PROLONG

Penalized Regression on Longitudinal Omics Data with Network and Group Lasso Constraints

Steve Broll [1]

Advised by Sumanta Basu [1], Martin Wells [1], and Myung Hee Lee [2]

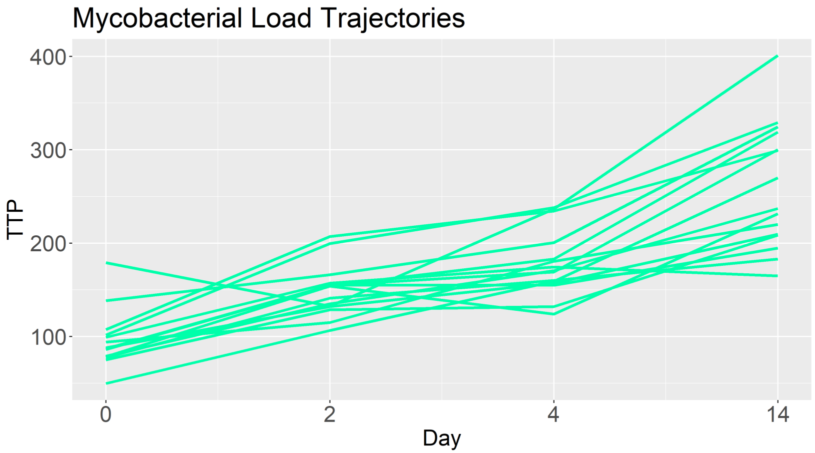
[1]Cornell Univeristy

[2]Weill Cornell Medicine

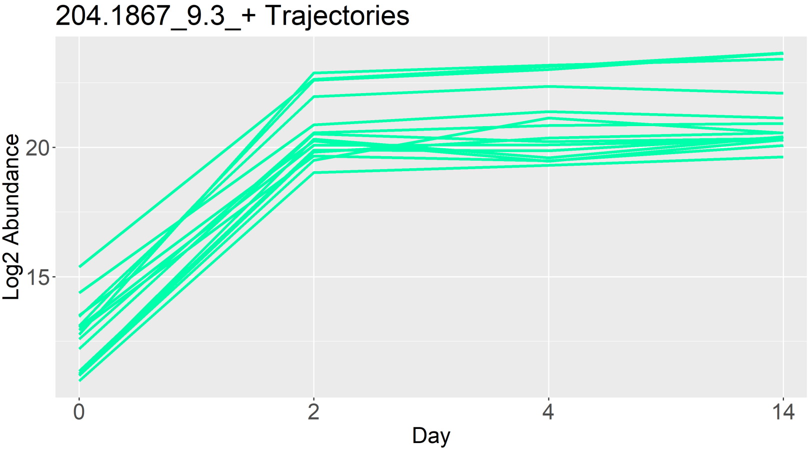
Motivation in Short

Clinician gives you a longitudinal clinical outcome, along with hundreds (or thousands) of longitudinal -omics variables, and asks which variables co-vary with the outcome?

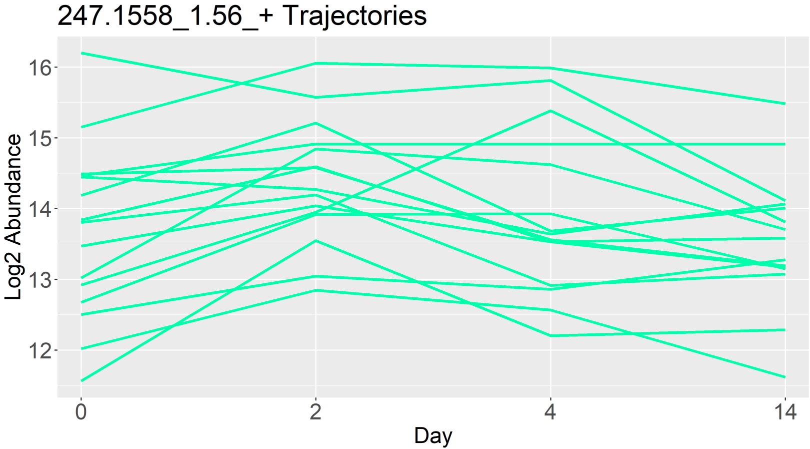
Motivation in Pictures - Outcome



Motivation in Pictures - Example Variable 1



Motivation in Pictures - Example Variable 2



Motivation in More Words

We have:

- Longitudinal measurements for some continuous outcome and for -omics variables with only a few time points
- Large amount of variables with relatively small number of subjects

We want to:

- Identify -omics variables that co-vary with the outcome
- Overcome time dependence, low signal, and high subject variability
- Incorporate correlation of the variables

Tuberculosis Data

- 15 subjects, TB patients treated with RHEZ [rifampin (R), isoniazid (H), ethambutol (E), and pyrazinamide (Z)]
- TB mycobacterial load measured by Time to Positivity (TTP) as our Y
- 352 urinary metabolites as our X
- 4 time points, days 0, 2, 4, 14

General Model Idea

- Take first difference of the data to deal with observed temporal dependence
- Stack our $t - 1$ first differenced value of X and Y so we have

  $$Y = |Y_4 - Y_3 \qquad Y_3 - Y_2 \qquad Y_2 - Y_1|^T$$

  And for each variable j we have

  $$X_j = |X_{j4} - X_{j3} \qquad X_{j3} - X_{j2} \qquad X_{j2} - X_{j1}|^T$$

- Set up design matrix so that each first differenced Y value is regressed on all prior first differenced values of X to account for potential lags
- Apply network and group lasso penalties to induce sparsity while utilizing correlation and inherent group structure

Vectorized Y

$$\tilde{Y} = \begin{bmatrix} \tilde{Y}_{11} & \cdots & \tilde{Y}_{1T} \\ & \vdots & \\ \tilde{Y}_{n1} & \cdots & \tilde{Y}_{nT} \end{bmatrix}_{n \times T} \rightarrow \begin{bmatrix} \Delta \tilde{Y}_{11} & \cdots & \Delta \tilde{Y}_{1(T-1)} \\ & \vdots & \\ \Delta \tilde{Y}_{n1} & \cdots & \Delta \tilde{Y}_{n(T-1)} \end{bmatrix}_{n \times (t-1)}$$

$$\rightarrow Y = \begin{bmatrix} \Delta \tilde{Y}_{11} \\ \vdots \\ \Delta \tilde{Y}_{n1} \\ \Delta \tilde{Y}_{1(T-1)} \\ \vdots \\ \Delta \tilde{Y}_{n(T-1)} \end{bmatrix}_{n(T-1) \times 1}$$

Moving X from Tensor to Matrix

$$\tilde{X}^{[j]} = \begin{bmatrix} \tilde{X}^{[j]}_{11} & \cdots & \tilde{X}^{[j]}_{1T} \\ & \vdots & \\ \tilde{X}^{[j]}_{n1} & \cdots & \tilde{X}^{[j]}_{nT} \end{bmatrix}_{n\times T} \rightarrow \begin{bmatrix} \Delta\tilde{X}^{[j]}_{11} & \cdots & \Delta\tilde{X}^{[j]}_{1(T-1)} \\ & \vdots & \\ \Delta\tilde{X}^{[j]}_{n1} & \cdots & \Delta\tilde{X}^{[j]}_{n(T-1)} \end{bmatrix}_{n\times(T-1)}$$

$$\rightarrow X^{[j]} = \left[\begin{array}{c|c|c|c} \begin{matrix} \Delta\tilde{X}^{[j]}_{11} \\ \vdots \\ \Delta\tilde{X}^{[j]}_{n1} \end{matrix} & 0 & 0 & 0 \\ \hline 0 & \begin{matrix} \Delta\tilde{X}^{[j]}_{11} & \Delta\tilde{X}^{[j]}_{12} \\ & \vdots & \\ \Delta\tilde{X}^{[j]}_{n1} & \Delta\tilde{X}^{[j]}_{n2} \end{matrix} & 0 & 0 \\ \hline 0 & 0 & \ddots & 0 \\ \hline 0 & 0 & 0 & \begin{matrix} \Delta\tilde{X}^{[j]}_{11} & \cdots & \Delta\tilde{X}^{[j]}_{1(T-1)} \\ & \vdots & \\ \Delta\tilde{X}^{[j]}_{n1} & \cdots & \Delta\tilde{X}^{[j]}_{n(T-1)} \end{matrix} \end{array}\right]_{n(T-1)\times T(T-1)/2}$$

Moving X from Tensor to Matrix

Now replace $\Delta\tilde{X}_{it}^{[j]}$ with row vector

$$\Delta\tilde{X}^{[j]} = |\Delta\tilde{X}_{ij}^{[1]}\Delta\tilde{X}_{ij}^{[2]}\dots\Delta\tilde{X}_{ij}^{[p]}|$$

$$\rightarrow X^{[j]} = \left[\begin{array}{c|c|c|c}
\begin{array}{c} \Delta\tilde{X}_{11}^{[j]} \\ \vdots \\ \Delta\tilde{X}_{n1}^{[j]} \end{array} & 0 & 0 & 0 \\
\hline
0 & \begin{array}{cc} \Delta\tilde{X}_{11}^{[j]} & \Delta\tilde{X}_{12}^{[j]} \\ \multicolumn{2}{c}{\vdots} \\ \Delta\tilde{X}_{n1}^{[j]} & \Delta\tilde{X}_{n2}^{[j]} \end{array} & 0 & 0 \\
\hline
0 & 0 & \ddots & 0 \\
\hline
0 & 0 & 0 & \begin{array}{ccc} \Delta\tilde{X}_{11}^{[j]} & \cdots & \Delta\tilde{X}_{1(T-1)}^{[j]} \\ & \vdots & \\ \Delta\tilde{X}_{n1}^{[j]} & \cdots & \Delta\tilde{X}_{n(T-1)}^{[j]} \end{array}
\end{array}\right]_{n(T-1)\times T(T-1)/2}$$

Group Lasso Laplacian Penalty

Given our stacked response vector Y and design matrix X we seek to minimize

$$(Y - X\beta)^T(Y - X\beta) + \lambda_1 \sum_{j=1}^{p} \left\|\boldsymbol{\beta}_{(j)}\right\|_2 + \lambda_2 \beta^T L \beta,$$

- λ_1 is the tuning parameter for our group lasso penalty, where each group j corresponds to all of the representations in the design matrix of the jth variable
- λ_2 is the tuning parameter for the network penalty
- L is the Laplacian matrix for the weighted graph where the edge weights between each pair of variables are their absolute correlation

Nice Properties of this Penalty

- Each variable is represented multiple times in the model, but the group lasso penalty results in either all zero or all non-zero coefficients for the representations of each variable, helping interpretability
- If two variables are highly correlated, and one is a strong enough predictor to be selected, the other variable is more likely to be selected than if they weren't correlated
- If two variables are identical, either both will be selected and have the same coefficient or neither will be selected

Models Compared

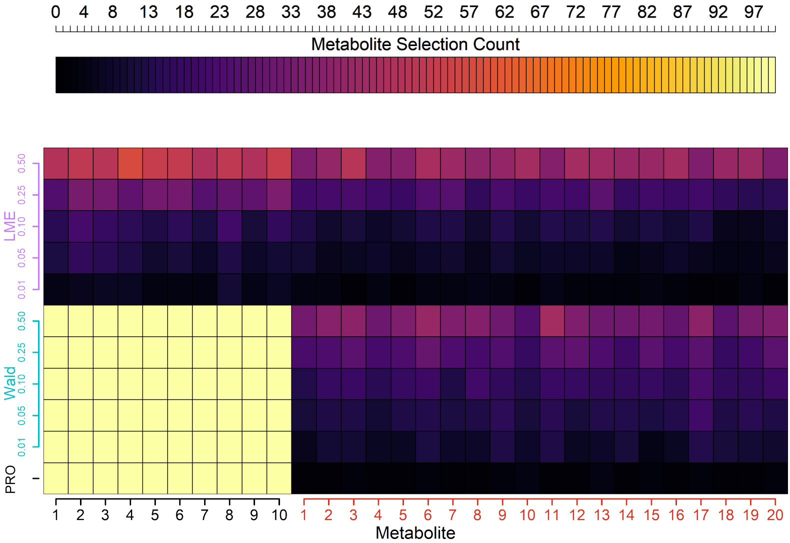
- Linear Mixed Effects Model, one variable at a time
- Wald test on the Δ scale, one variable at a time
- PROLONG

In the following simulations, the univariate models are evaluated at different FDR thresholds and compared to PROLONG

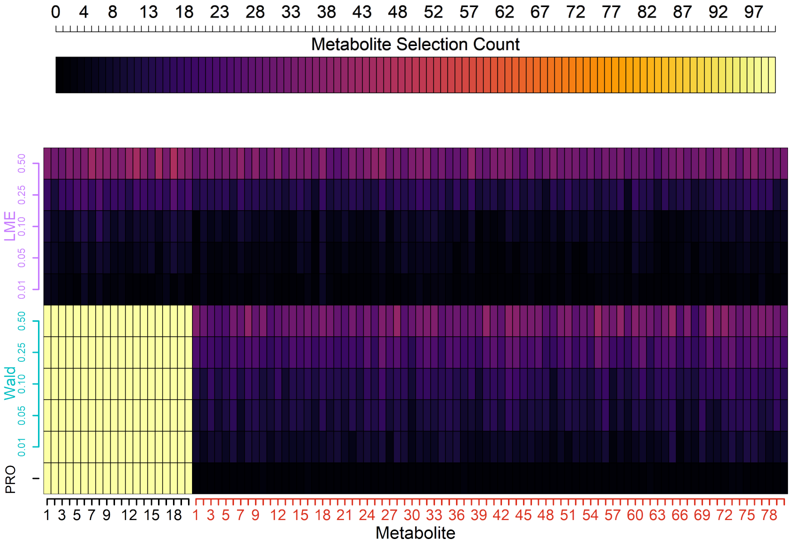
Simulation Scenarios

- Simulated data mimics real TB data in means, variances etc. but with specified relationships between X's and Y
- Outcome is generated by simulated, correlated target variables at varying dimensions with a SNR ranging from 1 to 2
    - 10, 20, and 50 target variables
    - 20, 80, and 300 noise variables
- Each scenario is run 100 times, and the models are evaluated by selection rate of target and noise variables

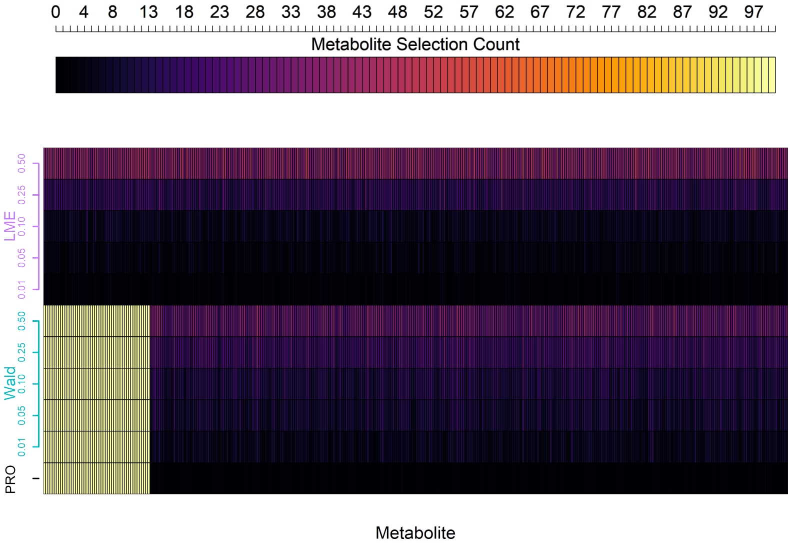
Performance in Simulations



Performance in Simulations



Performance in Simulations



Performance with Real Data

- Univariate mixed effect models do not pick up a single metabolite from our 352 at an FDR of 0.05
- Univariate Delta Wald tests pick 116 metabolites at an FDR of 0.05
- PROLONG selects 29 metabolites, including targets identified by our clinician collaborators and during our EDA

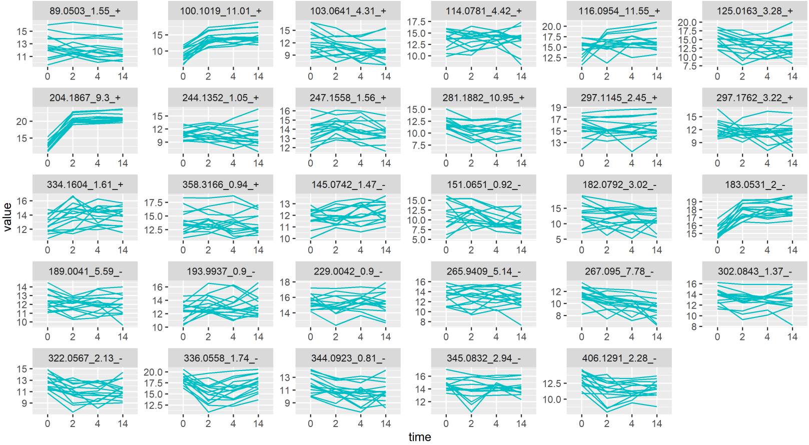
Summary

- PROLONG gets high sensitivity and specificity in simulations while 'competitor' mixed effects univariate models fail to distinguish between targets and noise across the board
- The univariate Wald version of our model does relatively well at each dimension, indicating the importance of the first-differencing and block structure
- PROLONG improves as dimension and sparsity increase and picks up significantly less noise than the univariate Wald

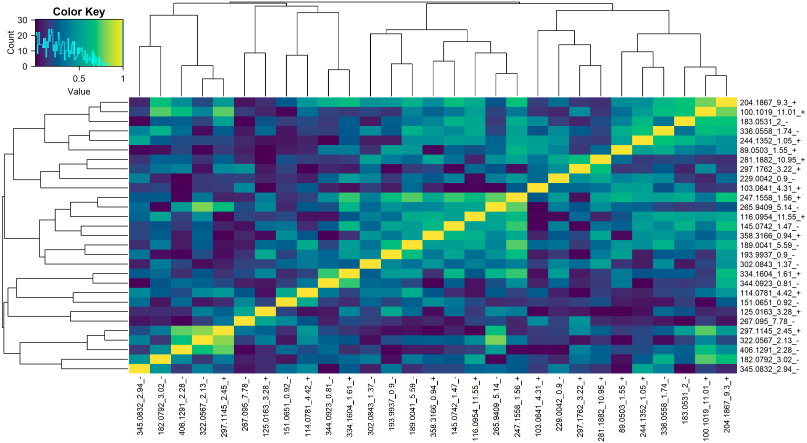
Applying PROLONG

- R package ‘prolong‘, available on Github currently, takes in raw time-scale data and
    - First-differences and shapes the data into the block design structure
    - Automatically selects hyper-parameters and fits the model
    - Provides visualizations for the full data and for selected variables
- Shiny app is in development and will be included within the ‘prolong‘ package, providing a point-and-click interface for users with less familiarity with R

R Package Selected Variable Trajectories



R Package Correlation Heatmap



Next Steps

- Fixed effects for multiple treatments, patient demographics, etc.
- Multi-omic data
- Tensor extension to avoid matricization and vectorization

Contact and Manuscript

Thank You!

R package available via Github:

https://github.com/stevebroll/prolong

Manuscript available via Biorxiv:

Steve Broll, Sumanta Basu, Myung Hee Lee, and Martin T. Wells.
PROLONG: Penalized regression for outcome guided longitudinal omics analysis with network and group constraints.
bioRxiv, 2023.

Email me at sb2643@cornell.edu

https://stevebroll.github.io