make_input_OMELETmouse_latest.m

2022/03/06更新。

Major changes

主な変更点一覧。

- `mut_rxn`の指定が不要: これまでは独立フラックスのセットをあらかじめ自分で指定する必要があったが、代謝ネットワークの定義（`S_*.csv`）で指定しなくても（指定が適当でも）、代謝ネットワーク構造から独立フラックスのセットを見つけるように変更。独立フラックスのセットを見つけるのは具体的には `check_indflux_latest.m` にて行っている。
- 出力される `.txt` ファイルの名前が一部変更（プログラム内で読み込まれるものばかりなので気にしなくても良いが、デバッグの際は要参照）。

Arguments

- `S_path`: 代謝ネットワークの定義（`S_*.csv`へのpath）
- `data_dir_path`: metabolome.csv, proteome.csv, transcriptome.csv があるディレクトリへのpath
- `idx_smplgrp`: metabolome.csv, proteome.csv, transcriptome.csv に含まれるサンプルのうち、どの条件のサンプルを読み込むか。上記のcsvファイルに Index という行があり、それは各条件に番号を振ったもの。その中から自分が解析したい条件のセットを選ぶ。現状では、1番目の条件についてG6PC flux（グルコース産生フラックス）が1になるよう固定する。
- `savedir`: `make_input_OMELETmouse_latest.m` で作られるRstanへの入力ファイル（`.txt`群）を保存するディレクトリ。あらかじめ作っておく必要がある。

Return

- `X`: 代謝ネットワークの定義から読み込んだネットワーク情報
- `D`: オミクスデータから読み込んだオミクスデータ情報
- `out`: Rstanの入力用に整形した代謝ネットワーク・オミクスデータ情報。基本的にこの中の変数が `.txt` に出力される。
- `init`: Rstanでのパラメータ推定の際に用いる μ_l^u の初期値の情報。

Overall structure

- `X = parse_tbl(S_path);`: 代謝ネットワークの定義を読み込み。
- `X = check_indflux_latest(X);`: 独立フラックスのセットが妥当かどうかチェック。
- `X = calc_kernel(X);`: $W^u, W^{\dot{x}}$ などを計算。
- `D = load_omics_data(X,data_dir_path,idx_smplgrp);`: オミクスデータを読み込み。
- `out = make_output(X,D);`: Rstanの入力用に代謝ネットワーク・オミクスデータ情報を整形。
- `fid = fopen([savedir '/int_list.txt'],'w');` から `fclose(fid);` まで: `out`の中身を `.txt` ファイルに出力
- `init = set_init(X);`: Rstanでのパラメータ推定の際に用いる μ_l^u の初期値の設定。
- `fnames = fieldnames(init);` から `end`まで: `init` の中身を `.txt` ファイルに出力。
- `copyfile(S_path,savedir);`: 代謝ネットワークの定義を `.csv` をコピー（どの代謝ネットワークの定義から `.txt` ファイルを作ったか忘れないようにするため）

- `save([savedir '/model_data.mat'],'X','D','out','init');`: `make_input_OMELETmouse_latest.m` で作った上記の変数群を `model_data.mat` として保存。

Standard output

- `Reactions with multiple substrate:`: 2つ以上の基質をもつ反応名を出力
- `Reactions with multiple products:`: 2つ以上の生成物をもつ反応名を出力
- `The following [metabolites/proteins.transcripts] were not found in <data_path>/[metabolome/proteome/transcriptome].csv`: オミクス（メタボローム/プロテオーム/トランスクリプトーム）データに含まれていない（計測対象になっていない）分子（代謝物/酵素タンパク質/転写物）名を出力。オミクスデータ内に書かれているのと同じ代謝物名を指定する必要がある（例えばメタボロームデータ内に`Acetyl-CoA`と書いてるなら`S_*.csv`でも`Acetyl-CoA`と書く必要がある）ので、もし誤って分子名を指定してしまった場合でもこの標準出力を見ればわかる。
- `The following [metabolites/proteins.transcripts] exist but too many NaN in <data_path>/[metabolome/proteome/transcriptome].csv`: オミクス（メタボローム/プロテオーム/トランスクリプトーム）データに含まれている（計測対象になっていない）分子（代謝物/酵素タンパク質/転写物）の中で、欠損値が「同一条件の半数以上」のサンプルで見られた場合に、その分子名と欠損値の数を出力。例えば `[metabolite/protein/transcript]: #NaN=[1 1 1 4]` と出たら、その分子名が指定した条件のインデックス（`idx_smplgrp`）の順に欠損値が1, 1, 1, 4サンプルで見られたという意味になる。
- `The following substrate in reactions are not measured:`: 基質が計測できていない反応・基質名を出力する。これらの代謝物量はパラメータとして推定される。
- `The following protein levels should be inferred from transcript levels:`: 酵素タンパク質量が計測できていないために転写物量から推定しなければいけない酵素名を出力する。
- `The following metabolite effectors were not found in <data_path>/metabolome.csv`: 上と同様だが、`metabolite effectors` とは2つ目以上の基質・生成物、cofactor、アロステリック制御因子を指す。上の代謝物とは基本的に各反応の1つの基質・1つの生成物。特に`metabolite`と`metabolite effectors`を分けている意味はないので気にしないでください。

Example

実行例。自分の環境に合わせてpathは変えてください。

```
savedir_ = '/home/suematsu/Git/FluxAnalysis_result/result20220306';
S_path = '/home/suematsu/Git/FluxAnalysis_result/result20220306/S_OMELETmouse_model105.csv';
savedir = [savedir_ '/model105_all'];
data_dir_path = '/home/suematsu/Git/DATA/formatted/data_fasting';
mkdir(savedir);
% for example: WT16h, WT0,2,4,6,8,12,24h, ob16h, ob0,2,4,6,8,12,24h
% choose indeces from Index in metabolome.csv, proteome.csv, transcriptome.csv
idx_smplgrp = [7 1:6 8 15 9:14 16];
make_input_OMELETmouse_latest(S_path,data_dir_path,idx_smplgrp,savedir);
```