Graph-Sparse Logistic Regression

Alexander LeNail[1], Ludwig Schmidt[2], Jonathan Li[1], Tobias Ehrenberger[1], Karen Sachs[1], Stefanie Jegelka[2], Ernest Fraenkel[1]

[1]MIT BE, [2]MIT CSAIL

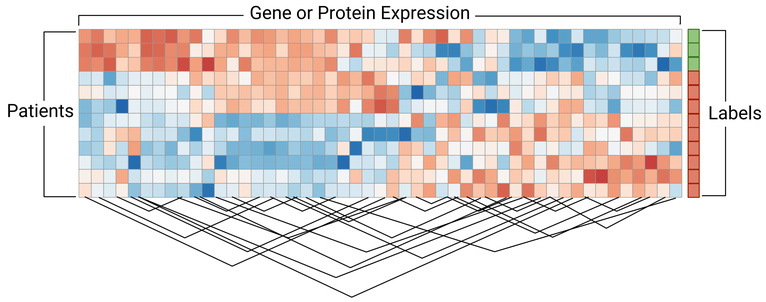
Introduction

Scientists often use machine learning to **learn more about their data**, not to predict it. They want to **interpret** their models.
The simplest way to build an interpretable model is to keep it **small**.
The classic way to accomplish this is with **LASSO** (L1-regularization).

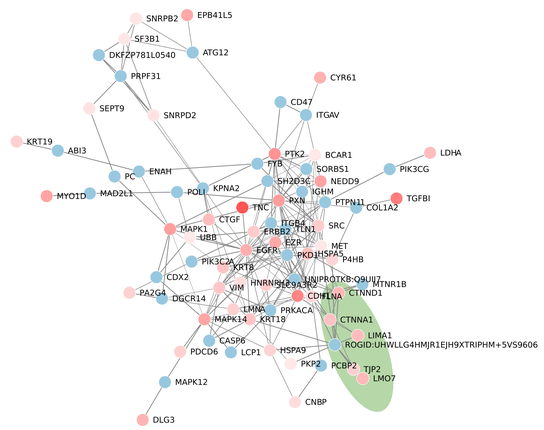
Problem setup

The LASSO does not always find the "right" model. In some cases, we can leverage side-information to help find the right model.

In our case, side-information is graphical structure among the features.



We can re-draw each instance as a labeling of the graph's nodes:



The "right" model is a "graph-sparse" model, i.e. a model with a sparse support which is connected on a graph. The goal is therefore to find the **most predictive connected subgraph**.
(possibly add the logistic loss function here?)

Our approach

function $\mathrm{GSLR}(X,\ y,\ G,\ s,\ \eta,\ k)$
  Let $f(X, y, \theta)$ be the logistic loss.
  $\hat{\theta}^0 \leftarrow 0$
  for $i \leftarrow 0, \ldots, k-1$ **do**
    $\tilde{\theta}^{i+1} \leftarrow \hat{\theta}^i - \eta \cdot \nabla f(X, y, \hat{\theta}^i)$
    $\hat{\theta}^{i+1} \leftarrow P_{G,s}(\tilde{\theta}^{i+1})$ ▷ Graph-sparse projection
  return $\hat{\theta}^k$

Main idea:

Sparse Projection Operator:
Given an arbitrary vector $p \in \mathbb{R}^d$, the projection operator $P_{G,s}$ returns a vector $q \in \mathbb{R}^d$ satisfying the following two properties:

- **Approximate projection:** The vector q is an approximate projection, i.e., instead of achieving the smallest distance to the input point p among points in the constraint set, the distance achieved by q is within a small constant factor:

$$\|p - q\|_2^2 \ \leq\ 2 \cdot \min_{q' \text{ is } (G,s)\text{-sparse}} \|p - q'\|_2^2 \,. \tag{1}$$

- **Approximate graph sparsity:** The support of the vector q forms a connected component of size at most $6s + 1$ in the graph G.

Graph-Sparse Projection through PCSF:
The Graph-Sparse projection operator $P_{G,s}$ is a carefully-tuned set of Prize-Collecting Steiner Forest instances.
The PCSF objective is to minimize:

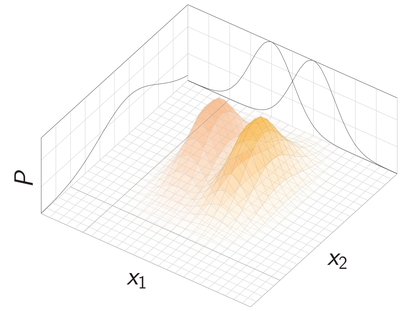
$$f(F) = \beta \sum_{v \notin V_F} p(v) + \sum_{e \in E_F} c(e) + \omega \cdot \kappa, \tag{2}$$

Intuitively, the goal is to pay as little edge cost to connect the highest prize nodes.
We project the parameter vector *theta* onto the graph by setting it as the prize for PCSF.

Experimental Setup

Since we don't have the ground truth subgraphs for the real data, we generate synthetic data by this procedure:

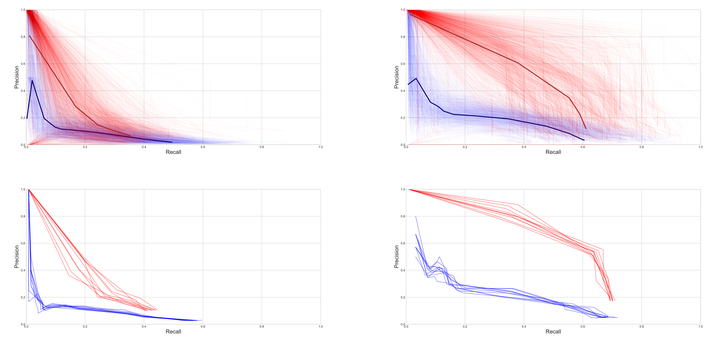
1. Determine μ and Σ from real data.
2. Sample from multivariate $\mathcal{N}(\vec{\mu}, \Sigma)$
3. Sample perturbation vector $\vec{x}$:
   scheme 1: $\vec{x_p} = \mathcal{N}(0, \sigma_p^2)$ if $p \in$ KEGG, 0 otherwise
   scheme 2: $\vec{x_p} = \mathcal{N}(\pm\sigma_p, \sigma_p^2)$ if $p \in$ KEGG, 0 otherwise
4. Translate "positive" samples by perturbation vector



Since we know the perturbation vector, we know the ground truth! We can then evaluate algorithms on the feature selection task.

Experimental Results

We benchmark our technique against the LASSO by how many of the truly "perturbed" features each uses in its support.



We then use our technique on real Ovarian Cancer data, and find that the support chosen by GSLR is qualitatively superior.