
Blacksheep

Release 0.0.1

Ruggleslab

Dec 02, 2019

CONTENTS:

1	blacksheep package	1
1.1	Submodules	4
1.2	blacksheep.catheat module	4
1.3	blacksheep.classes module	5
1.4	blacksheep.cli module	6
1.5	blacksheep.comparisons module	6
1.6	blacksheep.deva module	6
1.7	blacksheep.parsers module	8
1.8	blacksheep.visualization module	9
2	Indices and tables	11
3	Command Line Interface	13
3.1	Positional Arguments	13
3.2	Named Arguments	13
3.3	Sub-commands:	13
3.3.1	normalize	13
3.3.1.1	Positional Arguments	13
3.3.1.2	Named Arguments	13
3.3.2	outliers_table	14
3.3.2.1	Positional Arguments	14
3.3.2.2	Named Arguments	14
3.3.3	binarize	14
3.3.3.1	Positional Arguments	14
3.3.3.2	Named Arguments	15
3.3.4	compare_groups	15
3.3.4.1	Positional Arguments	15
3.3.4.2	Named Arguments	15
3.3.5	visualize	16
3.3.5.1	Positional Arguments	16
3.3.5.2	Named Arguments	16
3.3.6	deva	17
3.3.6.1	Positional Arguments	17
3.3.6.2	Named Arguments	17
3.3.7	simulations	18
3.3.7.1	Positional Arguments	19
3.3.7.2	Named Arguments	19
	Python Module Index	21

BLACKSHEEP PACKAGE

```
blacksheep.make_outliers_table(df: pandas.core.frame.DataFrame, iqrs: float = 1.5,
                               up_or_down: str = 'up', aggregate: bool = True,
                               save_outlier_table: bool = False, save_frac_table: bool =
                               False, output_prefix: str = 'outliers', ind_sep: str = '-') →
                               blacksheep.classes.OutlierTable
```

Converts a DataFrame of values into an OutlierTable object, which includes a DataFrame of outlier and non-outlier count values.

Parameters

- **df** – Input DataFrame with samples as columns and sites/genes as columns.
- **iqrs** – The number of inter-quartile ranges (IQRs) above or below the median to consider a value as an outlier.
- **up_or_down** – Whether to call up or down outliers. Up is above the median; down is below the median. Options “up” or “down”.
- **aggregate** – Whether to sum outliers across a grouping (e.g. gene-level) than individual sites. For instance if columns indicate phosphosites on proteins, with the format “RAG2-S365”, output will show counts of outliers per protein (e.g. RAG2) rather than on individual sites (e.g. RAG2-S365).
- **save_outlier_table** – Whether to write a file with the outlier count table.
- **save_frac_table** – Whether to write a file with the outlier fraction table.
- **output_prefix** – If files are written, a prefix for the files.
- **ind_sep** – The separator used in sites, for instance, to separate a gene and site. If just using genes (i.e. no separator), or not aggregating this parameter has no effect.

Returns: outliers Returns an OutlierTable object, with outlier and non-outlier counts and metadata about how the outliers were called.

```
blacksheep.compare_groups_outliers(outliers: blacksheep.classes.OutlierTable, annotations:
                                   pandas.core.frame.DataFrame, frac_filter: Optional[float]
                                   = 0.3, save_qvalues: bool = False, output_prefix: str =
                                   'outliers', save_comparison_summaries: bool = False) →
                                   blacksheep.classes.qValues
```

Takes an OutlierTable object and a sample annotation DataFrame and performs comparisons for any column in annotations with exactly 2 groups. For each group identified in the annotations DataFrame, this function will calculate the q-values of enrichment of outliers for each row in each group.

Parameters

- **outliers** – An OutlierTable, with a DataFrame of outlier and non-outlier counts, as well as parameters for how outliers were calculated.
- **annotations** – A DataFrame with samples as rows and annotations as columns. Each column must contain exactly 2 different categories, not counting missing values. Columns without 2 options will be ignored.
- **frac_filter** – The fraction of samples in the group of interest that must have an outlier value to be considered in the comparison. Float between 0 and 1 or None.
- **save_qvalues** – Whether to write a file with a table of qvalues.
- **output_prefix** – If files are written, a prefix for the files.
- **save_comparison_summaries** – Whether to write a file for each annotation column with the counts in the fisher table, pvalues and q values per row.

Returns: qvals A qValues object, which includes a DataFrame of q-values for each comparison, as well as some metadata about how the comparisons were performed.

```
blacksheep.deva(df: pandas.core.frame.DataFrame, annotations: pandas.core.frame.DataFrame,
               iqr: float = 1.5, up_or_down: str = 'up', aggregate: bool = True,
               save_outlier_table: bool = False, save_frac_table: bool = False, frac_filter:
               Optional[float] = 0.3, save_qvalues: bool = False, output_prefix: str = 'out-
               liers', ind_sep: str = '-', save_comparison_summaries: bool = False) → Tu-
               ple[blacksheep.classes.OutlierTable, blacksheep.classes.qValues]
```

Takes a DataFrame of values and returns OutlierTable and qValues objects. This command runs the whole outliers pipeline. The DataFrame in the OutlierTable object can be used to run more comparisons in future. The qValues object can be used for visualization, or writing significant gene lists.

Parameters

- **df** – Input DataFrame with samples as columns and sites/genes as rows.
- **annotations** – A DataFrame with samples as rows and annotations as columns. Each column must contain exactly 2 different values, not counting missing values. Other columns will be ignored.
- **iqr** – The number of interquartile ranges (IQRs) above or below the median to consider a value as an outlier.
- **up_or_down** – Whether to call up or down outliers. Up is above the median; down is below the median. Options “up” or “down”.
- **aggregate** – Whether to sum outliers across a grouping (e.g. gene-level) than individual sites. For instance if columns indicate phosphosites on proteins, with the format “RAG2-S365”, output will show counts of outliers per protein (e.g. RAG2) rather than on individual sites (e.g. RAG2-S365).
- **save_outlier_table** – Whether to write a file with the outlier count table.
- **save_frac_table** – Whether to write a file of the fraction of outliers.
- **frac_filter** – The fraction of samples in the group of interest that must have an outlier value to be considered in the comparison. Float between 0 and 1 or None.
- **save_qvalues** – Whether to write a file of qvalues.
- **output_prefix** – If files are written, a prefix for the files.
- **ind_sep** – The separator used in the columns, for instance, to separate a gene and site. If just using genes (i.e. no separator), or not aggregating this parameter has no effect.

- **save_comparison_summaries** – Whether to write a table for each comparison with the counts in the fisher table, pvalues and qvalues per row.

Returns: **outliers, qvals** Returns an OutlierTable object and qValues object.

```
blacksheep.plot_heatmap(annotations: pandas.core.frame.DataFrame, qvals: pandas.core.frame.DataFrame, col_of_interest: str, vis_table: pandas.core.frame.DataFrame, fdr: float = 0.05, red_or_blue: str = 'red', output_prefix: str = 'outliers', colors: Optional[str] = None, savefig: bool = False) → list
```

Plots a heatmap of significantly enriched values for a given comparison.

Parameters

- **annotations** – Annotations DataFrame, samples as rows, annotations as columns
- **qvals** – qvalues DataFrame with genes/sites as rows and comparisons as columns
- **col_of_interest** – Which column from qvalues should be used to find significant genes
- **vis_table** – Table to be visualized in heatmap. Index values should correspond to the annotation df index, column names should correspond to qvals df index
- **fdr** – FDR threshold to for significance
- **red_or_blue** – Whether heatmap should be in red or blue color scale
- **output_prefix** – If saving files, output prefix
- **colors** – File to find color map for annotation header
- **savefig** – Whether to save the plot to a pdf

Returns: **[annot_ax, vals_ax, cbar_ax, leg_ax]** List of matplotlib ax, can be further customized before saving. In order the axes contain: annotation header, the heatmap, the color bar, and the legend.

```
blacksheep.run_simulations(infile, ind_sep, thresh, reps, outfile, genes, pval)
```

```
blacksheep.binarize_annotations(df: pandas.core.frame.DataFrame) → pandas.core.frame.DataFrame
```

Takes an annotation DataFrame, checks each column for the number of possible values, and adjusts based on that. If the column has 0 or 1 options, it is dropped. Cols with 2 possible values are retained as-is. Cols with more than 2 values are expanded. For each value in that column, a new column is created with val and not_val options.

Parameters **df** – Annotations DataFrame.

Returns: **new_df** Refactored annotations DataFrame.

```
blacksheep.normalize(df: pandas.core.frame.DataFrame) → pandas.core.frame.DataFrame
```

Performs median of ratios normalization on a given dataframe, then a log2 transform.

Parameters **df** – Unnormalized values dataframe

Returns: Normalized dataframe

```
blacksheep.read_in_values(path: str) → pandas.core.frame.DataFrame
```

Figures out sep and parsing file into dataframe.

Parameters **path** – File path

Returns: **df** DataFrame from table in file

`blacksheep.read_in_outliers` (*path: str, updown: str, iqrs: float*) → `blacksheep.classes.OutlierTable`
 Parses a file into an OutlierTable object.

Parameters

- **path** – File path
- **updown** – Whether the outliers represent up or down outliers
- **iqrs** – How many IQRs were used to define an outlier

Returns: `outliers` OutlierTable object

class `blacksheep.qValues` (*df: pandas.core.frame.DataFrame, comps: list, frac_filter: Optional[float]*)
 Bases: `object`

Output from comparing groups using outliers.

make_signed_logqs () → `pandas.core.frame.DataFrame`

Create a DataFrame with signed log10 qvalues for each comparison. E.g. group1 qvalues will be positive, and group 2 qvalues will be negative. Assignment of positive group is based on order in qvalues, could be helpful to negate some columns in output depending on group of interest.

Returns: DataFrame with signed qvalues.

write_gene_lists (*fdr_cut_off: float = 0.01, output_prefix: str = 'outliers', comparisons: Optional[List] = None*)

Writes significant gene list files for every column in a qvalue table

Parameters

- **fdr_cut_off** – FDR threshold for significance
- **output_prefix** – Output prefix for files
- **comparisons** – which subset of qvalue columns to write gene lists for. Default will
- **for all columns** (*write*) –

Returns: None

class `blacksheep.OutlierTable` (*df: pandas.core.frame.DataFrame, updown: str, iqrs: Optional[float], samples: Optional[list], frac_table: Optional[pandas.core.frame.DataFrame]*)

Bases: `object`

Output of calling outliers.

1.1 Submodules

1.2 blacksheep.catheat module

`blacksheep.catheat.heatmap` (*data, cmap: Optional[dict] = None, palette: str = 'hls', ax=None, legend: bool = True, leg_pos: str = 'right', leg_ax=None, leg_kws: Optional[dict] = None, **sns_kws*)

Class to plot categorical heatmap using seaborn.

Parameters

- **data** (*rectangular dataset*) – 2D dataset that can be coerced into an ndarray. If a Pandas DataFrame is provided, the index/column information will be used to label the columns and rows.
- **cmap** (*dict, optional*) – Colors for each category in the dataset. Missing colors will be added from the palette.
- **palette** (*matplotlib/seaborn color palette name or object, optional*) – Palette to be used for heatmap.
- **ax** (*matplotlib ax, optional*) –
- **legend** (*bool, optional*) – If True, plot legend.
- **leg_ax** (*matplotlib axis, optional*) – By default, will add legend to same ax as heatmap. Use this argument to explicitly set legend ax.
- **leg_pos** (*{'right', 'top'}*) – Position of legend. Only relevant if legend ax is not explicitly provided via *leg_ax*.
- **leg_kws** (*dict, optional*) – Keyword arguments passed to `plt.legend()`
- ****sns_kws** – Keyword arguments passed through to `seaborn.heatmap()`

Returns

- *matplotlib axis* – Heatmap axis as returned by `seaborn.heatmap()`.
- **colormap** (*dict*) – Colormap mapping categorical values to RGB colours.

1.3 blacksheep.classes module

class `blacksheep.classes.OutlierTable` (*df: pandas.core.frame.DataFrame, updown: str, iqr: Optional[float], samples: Optional[list], frac_table: Optional[pandas.core.frame.DataFrame]*)

Bases: `object`

Output of calling outliers.

`blacksheep.classes.list_to_file` (*lis: Iterable, filename: str*)

Takes an iterable and a file path and writes a value per line from the iterable into the new file.

Parameters

- **lis** – Iterable to write to file
- **filename** – Filename to write to.

Returns `None`

`blacksheep.classes.make_frac_table` (*df, samples*)

Constructs the fraction table from the outliers table

Returns: A DataFrame with one column per sample, with the fraction of outliers per row per sample. This table is useful for visualization but not statistics.

class `blacksheep.classes.qValues` (*df: pandas.core.frame.DataFrame, comps: list, frac_filter: Optional[float]*)

Bases: `object`

Output from comparing groups using outliers.

make_signed_logqs () → pandas.core.frame.DataFrame

Create a DataFrame with signed log10 qvalues for each comparison. E.g. group1 qvalues will be positive, and group 2 qvalues will be negative. Assignment of positive group is based on order in qvalues, could be helpful to negate some columns in output depending on group of interest.

Returns: DataFrame with signed qvalues.

write_gene_lists (fdr_cut_off: float = 0.01, output_prefix: str = 'outliers', comparisons: Optional[List] = None)

Writes significant gene list files for every column in a qvalue table

Parameters

- **fdr_cut_off** – FDR threshold for significance
- **output_prefix** – Output prefix for files
- **comparisons** – which subset of qvalue columns to write gene lists for. Default will
- **for all columns** (write) –

Returns: None

1.4 blacksheep.cli module

1.5 blacksheep.comparisons module

blacksheep.comparisons.get_sample_lists (annotations: pandas.core.frame.DataFrame, col: str) → Tuple[Optional[str], Optional[List[str]], Optional[str], Optional[List[str]]]

Finds groupings of samples from an annotation DataFrame column.

Parameters

- **annotations** – A DataFrame with samples as the index and annotations as columns. Each
- **must contain exactly 2 different values, and optionally missing values. Columns** (column) –
- **less or more than 2 options will be ignored. (with)** –
- **col** – Which column for which to define groups.

Returns: A label for group0, the list of samples in group0, a label for group1 and the list of samples in group1.

1.6 blacksheep.deva module

blacksheep.deva.compare_groups_outliers (outliers: blacksheep.classes.OutlierTable, annotations: pandas.core.frame.DataFrame, frac_filter: Optional[float] = 0.3, save_qvalues: bool = False, output_prefix: str = 'outliers', save_comparison_summaries: bool = False) → blacksheep.classes.qValues

Takes an OutlierTable object and a sample annotation DataFrame and performs comparisons for any column in

annotations with exactly 2 groups. For each group identified in the annotations DataFrame, this function will calculate the q-values of enrichment of outliers for each row in each group.

Parameters

- **outliers** – An OutlierTable, with a DataFrame of outlier and non-outlier counts, as well as parameters for how outliers were calculated.
- **annotations** – A DataFrame with samples as rows and annotations as columns. Each column must contain exactly 2 different categories, not counting missing values. Columns without 2 options will be ignored.
- **frac_filter** – The fraction of samples in the group of interest that must have an outlier value to be considered in the comparison. Float between 0 and 1 or None.
- **save_qvalues** – Whether to write a file with a table of qvalues.
- **output_prefix** – If files are written, a prefix for the files.
- **save_comparison_summaries** – Whether to write a file for each annotation column with the counts in the fisher table, pvalues and q values per row.

Returns: qvals A qValues object, which includes a DataFrame of q-values for each comparison, as well as some metadata about how the comparisons were performed.

```
blacksheep.deva.deva(df: pandas.core.frame.DataFrame, annotations: pandas.core.frame.DataFrame, iqrs: float = 1.5, up_or_down: str = 'up',
aggregate: bool = True, save_outlier_table: bool = False, save_frac_table: bool = False, frac_filter: Optional[float] = 0.3, save_qvalues: bool = False,
output_prefix: str = 'outliers', ind_sep: str = '-', save_comparison_summaries: bool = False) → Tuple[blacksheep.classes.OutlierTable, blacksheep.classes.qValues]
```

Takes a DataFrame of values and returns OutlierTable and qValues objects. This command runs the whole outliers pipeline. The DataFrame in the OutlierTable object can be used to run more comparisons in future. The qValues object can be used for visualization, or writing significant gene lists.

Parameters

- **df** – Input DataFrame with samples as columns and sites/genes as rows.
- **annotations** – A DataFrame with samples as rows and annotations as columns. Each column must contain exactly 2 different values, not counting missing values. Other columns will be ignored.
- **iqrs** – The number of interquartile ranges (IQRs) above or below the median to consider a value as an outlier.
- **up_or_down** – Whether to call up or down outliers. Up is above the median; down is below the median. Options “up” or “down”.
- **aggregate** – Whether to sum outliers across a grouping (e.g. gene-level) than individual sites. For instance if columns indicate phosphosites on proteins, with the format “RAG2-S365”, output will show counts of outliers per protein (e.g. RAG2) rather than on individual sites (e.g. RAG2-S365).
- **save_outlier_table** – Whether to write a file with the outlier count table.
- **save_frac_table** – Whether to write a file of the fraction of outliers.
- **frac_filter** – The fraction of samples in the group of interest that must have an outlier value to be considered in the comparison. Float between 0 and 1 or None.
- **save_qvalues** – Whether to write a file of qvalues.

- **output_prefix** – If files are written, a prefix for the files.
- **ind_sep** – The separator used in the columns, for instance, to separate a gene and site. If just using genes (i.e. no separator), or not aggregating this parameter has no effect.
- **save_comparison_summaries** – Whether to write a table for each comparison with the counts in the fisher table, pvalues and qvalues per row.

Returns: **outliers, qvals** Returns an OutlierTable object and qValues object.

```
blacksheep.deva.make_outliers_table(df: pandas.core.frame.DataFrame, iqrs: float = 1.5,
                                     up_or_down: str = 'up', aggregate: bool = True,
                                     save_outlier_table: bool = False, save_frac_table: bool
                                     = False, output_prefix: str = 'outliers', ind_sep: str =
                                     '-') → blacksheep.classes.OutlierTable
```

Converts a DataFrame of values into an OutlierTable object, which includes a DataFrame of outlier and non-outlier count values.

Parameters

- **df** – Input DataFrame with samples as columns and sites/genes as columns.
- **iqrs** – The number of inter-quartile ranges (IQRs) above or below the median to consider a value as an outlier.
- **up_or_down** – Whether to call up or down outliers. Up is above the median; down is below the median. Options “up” or “down”.
- **aggregate** – Whether to sum outliers across a grouping (e.g. gene-level) than individual sites. For instance if columns indicate phosphosites on proteins, with the format “RAG2-S365”, output will show counts of outliers per protein (e.g. RAG2) rather than on individual sites (e.g. RAG2-S365).
- **save_outlier_table** – Whether to write a file with the outlier count table.
- **save_frac_table** – Whether to write a file with the outlier fraction table.
- **output_prefix** – If files are written, a prefix for the files.
- **ind_sep** – The separator used in sites, for instance, to separate a gene and site. If just using genes (i.e. no separator), or not aggregating this parameter has no effect.

Returns: **outliers** Returns an OutlierTable object, with outlier and non-outlier counts and metadata about how the outliers were called.

1.7 blacksheep.parsers module

```
blacksheep.parsers.binarize_annotations(df: pandas.core.frame.DataFrame) → pandas.core.frame.DataFrame
```

Takes an annotation DataFrame, checks each column for the number of possible values, and adjusts based on that. If the column has 0 or 1 options, it is dropped. Cols with 2 possible values are retained as-is. Cols with more than 2 values are expanded. For each value in that column, a new column is created with val and not_val options.

Parameters **df** – Annotations DataFrame.

Returns: **new_df** Refactored annotations DataFrame.

`blacksheep.parsers.normalize` (*df: pandas.core.frame.DataFrame*) → *pan-*
das.core.frame.DataFrame
 Performs median of ratios normalization on a given dataframe, then a log2 transform.

Parameters *df* – Unnormalized values dataframe

Returns: Normalized dataframe

`blacksheep.parsers.read_in_outliers` (*path: str, updown: str, iqrs: float*) → *black-*
sheep.classes.OutlierTable

Parses a file into an OutlierTable object.

Parameters

- **path** – File path
- **updown** – Whether the outliers represent up or down outliers
- **iqrs** – How many IQRs were used to define an outlier

Returns: *outliers* OutlierTable object

`blacksheep.parsers.read_in_values` (*path: str*) → *pandas.core.frame.DataFrame*
 Figures out sep and parsing file into dataframe.

Parameters *path* – File path

Returns: *df* DataFrame from table in file

`blacksheep.parsers.subset_by_genes` (*outliers: pandas.core.frame.DataFrame, ind_list:*
Iterable[str], ind_sep: str = None) → *pan-*
das.core.frame.DataFrame

1.8 blacksheep.visualization module

`blacksheep.visualization.plot_heatmap` (*annotations: pandas.core.frame.DataFrame, qvals:*
pandas.core.frame.DataFrame, col_of_interest: str,
vis_table: pandas.core.frame.DataFrame, fdr: float =
0.05, red_or_blue: str = 'red', output_prefix: str =
'outliers', colors: Optional[str] = None, savefig: bool
= False) → *list*

Plots a heatmap of significantly enriched values for a given comparison.

Parameters

- **annotations** – Annotations DataFrame, samples as rows, annotations as columns
- **qvals** – qvalues DataFrame with genes/sites as rows and comparisons as columns
- **col_of_interest** – Which column from qvalues should be used to find significant genes
- **vis_table** – Table to be visualized in heatmap. Index values should correspond to the annotation df index, column names should correspond to qvals df index
- **fdr** – FDR threshold to for significance
- **red_or_blue** – Whether heatmap should be in red or blue color scale
- **output_prefix** – If saving files, output prefix
- **colors** – File to find color map for annotation header

- **savefig** – Whether to save the plot to a pdf

Returns: [annot_ax, vals_ax, cbar_ax, leg_ax] List of matplotlib axes, can be further customized before saving. In order the axes contain: annotation header, the heatmap, the color bar, and the legend.

INDICES AND TABLES

- `genindex`
- `modindex`
- `search`

COMMAND LINE INTERFACE

```
usage: blacksheep [-h] [--version]
                  {normalize,outliers_table,binarize,compare_groups,visualize,deva,
↪simulations}
                  ...
```

3.1 Positional Arguments

which Possible choices: normalize, outliers_table, binarize, compare_groups, visualize, deva, simulations

3.2 Named Arguments

--version, -v show program's version number and exit

3.3 Sub-commands:

3.3.1 normalize

Takes an unnormalized values table and uses median of ratios normalization to normalize. Saves a log2 normalized table appropriate for BlackSheep analysis.

```
blacksheep normalize [-h] [--output_prefix OUTPUT_PREFIX] unnormed_values
```

3.3.1.1 Positional Arguments

unnormed_values Table of values to be normalized. Sites/genes as rows, samples as columns.

3.3.1.2 Named Arguments

--output_prefix Prefix for output file. Suffix will be '.normalized.tsv'
Default: "values"

3.3.2 outliers_table

Takes a table of values and converts to a table of outlier counts.

```
blacksheep outliers_table [-h] [--output_prefix OUTPUT_PREFIX] [--iqr IQR]
                          [--up_or_down {up,down}] [--ind_sep IND_SEP]
                          [--do_not_aggregate] [--write_frac_table]
                          values
```

3.3.2.1 Positional Arguments

values File path to input values. Columns must be samples, genes must be sites or genes. Only .tsv and .csv accepted.

3.3.2.2 Named Arguments

--output_prefix Output prefix for writing files. Default outliers.
Default: outliers

--iqr Number of interquartile ranges (IQRs) above or below the median to consider a value an outlier. Default is 1.5 IQRs.
Default: 1.5

--up_or_down Possible choices: up, down
Whether to look for up or down outliers. Choices are up or down. Default up.
Default: "true"

--ind_sep If site labels have a parent molecule (e.g. a gene name such as ATM) and a site identifier (e.g. S365) this is the delimiter between the two elements. Default is -
Default: "-"

--do_not_aggregate Use flag if you do not want to sum outliers based on site prefixes.
Default: False

--write_frac_table Use flag if you want to write a table with fraction of values per site, per sample that are outliers. Will not be written by default. Useful for visualization.
Default: False

3.3.3 binarize

Takes an annotation table where some columns may have more than 2 possible values (not including empty/null values) and outputs an annotation table with only two values per annotation. Propagates null values.

```
blacksheep binarize [-h] [--output_prefix OUTPUT_PREFIX] annotations
```

3.3.3.1 Positional Arguments

annotations Annotation table with samples as rows and annotation labels as columns.

3.3.3.2 Named Arguments

--output_prefix Output prefix for writing files. Default annotations. Suffix will be '.binarized.tsv'
Default: annotations

3.3.4 compare_groups

Takes an annotation table and outlier count table (output of outliers_table) and outputs qvalues from a statistical test that looks for enrichment of outlier values in each group in the annotation table. For each value in each comparison, the qvalue table will have 1 column, if there are any genes in that comparison.

```
blacksheep compare_groups [-h] [--ind_subset IND_SUBSET] [--ind_sep IND_SEP]
                          [--output_prefix OUTPUT_PREFIX]
                          [--frac_filter FRAC_FILTER]
                          [--write_comparison_summaries] [--iqrs IQRS]
                          [--up_or_down {up,down}] [--write_gene_list]
                          [--make_heatmaps] [--fdr FDR]
                          [--red_or_blue {red,blue}]
                          [--annotation_colors ANNOTATION_COLORS]
                          outliers_table annotations
```

3.3.4.1 Positional Arguments

outliers_table Table of outlier counts (output of outliers_table). Must be .tsv or .csv file, with outlier and non-outlier counts as columns, and genes/sites as rows.

annotations Table of annotations. Must be .csv or .tsv. Samples as rows and comparisons as columns. Comparisons must have only unique values (not including missing values). If there are more options than that, you can use binarize to prepare the table.

3.3.4.2 Named Arguments

--ind_subset File with subset of indexes to consider in comparison

--ind_sep Index separator for subsetting genes. Only needed if using ind_subset, and if rows of outliers are NOT aggregated.

--output_prefix Output prefix for writing files. Default outliers.
Default: outliers

--frac_filter The minimum fraction of samples per group that must have an outlier in a gene to consider that gene in the analysis. This is used to prevent a high number of outlier values in 1 sample from driving a low qvalue. Default 0.3
Default: 0.3

--write_comparison_summaries Use flag to write a separate file for each column in the annotations table, with outlier counts in each group, p-values and q-values in each group.
Default: False

--iqrs Number of IQRs used to define outliers in the input count table. Optional.

--up_or_down	Possible choices: up, down Whether input outlier table represents up or down outliers. Needed for output file labels. Default up
--write_gene_list	Use flag to write a list of significantly enriched genes for each value in each comparison. If used, need an fdr threshold as well. Default: False
--make_heatmaps	Use flag to draw a heatmap of significantly enriched genes for each value in each comparison. If used, need an fdr threshold as well. Default: False
--fdr	FDR cut off to use for significantly enriched gene lists and heatmaps. Default 0.05 Default: 0.05
--red_or_blue	Possible choices: red, blue If --make_heatmaps is called, color of values to draw on heatmap. Default red. Default: "red"
--annotation_colors	File with color map to use for annotation header if --make_heatmaps is used. Must have a 'value color' format for each value in annotations. Any value not represented will be assigned a new color.

3.3.5 visualize

Used to make custom heatmaps from significant genes.

```
blacksheep visualize [-h] [--output_prefix OUTPUT_PREFIX]
                    [--annotations_to_show ANNOTATIONS_TO_SHOW [ANNOTATIONS_TO_SHOW .
→...]]
                    [--fdr FDR] [--red_or_blue {red,blue}]
                    [--annotation_colors ANNOTATION_COLORS]
                    [--write_gene_list]
                    comparison_qvalues annotations visualization_table
                    comparison_of_interest
```

3.3.5.1 Positional Arguments

comparison_qvalues	Table of qvalues, output from compare_groups. Must be .csv or .tsv. Has genes/sites as rows and comparison values as columns.
annotations	Table of annotations used to generate qvalues.
visualization_table	Values to visualize in heatmap. Samples as columns and genes/sites as rows. Using outlier fraction table is recommended, but original values can also be used if no aggregation was used.
comparison_of_interest	Name of column in qvalues table from which to visualize significant genes.

3.3.5.2 Named Arguments

--output_prefix	Output prefix for writing files. Default outliers. Default: outliers
------------------------	---

--annotations_to_show	Names of columns from the annotation table to show in the header of the heatmap. Default is all columns.
--fdr	FDR threshold to use to select genes to visualize. Default 0.05 Default: 0.05
--red_or_blue	Possible choices: red, blue Color of values to draw on heatmap. Default red. Default: "red"
--annotation_colors	File with color map to use for annotation header. Must have a line with 'value color' format for each value in annotations. Any value not represented will be assigned a new color.
--write_gene_list	Use flag to write a list of significantly enriched genes for each value in each comparison. Default: False

3.3.6 deva

Runs whole outliers pipeline. Has options to output every possible output.

```
blacksheep deva [-h] [--output_prefix OUTPUT_PREFIX] [--iqrs IQRS]
                [--up_or_down {up,down}] [--do_not_aggregate]
                [--write_outlier_table] [--write_frac_table]
                [--ind_sep IND_SEP] [--frac_filter FRAC_FILTER]
                [--write_comparison_summaries] [--fdr FDR] [--write_gene_list]
                [--make_heatmaps] [--red_or_blue {red,blue}]
                [--annotation_colors ANNOTATION_COLORS]
                values annotations
```

3.3.6.1 Positional Arguments

values	File path to input values. Samples are columns and genes/sites are rows. Only .tsv and .csv accepted.
annotations	File path to annotation values. Rows are sample names, header is different annotations. e.g. mutation status.

3.3.6.2 Named Arguments

--output_prefix	Output prefix for writing files. Default outliers. Default: outliers
--iqrs	Number of inter-quartile ranges (IQRs) above or below the median to consider a value an outlier. Default is 1.5. Default: 1.5
--up_or_down	Possible choices: up, down Whether to look for up or down outliers. Choices are up or down. Default up. Default: "true"

- do_not_aggregate** Use flag if you do not want to sum outliers based on site prefixes.
Default: False
- write_outlier_table** Use flag to write a table of outlier counts.
Default: False
- write_frac_table** Use flag if you want to write a table with fraction of values per site per sample that are outliers. Useful for custom visualization.
Default: False
- ind_sep** If site labels have a parent molecule (e.g. a gene name such as ATM) and a site identifier (e.g. S365) this is the delimiter between the two elements. Default is -
Default: “-“
- frac_filter** The minimum fraction of samples per group that must have an outlier in a gene to consider that gene in the analysis. This is used to prevent a high number of outlier values in 1 sample from driving a low qvalue. Default 0.3
Default: 0.3
- write_comparison_summaries** Use flag to write a separate file for each column in the annotations table, with outlier counts in each group, p-values and q-values in each group.
Default: False
- fdr** FDR threshold to use to select genes to visualize. Default 0.05
Default: 0.05
- write_gene_list** Use flag to write a list of significantly enriched genes for each value in each comparison.
Default: False
- make_heatmaps** Use flag to draw a heatmap of significantly enriched genes for each value in each comparison. If used, need an fdr threshold as well.
Default: False
- red_or_blue** Possible choices: red, blue
Color of values to draw on heatmap. Default red.
Default: “red”
- annotation_colors** File with color map to use for annotation header. Must have a line with ‘value color’ format for each value in annotations. Any value not represented will be assigned a new color.

3.3.7 simulations

Add here.

```
blacksheep simulations [-h] [--ind_sep IND_SEP] [--iqrs IQRS] [--reps REPS]
                        [--output_prefix OUTPUT_PREFIX]
                        [--molecules MOLECULES [MOLECULES ...]] [--pval PVAL]
                        values
```

3.3.7.1 Positional Arguments

values File path to input values. Samples are columns and genes/sites are rows. Only .tsv and .csv accepted.

3.3.7.2 Named Arguments

--ind_sep Delimiter between the parent molecule (e.g. a gene name such as ATM) and a site identifier (e.g. S365). Default is -
Default: “-“

--iqr Number of inter-quartile ranges (IQRs) above or below the median to consider a value an outlier. Default is 1.5.
Default: 1.5

--reps Number of repetitions for the simulation to perform. Default is 1,000,000.
Default: 1000000

--output_prefix Output prefix for writing files. Default is ‘simulated_pvals’.
Default: simulated_pvals

--molecules List of parent molecules of interest. Empty list or absence of argument defaults to all parent molecules in input file.
Default: []

--pval p-value threshold for significant results. Must be between 0 and 1. Default is 0.05.
Default: 0.05

PYTHON MODULE INDEX

b

- `blacksheep`, 1
- `blacksheep.catheat`, 4
- `blacksheep.classes`, 5
- `blacksheep.cli`, 6
- `blacksheep.comparisons`, 6
- `blacksheep.deva`, 6
- `blacksheep.parsers`, 8
- `blacksheep.visualization`, 9

B

binarize_annotations() (in module *blacksheep*), 3
 binarize_annotations() (in module *blacksheep.parsers*), 8
 blacksheep (module), 1
 blacksheep.catheat (module), 4
 blacksheep.classes (module), 5
 blacksheep.cli (module), 6
 blacksheep.comparisons (module), 6
 blacksheep.deva (module), 6
 blacksheep.parsers (module), 8
 blacksheep.visualization (module), 9

C

compare_groups_outliers() (in module *blacksheep*), 1
 compare_groups_outliers() (in module *blacksheep.deva*), 6

D

deva() (in module *blacksheep*), 2
 deva() (in module *blacksheep.deva*), 7

G

get_sample_lists() (in module *blacksheep.comparisons*), 6

H

heatmap() (in module *blacksheep.catheat*), 4

L

list_to_file() (in module *blacksheep.classes*), 5

M

make_frac_table() (in module *blacksheep.classes*), 5
 make_outliers_table() (in module *blacksheep*), 1
 make_outliers_table() (in module *blacksheep.deva*), 8

make_signed_logqs() (*blacksheep.classes.qValues* method), 5
 make_signed_logqs() (*blacksheep.qValues* method), 4

N

normalize() (in module *blacksheep*), 3
 normalize() (in module *blacksheep.parsers*), 8

O

OutlierTable (class in *blacksheep*), 4
 OutlierTable (class in *blacksheep.classes*), 5

P

plot_heatmap() (in module *blacksheep*), 3
 plot_heatmap() (in module *blacksheep.visualization*), 9

Q

qValues (class in *blacksheep*), 4
 qValues (class in *blacksheep.classes*), 5

R

read_in_outliers() (in module *blacksheep*), 3
 read_in_outliers() (in module *blacksheep.parsers*), 9
 read_in_values() (in module *blacksheep*), 3
 read_in_values() (in module *blacksheep.parsers*), 9
 run_simulations() (in module *blacksheep*), 3

S

subset_by_genes() (in module *blacksheep.parsers*), 9

W

write_gene_lists() (*blacksheep.classes.qValues* method), 6
 write_gene_lists() (*blacksheep.qValues* method), 4