Ligand Optimisation using R

The purpose of these notes is to explain how to apply McGuigan's new extended Ligand Optimisation Method (LOME) using software written in the package R.

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.

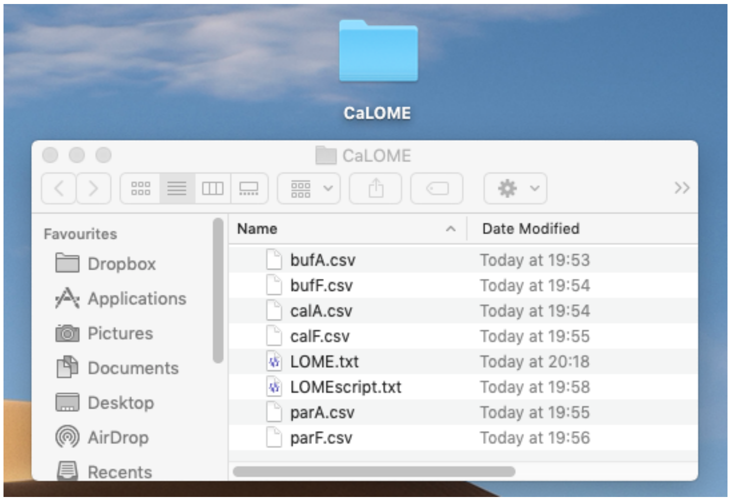
Also, download and install RStudio from:

https://www.rstudio.com/products/rstudio/download/

We will use RStudio in the following demonstration. The file.choose command used below works better in RStudio in that the working directory is made available to the user.

The Data Files

Let's assume that the folder CaLOME has been placed on the user's Desktop, as shown below.

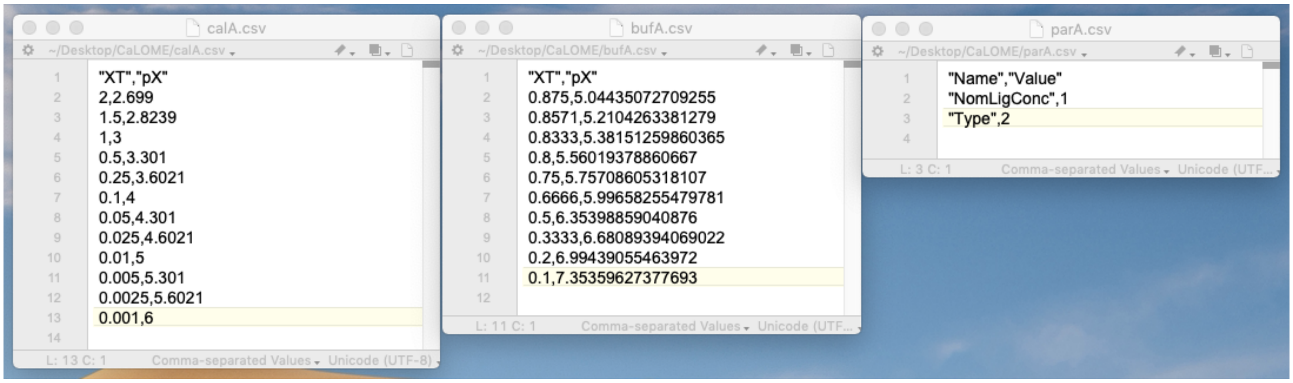


Name	Date Modified
bufA.csv	Today at 19:53
bufF.csv	Today at 19:54
calA.csv	Today at 19:54
calF.csv	Today at 19:55
LOME.txt	Today at 20:18
LOMEscript.txt	Today at 19:58
parA.csv	Today at 19:55
parF.csv	Today at 19:56

The folder CaLOME contains the files that are necessary to run the program LOME on the selected input data. There are two sets of files - one set for the Aequorin data and one set for the fluorescence data – with the contents shown in screenshots.

Aequorin Data Files

calA.csv	the data obtained using the calibrating solutions
bufA.csv	the data obtained from the buffer solutions
parA.csv	the parameter values for nominal ligand concentration and the type of data



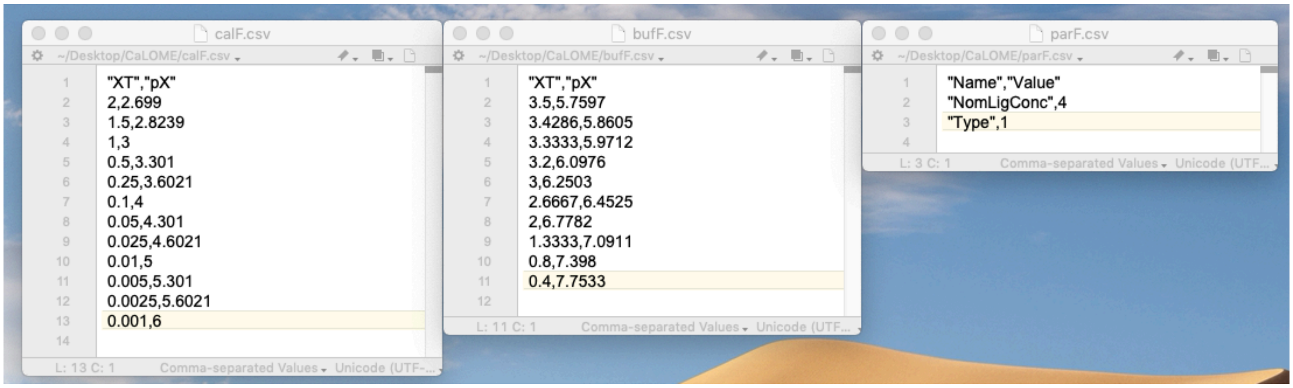
```
"XT","pX"
2,2.699
1.5,2.8239
1,3
0.5,3.301
0.25,3.6021
0.1,4
0.05,4.301
0.025,4.6021
0.01,5
0.005,5.301
0.0025,5.6021
0.001,6
```

```
"XT","pX"
0.875,5.04435072709255
0.8571,5.2104263381279
0.8333,5.38151259860365
0.8,5.56019378860667
0.75,5.75708605318107
0.6666,5.99658255479781
0.5,6.35398859040876
0.3333,6.68089394069022
0.2,6.99439055463972
0.1,7.35359627377693
```

```
"Name","Value"
"NomLigConc",1
"Type",2
```

Fluorescence Data Files

calF.csv	the data obtained using the calibrating solutions
bufF.csv	the data obtained from the buffer solutions
parF.csv	the parameter values for nominal ligand concentration and the type of data



```
"XT","pX"
2,2.699
1.5,2.8239
1,3
0.5,3.301
0.25,3.6021
0.1,4
0.05,4.301
0.025,4.6021
0.01,5
0.005,5.301
0.0025,5.6021
0.001,6
```

```
"XT","pX"
3.5,5.7597
3.4286,5.8605
3.3333,5.9712
3.2,6.0976
3,6.2503
2.6667,6.4525
2,6.7782
1.3333,7.0911
0.8,7.398
0.4,7.7533
```

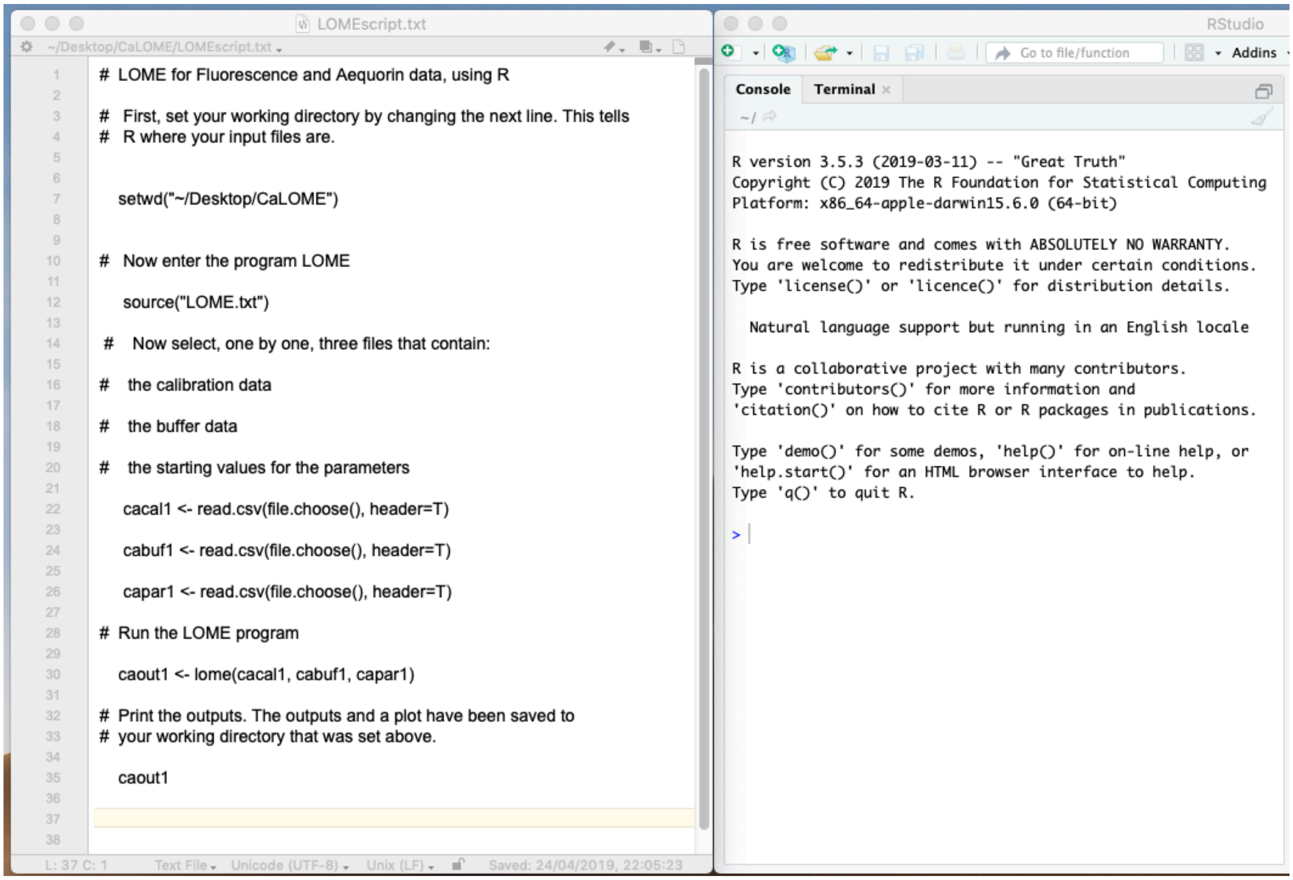
```
"Name","Value"
"NomLigConc",4
"Type",1
```

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 LOME program

1. Launch the application RStudio. This will open a window into which R commands can be entered.

2. Open the file LOMEscript.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.



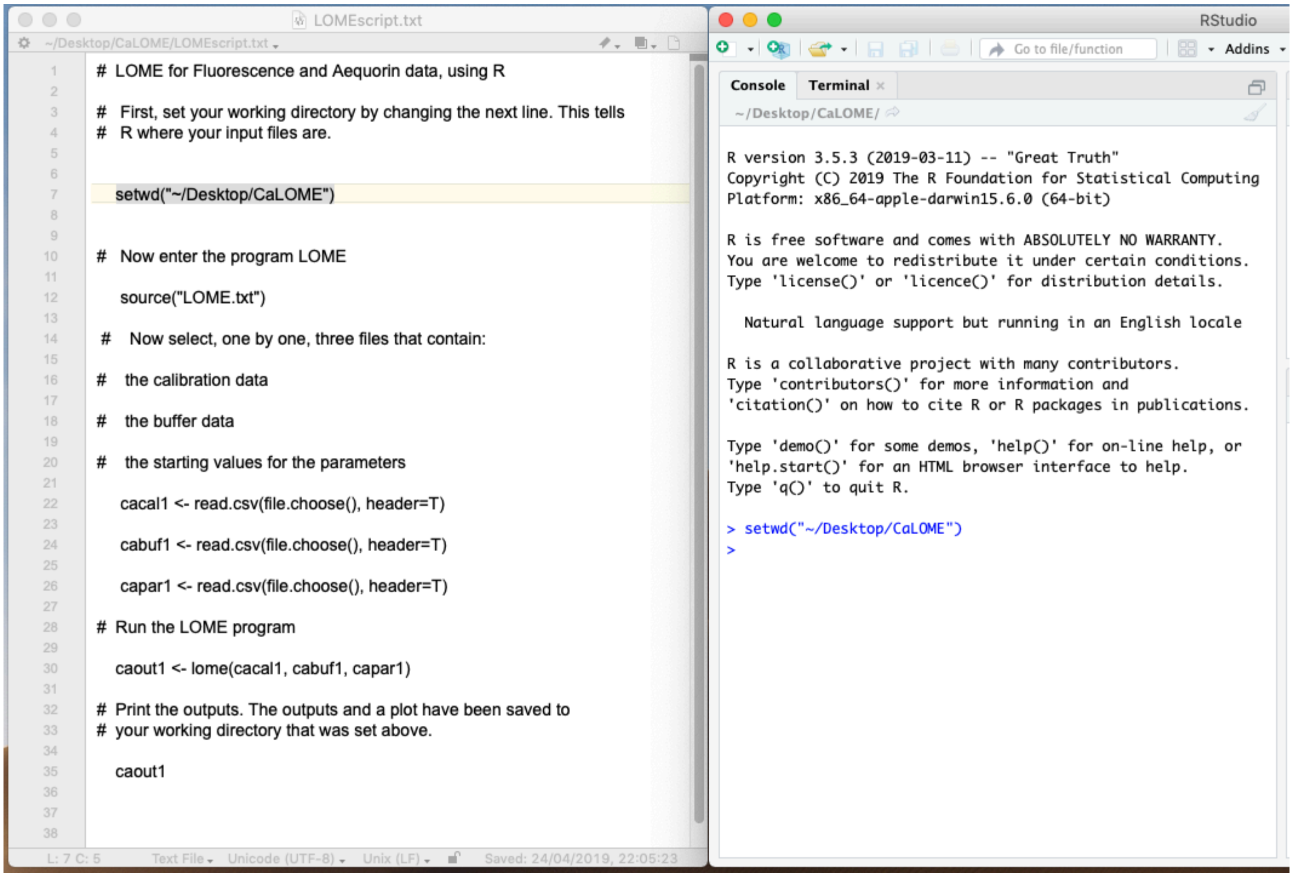
```
# LOME for Fluorescence and Aequorin data, using R

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


   setwd("~/Desktop/CaLOME")


#  Now enter the program LOME

    source("LOME.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 LOME program

   caout1 <- lome(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 3.5.3 (2019-03-11) -- "Great Truth"
Copyright (C) 2019 The R Foundation for Statistical Computing
Platform: x86_64-apple-darwin15.6.0 (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.

> 
```

3. Set the working directory to tell RStudio 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 CaLOME and it 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/CaLOME") and copy this to the R window at the command prompt >, and then press ENTER to run this command.



```
# LOME for Fluorescence and Aequorin data, using R

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


  setwd("~/Desktop/CaLOME")


#  Now enter the program LOME

   source("LOME.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 LOME program

  caout1 <- lome(cacal1, cabuf1, capar1)

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

  caout1
```

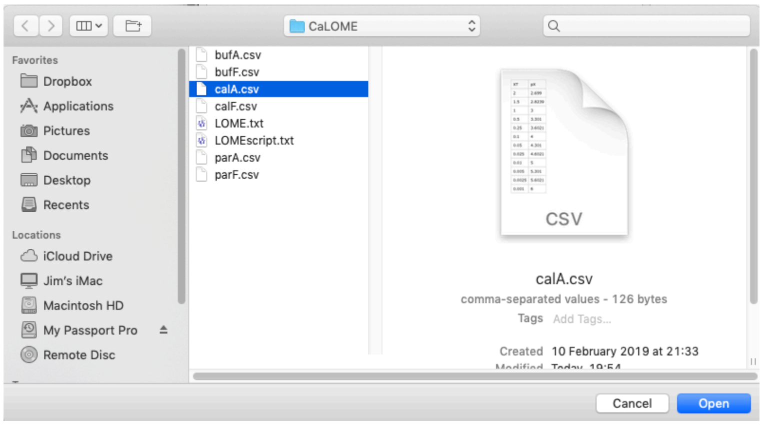
```
R version 3.5.3 (2019-03-11) -- "Great Truth"
Copyright (C) 2019 The R Foundation for Statistical Computing
Platform: x86_64-apple-darwin15.6.0 (64-bit)

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.

> setwd("~/Desktop/CaLOME")
>
```

4. Select the line source("LOME.txt") and copy and paste this into the RStudio window at the command prompt, and then press ENTER to run this command. This loads the program LOME into RStudio and this program will be run below once we have entered the data files. In what follows we will enter the data files for an application with Aequorin data.

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

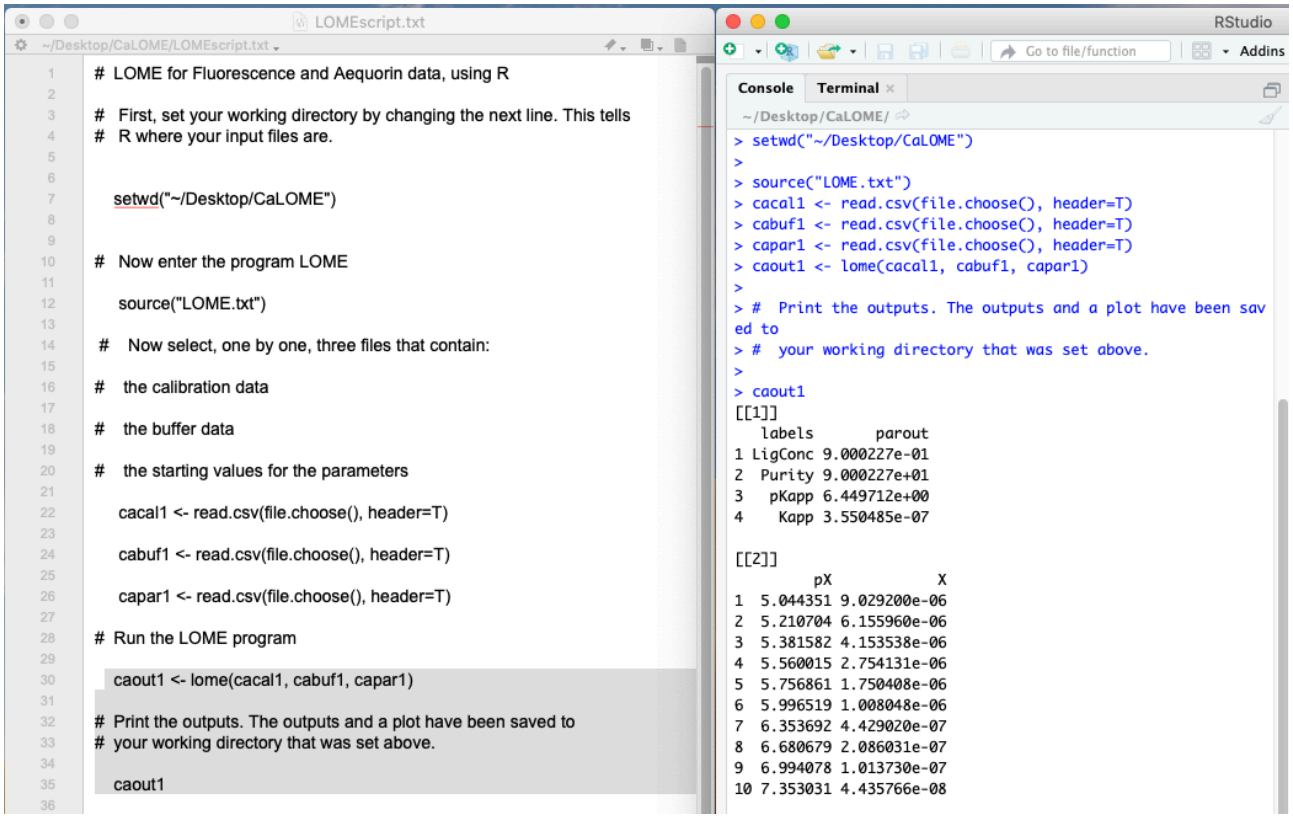
$$\text{cacal1} < - \text{read.csv(file.choose(), header=T)}$$

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 the file calA.csv. Then Click Open. The calibration data will then have been entered into the RStudio window.



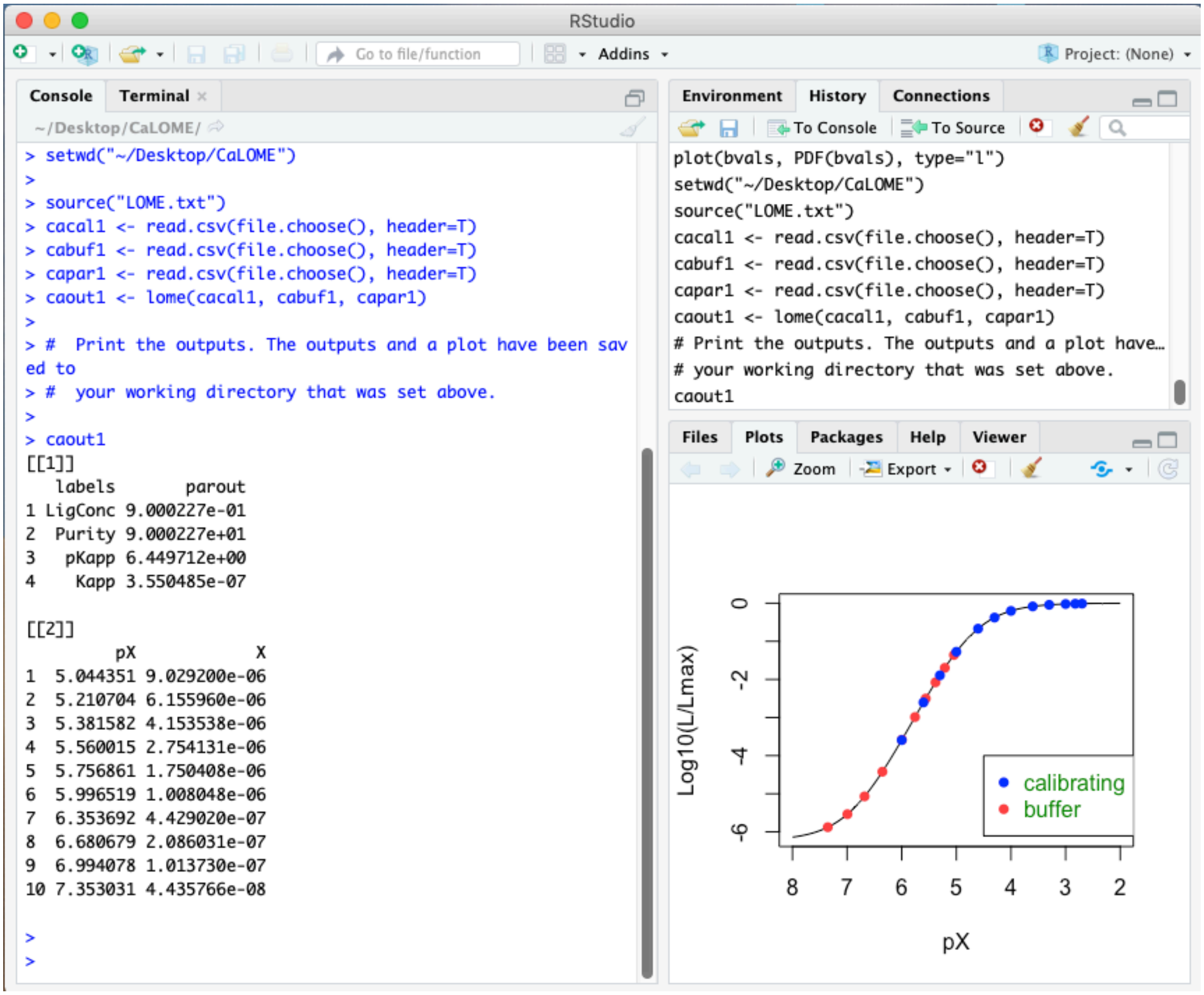
6. Repeat step 5 to enter the buffer data (bufA.csv) and the input parameters (parA.csv) into RStudio.

7. Now the scene has been set and we can run the LOME program. Select the lines containing caout1, as shown below, and copy and paste them into the RStudio window, and press ENTER. The output will appear in the RStudio window. Also a plot of the fitted model and data will appear in the plot window, as shown in the next plot below.



```
# LOME for Fluorescence and Aequorin data, using R

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


   setwd("~/Desktop/CaLOME")


#  Now enter the program LOME

    source("LOME.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 LOME program

   caout1 <- lome(cacal1, cabuf1, capar1)

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

   caout1
```

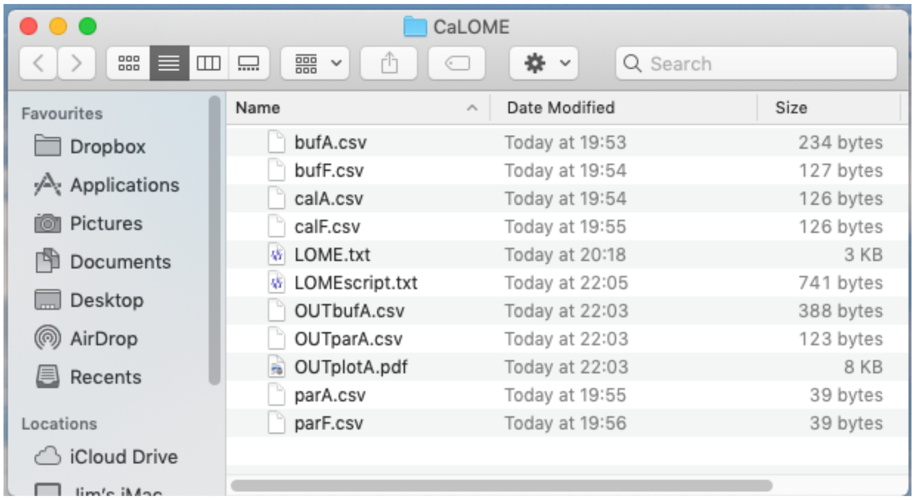
```
> setwd("~/Desktop/CaLOME")
>
> source("LOME.txt")
> cacal1 <- read.csv(file.choose(), header=T)
> cabuf1 <- read.csv(file.choose(), header=T)
> capar1 <- read.csv(file.choose(), header=T)
> caout1 <- lome(cacal1, cabuf1, capar1)
>
> #  Print the outputs. The outputs and a plot have been sav
ed to
> #  your working directory that was set above.
>
> caout1
[[1]]
   labels         parout
1 LigConc 9.000227e-01
2  Purity 9.000227e+01
3   pKapp 6.449712e+00
4    Kapp 3.550485e-07

[[2]]
         pX            X
1  5.044351 9.029200e-06
2  5.210704 6.155960e-06
3  5.381582 4.153538e-06
4  5.560015 2.754131e-06
5  5.756861 1.750408e-06
6  5.996519 1.008048e-06
7  6.353692 4.429020e-07
8  6.680679 2.086031e-07
9  6.994078 1.013730e-07
10 7.353031 4.435766e-08
```



```
> setwd("~/Desktop/CaLOME")
>
> source("LOME.txt")
> cacal1 <- read.csv(file.choose(), header=T)
> cabuf1 <- read.csv(file.choose(), header=T)
> capar1 <- read.csv(file.choose(), header=T)
> caout1 <- lome(cacal1, cabuf1, capar1)
>
> #  Print the outputs. The outputs and a plot have been sav
ed to
> #  your working directory that was set above.
>
> caout1
[[1]]
   labels       parout
1 LigConc 9.000227e-01
2  Purity 9.000227e+01
3   pKapp 6.449712e+00
4    Kapp 3.550485e-07

[[2]]
         pX            X
1  5.044351 9.029200e-06
2  5.210704 6.155960e-06
3  5.381582 4.153538e-06
4  5.560015 2.754131e-06
5  5.756861 1.750408e-06
6  5.996519 1.008048e-06
7  6.353692 4.429020e-07
8  6.680679 2.086031e-07
9  6.994078 1.013730e-07
10 7.353031 4.435766e-08

>
>
```

```
plot(bvals, PDF(bvals), type="l")
setwd("~/Desktop/CaLOME")
source("LOME.txt")
cacal1 <- read.csv(file.choose(), header=T)
cabuf1 <- read.csv(file.choose(), header=T)
capar1 <- read.csv(file.choose(), header=T)
caout1 <- lome(cacal1, cabuf1, capar1)
# Print the outputs. The outputs and a plot have…
# your working directory that was set above.
caout1
```

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



Name	Date Modified	Size
bufA.csv	Today at 19:53	234 bytes
bufF.csv	Today at 19:54	127 bytes
calA.csv	Today at 19:54	126 bytes
calF.csv	Today at 19:55	126 bytes
LOME.txt	Today at 20:18	3 KB
LOMEscript.txt	Today at 22:05	741 bytes
OUTbufA.csv	Today at 22:03	388 bytes
OUTparA.csv	Today at 22:03	123 bytes
OUTplotA.pdf	Today at 22:03	8 KB
parA.csv	Today at 19:55	39 bytes
parF.csv	Today at 19:56	39 bytes