

Profiler

Data Import Guide

How to prepare & upload your omics datasets

V1.2: extended format support



1. Overview

Profiler accepts tabular omics datasets in multiple formats: CSV, TSV, TXT, Excel (XLS/XLSX). The sidebar loader automatically detects separators, encodings, and standard column naming conventions. It now also supports specialised software exports from proteomics (MaxQuant, DIA-NN, Spectronaut, FragPipe, Proteome Discoverer, Progenesis QI, PEAKS Studio, Perseus), transcriptomics (DESeq2/edgeR counts, Salmon, kallisto, featureCounts, STAR, HTSeq), and metabolomics (MetaboAnalyst, XCMS, MZmine).

An **Auto-detect** engine inspects the file header and selects the correct parser automatically.

2. Column Naming Conventions

2.1 Target / Class Column

Profiler recognises any of the following column names as the classification or regression target. You do not need to rename your column to *Class*:

Column name	Note
Class	Standard Profiler name
class	Lowercase variant
Target	Common in ML datasets
target	Lowercase variant
Condition	Common in proteomics/genomics
condition	Lowercase variant
Label	Generic label column
Group	Group/cohort identifier
Status	Event status (disease/healthy)
Outcome	Clinical outcome

2.2 Sample ID Column

An ID column labels samples in PCA/UMAP hover tooltips, heatmaps, and enrichment tables. If absent, Profiler creates one automatically (Sample_1, Sample_2,...).

Column name	Note
ID	Standard identifier
id	Lowercase variant
SampleID	Combined form
Sample_ID	Underscore form
sample_id	Lowercase
SampleName	Text name
Sample_Name	Underscore form
Name	Generic name column
Patient	Clinical datasets
Subject	Clinical datasets

2.3 Clinical Metadata Columns (_meta suffix)

Any column whose name ends with **_meta** is treated as clinical metadata. These columns are available as alternative classification/regression targets and can be used to colour heatmap annotations or PCA/UMAP points. They are *excluded* from the feature matrix unless explicitly selected as target.

Column name	Typical use
treatment_meta	Drug arm / treatment group
age_meta	Patient age (numeric or categorical)
gender_meta	Biological sex
stage_meta	Disease stage (I, II, III, IV)
survival_meta	Overall survival (months)
response_meta	Response to therapy (Yes/No/Partial)
batch_meta	Batch / run identifier for QC
pfs_meta	Progression-free survival

3. Supported File Formats

3.1 Generic Tabular Formats

Extension	Format	Notes
.csv	Comma-Separated Values	Auto-detected: , ; \t
.tsv	Tab-Separated Values	Tab detected automatically
.txt	Plain text table	Any common delimiter
.xlsx/.xls	Excel Workbook	First sheet loaded, openpyxl engine

3.2 Proteomics— Protein Level

Software	Expected file	Feature source	Auto-detected by
MaxQuant	proteinGroups.txt	Gene names / Protein na	LFQ intensity cols
DIA-NN	pg_matrix.tsv	Genes / Protein.Names	Protein.Group col
Spectronaut	ProteinReport.tsv	PG.Genes / PG.ProteinG	PG* prefix
FragPipe / MSFragger	combined_protein.tsv	Gene column	MaxLFQ Intensity cols
Proteome Discoverer	Proteins.txt	Gene Symbol	Abundance: F* cols
Progenesis Q1	normalised_proteins.csv	Gene / Accession	Accession + normalised
PEAKS Studio	proteins.csv	Gene / Protein ID	Area: cols
Perseus	matrix.txt	T: Gene names row	T:/N:/C: prefix

3.3 Proteomics— Peptide Level

Software	Expected file	Feature source
MaxQuant	peptides.txt	Sequence / Modified sequence
DIA-NN	precursors.tsv	Stripped.Sequence / Precursor.Id
Spectronaut	PeptideReport.tsv	PEP.StrippedSequence / EG.ModifiedSequ

3.4 Transcriptomics— RNA-seq

Tool / Pipeline	Expected file	Feature source	Auto-detected by
DESeq2 / edgeR	counts_matrix.csv	gene_name / gene_id	gene_id/gene_name col
Salmon	quant.sf	Name (transcript/gene)	Name + TPM cols
kallisto	abundance.tsv	target_id	target_id + tpm cols
featureCounts	counts.txt	Geneid	Geneid + Chr cols
STAR	ReadsPerGene.out.tab	gene_id (Ensembl)	4-col + ENS* pattern
HTSeq-count	htseq_counts.txt	gene_id	2-col + __summary rows

3.5 Metabolomics

Software	Expected file	Feature source	Notes
MetaboAnalyst	data_table.csv	Metabolite columns	Sample-major (rows=sample)
XCMS	feature_table.csv	mz_rt feature identifiers	row m/z + row retention time
MZmine	feature_table.csv	row ID or mz_rt labels	Same as XCMS format

4. Auto-detect Engine

When you upload a file, Profiler reads the first 5 rows and checks column signatures against all registered parsers. The detected format is shown in the sidebar as "**Detected format:...**". If the detection is correct, click **Auto-detect** to parse immediately. You can always override by selecting a specific format from the dropdown.

Format label	Signature columns used for detection
MaxQuant proteins	Any column starting with LFQ intensity
MaxQuant peptides	Intensity * columns + Sequence column
DIA-NN proteins	Protein.Group or Protein.Ids column
DIA-NN peptides	Precursor.Id or Stripped.Sequence column
Spectronaut proteins	Any column with PG. prefix
Spectronaut peptides	Any column with PEP. or EG. prefix
FragPipe / MSFragger	Gene column + * MaxLFQ Intensity columns
Proteome Discoverer	Accession column + Abundance: F* columns
Salmon	Name column + TPM column
kallisto	target_id column + tpm column
featureCounts	Geneid column + Chr column
XCMS / MZmine	row m/z or mzmed column

5. Generic Format Examples

5.1 Minimal example (classification)

A file with only a Class column and numeric features:

```
ID, Class, Protein_A, Protein_B, Protein_C S01, Cancer, 1257.3, 0.45, 8892.1
S02, Healthy, 752.8, 1.30, 4431.0 S03, Cancer, 2103.5, 0.21, 9012.4
```

5.2 With clinical metadata (_meta columns)

```
ID, Class, Protein_A, Protein_B, treatment_meta, age_meta
S01, Responder, 1257.3, 0.45, drug_X, 58
S02, Non-responder, 752.8, 1.30, placebo, 62
```

In this example, *treatment_meta* and *age_meta* are available in heatmap annotations and can be selected as target columns for subgroup analysis.

5.3 Regression target (numeric Class)

```
ID, Class, Gene_1, Gene_2 P01, 2.4, 1890.1, 554.3 P02, 5.7, 3041.5, 812.0
```

Numeric values in the Class column trigger regression mode automatically.

6. How Sample ID Is Used Across Profiler

Module	How ID is used
PCA / UMAP / t-SNE	Hover tooltip shows ID + Class for each point
Heatmap	Row labels use ID instead of file path or row index
Enrichment gene table	Gene-pathway-class table references sample IDs
QC missing values heatmap	Y-axis labelled with sample ID
Real-time classification	Prediction results include the ID column
Survival analysis	Kaplan-Meier curves annotated with group IDs

7. Delimiter & Encoding Detection

Profiler counts occurrences of each candidate separator (, ; **tab** |) on the header line and picks the most frequent one. The following encodings are tried in order:

```
UTF-8-sig → UTF-8 → Latin-1 → ISO-8859-1 → CP1252
```

Column names are automatically stripped of invisible characters and stray quotation marks. All file content is read into memory as bytes first, so the file pointer is always reset correctly— even for multi-pass parsing.

8. Tips & Common Mistakes

✓ If your CSV uses semicolons (European Excel), just upload it— the delimiter is auto-detected. ✓ Numeric class columns (e.g. age, dose, survival) automatically activate regression mode.

✓ You can use `_meta` columns to stratify QC plots even if they are not the primary target.

✓ Keep feature column names short and free of special characters for best display in plots. ✓ For RNA-seq, upload the raw or normalised count matrix— not DESeq2 results tables.

✓ Salmon / kallisto files are single-sample by design. Upload them one at a time.

■ Columns named after numeric m/z values (e.g. 152.3, 873.1) must be numeric, not strings. ■ Do not include formula cells in Excel files— convert to values before uploading.

■ Avoid two columns with the same name; one will be dropped during loading.

■ Files with >50,000 features may be slow; consider binning m/z values beforehand.

■ DESeq2 results tables (log2FoldChange, padj) are NOT count matrices— use the raw counts. ■ Missing values should be blank or NaN, not filled with 0 unless they are truly zero.

9. Quick-Start Checklist

■ My file is CSV, TSV, TXT, XLS or XLSX

■ I have at least one column named Class / Target / Condition (or

similar) ■ Each row is one sample; each column (except metadata) is one

feature ■ Numeric features contain numbers— not text like 'Not detected'

■ I have an ID column (or I am happy for Profiler to create Sample_1,

Sample_2...) ■ Clinical variables end in `_meta` if I want them in Profiler metadata

features

■ Missing values are blank or NaN (not filled with 0 unless they are truly zero)

■ For specialised formats: I will use Auto-detect or select the correct parser from the dropdown

Profiler, Bioinformatics Analysis Platform | For questions, see the in-app help tooltips.

Contact : yanis.zirem@univ-lille.fr https://github.com/yanisZirem/Profiler_v1_requests_datatests/issues