Package: QFeaturesGUI (via r-universe)

November 25, 2026

Title A suite of shiny apps to use the main functionalities of the QFeatures package

Version 1.0.0

Description QFeaturesGUI is a suite of shiny apps that serve as graphical interfaces for the QFeatures package. The package provides three apps: import, process, and visualise.

biocViews Software, ShinyApps, GUI, Proteomics, SingleCell, DataImport, Preprocessing

URL https://rformassspectrometry.github.io/QFeaturesGUI/, https://github.com/rformassspectrometry/QFeaturesGUI

BugReports https://github.com/rformassspectrometry/QFeaturesGUI/issues

License GPL (>= 3)

Encoding UTF-8

LazyData false

Imports QFeatures, shiny, DT, shinydashboard, shinydashboardPlus, shinyalert, SummarizedExperiment, utils, htmltools, impute, plotly, MultiAssayExperiment, methods, shinyFeedback, stats, SingleCellExperiment, ggplot2, tidyr, shinyjs, rmarkdown, tibble, dplyr, matrixStats, MsCoreUtils, waiter, nipals

Depends R (>= 4.5.0)

Suggests knitr, BiocStyle, testthat (>= 3.0.0), shinytest2

Roxygen list(markdown = TRUE)

VignetteBuilder knitr

Config/testthat/edition 3

Config/roxygen2/version 8.1.0

Config/pak/sysreqs cmake libglpk-dev make libicu-dev libuv1-dev libxml2-dev libssl-dev zlib1g-dev

Repository Bioconductor 3.24

Date/Publication 2026-11-23 22:56:55 UTC

RemoteUrl https://github.com/bioc/QFeaturesGUI

RemoteRef RELEASE_3_24

RemoteSha a6096134be2c58e51b34c4c435a4c8414eacd544

Contents

import 2
inputTable 3
process 7
sampleTable 8
visualise 9

Index **11**

`import` *A shiny app to import QFeatures objects.*

Description

import is a simple graphical interface to import bulk and single-cell proteomics data. The app uses the `readQFeatures` function from the QFeatures package to convert simple tables (single or multiple, CSV or TSV) to a QFeatures object. The app allows users to convert tables to a QFeatures object.

Usage

```
import(colData = NULL, assayData = NULL, maxSize = 1000)
```

Arguments

`colData`	A data frame that contains the sample table.
`assayData`	A data frame that contains the input table.
`maxSize`	An integer that changes the shiny.maxRequestSize value, in MB.

Value

The "import" Shiny app object.

Examples

```
library(QFeaturesGUI)

data("sampleTable")
data("inputTable")
app <- import(colData = sampleTable, assayData = inputTable, maxSize = 100)

if (interactive()) {
    shiny::runApp(app)
}
```

`inputTable` *Example MaxQuant/SCoPE2 output*

Description

A `data.frame` with 1088 observations and 139 variables, as produced by reading a MaxQuant output file with `read.delim()`.

- Sequence: a character vector
- Length: a numeric vector
- Modifications: a character vector
- Modified.sequence: a character vector
- Deamidation..N..Probabilities: a character vector
- Oxidation..M..Probabilities: a character vector
- Deamidation..N..Score.Diffs: a character vector
- Oxidation..M..Score.Diffs: a character vector
- Acetyl..Protein.N.term.: a numeric vector
- Deamidation..N.: a numeric vector
- Oxidation..M.: a numeric vector
- Missed.cleavages: a numeric vector
- Proteins: a character vector
- Leading.proteins: a character vector
- protein: a character vector
- Gene.names: a character vector
- Protein.names: a character vector
- Type: a character vector
- Set: a character vector
- MS.MS.m.z: a numeric vector
- Charge: a numeric vector
- m.z: a numeric vector
- Mass: a numeric vector
- Resolution: a numeric vector
- Uncalibrated...Calibrated.m.z..ppm.: a numeric vector
- Uncalibrated...Calibrated.m.z..Da.: a numeric vector
- Mass.error..ppm.: a numeric vector
- Mass.error..Da.: a numeric vector
- Uncalibrated.mass.error..ppm.: a numeric vector
- Uncalibrated.mass.error..Da.: a numeric vector

- Max.intensity.m.z.0: a numeric vector
- Retention.time: a numeric vector
- Retention.length: a numeric vector
- Calibrated.retention.time: a numeric vector
- Calibrated.retention.time.start: a numeric vector
- Calibrated.retention.time.finish: a numeric vector
- Retention.time.calibration: a numeric vector
- Match.time.difference: a logical vector
- Match.m.z.difference: a logical vector
- Match.q.value: a logical vector
- Match.score: a logical vector
- Number.of.data.points: a numeric vector
- Number.of.scans: a numeric vector
- Number.of.isotopic.peaks: a numeric vector
- PIF: a numeric vector
- Fraction.of.total.spectrum: a numeric vector
- Base.peak.fraction: a numeric vector
- PEP: a numeric vector
- MS.MS.count: a numeric vector
- MS.MS.scan.number: a numeric vector
- Score: a numeric vector
- Delta.score: a numeric vector
- Combinatorics: a numeric vector
- Intensity: a numeric vector
- Reporter.intensity.corrected.0: a numeric vector
- Reporter.intensity.corrected.1: a numeric vector
- Reporter.intensity.corrected.2: a numeric vector
- Reporter.intensity.corrected.3: a numeric vector
- Reporter.intensity.corrected.4: a numeric vector
- Reporter.intensity.corrected.5: a numeric vector
- Reporter.intensity.corrected.6: a numeric vector
- Reporter.intensity.corrected.7: a numeric vector
- Reporter.intensity.corrected.8: a numeric vector
- Reporter.intensity.corrected.9: a numeric vector
- Reporter.intensity.corrected.10: a numeric vector
- RI1: a numeric vector
- RI2: a numeric vector

- RI3: a numeric vector
- RI4: a numeric vector
- RI5: a numeric vector
- RI6: a numeric vector
- RI7: a numeric vector
- RI8: a numeric vector
- RI9: a numeric vector
- RI10: a numeric vector
- RI11: a numeric vector
- Reporter.intensity.count.0: a numeric vector
- Reporter.intensity.count.1: a numeric vector
- Reporter.intensity.count.2: a numeric vector
- Reporter.intensity.count.3: a numeric vector
- Reporter.intensity.count.4: a numeric vector
- Reporter.intensity.count.5: a numeric vector
- Reporter.intensity.count.6: a numeric vector
- Reporter.intensity.count.7: a numeric vector
- Reporter.intensity.count.8: a numeric vector
- Reporter.intensity.count.9: a numeric vector
- Reporter.intensity.count.10: a numeric vector
- Reporter.PIF: a logical vector
- Reporter.fraction: a logical vector
- Reverse: a character vector
- Potential.contaminant: a logical vector
- id: a numeric vector
- Protein.group.IDs: a character vector
- Peptide.ID: a numeric vector
- Mod..peptide.ID: a numeric vector
- MS.MS.IDs: a character vector
- Best.MS.MS: a numeric vector
- AIF.MS.MS.IDs: a logical vector
- Deamidation..N..site.IDs: a numeric vector
- Oxidation..M..site.IDs: a logical vector
- remove: a logical vector
- dart_PEP: a numeric vector
- dart_qval: a numeric vector
- razor_protein_fdr: a numeric vector

- Deamidation..NQ..Probabilities: a logical vector
- Deamidation..NQ..Score.Diffs: a logical vector
- Deamidation..NQ.: a logical vector
- Reporter.intensity.corrected.11: a logical vector
- Reporter.intensity.corrected.12: a logical vector
- Reporter.intensity.corrected.13: a logical vector
- Reporter.intensity.corrected.14: a logical vector
- Reporter.intensity.corrected.15: a logical vector
- Reporter.intensity.corrected.16: a logical vector
- RI12: a logical vector
- RI13: a logical vector
- RI14: a logical vector
- RI15: a logical vector
- RI16: a logical vector
- Reporter.intensity.count.11: a logical vector
- Reporter.intensity.count.12: a logical vector
- Reporter.intensity.count.13: a logical vector
- Reporter.intensity.count.14: a logical vector
- Reporter.intensity.count.15: a logical vector
- Reporter.intensity.count.16: a logical vector
- Deamidation..NQ..site.IDs: a logical vector
- input_id: a logical vector
- rt_minus: a logical vector
- rt_plus: a logical vector
- mu: a logical vector
- muij: a logical vector
- sigmaij: a logical vector
- pep_new: a logical vector
- exp_id: a logical vector
- peptide_id: a logical vector
- stan_peptide_id: a logical vector
- exclude: a logical vector
- residual: a logical vector
- participated: a logical vector
- peptide: a character vector

Usage

```
data("inputTable")
```

Format

An object of class `data.frame` with 1361 rows and 149 columns.

Details

The dataset is a subset of the SCoPE2 dataset (version 2, Specht et al. 2019, BioRXiv). The input file `evidence_unfiltered.csv` was downloaded from a Google Drive repository. The MaxQuant evidence file was loaded and the data was cleaned (renaming columns, removing duplicate fields,...). MS runs that were selected in the `scp1` dataset (see `?scp1`) were kept along with a blank run. The data is stored as a `data.frame`.

See Also

`readSCP()` from the `scp` package for an example on how `mqScpData` is parsed into a QFeatures object.

---

process *Launch a Shiny application to process QFeatures objects*

Description

`process()` launches an interactive Shiny application that allows users to visually configure and apply pre-processing workflows to a QFeatures object.

The input `qfeatures` can be provided as an in-memory QFeatures object, as a path to an `.rds` file containing one, or omitted. If omitted, the application prompts the user to upload a QFeatures object from an `.rds` file or use the bundled demo dataset.

Usage

```
process(
  qfeatures = NULL,
  initialSets = NULL,
 prefilledSteps = c("sampleFiltering", "featureFiltering", "missingValuesFeatures",
  "missingValuesSamples", "normalisation", "aggregation", "join", "aggregation"),
  maxSize = 100
)
```

Arguments

`qfeatures`	Optional QFeatures object to be processed, or a character string specifying the path to a `.rds` file containing a QFeatures object. If omitted, the app starts without processing steps and displays a startup modal.
`initialSets`	An integer, logical, or character vector specifying which assays (feature sets) should be used as the starting point for processing. If `NULL` and `qfeatures` is provided, all assays in `qfeatures` are used. If `qfeatures` is omitted, the user chooses the initial sets after uploading the `.rds` file.

`prefilledSteps` A character vector specifying the initial workflow steps to display when the application launches. Available steps are:

- `"sampleFiltering"`
- `"featureFiltering"`
- `"normalisation"`
- `"zeroToNA"`
- `"logTransform"`
- `"imputation"`
- `"missingValuesFeatures"`
- `"missingValuesSamples"`
- `"aggregation"`
- `"join"`

`maxSize` An integer that changes the `shiny.maxRequestSize` value, in MB. This controls the maximum upload size for the startup `.rds` file upload modal.

Details

The application provides a drag-and-drop workflow builder that allows users to select, order, and configure processing steps such as filtering, normalization, and transformation. The configured workflow can then be applied to the selected assays.

Value

The process Shiny application.

Examples

```
library(QFeaturesGUI)


app <- process()

if (interactive()) {
    shiny::runApp(app)
}
```

---

sampleTable *Single-cell sample annotation*

Description

A data frame with 48 observations on the following 6 variables.

- Set: a character vector
- Channel: a character vector
- SampleType: a character vector

- lcbatch: a character vector
- sortday: a character vector
- digest: a character vector

Usage

```
data("sampleTable")
```

Format

An object of class `data.frame` with 64 rows and 6 columns.

Details

The dataset is a subset of the SCoPE2 dataset (version 2, Specht et al. 2019, BioRXiv). The input files `batch.csv` and `annotation.csv` were downloaded from a Google Drive repository. The two files were loaded and the column names were adapted for consistency with `mqScpData` table (see `?mqScpData`). The two tables were filtered to contain only sets present in `mqScpData`. The tables were then merged based on the run ID, hence merging the sample annotation and the batch annotation. Finally, annotation for the blank run was added manually. The data is stored as a `data.frame`.

See Also

`readSCP()` from the `scp` package to see how this file is used.

`visualise` *Launch a Shiny application to visualise QFeatures objects*

Description

`visualise()` launches an interactive Shiny application that allows users to visualise a QFeatures object.

The input `qfeatures` can be provided as an in-memory QFeatures object, as a path to an `.rds` file containing one, or omitted. If omitted, the application prompts the user to upload a QFeatures object from an `.rds` file or use the bundled demo dataset.

Usage

```
visualise(qfeatures = NULL, maxSize = 100)
```

Arguments

`qfeatures`	Optional QFeatures object to visualise, or a character string specifying the path to an `.rds` file containing one. If omitted or `NULL`, the app displays a startup modal for uploading a file or loading the bundled demo.
`maxSize`	An integer that changes the `shiny.maxRequestSize` value, in MB. This controls the maximum upload size for the startup `.rds` file upload modal.

Value

The visualise Shiny application.

Examples

```
library(QFeaturesGUI)


app <- visualise()

if (interactive()) {
    shiny::runApp(app)
}
```

Index

∗ **datasets**
    inputTable, 3
    sampleTable, 8

import, 2
inputTable, 3

process, 7

QFeatures, *7*, *9*

read.delim(), *3*
readQFeatures, *2*

sampleTable, 8

visualise, 9