IPCO: Inference of Pathways from Co-variance Analysis

IPCO is an open-source R library that infers functionality for a 16S dataset. IPCO can be implemented using internal or external reference data as per convenience. The references provided in IPCO are generated with UniRef90 database and the largest and manually curated MetaCyc mapping file provided along with HUMAnN2. It implements a double co-inertia methodology on the reference taxonomy and functionality database and co-varies with the queried data. The latest version can be downloaded from GitHub (https://github.com/IPCO-Rlibrary/IPCO).

For Correspondence Dr Ian B. Jeffery (i.jeffery@ucc.ie)

Requirements

1) R(v.3.5 or higher)
2) ade4 R library
3) devtools R library
4) A compatible taxonomic classification of the 16S data if using IPCO references
    a. Closed reference OTUs (GreenGenes 13.5)
    b. Species level classified data
    c. Genus level classified data

If using in-house reference and 16S dataset, then point number 4 is not mandatory.

Installation

1. Open R session
2. Load **library(devtools)**
3. Run **install_github("https://github.com/IPCO-Rlibrary/IPCO")**
4. If compatible version of ade4 library is not install then it needs to be install first.

IPCO commands

The prediction is a single step process in IPCO, however, precursor and optional steps involves loading the required reference data, normalisation/transformation of datasets, filtering functionality based on coverage and checking for covariance in reference if using in-house reference data.

1. Loading reference data

Following reference data are provided with IPCO

a) Genus and species level, MetaCyc pathways and KEGG pathways abundance and coverage table for a large cohort of healthy samples (n=1180)

b) HMP stool 16S (closed OTUs, genus and species level) and MetaCyc and KEGG pathways abundance and coverage datasets

We also provide other datasets, however, they are automatically available for loading into R, but can be accessed manually from the IPCO/data folder in GitHub. (Currently these datasets are not available and would be available soon)

Usage

data() loads the reference data for the cohort of healthy samples.

The reference data loaded into the R environment will have labelled as **IPCO_Healthy** which is a list object containing the following: **IPCO_Healthy$Species, IPCO_Healthy$Genus, IPCO_Healthy$KEGG, IPCO_Healthy$MetaCyc, IPCO_Healthy$KEGG_coverage** and **IPCO_Healthy$MetaCyc_coverage**. All objects in the list are matrices. Species and Genus matrices in the **IPCO_Healthy** list are generated from shotgun profiles.

If HMP data is load, the object is a list object labelled **IPCO_HMP** containing the following: **IPCO_HMP$closed_16S, IPCO_HMP$Species_16S, IPCO_HMP$Genus_16S, IPCO_HMP$Species, IPCO_HMP$Genus, IPCO_HMP$MetaCyc, IPCO_HMP$KEGG IPCO_HMP$KEGG_coverage** and **IPCO_HMP$MetaCyc_coverage**. All objects in the **IPCO_HMP** list are matrices. Species_16S and Genus_16S matrices are generated by mapping denovo OTUs to RDP database. By mapping the denovo OTUs to GreenGenes, closed_16S matrix is regenerated. Species and Genus labelled matrices are obtained from shotgun profiles.

2. **Normalisation/transformation of the datasets (optional)**

After loading the query and desired reference dataset, the datasets can be optionally normalised using **transform_data()** command which carries out Hellinger transformation as recommended. Alternatively the user is free to select any other transformation. The command **transform_data()** can only carry out Hellinger transformation from count, relative abundance or proportional dataset.

Usage

transform_data(dataset, type)

Arguments

dataset Dataset where rows are functions/pathways/OTUs and columns are samples.

type Enter the type of your dataset values: proportion, counts or relab (relative abundance).

Requires The dataset to be transformed and the type of values present in it.

Note: The provided reference functional and shotgun taxonomy datasets are proportional and relative abundance respectively and 16S HMP references are count data.

3. **Filtering functionality based on coverage (optional)**

The pathways with low coverage can be filtered using **filter_functionality()** based on the coverage information enabling better inference of the filtered pathways. Using the default threshold as recommended in the manuscript or a custom value, the filtering of pathways can be carried out.

Usage

filter_functionality(dataset, type, threshold_dataset, threshold)

Arguments

dataset dataset where rows are functions/pathways and columns are samples.

type Whether KEGG or MetaCyc dataset. type= “kegg” or type=“metacyc”

threshold_dataset dataset containing the information about pathway/function coverage. Threshold dataset should contain the same functions/pathways as present in dataset.

Optional

threshold Default is 0.01 for coverage for KEGG or the first quartile of the mean coverage for MetaCyc.

4. **Inferring functionality**

Inferring functionality is carried out by **IPCO()** command. It requires the reference taxonomic and functional datasets and the queried taxonomic dataset. Checking for significance in covariance between reference taxonomy and functional profiles is recommended if using in-house reference datasets. It requires the **rlq()** function from the ade4 library.

Usage

IPCO(R, L, Q)

Arguments

R R dataset where rows are functions/pathways and columns are samples

L L dataset where rows are taxa/OTUs and columns are the same samples as dataset R

Q Q dataset where rows are taxa/OTUs common between L and Q and columns are the samples for which functionality will be inferred

Requires The references functional (R) and its paired taxonomy dataset (L) and the taxonomy dataset (Q) for which functions need to be inferred

Note: Use of transformed/normalised and/or filtered datasets is optional however, it is recommended. IPCO internally check if there are zero abundance features and also check for common features (OTUs/genus/species) between reference taxonomy and queried dataset.

5. **Checking for significant covariance between reference taxonomic and functional datasets (optional)**

As discussed in the manuscript, the covariance between the taxonomic and functional datasets is in part responsible for inferring profiles accurately. When using in-house reference dataset, using the command **check_cointertia()**, significance between the taxonomic and functional reference data can be obtained. This is wrapper command containing the all the steps to calculate coinertia and its significance using ade4 library. It generates pca objects internally using **dudi.pca()** from the taxonomic and functional datasets, and uses the **coinertia()** function from ade4 to calculate the coinertia and the significance is determined using **randtest()** with 100 permutations. The coinertia object and randtest results are returned as a list object.

Usage

check_coinertia(R, L)

Arguments

R R table is the functions/pathways table where columns are samples

L L table is the taxonomic table where columns are the same samples as table R

Requires The references functional (R) and its paired taxonomy table (L) and the taxonomy table (Q) for which coinertia and its significance will be calculated.

Note: It does not transform/normalise the data automatically and scale=FALSE in **dudi.pca()** command.

An example of IPCO workflow

Step 1:Loading IPCO reference

```
data(IPCO_Healthy)

Or

data(IPCO_HMP)
```

Step 2: Normalisation

```
Ref_MetaCyc_norm <- transform_data(IPCO_Healthy$MetaCyc,“proportion”)

Ref_Species_norm <- transform_data(IPCO_Healthy$Species,“relab”)

                                        Or

Ref_HMP_closed_OTU_norm <- transform_data(IPCO_HMP$closed_16S,“count”)
```

Step 3: Filtering

```
Ref_MetaCyc_filter   <-   filter_functionality(Ref_MetaCyc_norm,   “metacyc”,
                          IPCO_Healthy$MetaCyc_coverage)

Or

Ref_MetaCyc_filter   <-   filter_functionality(Ref_MetaCyc_norm,   “metacyc”,
                          IPCO_Healthy$MetaCyc_coverage, threshold=0.4)
```

Note: Filtering out pathways whose mean coverage is below 40th percentile.

Step 4: Inference

```
IPCO_inferred <- IPCO(Ref_MetaCyc_filter, Ref_Species_norm, queried_dataset)
```

In case using in-house reference datasets

Step 5: Check coinertia

```
coinertia_resuls <- check_coinertia(taxonomy_table,functional_table)
```

Note: Ensure that the taxonomy and functional tables contains the same samples and in the same order.