Facilitate Analysis of Proteomic Data from Olink with Olink® Analyze

Olink® Analyze R Package Cheatsheet

Before generating data

Workflow

0 Before generating data
1 Generate data
2 Data input
3 Normalization
4 Preliminary Data Analysis
5 Statistical analysis
6 Biological visualization

Choosing a Statistical Model

The statistical model will depend on the study design. When choosing an analysis method it is important to answer the following questions:

- What is the question I am trying to answer? Ex: What proteins increase with time and with treatment?
- What is the sample size and assumed distribution? For sufficient sample sizes, NPX data can be used as is. To use parametric methods with absolute quantification, the data should be log transformed. Non-parametric methods also work for NPX or absolute quantification.
- How many variables are there? Ex: Time, Treatment
- How many levels are in each group? Ex: 3 timepoints, 2 treatments
- Are there any covariates to adjust for? (Other factors that may impact output variable unrelated for variables of interest) Ex: Age, Sex
- What comparisons are important? Ex: Time 1 vs Time 2 in Treated patients or Treated vs untreated patients at Time 2

Preliminary Data Analysis

Data overview, outlier detection, and identifying global trends

Data input

Reading in Olink data

read_NPX(filename) imports an NPX™ or Quant file exported from Olink NPX Signature, Olink NPX Explore, or MyData. The file should not be altered prior to import and will be imported in long format.

Column name Description
SampleID Sample ID name
Index Unique number for each SampleID
OlinkID Unique OlinkID for each protein
UniProt UniProt ID
Assay Protein symbol
MissingFreq Proportion of samples below LOD
Panel Version Panel version
PlateID Plate ID
QC_Warning QC status
LOD Limit of detection
NPX Normalized Protein Expression
Normalization Normalization method used

Normalization

Since NPX is a relative measurement, normalization is necessary to combine data across plates or across projects. An adjustment factor is calculated for each assay (protein) which is added to the NPX values from the second plate or project.

Normalization 2 projects
olink_normalization_bridge() - bridges 2 projects using overlapping samples using the median of paired differences
olink_normalization_subset() - normalizes 2 projects using the difference of group medians. The group can be a subset or all samples. No overlapping samples required.

Normalization more than 2 projects
olink_normalization_n() - normalize multiple projects based on provided normalization schema

Biostat tools

olink_bundleselector(df, sampleMissingFreq, n) selects n number of bridge samples based on input data. The samples are chosen based on passing QC, covering a large range of the data, and having less than sampleMissingFreq data below LOD per sample.

Olink® Analyze R Package

Olink® Analyze R Package

Olink Analyze and Shiny app support

Support

info@olink.com for more information

OLINK, NPX, PEA, PROXIMITY EXTENSION, INSIGHT and the Olink logotype are trademarks registered, or pending registration by Olink Proteomics AB. All company trademarks are the property of their respective owners.

© 2023 Olink Proteomics AB. Olink products and services are for Research Use Only and not for Use in Diagnostic Procedures. All information in this document is subject to change without notice. This document is intended to convey general information, representations and/or recommendations of any kind, unless such warranties, representations and/or recommendations are explicitly stated.
### Statistical analysis

#### Parametric Statistics

**Statistical analysis of normally distributed data**

- **olink_ttest(df, variable, pair_id)** performs a t-test for a given grouping variable as defined by `variable` argument. Paired t-test can be performed by defining `pair_id` argument. Variable must have 2 groups with more than 2 samples per group.

- **olink_anova(df, variable)** performs an ANOVA for each protein based on variables and covariates defined by `variable` and `covariate` arguments or by formula defined by `model_formula` argument.

- **olink_anova_posthoc(df, variable, olinkid_list, effect)** performs an ANOVA posthoc test per protein listed in `olinkid_list` based on variables and covariates defined by `variable` and `covariate` arguments or by formula defined by `model_formula` argument.

- **olink_lmer(df, variable)** performs a linear mixed model per protein based on variables and covariates defined by `variable` and `covariate` arguments or by formula defined by `model_formula` argument.

- **olink_lmer_posthoc(df, variable, olinkid_list, effect, random)** performs a linear mixed model posthoc per protein listed in `olinkid_list` based on variables and covariates defined by `variable` and `covariate` arguments or by formula defined by `model_formula` argument.

#### Non-parametric Statistics

**Statistical analysis of non-normally distributed data**

- **olink_wilcox(df, variable, pair_id)** performs Mann-Whitney U Test per protein for a given grouping variable as defined by `variable` argument. Wilcoxon signed-rank test can be performed by defining `pair_id` argument. Variable must have 2 groups with more than 2 samples per group.

- **olink_one_non_parametric(df, variable, dependence)** performs a one-way Kruskal-Wallis Test for each protein. When `dependence = TRUE`, a one-way Friedman Test will be performed for each protein. Model is defined by `variable` argument.

- **olink_one_non_parametric_posthoc(df, olinkid_list, variable)** performs a Wilcoxon posthoc test per protein listed in `olinkid_list` based on the variable defined by the `variable` argument.

- **olink_ordinalRegression(df, variable, covariates)** performs a 2-way ordinal ANOVA for each protein based on variables and covariates defined by `variable` and `covariate` arguments.

- **olink_ordinalRegression_posthoc(df, olinkid_list, variable, covariates, effect)** performs a 2-way ordinal ANOVA posthoc test per protein listed in `olinkid_list` based on variables and covariates defined by `variable` and `covariate` arguments.

### Statistical visualization

- **olink_volcano_plot(p.val_tbl)** generates a volcano plot using the results from **olink_ttest()**

- **olink_boxplot(df, variable, olinkid_list, posthoc_results, ttest_results)** generates boxplots for each protein in `olinkid_list` for `posthoc_results` or `ttest_results`

- **olink_lmer_plot(df, variable, random, olinkid_list, covariates, x_axis_variable, col_variable)** generates point-range plots of using `x_axis_variable` (x-axis) and `col_variable` (color) for each protein in `olinkid_list`.

### Biological interpretation

**Pathway Enrichment and Heatmap**

- **olink_pathway_enrichment(data, test_results)** performs pathway enrichment using over-representation analysis or gene-set enrichment analysis (based on `method` argument)

- **olink_pathway_heatmap(enrich_results, test_results)** generates a heatmap of proteins related to pathways using enrichment results for `olink_pathway_enrichment()

- **olink_pathway_visualization(enrich_results)** generates bargraph of top/selected enrichment terms based on enrichment results from `olink_pathway_enrichment()`

- **olink_heatmap_plot(df)** generates a heatmap of all samples from NPX data