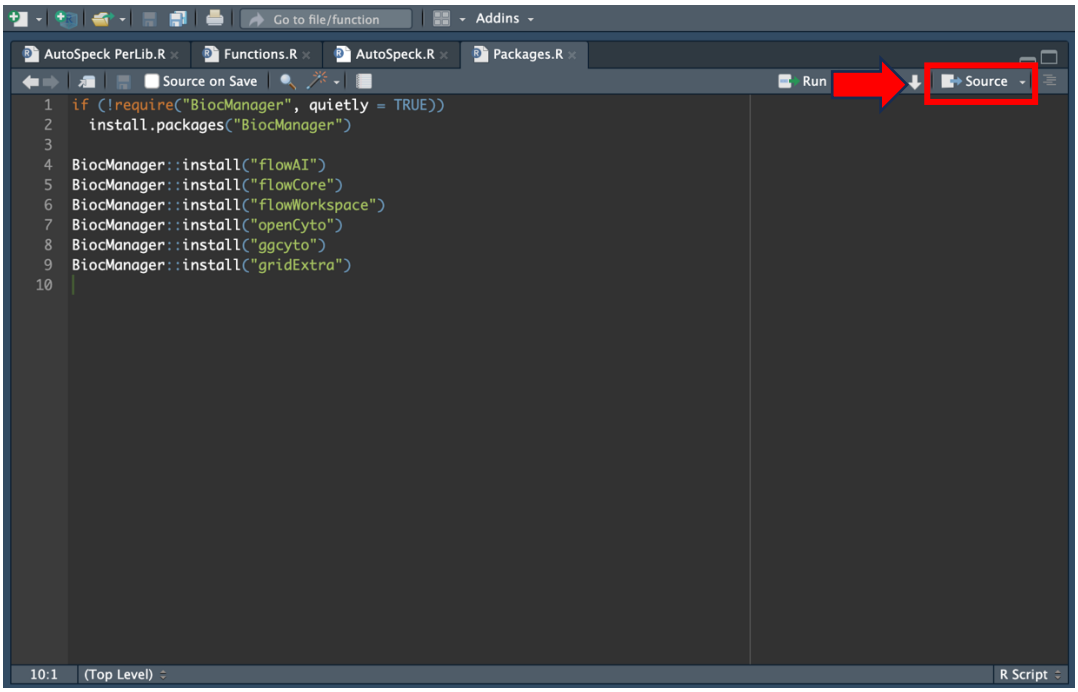
Automated Speck Assay Manual

Welcome! This manual will provide you with a guide on how to use our script for analysing your own speck assays, along with how to customise the settings to best suit your data and assay setup, even if you don't know much (or anything) about R.

<u>Setting up the program on your own machine</u>

1. First off, you'll need to have R and RStudio installed
2. You'll notice there are multiple files included in our repository, most of the time you won't need to worry about these, but for the first time you'll need to open "Packages.R" which should look like this:



3. Click 'Source' (highlighted in the top right) to run the code, which will install the packages required to run the program
4. Once this is done, you can close this file

Setting your parameters

1. Once you have all the required packages, we can then move on to opening the main program and setting your parameters
2. First, you should load your FCS files so that they may be accessible by the program. Take your files and move them to the “Data” folder. Each set of samples to be analysed together should be in their own folder.
3. Open “app.R”
4. Pick the folder containing your .fcs files
5. Select your ASC and Inflammasome activator (NLRP3 in our setup) fluorescence channels
6. You will then be prompted to input your control sample. This will be used as the control both for setting gates and generating an ASC50 value. If following our experimental setup, WT NLRP3 is used.
7. Once the program has completed, you should be able to see a new folder inside the ‘Results’ folder, which will contain graphical representations of all the gating applied, a summary graph of relative ec50 values, the fitted curves used to generate ec50 values, and an excel sheet of all the output data.

Function Guide

Here is a list of the custom functions used in this program, along with a guide on how to use them.

---

fcsImportLogicle Import FCS data and transform channels to logicle values

Usage

```
fcsImportLogicle(
        path,
        clean,
        logTrans
)
```

Arguments

path	A string value containing the directory of the FCS files to be imported
clean	A TRUE or FALSE value indicating if the user wants data cleaned using
logTrans	A TRUE or FALSE value indicating if the data should be transformed to a logicle or not

---

gate2dc Create a two-dimensional gate

Usage

```
gate2dc(
        gatingSet,
        parentPop,
        xchannel,
        ychannel,
        quantile,
        name,
        plot,
        kpop,
        save,
        target,
        controlSample
)
```

Arguments

gatingSet	Name of the “GatingSet” flowWorkspace object (gs by default)
parentPop	Name of the parent gate
xchannel	X axis fluorescence channel
ychannel	Y axis fluorescence channel
quantile	How restrictive the gate is to the population; higher value indicates more restricted gate.
name	Name of the gate
plot	TRUE or FALSE value indicating if the plots should be displayed in R
kpop	The expected number of populations present
save	TRUE or FALSE value indicating if the gating plots should be saved to the output folder
target	A list containing two values, the x and y values indicating the expected location of the population.
controlSample	The sample number to be used to set gating. Already assigned by the gatingControl variable

---

gate1dc Create a one-dimensional gate

Usage

```
gate1dc(
        gatingSet,
        parentPop,
        xchannel,
        range,
        name,
        plot,
        positive,
        smoothing,
        peaks,
        save,
        controlSample
)
```

Arguments

gatingSet	Name of the “GatingSet” flowWorkspace object (gs by default)
parentPop	Name of the parent gate
xchannel	X axis fluorescence channel
range	A list containing the range of data

name	Name of the gate
plot	TRUE or FALSE value indicating if the plots should be displayed in R
positive	TRUE or FALSE value indicating if the positive or negative population should be gated
smoothing	Degree of smoothing applied to the histogram
peaks	Numeric vector of the locations of peaks, usually left as NULL
save	TRUE or FALSE value indicating if the gating plots should be saved to the output folder
controlSample	The sample number to be used to set gating. Already assigned by the gatingControl variable

---

exportSingleCell Export single cell fluorescence values for a particular channel

Usage

```
exportSingleCell(
        speckPosGate,
        speckNegGate,
        ascGate,
        facsChannel
)
```

Exports the global variables:
speckName: List of well IDs for each sample
speckPosRaw/speckNegRaw: List of all single cell fluorescence values for speck positive/negative population
speckAll: Combination of speckPosRaw and speckNegRaw

Arguments

speckPosGate	String containing name of gate on speck positive population
speckNegGate	String containing name of gate on speck negative population
ascGate	String containing name of gate on total population, pre speck gating
facsChannel	String containing name of FACS channel to be exported

stepBin Bin single cell values using a single wide bin that steps through data

Usage

```
stepBin(
        index,
        stepLen,
        speckAll,
        speckPosRaw,
        speckNegRaw
)
```

Arguments

index	Numeric index of sample to be analysed
stepLen	Length of bin step
speckAll speckPosRaw speckNegRaw	Exported data from exportSingleCell