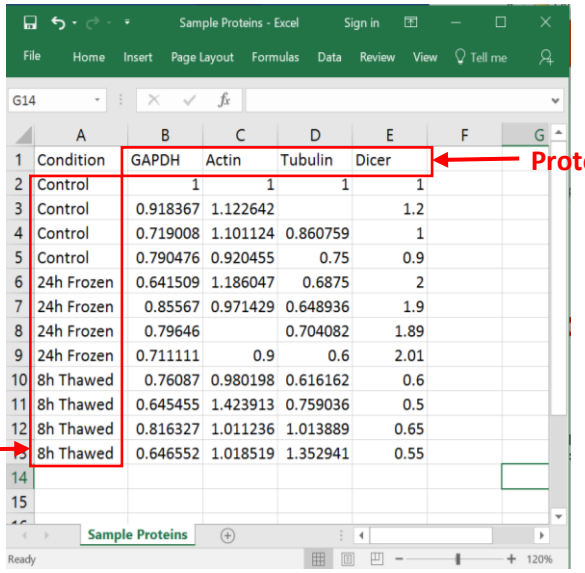
8) In the 'Files, Plots, and Packages' quadrant of the graph you should select 'Packages' and then find RBioplot in the list. Once you click on RBioplot you will be redirected to a page with the RBioplot documentation and all the help files for the different functions. If you get stuck using one of the commands use these help files to trouble shoot.

9) If your RBioplot stops working or there is an error that you have spent the last 3 weeks troubleshooting, then you should email the developer jzhangcad@gmail.com for help.

III. Statistical analysis

The rbiostats function is used to conduct statistical analysis. Before starting you should visit the **rbiostats** help page (or the **rbiostats mini-HELP! box** on page 6), this will outline all the available functions and how they work. All sample files used and generated can be downloaded from the website. **rbiostats** is a simple to use function for comprehensive statistical analyses that generates a statistical report for various statistical test including: t-tests, ANOVA and post-hoc tests (Tukey or Dunnett's). Make sure to read up on the various tests and to ensure you are using the one most appropriate for your data set and experimental design.

1) Your data must be formatted in a very specific manner to ensure the values are analyzed properly by RBioplot. **All excel files MUST be saved as a Microsoft Excel Comma Separated Values File (.csv)**.
2) The table on the right is my **Sample Proteins.csv** file, standardized western blot relative protein abundances. The values are standardized with Coomassie blue stained bands. Experimental conditions are listed in the first column. The A1 cell reads 'Condition' – or something similar. Protein names are listed in the first row. The values I report are an average of the relative protein abundance for all my technical replicates that correspond to each of my biological replicates.



	A	B	C	D	E
1	Condition	GAPDH	Actin	Tubulin	Dicer
2	Control	1	1	1	1
3	Control	0.918367	1.122642		1.2
4	Control	0.719008	1.101124	0.860759	1
5	Control	0.790476	0.920455	0.75	0.9
6	24h Frozen	0.641509	1.186047	0.6875	2
7	24h Frozen	0.85567	0.971429	0.648936	1.9
8	24h Frozen	0.79646		0.704082	1.89
9	24h Frozen	0.711111	0.9	0.6	2.01
10	8h Thawed	0.76087	0.980198	0.616162	0.6
11	8h Thawed	0.645455	1.423913	0.759036	0.5
12	8h Thawed	0.816327	1.011236	1.013889	0.65
13	8h Thawed	0.646552	1.018519	1.352941	0.55
14					
15					

***Note:** You can analyze and graph your data even with outliers taken out in your table – there is no need to rearrange the table.

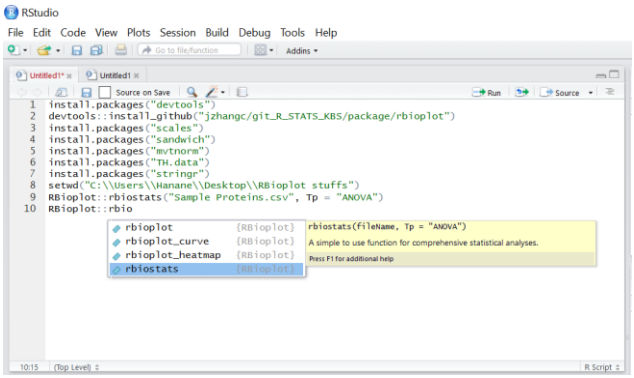
***Note:** Even though there are 4 different biological replicates for each experimental condition (ie. Control 1, 2, 3, and 4) in the example, make sure to keep the exact same identifier for each condition or the program will separate them into different conditions.

3) To run stats on your data using **rbiostats**, run the following command and replace "**Sample Proteins.csv**" with the file name for your .csv data file. Make sure to include the full file name (with .csv) in the quotations, and select the stats test you need. (ANOVA is the default, meaning that if you don't type the "Tp = " part, i.e. `RBioplot::rbiostats("Sample Proteins.csv")`, the function will run ANOVA)

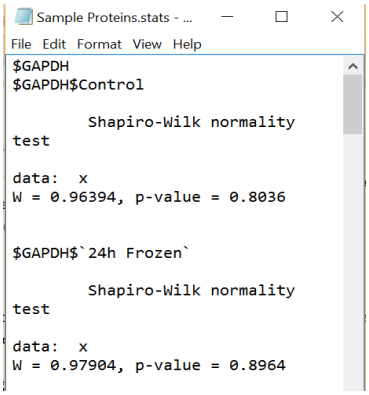
```
RBioplot::rbiostats("Sample Proteins.csv", Tp = "ANOVA")
```

***Note:** everything is case sensitive and has to be exact, meaning only "ANOVA" will work for ANOVA test. When you are not sure what to type, run command `?rbiostats` to view the help page for this function, or see the section below.

4) As you are typing your command you will notice that RStudio has an auto-fill setting that you will find quite useful. If you hover your mouse over one of the options that appear, you will get a description of the function and the various arguments and customizations available.



7) Outputs a **stats.txt** file with Shapiro-Wilk normality test results, Bartlett test of homogeneity (equal variance) results, and the results of the statistical analysis of interest, this is automatically exported to your working directory. In this case, my file (on the left) was called **Sample Proteins.stats.txt**, the complete report can be downloaded from the website. Briefly, you need ****p values greater than 0.05**** for both Shapiro-Wilk normality test and Bartlett test to proceed to the follow up statistical tests (e.g. t-test, ANOVA). For a detailed break-down of the statistical report refer to Zhang and Storey (2016).



rbiostats mini-HELP!

Usage

```
rbiostats(fileName, Tp = "ANOVA")
```

Arguments

`fileName`	Input file name. Case sensitive and be sure to type with quotation marks. Currently only takes `.csv` files.
`Tp`	Type of the intended statistical test. Case sensitive and type with quotation marks. Options are: "t-test", "ANOVA", "Tukey" and "Dunnett". Default is "ANOVA".

IV. Plotting

You can plot various types of graphs with your data using the corresponding functions in each of the RBioplot packages. They are excellent for applications such as western blotting, protein abundance levels, transcription factor-DNA binding activity, miRNA expression analyses, mRNA abundance, enzyme studies and many others. All sample files used and generated can be downloaded from the website.

***Note:** The following case studies don't explore all of the customizable arguments and settings available for each function. There are many additional functions that you can use to tailor your dataset graph requirements such as; plot width, plot height, using standard deviation for your error bars, etc.

A. Histogram

rbioplot is a simple to use function for plotting basing on the statistical analysis of choice. Before starting visit the **rbioplot** help page, this will outline the different functions and arguments and how to properly use them. You can also use the **rbioplot mini-HELP! box** on page 10.

1) For this example I used the western blotting data set from **Sample Proteins.csv** (check out rbiostats, step1). The protein abundance, PCR quantification, activity levels, etc. should all be in this format for **rbioplot.**

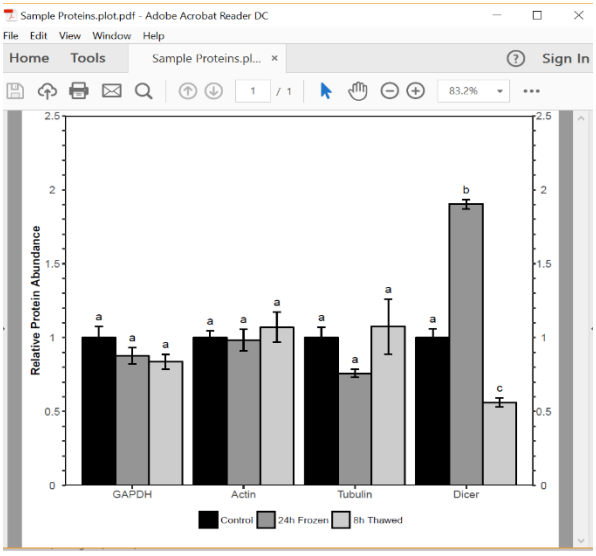
2) To generate a histogram run the following **rbioplot** command:

```
RBioplot::rbioplot("Sample Proteins.csv", Tp = "Tukey", Title = NULL, errorbar = "SEM", errorbarWidth = 0.2, fontType = "sans", xLabel = NULL, xTickLblSize = 10, xTickItalic = FALSE, xAngle = 0, xAlign = 0.5, yLabel = NULL, yTickLblSize = 10, yTickItalic = FALSE, legendTtl = FALSE, plotWidth = 170, plotHeight = 150, y_custom_tick_range = FALSE, y_lower_limit = 0, y_upper_limit, y_major_tick_range, y_n_minor_ticks = 4)
```

***Note:** Since most of the arguments have a default value, you can run something like rbioplot("Sample Protein") with all the default values. However, it is recommended that you play with the settings to get the graph that you are want. This is true of all the functions in the package.

3) The above command is very versatile and should be tailored to your specific dataset. You can use the customizable arguments to design what your graph looks like. Some of the customizable options include: Type of statistical test, Title, error bars, font, axes labels, Y-axis ticks, and plot height and width. For more information about the individual arguments go to the help pages described above.

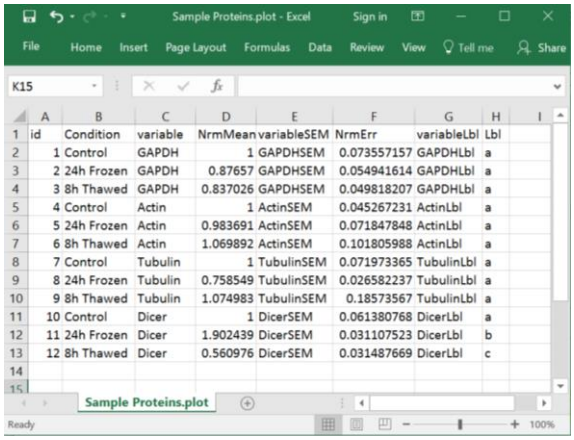
4) Once you run the command, a preview of your graph will appear in the lower right quadrant of RStudio. A pdf version of your graph (600 dpi resolution.) will also be automatically exported into your working directory. In this example the pdf generated was titled **Sample Proteins.plot.pdf**.



***Note:** If you re-run the above command using the same input file name then the new generated graph will overwrite your old graph.

***Note:** Make sure the **.pdf** and **.plot.pdf** file that were generated are closed before re-running. You cannot generate a new file with the exact same name if the old version is active.

5) The above command also outputs a .csv file with detailed metrics for the plot, including Mean, SEM and significance labels. In this sample the exported file was named **Sample Proteins.plot.csv**:



	A	B	C	D	E	F	G	H
1	id	Condition	variable	NrmMean	variableSEM	NrmErr	variableLbl	Lbl
2	1	Control	GAPDH	1	GAPDHSEM	0.073557157	GAPDHLbl	a
3	2	24h Frozen	GAPDH	0.87657	GAPDHSEM	0.054941614	GAPDHLbl	a
4	3	8h Thawed	GAPDH	0.837026	GAPDHSEM	0.049818207	GAPDHLbl	a
5	4	Control	Actin	1	ActinSEM	0.045267231	ActinLbl	a
6	5	24h Frozen	Actin	0.983691	ActinSEM	0.071847848	ActinLbl	a
7	6	8h Thawed	Actin	1.069892	ActinSEM	0.101805988	ActinLbl	a
8	7	Control	Tubulin	1	TubulinSEM	0.071973365	TubulinLbl	a
9	8	24h Frozen	Tubulin	0.758549	TubulinSEM	0.026582237	TubulinLbl	a
10	9	8h Thawed	Tubulin	1.074983	TubulinSEM	0.18573567	TubulinLbl	a
11	10	Control	Dicer	1	DicerSEM	0.061380768	DicerLbl	a
12	11	24h Frozen	Dicer	1.902439	DicerSEM	0.031107523	DicerLbl	b
13	12	8h Thawed	Dicer	0.560976	DicerSEM	0.031487669	DicerLbl	c

6) If you notice that the default axes on your newly generated graph are not optimal or do not match the criteria required for publication in your journal of interest, then you can use the **autorange_bar_y** function. It is a function that allows you to get custom lower/upper limit, major tick range, as well as minor tick options for y axis, based on a user-defined major tick number. **The purpose of this function is to allow the user to determine the optimal number of minor ticks based on a user-defined major tick number.**

7) For the **autorange_bar_y** function, data should be arranged as the same input file for **rbioplots**, in this example the file "Sample Proteins.csv" was used, remember to replace this with your "file name.csv". To find the perfect tick numbers for your y-axis run the following **autorange_bar_y** command:

```
RBioplot::autorange_bar_y("Sample Proteins.csv", Nrm = TRUE, errorbar = "SEM", nMajorTicks = 5, DfltZero = TRUE)
```

***Note:** If you have negative values then you must select 'DfltZero = FALSE'.

***Note:** If you do not require a specific number of major ticks then the default setting of 5 will be selected.

8) The output is a list containing lower_limit, upper_limit, major_tick_range, and minor_tick_options. This list will appear directly in your R Console:

```
$y_axis_range
  lower_limit   upper_limit major_tick_range     $minor_tick_options
          0.0           2.5             0.5                 [1] 0 4
```

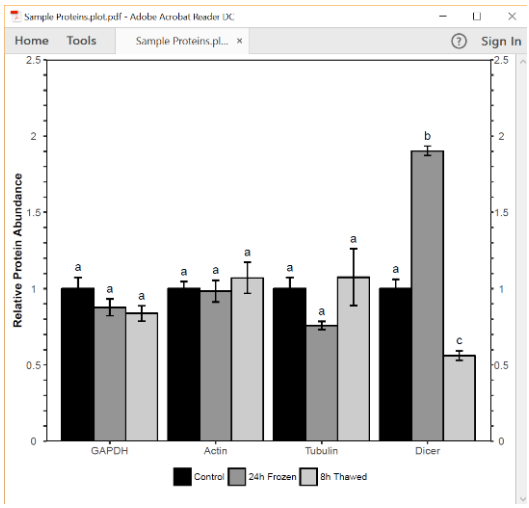
***Note:** The above values are just recommendations. The minor_tick_options sections shows all the available numbers of minor ticks (0 or 4) for this particular data set with the user-defined number of major ticks set to 5.

9) Once **autorange_bar_y** has detected the y-axis range that best represents your data you can input the above values and arguments into the main **rbioplot** command, see following (highlighted):

```
RBioplot::rbioplot("Sample Proteins.csv", Tp = "Tukey", Title = NULL, errorbar = "SEM", errorbarWidth = 0.2, fontType = "sans", xLabel = NULL, xTickLblSize = 10, xTickItalic = FALSE, xAngle = 0, xAlign = 0.5, yLabel = "Relative Protein Abundance", yTickLblSize = 10, yTickItalic = FALSE, legendTtl = FALSE, plotWidth = 170, plotHeight = 150, y_custom_tick_range = TRUE, y_lower_limit = 0, y_upper_limit = 2.5, y_major_tick_range = 0.5, y_n_minor_ticks = 4)
```

***Note:** Generally, the default graphing settings will try to optimize the plot area to our dataset, which means you only need to use the **autorange_bar_y** function if you have specific major and minor tick values you would like to use.

10) The following **Sample Proteins.pdf** was generated. In this case, we can see that the new graph is identical to the old graph, that's because the predicted y_axis_range was the same as the default settings. This is not always the case and I recommend that you run the **autorange_bar_y** function for all your histograms to ensure that your graphs are visually appealing and easy to read.



11) When customizing the command arguments, you could be forcing the graphing functions to generate a non-optimal graph. Be sure to visually inspect your plots and verify that they make sense. You may need to manually change the settings accordingly.

12) For more information on what the individual arguments mean and how to properly use them read the **autorange_bar_y** help page on RStudio.

rbioplot mini-HELP!

Usage

```
rbioplot(fileName, Tp = "Tukey", Title = NULL, errorbar = "SEM",
  errorbarWidth = 0.2, fontType = "sans", xLabel = NULL,
  xTickLblSize = 10, xTickItalic = FALSE, xAngle = 0, xAlign = 0.5,
  yLabel = NULL, yTickLblSize = 10, yTickItalic = FALSE,
  legendTtl = FALSE, plotWidth = 170, plotHeight = 150,
  y_custom_tick_range = FALSE, y_lower_limit = 0, y_upper_limit,
  y_major_tick_range, y_n_minor_ticks = 4)
```

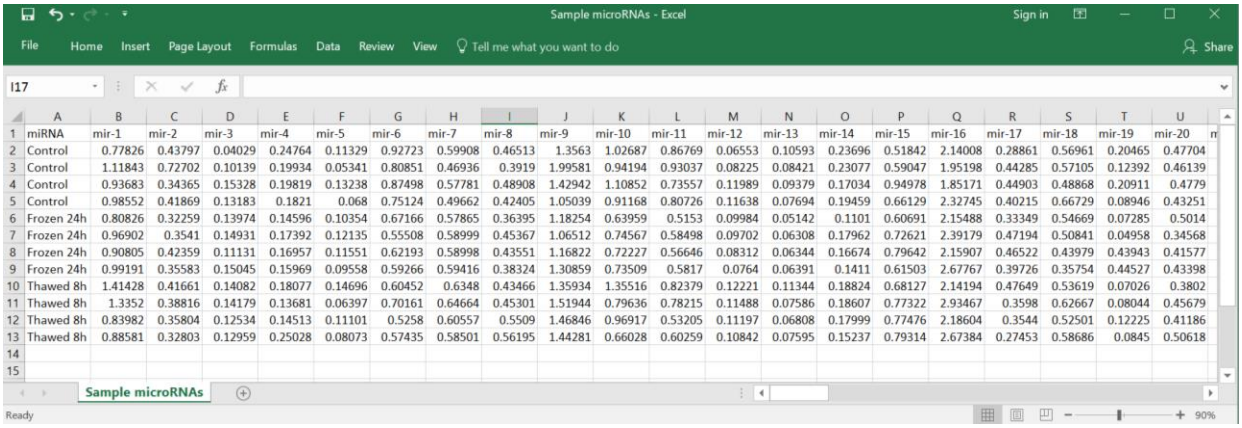
Arguments

`fileName`	Input file name. Case sensitive and be sure to type with quotation marks. Currently only takes `.csv` files.
`Tp`	Type of the intended statistical test. Case sensitive and be sure to type with quotation marks. Options are: "t-test", "Tukey" and "Dunnett". Default is "Tukey".
`Title`	The displayed title on top of the plot. Be sure to type with quotation marks. Default is `NULL`.
`errorbar`	Set the type of errorbar. Options are standard error of mean (`"SEM"`), or standard deviation (`"SD"`). Default is `"SEM"`.
`errorbarWidth`	Set the width for errorbar. Default is `0.2`.
`fontType`	The type of font in the figure. Default is "sans". For all options please refer to R font table, which is avaiable on the website: http://kenstoreylab.com/?page_id=2448.
`xLabel`	x axis label. Type with quotation marks. Default is `NULL`.
`xTickLblSize`	Font size of x axis ticks. Default is 10.
`xTickItalic`	Set x axis tick font to italic. Default is `FALSE`.
`xAngle`	The rotation angle (degrees) of the x axis marks. Default is `0` - horizontal.
`xAlign`	The alignment type of the x axis marks. Options are `0`, `0.5` and `1`. The default value at `0`is especially useful when `xAngle = 90`.
`yLabel`	y axis label. Type with quotation marks. Default is `NULL`.
`yTickLblSize`	Font size of y axis ticks. Default is 10.
`yTickItalic`	Set y axis tick font to italic. Default is `FALSE`.
`legendTtl`	Hide/Display legend title. If `TRUE` or `T`, the name of the first column of the raw data file will display as the legend title. Default is `FALSE`.
`plotWidth`	The width of the plot (unit: mm). Default is 170. Default will fit most of the cases.
`plotHeight`	The height of the plot (unit: mm). Default is 150. Default will fit most of the cases.
`y_custom_tick_range`	To initiate setting the custom `y_upper_limit`, `y_lower_limit`,`y_major_tick_range`, `y_n_minor_ticks`. Default is `FALSE`.
`y_lower_limit`	Can only be set when `y_custom_tick_range = TRUE`. Set custom lower limt for y axis. Default is `0`. Value can be obtained from `autorange_bar_y`.
`y_upper_limit`	Can only be set when `y_custom_tick_range = TRUE`. Set custom upper limt for y axis. Value can be obtained from `autorange_bar_y`.
`y_major_tick_range`	Can only be set when `y_custom_tick_range = TRUE`. Set custom major tick range for y axis. Value can be obtained from `autorange_bar_y`.
`y_n_minor_ticks`	Can only be set when `y_custom_tick_range = TRUE`. Set custom numbers of minor ticks. Default is `4`. Value can be obtained from `autorange_bar_y`.

B. Heatmap

The **rbioplot_heatmap** function can be used to analyze and graph largescale high-throughput studies such as large explorations of microRNA differential expression. **rbioplot_heatmap** is a function for plotting simple heatmap based on the statistical analysis of your choice. Before starting visit the **rbioplot_heatmap** help page, this will outline the different functions and arguments and how to properly use them. You can also use the **rbioplot_heatmap mini-HELP! box** on page 13 for a quick reference.

1) The table below is my **Sample miRNAs.csv** file, the values are standardized miRNA relative abundance levels (obtained from qRT-PCR and normalized to the reference gene). The A1 cell should contain the word **microRNA** (or whatever you are measuring). The values I report are an average of the relative miRNA abundance for all my technical replicates that correspond to each of my biological replicates. This is just a small portion of the file- the full file contains 152 microRNAs.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
1	miRNA	mir-1	mir-2	mir-3	mir-4	mir-5	mir-6	mir-7	mir-8	mir-9	mir-10	mir-11	mir-12	mir-13	mir-14	mir-15	mir-16	mir-17	mir-18	mir-19	mir-20
2	Control	0.77826	0.43797	0.04029	0.24764	0.11329	0.92723	0.59908	0.46513	1.3563	1.02687	0.86769	0.06553	0.10593	0.23696	0.51842	2.14008	0.28861	0.56961	0.20465	0.47704
3	Control	1.11843	0.72702	0.10139	0.19934	0.05341	0.80851	0.46936	0.3919	1.99581	0.94194	0.93037	0.08225	0.08421	0.23077	0.59047	1.95198	0.44285	0.57105	0.12392	0.46139
4	Control	0.93683	0.34365	0.15328	0.19819	0.13238	0.87498	0.57781	0.48908	1.42942	1.10852	0.73557	0.11989	0.09379	0.17034	0.94978	1.85171	0.44903	0.48868	0.20911	0.4779
5	Control	0.98552	0.41869	0.13183	0.1821	0.068	0.75124	0.49662	0.42405	1.05039	0.91168	0.80726	0.11638	0.07694	0.19459	0.66129	2.32745	0.40215	0.66729	0.08946	0.43251
6	Frozen 24h	0.80826	0.32259	0.13974	0.14596	0.10354	0.67166	0.57865	0.36395	1.18254	0.63959	0.5153	0.09984	0.05142	0.1101	0.60691	2.15488	0.33349	0.54669	0.07285	0.5014
7	Frozen 24h	0.96902	0.3541	0.14931	0.17392	0.12135	0.55508	0.58999	0.45367	1.06512	0.74567	0.58498	0.09702	0.06308	0.17962	0.72621	2.39179	0.47194	0.50841	0.04958	0.34568
8	Frozen 24h	0.90805	0.42359	0.11131	0.16957	0.11551	0.62193	0.58998	0.43551	1.16822	0.72227	0.56646	0.08312	0.06344	0.16674	0.79642	2.15907	0.46522	0.43979	0.43943	0.41577
9	Frozen 24h	0.99191	0.35583	0.15045	0.15969	0.09558	0.59266	0.59416	0.38324	1.30859	0.73509	0.5817	0.0764	0.06391	0.1411	0.61503	2.67767	0.39726	0.35754	0.44527	0.43398
10	Thawed 8h	1.41428	0.41661	0.14082	0.18077	0.14696	0.60452	0.6348	0.43466	1.35934	1.35516	0.82379	0.12221	0.11344	0.18824	0.68127	2.14194	0.47649	0.53619	0.07026	0.3802
11	Thawed 8h	1.3352	0.38816	0.14179	0.13681	0.06397	0.70161	0.64664	0.45301	1.51944	0.79636	0.78215	0.11488	0.07586	0.18607	0.77322	2.93467	0.3598	0.62667	0.08044	0.45679
12	Thawed 8h	0.83982	0.35804	0.12534	0.14513	0.11101	0.5258	0.60557	0.5509	1.46846	0.96917	0.53205	0.11197	0.06808	0.17999	0.77476	2.18604	0.3544	0.52501	0.12225	0.41186
13	Thawed 8h	0.88581	0.32803	0.12959	0.25028	0.08073	0.57435	0.58501	0.56195	1.44281	0.66028	0.60259	0.10842	0.07595	0.15237	0.79314	2.67384	0.27453	0.58686	0.0845	0.50618
14																					
15																					

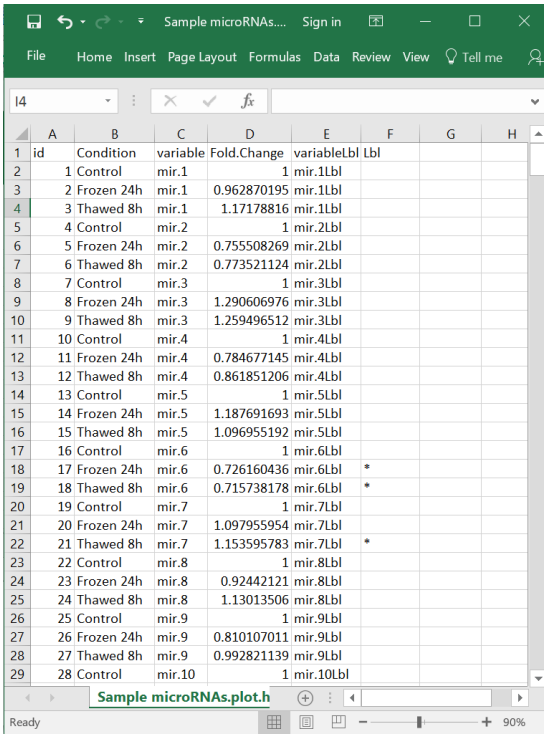
2) To generate a heatmap run the following **rbioplot** command:

```
RBioplot::rbioplot_heatmap("Sample microRNAs.csv", Tp = "Dunnett", rmCntl = FALSE, Title = NULL, fontType = "sans", tileLow = "firebrick1", tileHigh = "green2", tileLbl = TRUE, tileLblSize = 5, tileTxtColour = "white", tileLblPos = 0.5, xLabel = NULL, xTickLblSize = 10, xTickItalic = FALSE, xAngle = 0, xAlign = 0.5, yLabel = NULL, yTickLblSize = 10, yTickItalic = FALSE, legendTtl = FALSE, legendPos = "right", plotWidth = 170, plotHeight = 600)
```

***Note:** In this case, the plotHeight argument was adjusted to 600 fit all make the individual heatmap cells bigger.

3) The above command is very versatile and should be tailored to your specific dataset. You can also easily change the colour scheme of your heatmap, refer to **V. RBioplot Resources** for the full colour palette available on with R. For more information on how to customize the arguments go to the help page or the mini-help box.

4) The above **rbioplot_heatmap** command outputs a **.plot.csv** file with detailed metrics for the plot, including normalized mean and significance labels, as well as a plot image file (**.pdf**), with 600 dpi resolution.



	A	B	C	D	E	F	G	H
1	id	Condition	variable	Fold.Change	variableLbl	Lbl		
2	1	Control	mir.1	1	mir.1Lbl			
3	2	Frozen 24h	mir.1	0.962870195	mir.1Lbl			
4	3	Thawed 8h	mir.1	1.17178816	mir.1Lbl			
5	4	Control	mir.2	1	mir.2Lbl			
6	5	Frozen 24h	mir.2	0.755508269	mir.2Lbl			
7	6	Thawed 8h	mir.2	0.773521124	mir.2Lbl			
8	7	Control	mir.3	1	mir.3Lbl			
9	8	Frozen 24h	mir.3	1.290606976	mir.3Lbl			
10	9	Thawed 8h	mir.3	1.259496512	mir.3Lbl			
11	10	Control	mir.4	1	mir.4Lbl			
12	11	Frozen 24h	mir.4	0.784677145	mir.4Lbl			
13	12	Thawed 8h	mir.4	0.861851206	mir.4Lbl			
14	13	Control	mir.5	1	mir.5Lbl			
15	14	Frozen 24h	mir.5	1.187691693	mir.5Lbl			
16	15	Thawed 8h	mir.5	1.096955192	mir.5Lbl			
17	16	Control	mir.6	1	mir.6Lbl			
18	17	Frozen 24h	mir.6	0.726160436	mir.6Lbl	*		
19	18	Thawed 8h	mir.6	0.715738178	mir.6Lbl	*		
20	19	Control	mir.7	1	mir.7Lbl			
21	20	Frozen 24h	mir.7	1.097955954	mir.7Lbl			
22	21	Thawed 8h	mir.7	1.153595783	mir.7Lbl	*		
23	22	Control	mir.8	1	mir.8Lbl			
24	23	Frozen 24h	mir.8	0.92442121	mir.8Lbl			
25	24	Thawed 8h	mir.8	1.13013506	mir.8Lbl			
26	25	Control	mir.9	1	mir.9Lbl			
27	26	Frozen 24h	mir.9	0.810107011	mir.9Lbl			
28	27	Thawed 8h	mir.9	0.992821139	mir.9Lbl			
29	28	Control	mir.10	1	mir.10Lbl			

rbioplot_heatmap mini-HELP!

Usage

```
rbioplot_heatmap(fileName, Tp = "Dunnett", rmCntl = FALSE, Title = NULL,
  fontType = "sans", tileLow = "skyblue", tileHigh = "midnightblue",
  tileLbl = TRUE, tileLblSize = 10, tileTxtColour = "white",
  tileLblPos = 0.5, xLabel = NULL, xTickLblSize = 10,
  xTickItalic = FALSE, xAngle = 0, xAlign = 0.5, yLabel = NULL,
  yTickLblSize = 10, yTickItalic = FALSE, legendTtl = FALSE,
  legendPos = "bottom", plotWidth = 170, plotHeight = 150)
```

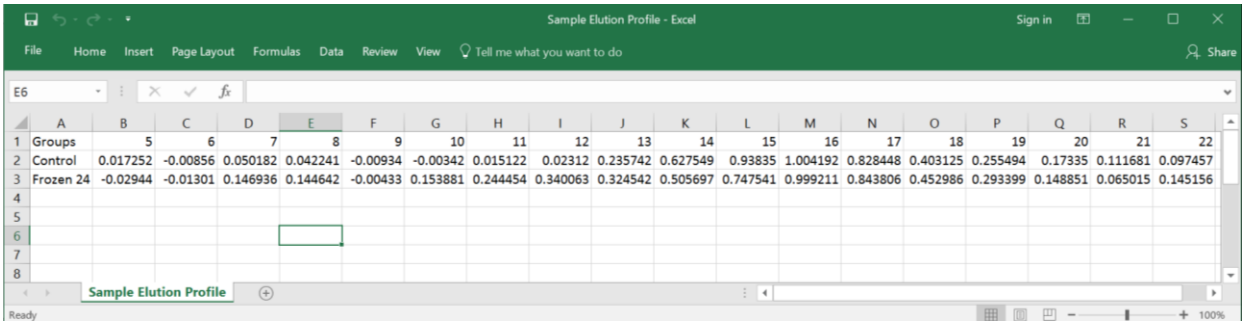
Arguments

`fileName`	Input file name. Case sensitive and be sure to type with quotation marks. Currently only takes `.csv` files.
`Tp`	Type of the intended statistical test. Case sensitive and be sure to type with quotation marks. Options are: "t-test", "Tukey" and "Dunnett". Default is "Dunnett".
`rmCntl`	Remove the first column (i.e., control). Default is `FALSE`.
`Title`	The displayed title on top of the plot. Be sure to type with quotation marks. Default is `NULL`.
`fontType`	The type of font in the figure. Default is "sans". For all options please refer to R font table, which is avaiable on the website:http://kenstoreylab.com/?page_id=2448.
`tileLow`	Set the colour for the lower limit of the heatmap. Default is `skyblue`. For full colour options and names, refer to the websitehttp://kenstoreylab.com/?page_id=2448.
`tileHigh`	Set the colour for the upper limit of the heatmap. Default is`midnightblue`. For full colour options and names, refer to the website http://kenstoreylab.com/?page_id=2448.
`tileLbl`	Enable or disable significant notation on the tiles. Default is `TRUE`.
`tileLblSize`	Set the font size of the tile label. Default is `10`.
`tileTxtColour`	Set the colour of the on tile label. Default is `"white"`. For full colour options and names, refer to the website http://kenstoreylab.com/?page_id=2448.
`tileLblPos`	Set the position of the tile lables. Options are `0`, `0.5` and `1`. Default is `0.5`.
`xLabel`	x axis label. Type with quotation marks. Default is `NULL`.
`xTickLblSize`	Font size of x axis ticks. Default is 10.
`xTickItalic`	Set x axis tick font to italic. Default is `FALSE`.
`xAngle`	The rotation angle (degrees) of the x axis marks. Default is `0` - horizontal.
`xAlign`	The alignment type of the x axis marks. Options are `0`, `0.5` and `1`. The default value at `0` is especially useful when `xAngle = 90`.
`yLabel`	y axis label. Type with quotation marks. Default is `NULL`.
`yTickLblSize`	Font size of y axis ticks. Default is 10.
`yTickItalic`	Set y axis tick font to italic. Default is `FALSE`.
`legendTtl`	Hide/Display legend title. Default is `FALSE`.
`legendPos`	Set the legend position. Options are `"top"`, `"bottom"`, `"left"`and `"right"`. Default is `"bottom"`.
`plotWidth`	The width of the plot (unit: mm). Default is 170. Default will fit most of the cases.
`plotHeight`	The height of the plot (unit: mm). Default is 150. Default will fit most of the cases.
`y_custom_tick_range`	To initiate setting the custom `y_upper_limit`, `y_lower_limit`,`y_major_tick_range`, `y_n_minor_ticks`. Default is `FALSE`.
`y_upper_limit`	Can only be set when `y_custom_tick_range = TRUE`. Set custom upper limt for y axis. Value can be obtained from`autorange_bar_y`.
`y_lower_limit`	Can only be set when `y_custom_tick_range = TRUE`. Set custom lower limt for y axis. Default is `0`. Value can be obtained from`autorange_bar_y`.
`y_major_tick_range`	Can only be set when `y_custom_tick_range = TRUE`. Set custom major tick range for y axis. Value can be obtained from`autorange_bar_y`.
`y_n_minor_ticks`	Can only be set when `y_custom_tick_range = TRUE`. Set custom numbers of minor ticks. Default is `4`. Value can be obtained from `autorange_bar_y`.

C. Joint-point curve

rbioplot_curve is a simple to use function for plotting joining-point curve figures with continuous X-axis and Y-axis values. Before starting visit the **rbioplot_curve** help page, this will outline the different functions and arguments and how to properly use them. You can also use the **rbioplot_curve mini-HELP! box** on page 17.

1) In this example below I plotted my imaginary enzyme's activity for 18 eluted fractions from both control and 24 h frozen. I used the **Sample Elution Profile.csv** (below). The independent variables are listed in column A.



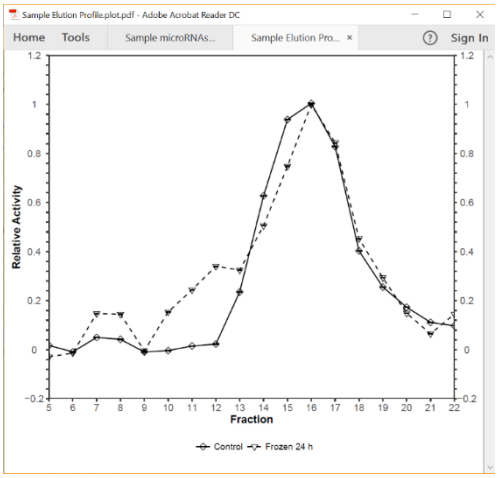
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
1	Groups	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
2	Control	0.017252	-0.00856	0.050182	0.042241	-0.00934	-0.00342	0.015122	0.02312	0.235742	0.627549	0.93835	1.004192	0.828448	0.403125	0.255494	0.17335	0.111681	0.097457
3	Frozen 24	-0.02944	-0.01301	0.146936	0.144642	-0.00433	0.153881	0.244454	0.340063	0.324542	0.505697	0.747541	0.999211	0.843806	0.452986	0.293399	0.148851	0.065015	0.145156

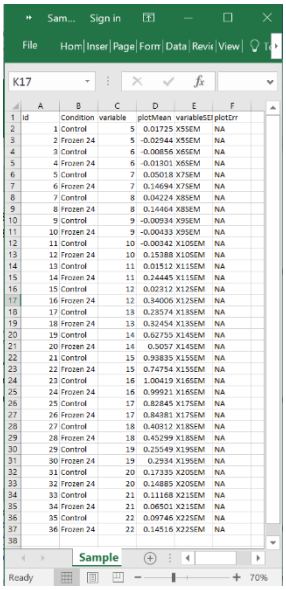
***Note:** biological replicates can also be accepted by this function. The format is the same as functions rbioplot and rbioplot_heatmap. The rbioplot_curve function will automatically detect the replicates and add appropriate error bars to the graph.

2) To generate a histogram run the following **rbioplot** command:

```
RBioplot::rbioplot_curve("Sample Elution Profile.csv", Title = NULL,
errorbar = "SEM", errorbarWidth = 0.2, fontType = "sans", symbolSize = 2,
xLabel = "Fraction", xTickLblSize = 10, xTickItalic = FALSE, xAngle = 0,
xAlign = 0.5, yLabel = "Relative Activity", yTickLblSize = 10, yTickItalic
= FALSE, legendTtl = FALSE, plotWidth = 170, plotHeight = 150,
x_custom_tick_range = TRUE, x_lower_limit = 5, x_upper_limit = 23,
x_major_tick_range = 1, x_n_minor_ticks = 0, y_custom_tick_range = TRUE,
y_lower_limit = -0.2, y_upper_limit = 1.2, y_major_tick_range = 0.2,
y_n_minor_ticks = 4)
```

3) The above **rbioplot_curve** command outputs a **.plot.csv** file with detailed metrics for the plot, including Mean and SEM, as well as a plot image file (**.pdf**), with 600 dpi resolution.





4) With a joint-point curve both the y- and x- axes are continuous which means that both of their ranges can be tailored to your dataset of interest. To do this automatically you can use the **autorange_curve** function. This function allows you to get custom lower/upper limit, major tick range, as well as minor tick options for both axes of a joint-point curve with continuous x AND y values, based on a user-defined major tick number. **The purpose of this function is to allow the user to determine the optimal number of x and y minor ticks based on a user-defined major tick number.**

*Note: Generally, the default graphing settings will try to optimize the plot area to the dataset, which means you only need to use the **autorange_curve** function if you have specific major and minor tick values you would like to use.

5) For the **autorange_curve** function, data should be arranged as the same input file for **rbiocurve**, in this example the file "Sample Elution Profile.csv" was used, remember to replace this with your "file name.csv". To find the perfect axes for your dataset run the following **autorange_curve** command:

```
RBioplot::autorange_curve("Sample Elution Profile.csv", errorbar = "SEM", x_nMajorTicks = 5, x_DfltZero = FALSE, y_nMajorTicks = 10, y_DfltZero = FALSE)
```

*Note: The DfltZero argument is set to "TRUE" by default which forces the axis to start from 0. You might need to set the x_DfltZero and y_DltZero arguments to 'FALSE' if the dataset contains negative values and/or large initial points into account.

*Note: If you do not require a specific number of major ticks then the default setting of 5 and 10 will be selected for x – and y- axis, respectively.

6) The output is a list containing lower_limit, upper_limit, major_tick_range, and minor_tick_options. This list will appear directly in your R Console:

```
$x_axis_range
     x_lower_limit  x_upper_limit  x_major_tick_range   $x_minor_tick_options
               3.5           24.5                 3.5         [1]  0  4  6  34

$y_axis_range
     y_lower_limit   y_upper_limit  y_major_tick_range  $y_minor_tick_options
              -0.6             1.6                 0.2       [1]  0 1 3 4 9 19
```

*Note: The above values are just recommendations.

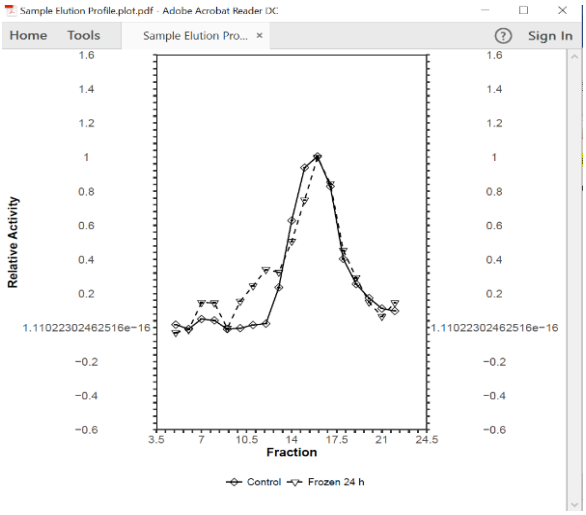
7) Once **autorange_curve** has detected the x- and y-axes range that best represents your data you can input the above values and arguments into the main **rbioplot** command, see following (axis settings are highlighted):

```
RBioplot::rbioplot_curve("Sample Elution Profile.csv", Title = NULL, errorbar = "SEM", errorbarWidth = 0.2, fontType = "sans", symbolSize = 2, xLabel = "Fraction", xTickLblSize = 10, xTickItalic = FALSE, xAngle = 0, xAlign = 0.5, yLabel = "Relative Activity", yTickLblSize = 10, yTickItalic = FALSE, legendTtl = FALSE, plotWidth = 170, plotHeight = 150, x_custom_tick_range = TRUE, x_lower_limit = 3.5, x_upper_limit = 24.5, x_major_tick_range = 3.5, x_n_minor_ticks = 4, y_custom_tick_range = TRUE, y_lower_limit = -0.6, y_upper_limit = 1.6, y_major_tick_range = 0.2, y_n_minor_ticks = 4)
```

*Note: When manually changing your x and y axis values you may get this error message, **Error: `breaks` and `labels` must have the same length**. This means that the upper and lower limit values you set do not match the major and minor tick values. For example, if your major tick value is 5 then you cannot set your lower limit to 3 as this is not a multiple of 5.

8) The following **Sample Elution Profile.pdf** was generated with the recommended x and y axes. As you can see, the graphing conditions were not optimal: the x axis is a fraction number and setting the lower limit to 3.5 does not make much sense since my first fraction was 5; whereas the default 10 major ticks setting produced an irregular y-axis tick setup. To change this, you may need to manually type in values for the x-axis, and set another major tick number other than 10 using the **autorange_curve** function.

9) For more information on what the individual arguments mean and how to properly use them read the **autorange_curve** help page on RStudio.



rbioplot_curve mini-HELP!

Usage

```
rbioplot_curve(fileName, Title = NULL, errorbar = "SEM",
  errorbarWidth = 0.2, fontType = "sans", symbolSize = 2, xLabel = NULL,
  xTickLblSize = 10, xTickItalic = FALSE, xAngle = 0, xAlign = 0.5,
  yLabel = NULL, yTickLblSize = 10, yTickItalic = FALSE,
  legendTtl = FALSE, plotWidth = 170, plotHeight = 150,
  x_custom_tick_range = FALSE, x_lower_limit = 0, x_upper_limit,
  x_major_tick_range, x_n_minor_ticks = 0, y_custom_tick_range = FALSE,
  y_lower_limit = 0, y_upper_limit, y_major_tick_range, y_n_minor_ticks = 4)
```

Arguments

`fileName`	Input file name. Case sensitive and be sure to type with quotation marks. Currently only takes `.csv` files. Note that the column names (excluding the first column) need to be numeric.
`Title`	The displayed title on top of the plot. Be sure to type with quotation marks. Default is `NULL`.
`errorbar`	Set the type of errorbar. Options are standard error of mean (`"SEM"`), or standard deviation (`"SD"`). Default is `"SEM"`.
`errorbarWidth`	Set the width for errorbar. Default is `0.2`.
`fontType`	The type of font in the figure. Default is "sans". For all options please refer to R font table, which is avaiable on the website:http://kenstoreylab.com/?page_id=2448.
`symbolSize`	Set the size of symbols. Default is `2`.
`xLabel`	x axis label. Type with quotation marks. Default is `NULL`.
`xTickLblSize`	Font size of x axis ticks. Default is `10`.
`xTickItalic`	Set x axis tick font to italic. Default is `FALSE`.
`xAngle`	The rotation angle (degrees) of the x axis marks. Default is `0` - horizontal.
`xAlign`	The alignment type of the x axis marks. Options are `0`, `0.5` and `1`. The default value at `0` is especially useful when `xAngle = 90`.
`yLabel`	y axis label. Type with quotation marks. Default is `NULL`.
`yTickLblSize`	Font size of y axis ticks. Default is 10.
`yTickItalic`	Set y axis tick font to italic. Default is `FALSE`.
`legendTtl`	Hide/Display legend title. If `TRUE` or `T`, the name of the first column of the raw data file will display as the legend title. Default is `FALSE`.
`plotWidth`	The width of the plot (unit: mm). Default is 170. Default will fit most of the cases.
`plotHeight`	The height of the plot (unit: mm). Default is 150. Default will fit most of the cases.
`x_custom_tick_range`	To initiate setting the custom `x_upper_limit`, `x_lower_limit`,`x_major_tick_range`, `x_n_minor_ticks`. Default is `FALSE`.
`x_lower_limit`	Can only be set when `x_custom_tick_range = TRUE`. Set custom lower limt for x axis. Default is `0`. Value can be obtained from`autorange_curve`.
`x_upper_limit`	Can only be set when `x_custom_tick_range = TRUE`. Set custom upper limt for x axis. Value can be obtained from`autorange_curve`.
`x_major_tick_range`	Can only be set when `x_custom_tick_range = TRUE`. Set custom major tick range for x axis. Value can be obtained from`autorange_curve`.
`x_n_minor_ticks`	Can only be set when `x_custom_tick_range = TRUE`. Set custom numbers of minor ticks. Default is `4`. Value can be obtained from `autorange_curve`.
`y_custom_tick_range`	To initiate setting the custom `y_upper_limit`, `y_lower_limit`,`y_major_tick_range`, `y_n_minor_ticks`. Default is `FALSE`.
`y_lower_limit`	Can only be set when `y_custom_tick_range = TRUE`. Set custom lower limt for y axis. Default is `0`. Value can be obtained from`autorange_curve`.
`y_upper_limit`	Can only be set when `y_custom_tick_range = TRUE`. Set custom upper limt for y axis. Value can be obtained from `autorange_curve`.
`y_major_tick_range`	Can only be set when `y_custom_tick_range = TRUE`. Set custom major tick range for y axis. Value can be obtained from`autorange_curve`.
`y_n_minor_ticks`	Can only be set when `y_custom_tick_range = TRUE`. Set custom numbers of minor ticks. Default is `4`. Value can be obtained from `autorange_curve`.

V. RBioplot Resources

To customize your graphs and make them more visually appealing (or usually you have to make them more boring for journals), you can refer to the R colour palette and R font table below.

R colour palette

brown4	darkorange4	gray	gray57	hotpink3	lightsalmon4	navajowhite1	plum3	slategray3	antiquewhite
brown3	darkorange3	goldenrod4	gray56	hotpink2	lightsalmon3	navajowhite	plum2	slategray2	aliceblue
brown2	darkorange2	goldenrod3	gray55	hotpink1	lightsalmon2	moccasin	plum1	slategray1	white
brown1	darkorange1	goldenrod2	gray54	hotpink	lightsalmon1	mistyrose4	plum	slategray	yellowgreen
brown	darkorange	goldenrod1	gray53	honeydew4	lightsalmon	mistyrose3	pink4	slateblue4	yellow4
blueviolet	darkolivegreen4	goldenrod	gray52	honeydew3	lightpink4	mistyrose2	pink3	slateblue3	yellow3
blue4	darkolivegreen3	gold4	gray51	honeydew2	lightpink3	mistyrose1	pink2	slateblue2	yellow2
blue3	darkolivegreen2	gold3	gray50	honeydew1	lightpink2	mistyrose	pink1	slateblue1	yellow1
blue2	darkolivegreen1	gold2	gray49	honeydew	lightpink1	mintcream	pink	slateblue	yellow
blue1	darkolivegreen	gold1	gray48	greenyellow	lightpink	midnightblue	peru	skyblue4	whitesmoke
blue	darkmagenta	gold	gray47	green4	lightgrey	mediumvioletred	peachpuff4	skyblue3	wheat4
blanchedalmond	darkkhaki	ghostwhite	gray46	green3	lightgreen	mediumturquoise	peachpuff3	skyblue2	wheat3
black	darkgrey	gainsboro	gray45	green2	lightgray	mediumspringgreen	peachpuff2	skyblue1	wheat2
bisque4	darkgreen	forestgreen	gray44	green1	lightgoldenrodyellow	mediumslateblue	peachpuff1	skyblue	wheat1
bisque3	darkgray	floralwhite	gray43	green	lightgoldenrod4	mediumseagreen	peachpuff	sienna4	wheat
bisque2	darkgoldenrod4	firebrick4	gray42	gray100	lightgoldenrod3	mediumpurple4	papayawhip	sienna3	violetred4
bisque1	darkgoldenrod3	firebrick3	gray41	gray99	lightgoldenrod2	mediumpurple3	palevioletred4	sienna2	violetred3
bisque	darkgoldenrod2	firebrick2	gray40	gray98	lightgoldenrod1	mediumpurple2	palevioletred3	sienna1	violetred2
beige	darkgoldenrod1	firebrick1	gray39	gray97	lightgoldenrod	mediumpurple1	palevioletred2	sienna	violetred1
azure4	darkgoldenrod	firebrick	gray38	gray96	lightcyan4	mediumpurple	palevioletred1	seashell4	violetred
azure3	darkcyan	dodgerblue4	gray37	gray95	lightcyan3	mediumorchid4	palevioletred	seashell3	violet
azure2	darkblue	dodgerblue3	gray36	gray94	lightcyan2	mediumorchid3	paleturquoise4	seashell2	turquoise4
azure1	cyan4	dodgerblue2	gray35	gray93	lightcyan1	mediumorchid2	paleturquoise3	seashell1	turquoise3
azure	cyan3	dodgerblue1	gray34	gray92	lightcyan	mediumorchid1	paleturquoise2	seashell	turquoise2
aquamarine4	cyan2	dodgerblue	gray33	gray91	lightcoral	mediumorchid	paleturquoise1	seagreen4	turquoise1
aquamarine3	cyan1	dimgrey	gray32	gray90	lightblue4	mediumblue	paleturquoise	seagreen3	turquoise
aquamarine2	cyan	dimgray	gray31	gray89	lightblue3	mediumaquamarine	palegreen4	seagreen2	tomato4
aquamarine1	cornsilk4	deepskyblue4	gray30	gray88	lightblue2	maroon4	palegreen3	seagreen1	tomato3
aquamarine	cornsilk3	deepskyblue3	gray29	gray87	lightblue1	maroon3	palegreen2	seagreen	tomato2
antiquewhite4	cornsilk2	deepskyblue2	gray28	gray86	lightblue	maroon2	palegreen1	sandybrown	tomato1
antiquewhite3	cornsilk1	deepskyblue1	gray27	gray85	lemonchiffon4	maroon1	palegreen	salmon4	tomato
antiquewhite2	cornsilk	deepskyblue	gray26	gray84	lemonchiffon3	maroon	palegoldenrod	salmon3	thistle4
antiquewhite1	cornflowerblue	deeppink4	gray25	gray83	lemonchiffon2	magenta4	orchid4	salmon2	thistle3
antiquewhite	coral4	deeppink3	gray24	gray82	lemonchiffon1	magenta3	orchid3	salmon1	thistle2
aliceblue	coral3	deeppink2	gray23	gray81	lemonchiffon	magenta2	orchid2	salmon	thistle1
white	coral2	deeppink1	gray22	gray80	lawngreen	magenta1	orchid1	saddlebrown	thistle
bisque3	coral1	deeppink	gray21	gray79	lavenderblush4	magenta	orchid	royalblue4	tan4
bisque2	coral	darkviolet	gray20	gray78	lavenderblush3	linen	orangered4	royalblue3	tan3
bisque1	chocolate4	darkturquoise	gray19	gray77	lavenderblush2	limegreen	orangered3	royalblue2	tan2
bisque	chocolate3	darkslategrey	gray18	gray76	lavenderblush1	lightyellow4	orangered2	royalblue1	tan1
beige	chocolate2	darkslategray4	gray17	gray75	lavenderblush	lightyellow3	orangered1	royalblue	tan
azure4	chocolate1	darkslategray3	gray16	gray74	lavender	lightyellow2	orangered	rosybrown4	steelblue4
azure3	chocolate	darkslategray2	gray15	gray73	khaki4	lightyellow1	orange4	rosybrown3	steelblue3
azure2	chartreuse4	darkslategray1	gray14	gray72	khaki3	lightyellow	orange3	rosybrown2	steelblue2
azure1	chartreuse3	darkslategray	gray13	gray71	khaki2	lightsteelblue4	orange2	rosybrown1	steelblue1
azure	chartreuse2	darkslateblue	gray12	gray70	khaki1	lightsteelblue3	orange1	rosybrown	steelblue
aquamarine4	chartreuse1	darkseagreen4	gray11	gray69	khaki	lightsteelblue2	orange	red4	springgreen4
aquamarine3	chartreuse	darkseagreen3	gray10	gray68	ivory4	lightsteelblue1	olivedrab4	red3	springgreen3
aquamarine2	cadetblue4	darkseagreen2	gray9	gray67	ivory3	lightsteelblue	olivedrab3	red2	springgreen2
aquamarine1	cadetblue3	darkseagreen1	gray8	gray66	ivory2	lightslategrey	olivedrab2	red1	springgreen1
aquamarine	cadetblue2	darkseagreen	gray7	gray65	ivory1	lightslategray	olivedrab1	red	springgreen
antiquewhite4	cadetblue1	darksalmon	gray6	gray64	ivory	lightslateblue	olivedrab	purple4	snow4
antiquewhite3	cadetblue	darkred	gray5	gray63	indianred4	lightskyblue4	oldlace	purple3	snow3
antiquewhite2	burlywood4	darkorchid4	gray4	gray62	indianred3	lightskyblue3	navyblue	purple2	snow2
antiquewhite1	burlywood3	darkorchid3	gray3	gray61	indianred2	lightskyblue2	navy	purple1	snow1
antiquewhite	burlywood2	darkorchid2	gray2	gray60	indianred1	lightskyblue1	navajowhite4	purple	snow
aliceblue	burlywood1	darkorchid1	gray1	gray59	indianred	lightskyblue	navajowhite3	powderblue	slategrey
white	burlywood	darkorchid	gray0	gray58	hotpink4	lightseagreen	navajowhite2	plum4	slategray4

R font table