Appendix A: the graphical user interface (GUI)

The GUI can either be accessed online or installed offline following instructions that can be found on `http://isoplotr.london-geochron.com`.

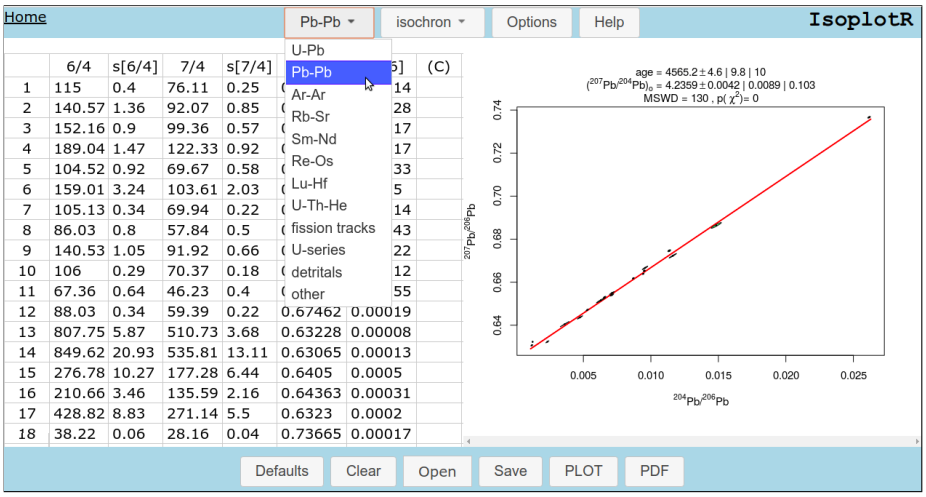

Figure A.1: The GUI has four components: a top bar with selection menus for the various chronometers and plot devices, optional settings and documentation; an input table into which data can be pasted from spreadsheet applications; an output window displaying the graphical or numerical results; and a lower bar to import and export data and results.

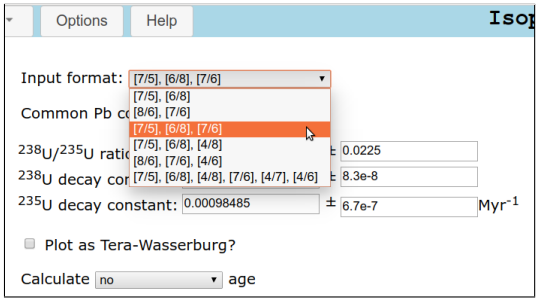

Figure A.2: Alternative input formats (shown here for the U-Pb method) are available through the `Options` menu.

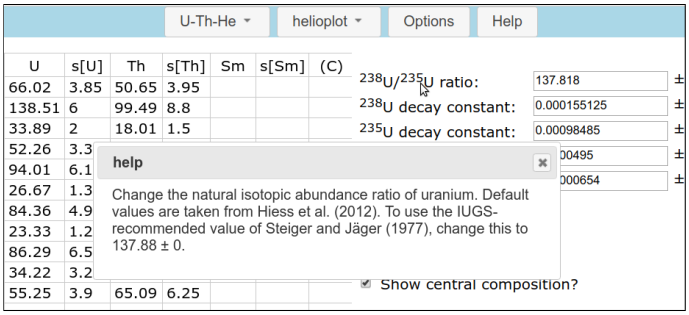

Figure A.3: Contextual help (shown here for U-Th-He data) can be accessed by clicking on any text within the `Options` menu.

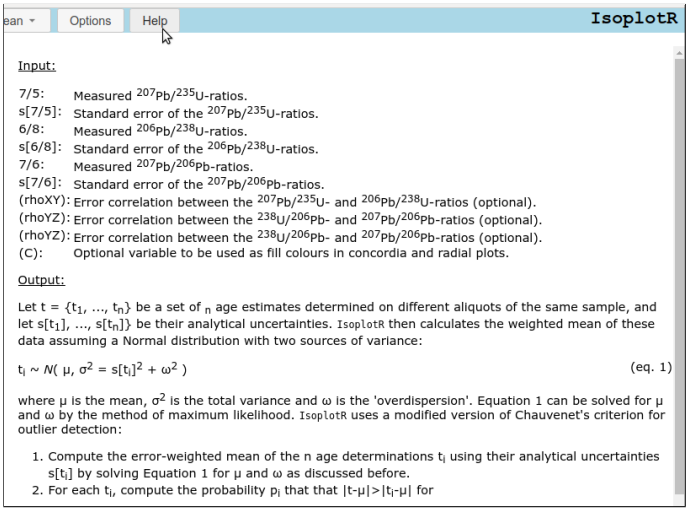

Figure A.4: Further documentation (shown here for the weighted mean of U-Pb data) is provided under the `Help` menu. This includes information about the input and output parameters, and a brief summary of the theoretical background with essential references.

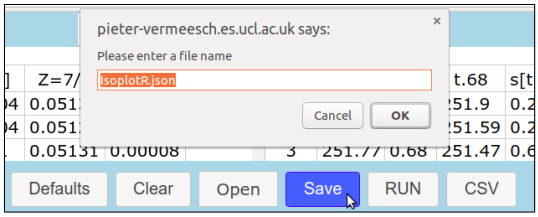

Figure A.5: Data and settings can be saved in a `.json` database format for future use.

Appendix B: command-line functionality

Running the following commands at the `R` command prompt reproduces all the figures in this paper. Everything that follows the hashtag ('#') is a comment and is ignored during execution:

```
# load the IsoplotR package:
library(IsoplotR)
# for this tutorial we will navigate to the system
# directory that stores the built-in data files:
setwd(system.file(package='IsoplotR'))
# Fig 1.a
RbSr <- read.data('RbSr1.csv',method='Rb-Sr',format=1)
isochron(RbSr)
# Fig 1.b
meandat <- read.data('LudwigMean.csv',method='other')
weightedmean(meandat)
# Fig 1.c
densdat <- read.data('LudwigKDE.csv',method='other')
cad(densdat)
# Fig 1.d
mixture <- read.data('LudwigMixture.csv',method='other')
kde(densdat,pch='|')
# Fig 1.e
radialplot(mixture,k='min',bg='yellow')
# Fig 1.f
ArAr <- read.data('ArAr3.csv',method='Ar-Ar',format=3)
agespectrum(ArAr)
# Fig 2.a
UPb <- read.data('UPb6.csv',method='U-Pb',format=6)
concordia(UPb,common.Pb=2,show.age=1,exterr=TRUE)
# Fig 2.b
ThU <- read.data('ThU1.csv',method='Th-U',format=1)
evolution(ThU,levels=ThU$x[,'U238Th232'],
          clabel=expression(paste(""^"238","U/"^"232","Th")))
# Fig 2.c
evolution(ThU,transform=TRUE,detrital=TRUE,
          ellipse.col=rgb(1,0,0,0.2),
          show.numbers=TRUE,isochron=TRUE)
# Fig 2.d
UThSmHe <- read.data('UThSmHe.csv',method='U-Th-He')
helioplot(UThSmHe,model=3,
          levels=log10(UThSmHe[,'Sm']),
          clabel=expression("log[Sm]"))
# Fig 2.e
helioplot(UThSmHe,model=1,logratio=FALSE,ellipse.col='lightblue')
# Fig 2.f
DZ <- read.data('DZ.csv',method='detritals')
mds(DZ)
```