Datasets Settings Explained

This document describes the practical effect of each option in the **datasets** section of the integration workflow configuration file (`/input_data/config/project_config.yml`), including the data processing tag and the four normalization steps used in omics data processing (filtering, devariancing, scaling, and replicability). Each section lists available methods, their options, and default values, and at the end there is an example of a full data processing workflow.

Setup Parameters

1. Tagging

- **data_processing_tag**

  Allows user to create a new data processing folder (`Data_Processing--`**`TAG`**) to store results. This is useful if data processing and normalization settings are changed (see below) and a new set of outputs should be produced that is separate from previous runs. Default: "0".

  Note: If a data processing folder already exists with the supplied tag and overwriting is disabled, the workflow will return an error message indicating that you should change the tag

- **dataset_dir**

  This is currently fixed based on the workflow structure, so do not change it.

Normalization Parameters

1. Filtering

Options:

- **method**
  - *minimum* [default]

    Removes features whose average observed value across samples is below the specified threshold. Useful for excluding low-abundance or low-count features that may be noise or low confidence observations.
  - *proportion*

    Removes features that are observed at higher abundance than the detection limit in fewer than a specified percentage of samples. Helps filter out features that are rarely detected.
  - *none*

    No filtering is performed; all features are retained regardless of abundance or prevalence.
- **value**
  - Value determines the minimum or proportion value for filtering. It represents either observed quantitative values from the raw data (*minimum*; real numbers greater than 0) or percentage of samples (*proportion*; real numbers 0-100).

2. Devariancing

Options:

- **method**

    - *percent* [default]

      Removes a specified percentage of features with the lowest variance across samples. This keeps only the most variable features, which are more likely to be informative.

    - *none*

      No variance-based filtering is performed; all features are retained regardless of their variance.

- **value**
    - Value determines percent value for removing low variance features (real numbers 0-100). It represents percent of total features.

3. Scaling

Options:

- **log2** [default]

  If enabled, applies a log2(x+1) transformation to all values before scaling. This reduces skewness and compresses large values.

- **modified_zscore** [default]

  Standardizes each feature using the median and median absolute deviation, making it more robust to outliers than standard z-score and makes features comparable regardless of their original scale.

- **zscore**

  Standardizes each feature to have mean zero and unit variance across samples. This makes features comparable regardless of their original scale.

- **none**

  No scaling is performed; features retain their original values. Not recommended because datasets will not fall along the same distribution and integration will be difficult to interpret.

4. Replicate Handling

Options:

- **method**
    - *variance* [default]

      Removes features with high variability among replicates within each group (or specified metadata category - see below). Only features with consistent observed values within a replicate group are retained.

    - *none*

      No replicate handling is performed; all features are retained.

- **group**

  If method is *variance*, **group** is any column from the dataset metadata (i.e., a variable in the `user_settings->variable_list` in the configuration file) for grouping samples as replicates - defaults to the meta-variable 'group', which is the combination of all listed metadata categories.

- **value**

  If method is *variance,* this sets the threshold for maximum allowable within-group variability. Features with variability above this threshold are removed (default: 0.5). *Note*: this step typically is performed after data scaling, so mean and variance will be standardized.

Example

Suppose you start with the following transcriptomics dataset (features as rows, samples as columns), with two sample groupings (high or low).

Feature	**High_1**	**High_2**	**High_3**	**High_4**	**High_5**	**Low_1**	**Low_2**	**Low_3**	**Low_4**	**Low_5**
FeatureA	900	850	920	870	910	20	25	22	18	24
FeatureB	5	8	7	6	9	4	3	5	6	4
FeatureC	100	120	110	130	115	90	95	85	100	92
FeatureD	500	520	510	530	515	480	490	470	495	485
FeatureE	10	10	10	10	10	10	10	10	10	10
FeatureF	700	750	720	710	740	680	690	670	700	685
FeatureG	50	55	52	54	53	51	56	53	55	52
FeatureH	15	18	17	16	19	14	13	15	16	14
FeatureI	300	320	310	330	315	290	295	285	300	292
FeatureJ	12	12	12	12	12	12	12	12	12	1

And the following dataset configuration:

```
tx:
  dataset_dir: transcriptomics
    normalization_parameters:
    filtering:
      method: minimum
      value: 10
    devariancing:
      method: percent
      value: 20
    scaling:
      log2: true
      method: modified_zscore
    replicate_handling:
      method: variance
      group: group
      value: 0.5
```

Step 1: Filtering (`filter_data`)

- Options: minimum, value = 10
- Result: Remove FeatureB (average observed = 5.7, below threshold)

Feature	**High_1**	**High_2**	**High_3**	**High_4**	**High_5**	**Low_1**	**Low_2**	**Low_3**	**Low_4**	**Low_5**
FeatureA	900	850	920	870	910	20	25	22	18	24
FeatureC	100	120	110	130	115	90	95	85	100	92
FeatureD	500	520	510	530	515	480	490	470	495	485
FeatureE	10	10	10	10	10	10	10	10	10	10
FeatureF	700	750	720	710	740	680	690	670	700	685
FeatureG	50	55	52	54	53	51	56	53	55	52
FeatureH	15	18	17	16	19	14	13	15	16	14
FeatureI	300	320	310	330	315	290	295	285	300	292
FeatureJ	12	12	12	12	12	12	12	12	12	1

Step 2: Devariancing (`devariance_data`)

- Options: percent, value = 20 (removes 20% of 9 feature rounded down, i.e., 1 feature with lowest variance)
- Result: Remove FeatureE (variance = 0)

Feature	High_1	High_2	High_3	High_4	High_5	Low_1	Low_2	Low_3	Low_4	Low_5
FeatureA	900	850	920	870	910	20	25	22	18	24
FeatureC	100	120	110	130	115	90	95	85	100	92
FeatureD	500	520	510	530	515	480	490	470	495	485
FeatureF	700	750	720	710	740	680	690	670	700	685
FeatureG	50	55	52	54	53	51	56	53	55	52
FeatureH	15	18	17	16	19	14	13	15	16	14
FeatureI	300	320	310	330	315	290	295	285	300	292
FeatureJ	12	12	12	12	12	12	12	12	12	1

Step 3: Scaling

- Option: log2 + modified_zscore
- Result: scaled dataset

First, apply log2(x+1) transformation:

Feature	High_1	High_2	High_3	High_4	High_5	Low_1	Low_2	Low_3	Low_4	Low_5
FeatureA	9.813	9.741	9.842	9.770	9.831	4.392	4.700	4.523	4.247	4.643
FeatureC	6.658	6.918	6.797	7.044	6.857	6.507	6.614	6.426	6.658	6.523
FeatureD	8.967	9.025	8.995	9.053	9.010	8.918	8.965	8.888	8.977	8.931
FeatureF	9.454	9.561	9.492	9.470	9.545	9.419	9.453	9.398	9.454	9.423
FeatureG	5.672	5.807	5.700	5.779	5.740	5.700	5.857	5.740	5.807	5.700
FeatureH	4.000	4.322	4.247	4.170	4.392	3.907	3.807	4.000	4.170	3.907
FeatureI	8.233	8.330	8.285	8.375	8.309	8.201	8.236	8.154	8.233	8.207
FeatureJ	3.700	3.700	3.700	3.700	3.700	3.700	3.700	3.700	3.700	1.000

Then, apply modified z-score:

Feature	High_1	High_2	High_3	High_4	High_5	Low_1	Low_2	Low_3	Low_4	Low_5
FeatureA	0.009	-0.009	0.034	-0.028	0.025	-1.022	-0.622	-0.883	-1.263	-0.713
FeatureC	0.073	0.442	0.253	0.701	0.326	-0.179	0.011	-0.357	0.073	-0.126
FeatureD	0.025	0.093	0.062	0.130	0.087	-0.025	0.022	-0.055	0.034	-0.012
FeatureF	0.044	0.186	0.089	0.047	0.179	-0.022	0.044	-0.067	0.044	-0.015
FeatureG	-0.067	0.179	-0.022	0.134	0.067	-0.022	0.224	0.067	0.179	-0.022
FeatureH	-0.134	0.224	0.134	0.044	0.313	-0.224	-0.313	-0.134	0.044	-0.224
FeatureI	0.044	0.186	0.089	0.228	0.120	-0.022	0.044	-0.067	0.044	-0.015
FeatureJ	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	-2.494

Step 4: Replicate Handling

- Option: variance, value = 0.5, group = group (high vs low)
- Result: Remove FeatureJ (Low_5 sample value is an outlier, causing high within-group variance in "low" group)

Feature	High_1	High_2	High_3	High_4	High_5	Low_1	Low_2	Low_3	Low_4	Low_5
FeatureA	0.009	-0.009	0.034	-0.028	0.025	-1.022	-0.622	-0.883	-1.263	-0.713
FeatureC	0.073	0.442	0.253	0.701	0.326	-0.179	0.011	-0.357	0.073	-0.126
FeatureD	0.025	0.093	0.062	0.130	0.087	-0.025	0.022	-0.055	0.034	-0.012
FeatureF	0.044	0.186	0.089	0.047	0.179	-0.022	0.044	-0.067	0.044	-0.015
FeatureG	-0.067	0.179	-0.022	0.134	0.067	-0.022	0.224	0.067	0.179	-0.022
FeatureH	-0.134	0.224	0.134	0.044	0.313	-0.224	-0.313	-0.134	0.044	-0.224
FeatureI	0.044	0.186	0.089	0.228	0.120	-0.022	0.044	-0.067	0.044	-0.015

Final Output: After all normalization steps, the dataset contains 7 features and all 10 samples, with all values log2-transformed and modified z-score standardized. This dataset now has a quality-controlled and standardized quantitative distribution and can be integrated with other datasets that have undergone the same treatment.