Umbrella Tutorial 1D

Bart K. M. Jacobs

5 april 2017

==

This tutorial explains how to run Umbrella when single-channel data are available in an RData file in the correct format.

To create an RData file from .csv files, please consult `Tutorial_DataCreation`.

To analyse multi-channel data, please consult `Tutorial_Umbrella_2D`.

Introduction to R

First, load the RData file `Tutorial_1D.RData`. You can do this through File -> Load Workspace in Base R, or File -> Open File… in RStudio. Alternatively, execute the following command after changing the path accordingly

```
load("C:\\temp\\Tutorial_1D.RData")
```

With the `ls()` command, you can see all objects currently loaded.
In RStudio, such list can also be consulted in the Global Environment, in the upper right panel.

```
ls()
```

```
## [1] "PlotUmbrella1d"    "Tutorial.Plate.1D" "Umbrella1d"
```

This file contains three objects:

- "Tutorial.Plate.1D", a simulated example dataset
- "Umbrella1d", the Umbrella function for single-channel data
- "PlotUmbrella1d", a routine to create graphics

You can check the structure of an R object with `str()` and `attributes()`

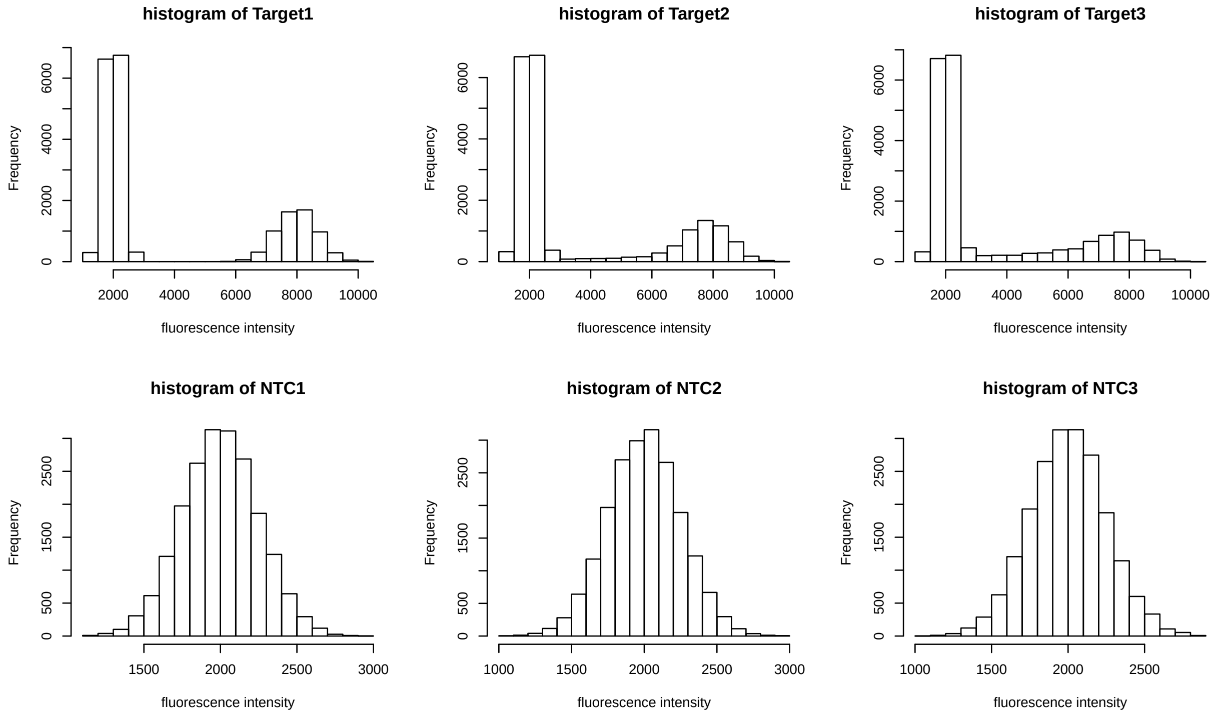
```
str(Tutorial.Plate.1D)
```

```
## 'data.frame':    20000 obs. of  6 variables:
##  $ Target1: num  2086 1821 1744 2198 8712 ...
##  $ Target2: num  1827 8848 1710 2018 1530 ...
##  $ Target3: num  2139 1720 2018 2376 7485 ...
##  $ NTC1   : num  1919 1843 2268 2334 2020 ...
##  $ NTC2   : num  2159 2012 2182 1485 2086 ...
##  $ NTC3   : num  1725 1989 1763 2270 2042 ...
```

Input data for Umbrella needs to be structured as a list with a single, named entry for each partition set.

Let's make histograms of the data:

```
par(mfrow=c(2,3))
for(i in 1:6){
hist(Tutorial.Plate.1D[[i]],xlab="fluorescence intensity",
  main=paste("histogram of",names(Tutorial.Plate.1D)[i]))
}
```



```r
par(mfrow=c(1,1))
```

We see that the clusters are nicely separated in Target 1, while Target2 and Target3 show increasing rain.

Running Umbrella

In general, you want to open the `Umbrella_1D.R` file and run all the code in it. Note that this file will have a version number which can change as the software gets updated: e.g. `Umbrella_1d_V0.R`

For this tutorial, the functions have already been included in the Rdata file.

Umbrella requires 5 other packages to be loaded, these are: MASS, mgcv, robust, modeest, OrdMonReg

Please intall them before running the procedure. This can be done through Packages -> Install Package(s)… in Base R or with Tools -> Install Packages… in RStudio. In most installations, running the following code will work as well:

```r
install.packages(c("MASS","mgcv","robust","modeest","OrdMonReg"))
```

Run Umbrella on the tutorial data set with the following code:

```r
ResTut <- Umbrella1d(Tutorial.Plate.1D,NTC=c("NTC1","NTC2","NTC3"))
```

```
## Step 1: Identify negative cluster centre per well - completed
## Step 2: Check centres with plate info - completed
## Step 3: NTC identification - completed
## Estimation Target1 completed in 20.25803 secs .
##
## Well Target1 :
##
## 20000 partitions
## Robust estimator:
```

```
##    6008 positive, p = 0.3004 ( CI: [ 0.2898 , 0.311 ] )
##    420.3 copies/mul ( CI: [ 399.5 , 441 ] )
## Threshold estimator:
##    6021 positive, p = 0.301 ( CI: [ 0.301 , 0.301 ] )
##    421.4 copies/mul ( CI: [ 410.7 , 432.1 ] )
## CI's for concentration include Poisson variability.
## __________________________________________________
## Estimation Target2 completed in 21.08511 secs .
##
## Well Target2 :
##
## 20000 partitions
## Robust estimator:
##    5914 positive, p = 0.2957 ( CI: [ 0.284 , 0.3074 ] )
##    412.4 copies/mul ( CI: [ 390.3 , 434.6 ] )
## Threshold estimator:
##    5893 positive, p = 0.2947 ( CI: [ 0.2934 , 0.2964 ] )
##    410.7 copies/mul ( CI: [ 399.7 , 421.6 ] )
## CI's for concentration include Poisson variability.
## __________________________________________________
## Estimation Target3 completed in 21.91419 secs .
##
## Well Target3 :
##
## 20000 partitions
## Robust estimator:
##    5697 positive, p = 0.2848 ( CI: [ 0.2721 , 0.2976 ] )
##    394.4 copies/mul ( CI: [ 371 , 417.8 ] )
## Threshold estimator:
##    5717 positive, p = 0.2858 ( CI: [ 0.2836 , 0.2878 ] )
##    396.1 copies/mul ( CI: [ 385 , 407.1 ] )
## CI's for concentration include Poisson variability.
## __________________________________________________
## Estimation NTC1 completed in 24.52645 secs .
##
## Well NTC1 :
##
## 20000 partitions
## Robust estimator:
##    0 positive, p = 0 ( CI: [ 0 , 0 ] )
##    0 copies/mul ( CI: [ 0 , 0 ] )
## Threshold estimator:
##    0 positive, p = 0 ( CI: [ 0 , 0 ] )
##    0 copies/mul ( CI: [ 0 , 0 ] )
## CI's for concentration include Poisson variability.
## __________________________________________________
## Estimation NTC2 completed in 16.48365 secs .
##
## Well NTC2 :
##
## 20000 partitions
## Robust estimator:
##    79 positive, p = 0.0039 ( CI: [ 0 , 0.0121 ] )
##    4.6 copies/mul ( CI: [ 0 , 14.4 ] )
## Threshold estimator:
##    0 positive, p = 0 ( CI: [ 0 , 0 ] )
##    0 copies/mul ( CI: [ 0 , 0 ] )
## CI's for concentration include Poisson variability.
## __________________________________________________
## Estimation NTC3 completed in 23.08631 secs .
##
## Well NTC3 :
##
## 20000 partitions
## Robust estimator:
##    0 positive, p = 0 ( CI: [ 0 , 0.0023 ] )
##    0 copies/mul ( CI: [ 0 , 3.7 ] )
## Threshold estimator:
##    0 positive, p = 0 ( CI: [ 0 , 0 ] )
##    0 copies/mul ( CI: [ 0 , 0 ] )
## CI's for concentration include Poisson variability.
## __________________________________________________
## Step 4: Estimation - complete
```

Only the first argument is needed. This should be a list of partition sets with single channel fluorescence intensities.

Currently, there are two optional arguments:

- vol= for the volume in nanoliter, default value 0.85 for Bio-Rad's QX100.
- NTC= for a vector with names of the NTCs.

In this example, we keep the default value for the volume and supply three NTCs with names NTC1, NTC2, NTC3.
Note: Umbrella can run without NTCs, but underlying distributional assumptions are likely violated in this case! Please do provide NTCs.

Extensive output is stored in the "ResTut" object. The name of the result object can be freely chosen.

While the procedure is running, some output is already shown in the R console. This includes:

- During the first run: messages about all packages that are loaded.
- The steps when they are concluded.
- Per partition set:
- The time to complete the estimation step.
- The number of partitions
- The number of positives and the concentration for both the overall Umbrella estimator and a probabilistic threshold estimator
- A confidence interval for this number and the concentration

A new object, in this case named "ResTut" is created.
The structure of this object can be summarized with `str()`. The object is a list with a separate entry for each partition set. The structure is the same over all partition sets. In what follows, we'll look at the details for Target3.

```r
str(ResTut$Target3,max.level=1)
```

```
## List of 11
##  $ conc    : num [1:4, 1:4] 2.85e-01 5.64e-03 2.72e-01 2.98e-01 3.94e+02 ...
##   ..- attr(*, "dimnames")=List of 2
##  $ tresh   : num [1:3, 1:6] 0.28585 0.7068 0.00735 0.2836 0.70615 ...
##   ..- attr(*, "dimnames")=List of 2
##  $ fits    :List of 3
##  $ data    : num [1:20000] 2139 1720 2018 2376 7485 ...
##  $ droppi  : num [1:20000] 1.00 1.00 1.00 1.00 3.11e-12 ...
##  $ dropci  : num [1:20000, 1:2] 0.93 1 0.961 0.652 0 ...
##   ..- attr(*, "dimnames")=List of 2
##  $ reppi   : num [1:20000, 1:3] 1.02 1.00 1.02 1.01 1.66e-12 ...
##  $ densfits:List of 3
##  $ ntcfits :List of 3
##  $ pifits  :List of 3
##  $ name    : chr "Target3"
```

For each partition set, a list of 11 pieces of information is created. In what follows, we'll go through all of them, although most likely you will only want to use the first few.

1. $ conc

```
ResTut$Target3$conc
```

```
##       prob_robust conc_robust  prob_thres conc_thres
## est   0.284835109  394.402524 0.285850000 396.073241
## sd    0.005644342    9.285135 0.001061053   1.747949
## CI LB 0.272060995  371.004610 0.283600000 385.025259
## CI UB 0.297609222  417.800439 0.287750000 407.121222
```

These are the estimates of (i) the probability to be positive and (ii) the concentration of the sample (in copies/microliter) for both Umbrella estimators (overall and probability threshold). (first line) Additionally, their standard deviation (second line) and confidence intervals are added (third and fourth line).
Note: proper standard deviations and confidence intervals can only be calculated when multiple NTCs are provided

2. $ thres

```
ResTut$Target3$tresh
```

```
##         prob CI low p CI high p count CI low n CI high n
## pos  0.28585  0.28360   0.28775  5717     5672      5755
## neg  0.70680  0.70615   0.70760 14136    14123     14152
## rain 0.00735  0.00490   0.01000   147       98       200
```

These are the proportion (probability) of partitions that are positive (pos), negative (neg) or rain (rain) with a confidence interval on each, as well as the associated absolute numbers.
Proportions are calculated as absolute numbers divided by the total number of partitions.

3. $ fits

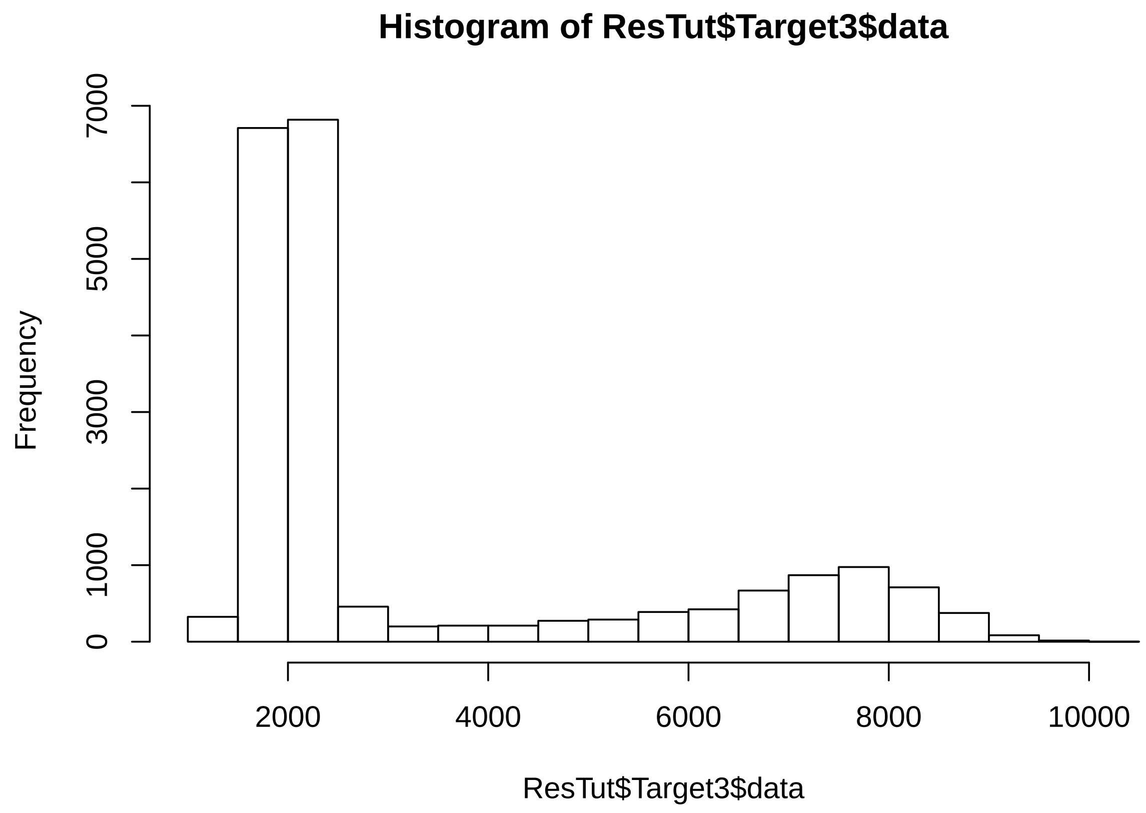
```
ResTut$Target3$fits
```

```
## $mu
## [1] 2008.714 2008.714 2008.714
##
## $sig
## [1] 248.1197 250.3868 246.7626
##
## $pi
## [1] 0.7093865 0.7206648 0.7154434
```

Estimates used for the mode (mu), robust deviation (sig) and the probability that a random target partition is negative (pi). These estimates are recalculated for each combination with a different NTC. I.e.: the first number refers to the estimate calculated and used in the joint model with Target3 and NTC1, the second number is the estimate from the model with Target3 and NTC2, and so on.

4. $ data

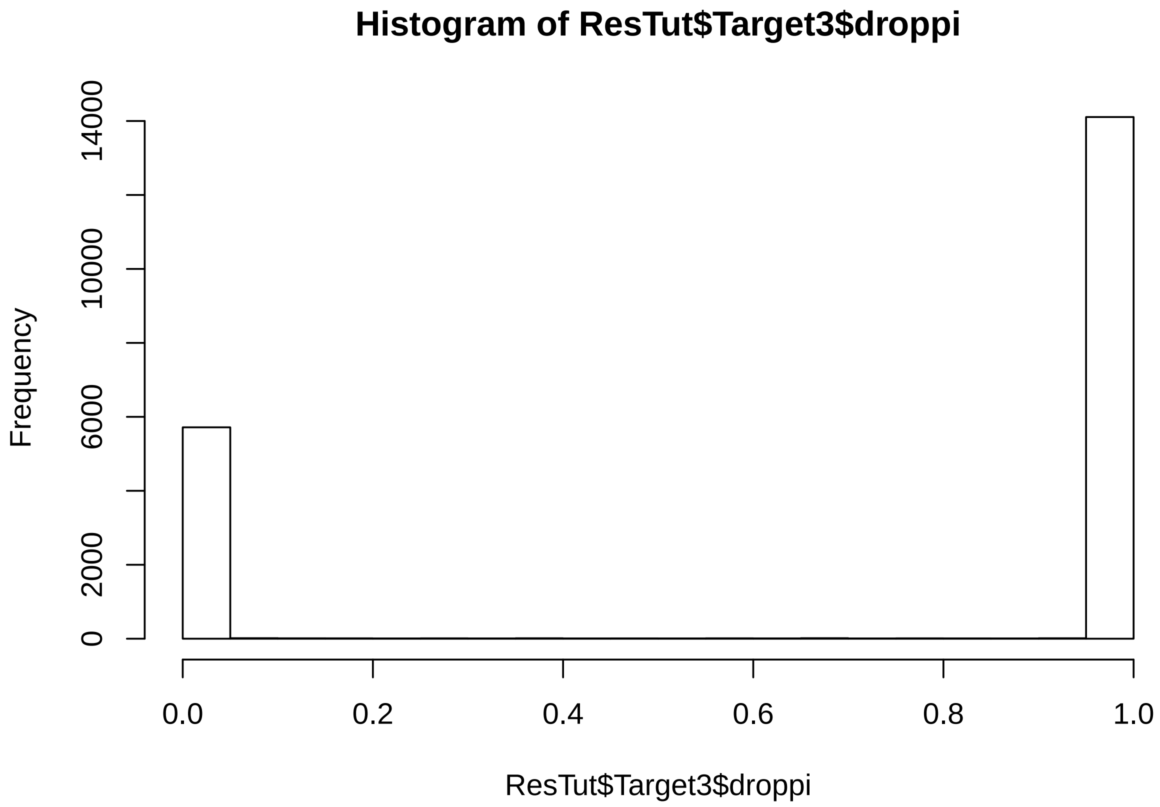
```
hist(ResTut$Target3$data)
```



The original data

5. $ droppi

```
hist(ResTut$Target3$droppi)
```



The estimates for the individual partition probabilities

6. $ dropci

```
str(ResTut$Target3$dropci)
```

```
##  num [1:20000, 1:2] 0.93 1 0.961 0.652 0 ...
##  - attr(*, "dimnames")=List of 2
##   ..$ : NULL
##   ..$ : chr [1:2] "probciL" "probciU"
```

Confidence intervals for the individual droplet probabilities

7. $ reppi

```
str(ResTut$Target3$reppi)
```

```
##  num [1:20000, 1:3] 1.02 1.00 1.02 1.01 1.66e-12 ...
```

The original partition probabilities per NTC before antitonic regression and averaging over NTCs. Mainly used for quality control (QC).

8. $ densfits

```
str(ResTut$Target3$densfits)
```

```
## List of 3
##  $ : num [1:20000(1d)] 9.63e-04 6.07e-04 1.12e-03 3.82e-04 9.47e-05 ...
##   ..- attr(*, "dimnames")=List of 1
##   .. ..$ : chr [1:20000] "10026" "1836" "7425" "13176" ...
##  $ : num [1:20000(1d)] 9.64e-04 6.05e-04 1.11e-03 3.81e-04 9.46e-05 ...
##   ..- attr(*, "dimnames")=List of 1
##   .. ..$ : chr [1:20000] "10026" "1836" "7425" "13176" ...
##  $ : num [1:20000(1d)] 9.63e-04 6.03e-04 1.12e-03 3.87e-04 9.47e-05 ...
##   ..- attr(*, "dimnames")=List of 1
##   .. ..$ : chr [1:20000] "10026" "1836" "7425" "13176" ...
```

The fitted density of the target when jointly modelled with each NTC.

9. $ ntcfits

```
str(ResTut$Target3$ntcfits)
```

```
## List of 3
##  $ : num [1:20000(1d)] 9.79e-04 5.85e-04 1.14e-03 3.86e-04 1.58e-16 ...
##   ..- attr(*, "dimnames")=List of 1
##   .. ..$ : chr [1:20000] "10026" "1836" "7425" "13176" ...
##  $ : num [1:20000(1d)] 1.01e-03 5.89e-04 1.14e-03 4.03e-04 1.60e-16 ...
##   ..- attr(*, "dimnames")=List of 1
##   .. ..$ : chr [1:20000] "10026" "1836" "7425" "13176" ...
##  $ : num [1:20000(1d)] 1.02e-03 5.78e-04 1.14e-03 3.57e-04 1.59e-16 ...
##   ..- attr(*, "dimnames")=List of 1
##   .. ..$ : chr [1:20000] "10026" "1836" "7425" "13176" ...
```

The fitted density of each NTC when jointly modelled with the target.

10. $ pifits

```
str(ResTut$Target3$pifits)
```

```
## List of 3
##  $ : num [1:51(1d)] 0.775 0.762 0.753 0.748 0.746 ...
##   ..- attr(*, "dimnames")=List of 1
##   .. ..$ : chr [1:51] "52" "53" "54" "55" ...
##  $ : num [1:51(1d)] 0.677 0.69 0.704 0.719 0.734 ...
##   ..- attr(*, "dimnames")=List of 1
##   .. ..$ : chr [1:51] "52" "53" "54" "55" ...
##  $ : num [1:51(1d)] 0.629 0.657 0.684 0.709 0.731 ...
##   ..- attr(*, "dimnames")=List of 1
##   .. ..$ : chr [1:51] "52" "53" "54" "55" ...
```

Ratio between target and NTC around the mode. Can be used for QC.

11. $ name

Name of the partition set, mainly used for internal automatic calling and matching and when displaying the results.

Summarizing Output

Automatic Figures

We created a file to provide some useful figures automatically. To use this, you generally want to open `Graphics_Umbrella_1d.R` and execute all the code. For this tutorial, we included the function in the already loaded RData file.

Change the path below to the folder where you want figures to appear and then run the code.

```
setwd("C:/temp")
```

Run the code below to generate and save some graphics directly onto your hard disk.

```
PlotUmbrella1d(ResTut,"Tutorial1D")
```

```
## pdf
##   2
```

The command has three arguments of which two are required.

- The name of an object created by the Umbrella1d procedure (here: ResTut)
- A name for the files that can be freely chosen (here: Tutorial1D)

- Optionally, the resolution can be chosen with the `res=` command (default 120 dpi)

A plot similar to the middle panel of Figure 2 of our manuscript will be created for each partition set. Important numeric output is added to this plot.
A single QC figure is added as well.

As an example, for Target3, Figure 1, below, is generated.

Using the R object (ResTut here), it is possible to create custom output. We give a few handy examples similar to figures in the manuscript.

Table with concentration estimates

Creating a table with concentration estimates for the complete plate / experiment can be done with the following code:

```
conrob <- t(sapply(ResTut,function(x){x$conc[c(1,3,4),2]}))
conrob
```

```
##                est    CI LB      CI UB
## Target1 420.276737 399.5499 441.003588
## Target2 412.418551 390.2588 434.578296
## Target3 394.402524 371.0046 417.800439
## NTC1      0.000000   0.0000   0.000000
## NTC2      4.633868   0.0000  14.377544
## NTC3      0.000000   0.0000   3.720962
```

The table above displays the results of the main Umbrella estimator.

```
conptr <- t(sapply(ResTut,function(x){x$conc[c(1,3,4),4]}))
conptr
```

```
##              est    CI LB    CI UB
## Target1 421.3836 410.6830 432.0843
## Target2 410.6602 399.7396 421.5807
## Target3 396.0732 385.0253 407.1212
## NTC1      0.0000   0.0000   0.0000
## NTC2      0.0000   0.0000   0.0000
## NTC3      0.0000   0.0000   0.0000
```

This table shows the results of the probability threshold estimator.

We can now save these together in a handy comma separated value file, which can be opened by most data processing programs, but also in Excel, Notepad, etc.

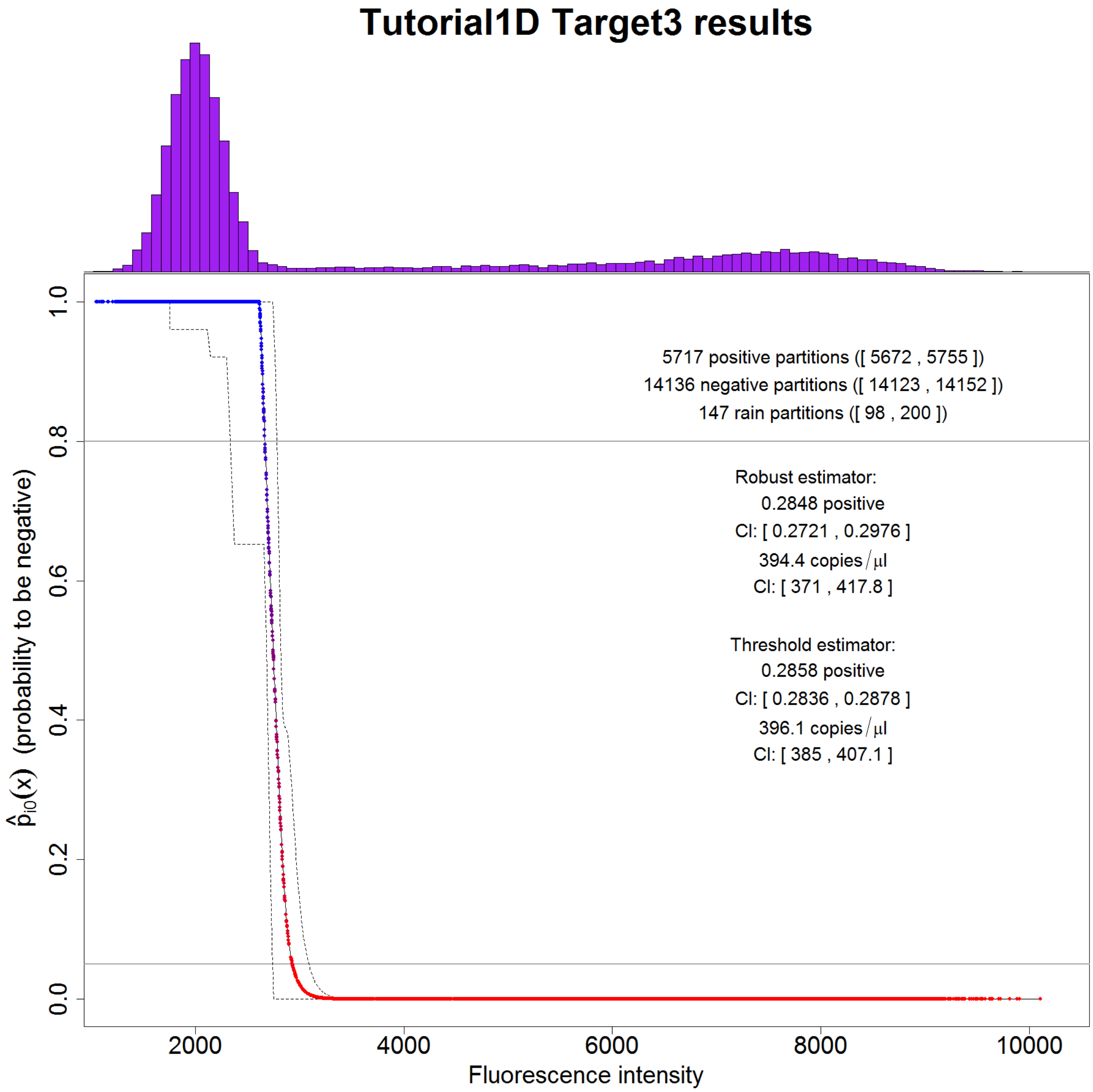


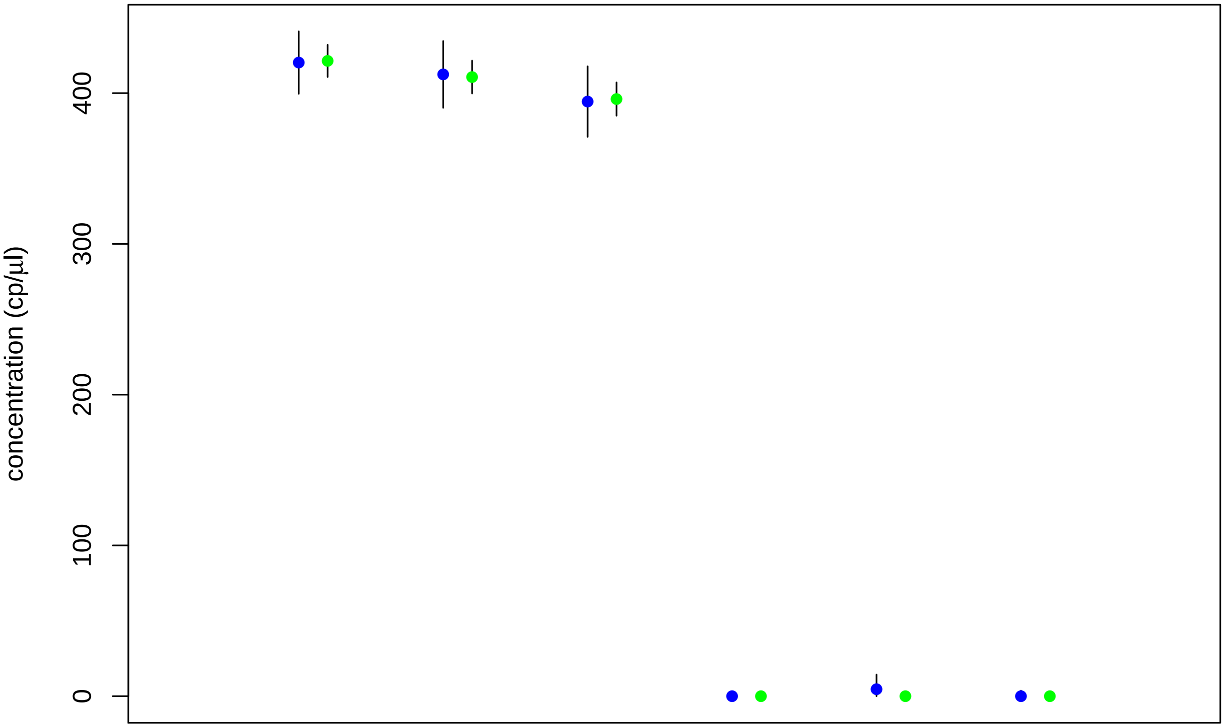
Figure 1: Output of PlotUmbrella1d

```
concdat <- t(sapply(ResTut,function(x){x$conc[c(1,3,4),c(2,4)]}))
colnames(concdat) <- c("est_main","CI_LB_main","CI_UB_main","est_thres","CI_LB_thres","CI_UB_thres")
write.csv(concdat,file="Tutorial_Concentrations.csv")
```

Plotting the concentrations

This creates output similar to Figure 3 in the manuscript.

```
ymaxs <- max(conrob[conrob<Inf],conptr[conrob<Inf])
npset <- nrow(conrob)
par(mar=c(1,4,1,1))
plot(NULL,ylim=c(0,ymaxs),xlim=c(0,npset+1),
    main="",xlab="",xaxt="n",
    ylab=expression(paste("concentration (cp/",mu,"l)",sep="")))
for(i in 1:npset){
lines(c(i-0.1,i-0.1),c(conrob[i,2],conrob[i,3]))
lines(c(i+0.1,i+0.1),c(conptr[i,2],conptr[i,3]))
points(i-0.1,conrob[i,1],col="blue",pch=16)
points(i+0.1,conptr[i,1],col="green",pch=16)
}
```


concentration (cp/µl)