

Ligand Optimisation using R

The purpose of these notes is to explain how to apply McGuigan's Ligand Optimisation Method (LOM) using software written in the package R. The software provides an automated version of the latest improved version of LOM which includes McGuigan's¹ modification of the Nicolsky-Eisenman equation.

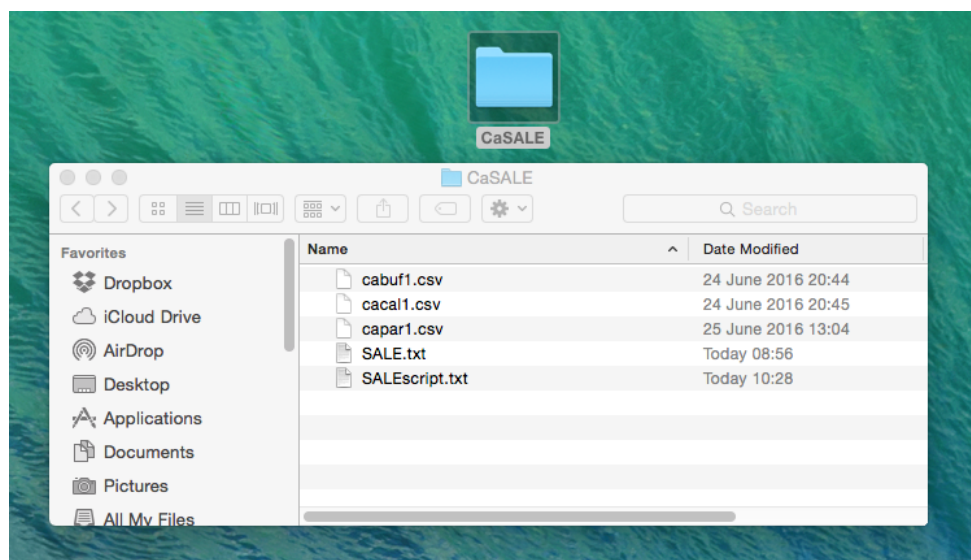
The R package

The user is required to download and install the free package R from <http://cran.r-project.org>, if necessary.

Once the user has opened R, running the code involves copying and pasting some commands in a script file into the R window and pressing ENTER, as explained below. It is not necessary for the user to know anything about R itself.

The Data Files

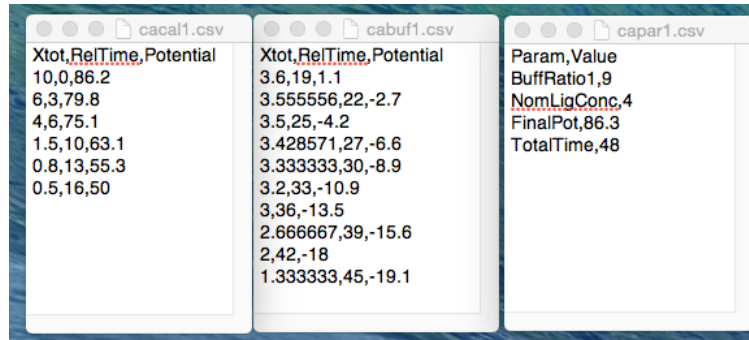
Let's assume that the folder CaSALE (as sent by email) has been placed on the user's Desktop, as shown below.



The folder CaSALE contains the five files that are necessary to run the program SALE on the input data. The three files with .csv extensions contain the input data. The file cacal1.csv contains the data obtained using the calibrating solutions. The file cabuf1.csv has the data obtained from the buffer solutions. The file capar1.csv contains the values of the input parameters.

¹McGuigan et al., Prog. Biophys. Mol. Biol., 116, 203-2011

Here is a screenshot of the contents of the files

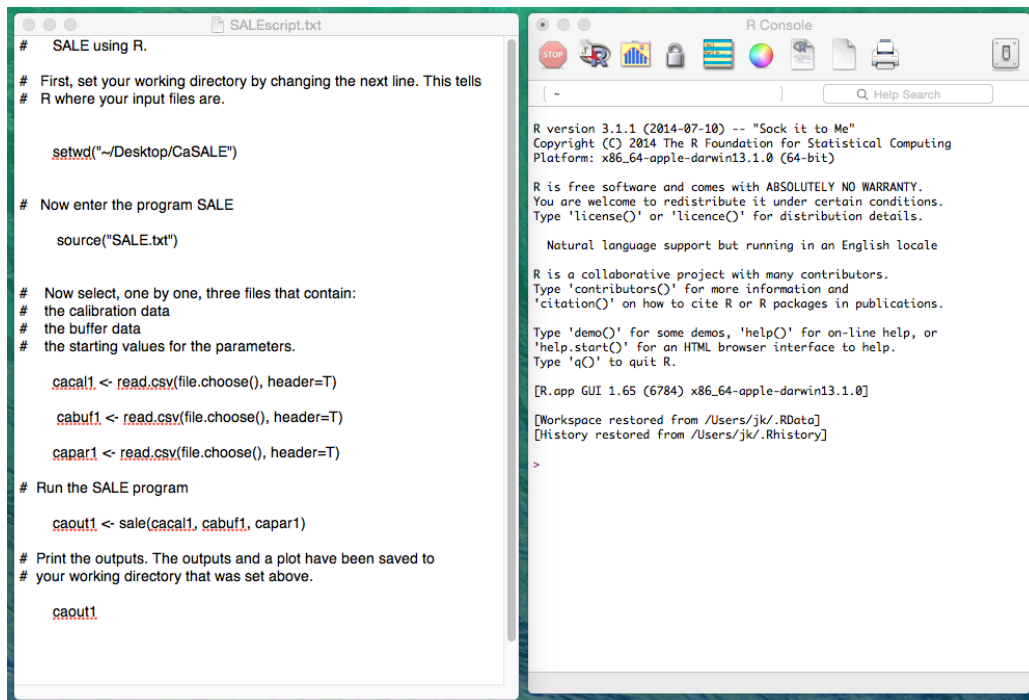


Xtot,RelTime,Potential	Xtot,RelTime,Potential	Param,Value
10,0,86.2	3.6,19,1.1	BuffRatio,9
6,3,79.8	3.555556,22,-2.7	NomLigConc,4
4,6,75.1	3.5,25,-4.2	FinalPot,86.3
1.5,10,63.1	3.428571,27,-6.6	TotalTime,48
0.8,13,55.3	3.333333,30,-8.9	
0.5,16,50	3.2,33,-10.9	
	3,36,-13.5	
	2.666667,39,-15.6	
	2,42,-18	
	1.333333,45,-19.1	

These files are in comma separated value (csv) format. A simple way to make data files in this format is to use Excel and then save the file in csv format. Alternatively, a text editor could be used to place the data in csv format; in each row the entries must be separated by commas rather than white space.

Running the SALE program

1. Run the application R. This will open a window into which R commands can be entered.
2. Open the file SALEscript.txt. This contains the the required R commands. The lines beginning with # are comment lines are are ignored by the R interpreter. Now, both windows will be open on the desktop, as shown below.



```

# SALE using R.

# First, set your working directory by changing the next line. This tells
# R where your input files are.

setwd("~/Desktop/CaSALE")

# Now enter the program SALE

source("SALE.txt")

# Now select, one by one, three files that contain:
# the calibration data
# the buffer data
# the starting values for the parameters.

cacal1 <- read.csv(file.choose(), header=T)
cabuf1 <- read.csv(file.choose(), header=T)
capar1 <- read.csv(file.choose(), header=T)

# Run the SALE program

caout1 <- sale(cacal1, cabuf1, capar1)

# Print the outputs. The outputs and a plot have been saved to
# your working directory that was set above.

caout1
  
```

```

R version 3.1.1 (2014-07-10) -- "Suck it to Me"
Copyright (C) 2014 The R Foundation for Statistical Computing
Platform: x86_64-apple-darwin13.1.0 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

[R.app GUI 1.65 (6784) x86_64-apple-darwin13.1.0]

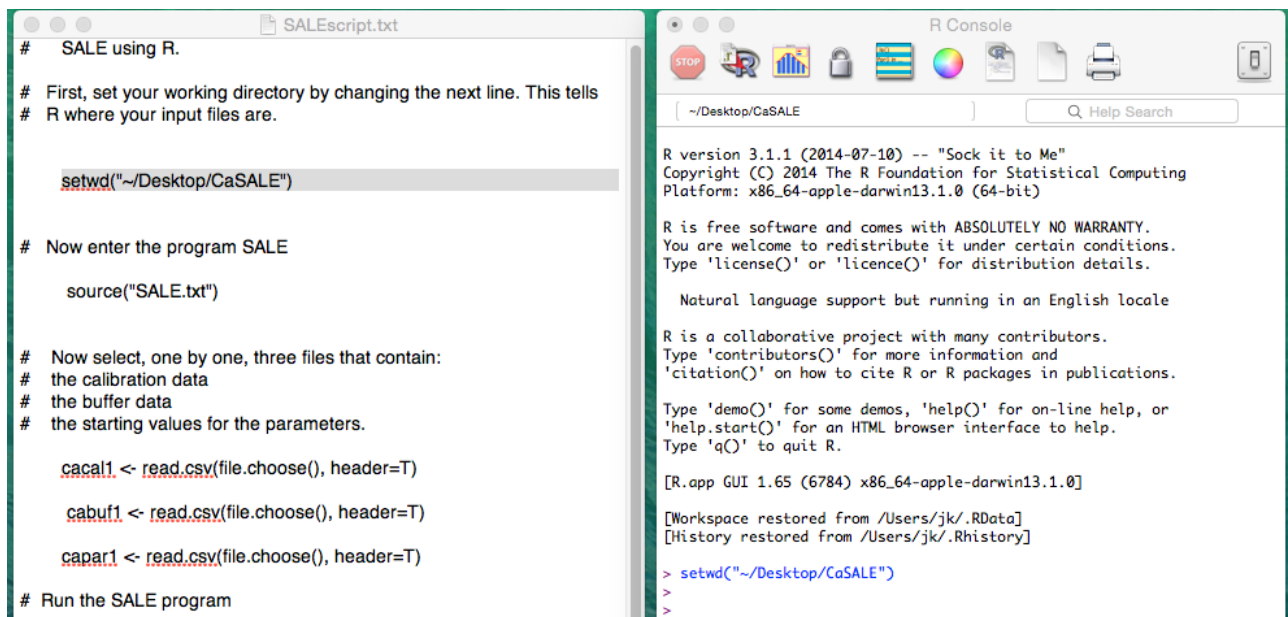
[Workspace restored from /Users/jk/.RData]
[History restored from /Users/jk/.Rhistory]
>
  
```

3. Set the working directory to tell R where the data files are; output will also be sent to this folder. For this illustration it has been assumed that the files are in a folder

named CaSALE and this is placed on the Desktop. The user can place the files in a folder with a name of their choosing and place the folder anywhere in their Documents folder; if so, the `setwd()` command needs editing.

NB. The form of the folder name given here is suitable for a Mac, but in Windows the slashes would be different.

Select `setwd("~/Desktop/CaSALE")` and copy this to the R window at the command prompt `>`, and then press ENTER to run this command.

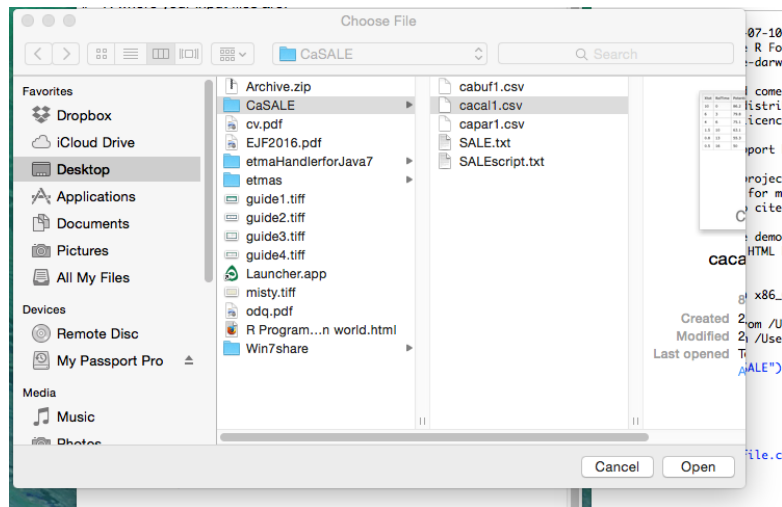


4. Select the line `source("SALE.txt")` and copy and paste this into the R window at the command prompt, and then press ENTER to run this command. This loads the program SALE into R, and this program will be run below once we have entered the data files.

5. Enter the data files one by one. Select the line

```
cacal1 <- read.csv(file.choose(), header=T)
```

and copy and paste this into the R window at the command prompt; then press ENTER. This command opens a dialogue box and gives the user access to their filesystem. Select Desktop, then CaSALE and then `cacal1.csv`. Then Click Open. The calibration data will have been entered into R.



6. Repeat step 5 to enter the buffer data (cabuf1.csv) and the input parameters (capar1.csv) into R.
7. Now the scene has been set and we can run the SALE program. Select the lines containing caout1, as shown below, and copy and paste them into the R window, and press ENTER. The output will appear in the R window. Also a plot of the fitted model and data will appear in a separate window.

```

# SALE using R.

# First, set your working directory by changing the next line. This tells
# R where your input files are.

setwd("~/Desktop/CaSALE")

# Now enter the program SALE

source("SALE.txt")

# Now select, one by one, three files that contain:
# the calibration data
# the buffer data
# the starting values for the parameters.

cacal1 <- read.csv(file.choose(), header=T)
cabuf1 <- read.csv(file.choose(), header=T)
capar1 <- read.csv(file.choose(), header=T)

# Run the SALE program

caout1 <- sale(cacal1, cabuf1, capar1)

# Print the outputs. The outputs and a plot have been saved to
# your working directory that was set above.

caout1

```

```

R Console

~/Desktop/CaSALE

> cacal1 <- read.csv(file.choose(), header=T)
> cabuf1 <- read.csv(file.choose(), header=T)
> capar1 <- read.csv(file.choose(), header=T)
> caout1 <- sale(cacal1, cabuf1, capar1)
> # Print the outputs. The outputs and a plot have been saved to
> # your working directory that was set above.
> caout1
[[1]]
  labels      parout
1      E0 5.606965e+01
2      s 2.803578e+01
3      Sigma 1.571457e-06
4      pKapp 6.519974e+00
5      Kapp 3.020133e-07
6      LigConc 3.741434e+00
7      Purity 9.353586e+01
8      RSS 9.260971e-01
9      r 9.988865e-01

[[2]]
      pX      X
1 5.136951 7.295406e-06
2 5.251869 5.599264e-06
3 5.366899 4.296364e-06
4 5.485149 3.272284e-06
5 5.610782 2.450290e-06
6 5.750035 1.778135e-06
7 5.913816 1.219507e-06
8 6.125744 7.486107e-07
9 6.460013 3.467266e-07
10 6.776795 1.671881e-07

```

8. The program will have written the output data and saved the plot to the folder CaSALE, in files with names starting with OUT, as shown below.

