MALDIquantForeign: Import/Export routines for MALDIquant

Sebastian Gibb[*]

April 11, 2014

Abstract

MALDIquantForeign provides routines for importing/exporting different file formats into/from MALDIquant.
This vignette describes the usage of the MALDIquantForeign package.

[*] mail@sebastiangibb.de

Contents

Foreword

`MALDIquantForeign` is free and open source software for the `R` (R Core Team, 2014) environment and under active development. If you use it, please support the project by citing it in publications:

> Gibb, S. and Strimmer, K. (2012). MALDIquant: a versatile R package for the analysis of mass spectrometry data. *Bioinformatics*, 28(17):2270–2271

If you have any questions, bugs, or suggestions do not hesitate to contact me (`mail@sebastiangibb.de`).
Please visit `http://strimmerlab.org/software/maldiquant/`.

1 Introduction

`MALDIquant` should be device and platform independent. That's why it has not any import/export functions.
`MALDIquantForeign` fills this gap and provides import/export routines for various file formats:

```
> supportedFileFormats()
```

```
$import
 [1] "txt"       "tab"       "csv"       "fid"       "ciphergen"
 [6] "mzxml"     "mzml"      "imzml"     "analyze"   "cdf"

$export
[1] "tab"  "csv"  "msd"  "mzml"
```

2 Setup

After starting R we could install MALDIquant and MALDIquantForeign directly from CRAN using install.packages:

```
> install.packages(c("MALDIquant", "MALDIquantForeign"))
```

Before we can use MALDIquant and MALDIquantForeign we have to load the packages.

```
> library("MALDIquant")
> library("MALDIquantForeign")
```

3 Import

MALDIquantForeign provides an import function that tries to auto-detect the correct file type. Because this would never be perfect MALDIquantForeign offers also many import* functions like importBrukerFlex, importMzMl, etc. Please see the manual page of import for a complete list (?import).

First we try to import some example data in Bruker Daltonics *flex-series file format using the import function.

```
> ## get the example directory
> exampleDirectory <- system.file("exampledata",
+                                 package="MALDIquantForeign")
>
```

```
> spectra <- import(file.path(exampleDirectory,
+                                  "brukerflex"),
+                    verbose=FALSE)
> spectra[[1]]

S4 class type              : MassSpectrum
Number of m/z values       : 5
Range of m/z values        : 226.762 - 230.51
Range of intensity values: 1e+00 - 5e+00
Memory usage               : 7.773 KiB
Name                       : brukerflex.
File                       : /tmp/RtmpP1PzXd/Rinst538d7419a7c3/MALDIquantForeign/exa
```

Next we use the importBrukerFlex function (the result is the same as above).

```
> spectra <- importBrukerFlex(file.path(exampleDirectory,
+                                            "brukerflex"),
+                              verbose=FALSE)
> spectra[[1]]

S4 class type              : MassSpectrum
Number of m/z values       : 5
Range of m/z values        : 226.762 - 230.51
Range of intensity values: 1e+00 - 5e+00
Memory usage               : 7.773 KiB
Name                       : brukerflex.
File                       : /tmp/RtmpP1PzXd/Rinst538d7419a7c3/MALDIquantForeign/exa
```

MALDIquantForeign supports compressed files, too (*zip*, *tar*.{*bz2*, *gz*,*xz*}).

```
> spectra <- importCsv(file.path(exampleDirectory, "compressed",
+                                  "csv.tar.gz"), verbose=FALSE)
> spectra[[1]]

S4 class type              : MassSpectrum
Number of m/z values       : 5
```

```
Range of m/z values      : 1 - 5
Range of intensity values: 6 - 10
Memory usage             : 1.406 KiB
File                     : /tmp/RtmpZPb3Iq/MALDIquantForeign_uncompress/csv_53a039

>
> spectra <- importCsv(file.path(exampleDirectory, "compressed",
+                                "csv.zip"), verbose=FALSE)
> spectra[[1]]

S4 class type            : MassSpectrum
Number of m/z values     : 5
Range of m/z values      : 1 - 5
Range of intensity values: 6 - 10
Memory usage             : 1.406 KiB
File                     : /tmp/RtmpZPb3Iq/MALDIquantForeign_uncompress/csv_53a06c
```

Remote files are supported as well.

```
> githubUrl <- paste0("https://raw.github.com/sgibb/",
+                     "MALDIquantForeign/master/",
+                     "inst/exampledata/tiny1.mzML1.1.mzML")
> spectra <- import(githubUrl, verbose=FALSE)
> spectra[[1]]

S4 class type            : MassSpectrum
Number of m/z values     : 5
Range of m/z values      : 1 - 5
Range of intensity values: 6e+00 - 1e+01
Memory usage             : 2.008 KiB
File                     : /tmp/RtmpZPb3Iq/MALDIquantForeign_download/tiny1.mzML1.

>
> spectra <- importMzMl(githubUrl, verbose=FALSE)
> spectra[[1]]

S4 class type            : MassSpectrum
Number of m/z values     : 5
```

```
Range of m/z values      : 1 - 5
Range of intensity values: 6e+00 - 1e+01
Memory usage             : 2.008 KiB
File                     : /tmp/RtmpZPb3Iq/MALDIquantForeign_download/tiny1.mzML1
```

4 Export

The export routines in MALDIquantForeign are very similar to the import routines. Please see manual page of export for a complete list of supported export routines (?export).

First we create a simple list of MassSpectrum objects using createMassSpectrum.

```
> spectra <- list(
+   createMassSpectrum(mass=1:5, intensity=1:5),
+   createMassSpectrum(mass=1:5, intensity=6:10))
```

Now we want to export the first spectrum into a CSV file.

```
> export(spectra[[1]], file="spectrum1.csv")
```

Exporting every file by hand is cumbersome. We want to export the whole list of spectra. Instead of file we use path now to specify a directory. Please note that we have to add the file type/format information now (we can use the type argument or the corresponding export* function).

```
> ## force=TRUE creates the path if it does not exist
> ## (or overwrites it)
> export(spectra, type="csv", path="spectra", force=TRUE)
> list.files("spectra")

[1] "1.csv" "2.csv"
```

5 Analyse Mass Spectrometry Data

Please have a look at the corresponding vignette shipped with `MALDIquant` and the `MALDIquant` website: `http://strimmerlab.org/software/maldiquant/`.

```
> vignette(topic="MALDIquant", package="MALDIquant")
```

6 Session Information

- R version 3.0.3 (2014-03-06), `x86_64-pc-linux-gnu`
- Base packages: base, datasets, grDevices, graphics, methods, stats, utils
- Other packages: MALDIquant~1.10, MALDIquantForeign~0.7, knitr~1.5
- Loaded via a namespace (and not attached): XML~3.98-1.1, base64enc~0.1-1, digest~0.6.4, downloader~0.3, evaluate~0.5.3, formatR~0.10, highr~0.3, readBrukerFlexData~1.7, readMzXmlData~2.7, stringr~0.6.2, tools~3.0.3

References

Gibb, S. and Strimmer, K. (2012). MALDIquant: a versatile R package for the analysis of mass spectrometry data. *Bioinformatics*, 28(17):2270–2271.

R Core Team (2014). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.