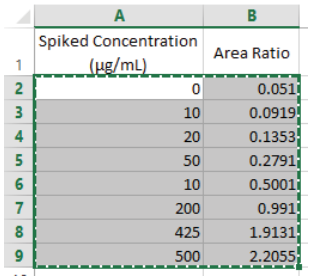
Supplementary Data 1

R Script for Automatic Correction of Endogenous Concentration

The functional version of the "Endogenous_Correction.R" script can be found here: `https://www.dropbox.com/sh/8agalof79qfwe81/AADYzwHQqD4z9Ix8qSsVYd1Sa?dl=0`.
Users can watch a video tutorial detailing its use in R Studio here: `https://youtu.be/coPCWYQ5DUM`.
To summarize, the user should copy to the clipboard a table containing the spiked concentration of calibration standards in the first column, and measurements (area, area ratio or response ratio) in the second column, without any header. For example, in Excel, the zone in grey below should be copied (Ctrl+C):



	A	B
1	Spiked Concentration (µg/mL)	Area Ratio
2	0	0.051
3	10	0.0919
4	20	0.1353
5	50	0.2791
6	10	0.5001
7	200	0.991
8	425	1.9131
9	500	2.2055

The script is then sourced in R Studio. At the end of the compilation procedure, the corrected concentrations will be sent to the clipboard; the user can paste these concentration in their data analysis software (Mass Hunter, MultiQuant, etc.).

```r
# Endogenous_Correction.R
# A script correcting the concentrations of calibrants and QCs for the
# endogenous analyte present in the spiking matrix.
# By Brigitte Desharnais, 2018-08-21.


#### Parameters to be set by the user ####

# Set the working directory.
setwd("~/R/Endogene_2018-09-21")

# Set the weighting to be used: none (A), 1/x (B), 1/x^2 (C)
WM <- "C"

# Set the calibration order to be used: linear (L) or quadratic (Q)
CO <- "Q"

##########################################################################


# Load the necessary packages
library(dplyr)
```

```
23
24 # Load data in the clipboard.
25 Data <- read.table(file="clipboard", sep="\t", header=FALSE)
26 colnames(Data) <- c("Concentration", "Measure")
27
28 # Sterilize data set to remove "missing" standards.
29 Data <- tbl_df(Data) %>% filter(Measure > 0)
30
31 # Sterilize data set to prevent NaN in weighted functions,
32 # i.e. replace concentrations of 0 by 1e-8.
33 if(WM!="A"){
34   for (k in 1:length(Data$Concentration)){
35     if (Data$Concentration[k] == 0){
36       Data$Concentration[k] <- 1E-8
37     }
38   }
39 }
40
41 # Load the function OptParam
42 # (Taken from CodeV5.R of the calibration project, written by FCL)
43 OptParam <- function(X,Y,poids,indice){
44   # Finds the calibration parameters for
45   # Weight (poids) 1 (=0) or 1/x (=1) or 1/(x^2) (=2).
46   # Model order (indice) linear (1) or quadratic (2).
47   W <- diag(1/abs(X)^poids)
48   xf <- matrix(rep(X,each = indice +1),ncol=indice+1,byrow=T)
49   ex <- matrix(rep(0:(indice),length(X)),ncol=indice +1 ,byrow=T)
50   Xp = xf^ex
51   matInv <- solve(t(Xp)%*%W%*%Xp)
52   param.optimaux <- rev(matInv%*%t(Xp)%*%W%*%Y)
53   return(param.optimaux)
54 }
55
56 # Calculation of the endogenous concentration with the linear calibration model.
57 if (CO == "L"){
58
59   # Calculations with no weighting applied.
60   if (WM == "A"){
61
62     # Perform the regression and extract the curve parameters.
63     Reg_Params <- OptParam(Data$Concentration, Data$Measure, 0, 1)
64     B0 <- Reg_Params[2]
65     B1 <- Reg_Params[1]
66
67     }
68
69   # Calculations with 1/x weighting applied.
70   if (WM == "B"){
71
72     # Perform the regression and extract the curve parameters.
73     Reg_Params <- OptParam(Data$Concentration, Data$Measure, 1, 1)
74     B0 <- Reg_Params[2]
75     B1 <- Reg_Params[1]
76
```

```
  }

  # Calculations with 1/(x^2) weighting applied.
  if (WM == "C"){

    # Perform the regression and extract the curve parameters.
    Reg_Params <- OptParam(Data$Concentration, Data$Measure, 2, 1)
    B0 <- Reg_Params[2]
    B1 <- Reg_Params[1]

  }

  # Calculate the endogenous concentration.
  XE <- B0/B1
}

# Calculation of the endogenous concentration with the quadratic calibration model.
if (CO == "Q"){
  if (WM == "A"){

    # Perform the regression and extract the curve parameters.
    Reg_Params <- OptParam(Data$Concentration, Data$Measure, 0, 2)
    B0 <- Reg_Params[3]
    B1 <- Reg_Params[2]
    B2 <- Reg_Params[1]

  }

  if (WM == "B"){

    # Perform the regression and extract the curve parameters.
    Reg_Params <- OptParam(Data$Concentration, Data$Measure, 1, 2)
    B0 <- Reg_Params[3]
    B1 <- Reg_Params[2]
    B2 <- Reg_Params[1]

  }

  if (WM == "C"){

    # Perform the regression and extract the curve parameters.
    Reg_Params <- OptParam(Data$Concentration, Data$Measure, 2, 2)
    B0 <- Reg_Params[3]
    B1 <- Reg_Params[2]
    B2 <- Reg_Params[1]

  }

  # Calculate the endogenous concentration.
  # Calculate the two possible solutions to a quadratic equation.
  XE_a <- ((-B1)-sqrt((B1^2)-(4*B2*B0)))/(2*B2)
  XE_b <- ((-B1)+sqrt((B1^2)-(4*B2*B0)))/(2*B2)

  # Keep as XE the negative/smallest absolute solution.
```

```
  if(abs(XE_a)<abs(XE_b)){
    XE <- abs(XE_a)
  }else{
    XE <- abs(XE_b)
  }

}

# Compute the vector of corrected concentrations.
Corr_Conc <- Data$Concentration + XE

# Print out the calculated endogenous concentration.
print("The endogenous concentration estimated from the data set is:")
print(XE)
print("Corrected concentrations are in the clipboard, ready to be pasted!")

# Output of the results in the clipboard.
writeClipboard(as.character(Corr_Conc))
```