1. DREAM4 PIPELINE PSEUDO CODE

Note that this pseudo code is not intended to be comprehensive. But rather to point out the important steps in the DREAM4 pipeline. For more details consult the DREAM3 or DREAM4 (in preparation) plosOne papers. To begin, we assume that:

(1) modeled transcriptional entities are genes (and not biclusters as was in the original inferelator).
(2) all genes are possible predictors (but not environmental factors).
(3) dataset of mRNA levels is already properly normalized.
(4) When resmapling is used. We use 100% resampling, i.e. the size of resampled data sets is the same size of original data set.
(5) creating response and design matrices is not covered in pseudo code, i.e. pseudo code is not specific for ODE model. Covered in the DREAM4 and DREAM3 plosONE papers.
(6) only single predictors are allowed, i.e. we don't allow couplings (as predicting this interactions was not part of DREAM4 challenge).
(7) we use Mutual Information (MI), as a similarity metric between predictors and genes.
(8) LARS algorithm is abstracted as a function call. It takes as input response and design matrices, and outputs a sparse weight vector (choosing a shrinkage parameter, automatically, based on cross validation).
(9) Median Corrected Z-scores (MCZ) method is abstracted as a function call. It takes as input the KO data, and outputs a matrix of mcz scores for every possible regulatory interaction.
(10) Combining ranked regulatory interactions lists (implemented as combining two matrices of confidence scores in every putative regulatory interaction) is not discussed. Discussion can be found in DREAM4 and DREAM3 plosONE papers.

All of these assumptions are easy to remove and are used to make the pseudo code concise and clear.

Notations:

N_g - number of modeled entities (mRNA levels)
N_c - number of conditions in our dataset (time series and steady state conditions)
$N_p = N_g$ - number of predictors (see assumptions, could choose N_p to be number of TFs)
N_b - number of re-samplings (bootstraps), i.e. number of output regulatory networks
b - a counter that follows the current re-sample/iteration number ($b = 1, \dots, N_b$)
D - data set matrix holding the observed mRNA levels over the various experimental conditions, $N_g \times N_c$
Y - response matrix, $N_g \times N_c$
X - design matrix, $N_p \times N_c$
X^{ko} - the Knock Out data, where each gene at a time is KO'd and the system observed at steady state
Π - re-sampling vector, $1 \times N_c$, where $\pi_i \in \{1, \dots, N_c\}$, i.e. we are re-sampling the conditions
Y^π, X^π - bootstrapped response and design matrices respectively, $N_g \times N_c$ and $N_p \times N_c$ respectively
M^{s} - static mutual information matrix, holding MI scores between X^π and X^π (i.e. pairwise MI scores)
M^{d} - mutual information matrix, holding MI scores between Y^π and X^π (i.e. pairwise MI scores between response and
and putative predictors (columns), $N_g \times N_p$
$\boldsymbol{\beta}_b$ - regulatory model (result of re-sample run b), $N_g \times N_c$
Z_b - mixed-CLR z-scores (result of re-sample run b), $N_g \times N_c$
$\boldsymbol{\beta}$ - regulatory models, $N_g \times N_c \times N_b$ (output)
Z - mixed-CLR z-scores, $N_g \times N_c \times N_b$ (output)
MCZ - Median corrected Z-scores matrix (the same for each bootstrap run).
S - final list of confidence scores for regulatory interactions. The result of combining MCZ, with the "median" bootstra

Note that for each permutation vector, Π, we learn a complete RN network, that is we learn an ODE model for all of the modeled entities. As pseudo-code conventions we use indentation for program blocks;

we use the symbol, $\lhd$, for comments. Sometimes, throughout the pseudo-code we use MATLAB syntax.

Inferelator algorithm:

```
Input: $D$
Output: $\boldsymbol{\beta}, Z$

0. *Define parameters*:
    $max_p$ $\lhd$ the max number of predictors that are allowed to regulate any one gene
    set $N_g, N_c, N_p, N_b$

1. *Initialize*:
    create $X, Y$ $\lhd$ design and response matrices respectively
    $b = 1$

2. *Calculate Median Corrected Z scores and store them in $MCZ$ matrix*
    call MCZ function with $X^{\mathrm{ko}}$ data as input. Result in $MCZ$ matrix of confidence scores.

3. *Setup for bootstrap*:
    create $\Pi$ $\lhd$ a permutation vector
    create $Y^{\pi} = Y[:, \Pi], X^{\pi} = X[:, \Pi]$

4. *Pass one: fill $M^d$ - mutual information table*
    for i goes from 1 to $N_g$
        for j goes from 1 to $N_p$
            $M^{\mathrm{d}}[i, j]$ = MI between response $y_i^{\pi}$ (i.e. $Y^{\pi}[i, :]$) and predictor $x_j^{\pi}$ (i.e. $X^{\pi}[j, :]$)

5. *Pass one: fill $M^s$ - mutual information table*
    for i goes from 1 to $N_g$
        for j goes from 1 to $N_p$
            $M^{\mathrm{s}}[i, j]$ = MI between $X_i^{\pi}$ (i.e. $X^{\pi}[i, :]$) and predictor $x_j^{\pi}$ (i.e. $X^{\pi}[j, :]$)

6. *Calculate mixed-CLR matrix*
    for i goes from 1 to $N_g$
        for j goes from 1 to $N_p$
            $Z_b[i, j] = \sqrt{z_{row}^2 + z_{col}^2}$,
            where $z_{row}$ is the Z_score of entry $M^{\mathrm{d}}[i, j]$ wrt. the histogram of row i (i.e. $M^{\mathrm{d}}[i, :]$),
            and $z_{col}$ is the Z_score of entry $M^{\mathrm{d}}[i, j]$ wrt. the histogram of column j (i.e. $M^{\mathrm{s}}[:, j]$).

            Note that in $M^{\mathrm{d}}$ and $M^{\mathrm{s}}$ the diagonal entries is set to zero (autoregulaotry interac-
            tions are not considered).

7. *Pass two: compute LARS-generated ODE-models for each of the modeled entities*
    for i goes from 1 to $N_g$
        - create a predictor index vector, $P^{\pi} \subset \{1, \dots, N_p\}$, by choosing the $max_p$
        column corresponding to the highest MI scores in $M[i, :]$.
        - call LARS with response input $y_i^\pi$ and design matrix input $X^\pi[P^\pi, :]$, and store
        resulting weight vector in $\boldsymbol{\beta}_b[i, :]$.

8. *Store current re-sampling results*
    $\boldsymbol{\beta}[:, :, b] = \boldsymbol{\beta}_b[:, :]$
    $Z[:, :, b] = Z_b[:, :]$

9. *while* $b < N_b$ *increment* $b$ *by 1 and repeat steps* $2 - 6$

10. *Combine* $MCZ$ *with median network from combining* $\boldsymbol{\beta}$ *and* $Z$.
```