

Inverse Scale Space Method and Statistical Properties

Yuan Yao

HKUST

June 25, 2022

Acknowledgements

- Theory
 - *Stanley Osher, Wotao Yin* (UCLA)
 - *Feng Ruan* (Stanford & PKU)
 - *Jiechao Xiong, Chendi Huang, Xinwei Sun* (PKU)
 - *Yang Cao, Jinshan Zeng* (HKUST)
- Applications:
 - *Xinwei Sun* (Fudan/PKU)
 - *Qianqian Xu, Jiechao Xiong, Chendi Huang* (PKU)
 - *Yanwei Fu* (Fudan University)
 - *Chen Liu, Donghao Li* (Fudan and HKUST)
 - *Wenqi Zeng* (HKUST)
 - *Lingjing Hu* (BCMU)

1 Libra (R) and DensiLBI (Python)

- Libra: Linear/Logistic Regression, Ising graphical models
- DensiLBI: Deep structurally splitting Linearized Bregman Iteration

2 From LASSO to Inverse Scale Space

- LASSO and Bias
- Differential Inclusion of Inverse Scale Space
- Statistical Path Consistency with Early Stopping
- Large Scale Algorithm: Linearized Bregman Iteration (LBI)

3 Variable Splitting: Split LBI

- A Weaker Irrepresentable/Incoherence Condition
- Applications: Alzheimer's Disease, Ranking, Deep Learning
- Data Adaptive Early Stopping Rule: Split Knockoffs

4 Summary

Cran R package: Libra

<http://cran.r-project.org/web/packages/Libra/>



Libra: Linearized Bregman Algorithms for Generalized Linear Models

Efficient procedures for fitting the regularization path for linear, binomial, multinomial, Ising and Potts models with lasso, group lasso or column lasso(only for multinomial) penalty. The package uses Linearized Bregman Algorithm to solve the regularization path through iterations. Bregman Inverse Scale Space Differential Inclusion solver is also provided for linear model with lasso penalty.

Version: 1.5
 Depends: R (\geq 3.0), [mnl](#)
 Suggests: [lars](#), [MASS](#), [igraph](#)
 Published: 2016-02-17
 Author: Feng Ruan, Jiechao Xiong and Yuan Yao
 Maintainer: Jiechao Xiong <xiongjiechao@pku.edu.cn>
 License: [GPL-2](#)
 URL: <http://arxiv.org/abs/1406.7728>
 NeedsCompilation: yes
 SystemRequirements: GNU Scientific Library (GSL)
 CRAN checks: [Libra results](#)

Downloads:

Reference manual: [Libra.pdf](#)
 Package source: [Libra_1.5.tar.gz](#)
 Windows binaries:
 r-devel: [Libra_1.5.zip](#), r-release: [Libra_1.5.zip](#), r-oldrel: [Libra_1.5.zip](#)
 OS X Snow Leopard binaries: r-release: [Libra_1.5.tgz](#), r-oldrel: not available
 OS X Mavericks binaries: r-release: [Libra_1.5.tgz](#)
 Old sources: [Libra archive](#)



Libra (1.6) currently includes

Sparse statistical models:

- linear regression: ISS (differential inclusion), LB
- logistic regression (binomial, multinomial): LB
- graphical models (Gaussian, Ising, Potts): LB

Two types of regularization:

- LASSO: l_1 -norm penalty
- Group LASSO: $l_2 - l_1$ penalty

Libra computes regularization paths via Linearized Bregman Iteration (LB)

for $\theta_0 = z_0 = \mathbf{0}$ and $k \in \mathbb{N}$,

$$z_{k+1} = z_k - \frac{\alpha_k}{n} \sum_{i=1}^n \nabla_{\theta} \ell(x_i, \theta_k) \quad (1a)$$

$$\theta_{k+1} = \kappa \cdot \text{prox}_{\|\cdot\|_*}(z_{k+1}) \quad (1b)$$

where

- $\ell(x, \theta)$ is the *loss* function to minimize
- $\text{prox}_{\|\cdot\|_*}(z) := \arg \min_u \left(\frac{1}{2} \|u - z\|^2 + \|u\|_* \right)$
- $\alpha_k > 0$ is step-size
- $\kappa > 0$ while $\alpha_k \kappa \|\nabla_{\theta}^2 \hat{\mathbb{E}} \ell(x, \theta)\| < 2$
- as simple as ISTA (easy to parallel implementation), yet different limit dynamics

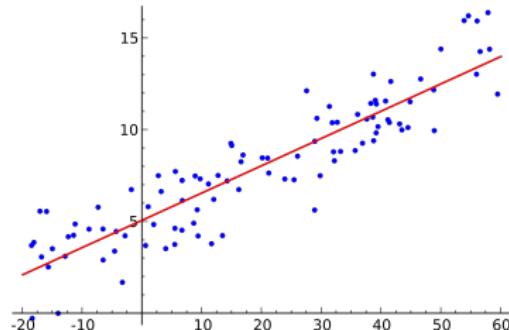
Linear Regression

Linear Regression:

$$y = X\beta + \epsilon$$

β is sparse or group sparse, with two types of penalty:

- "ungrouped": $\sum_i |\beta_i|$
- "grouped": $\sum_g \sqrt{\sum_{g_i=g} \beta_i^2}$

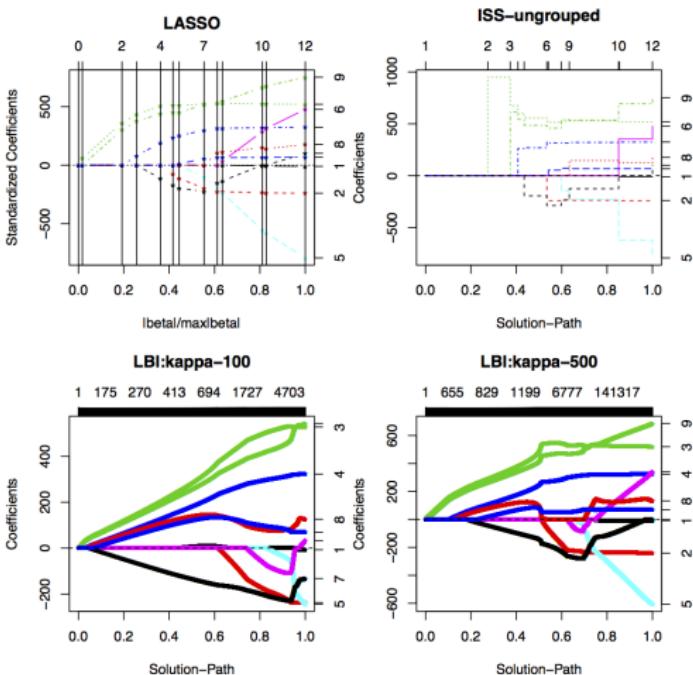


Linear Regression Example: Diabetes Data

```
data('diabetes')
attributes(x)
##dim
# [1] 442 10
##dimnames[[2]]
# [1] "age" "sex" "bmi" "map" "tc"   "ldl" "hdl" "tch" "ltg" "glu"

lassopath = lars(x,y)
isspath = iss(x,y)
lb(x,y,kappa=100,alpha=0.005,family="gaussian",group="ungrouped",
    intercept=FALSE,normalize=FALSE)
lb(x,y,kappa=500,alpha=0.001,family="gaussian",group="ungrouped",
    intercept=FALSE,normalize=FALSE)
```

LB generates iterative regularization paths



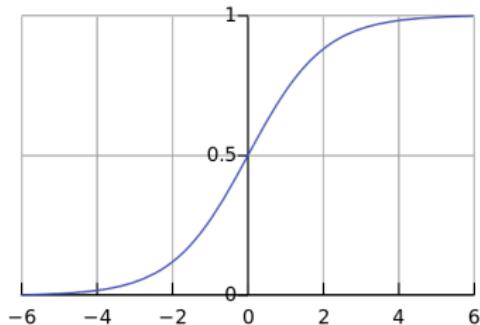
Logistic Regression

Logistic Regression:

$$\log \frac{P(y = 1|X)}{P(y = -1|X)} = X\beta \Leftrightarrow P(y = 1|X) = \frac{e^{X\beta}}{1 + e^{X\beta}} =: \sigma(X\beta)$$

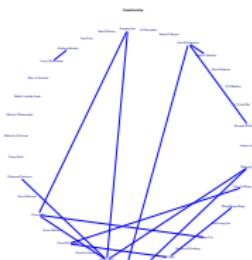
β is sparse or group sparse, with two types of penalty:

- "ungrouped": $\sum_i |\beta_i|$
- "grouped": $\sum_g \sqrt{\sum_{g_i=g} \beta_i^2}$



Example: Publications of COPSS Award Winners

- dataset is provided by Prof. [Jiashun Jin](#) @CMU
- 3248 papers by 3607 authors between 2003 and the first quarter of 2012 from:
 - the Annals of Statistics, Journal of the American Statistical Association, Biometrika and Journal of the Royal Statistical Society Series B
- a subset of 382 papers by 35 COPSS award winners
- Question: can we **model the coauthorship structure to predict the out-of-sample behavior?**



A logistic regression path with early stopping regularization

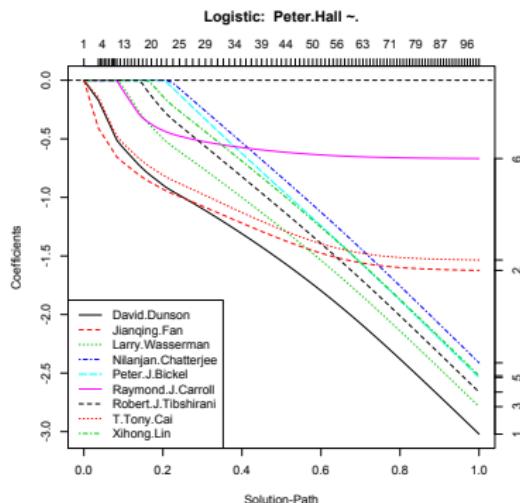


Figure: Peter Hall vs. other COPSS award winners in sparse logistic regression [papers from AoS/JASA/Biometrika/JRSSB, 2003-2012]: true coauthors are merely Tony Cai, R.J. Carroll, and J. Fan

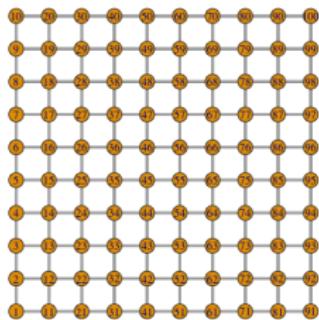
Sparse Ising Model

All models are wrong, but some are useful ([George Box](#)):

$$P(x_1, \dots, x_p) \sim \exp \left(\sum_i H_i x_i + \sum_{i,j} J_{ij} x_i x_j \right)$$

- Ising model: $x_i = 1$ if author i appears in a paper, otherwise 0
- H_i describes the mean publication rate of author i
- J_{ij} describes the interactions between author i and j
 - $J_{ij} > 0$: author i and j collaborate more often than others
 - $J_{ij} < 0$: author i and j collaborate less frequently than others
 - sparsity: $J_{ij} = 0$ mostly, a model of collaboration network
 - learned by maximum composite conditional likelihood with LB

Early stopping against overfitting in sparse Ising model learning



a true Ising model of 2-D grid

a movie of LB path

Application: Sparse Ising Model of COPSS Award Winners

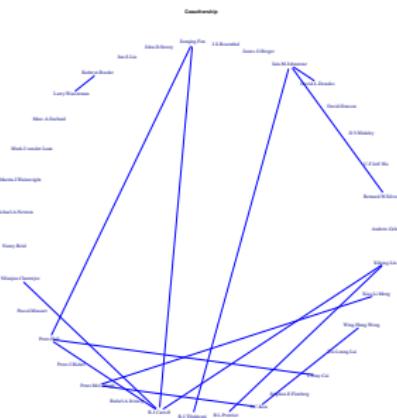


Figure: Left: LB path of Ising Model learning; Right: coauthorship network of existing data. Typically COPSS winners do not like working together; Peter Hall (1951-2016) is the hub of statisticians, like Erdős for mathematicians

Sparse Filters Learned on MNIST

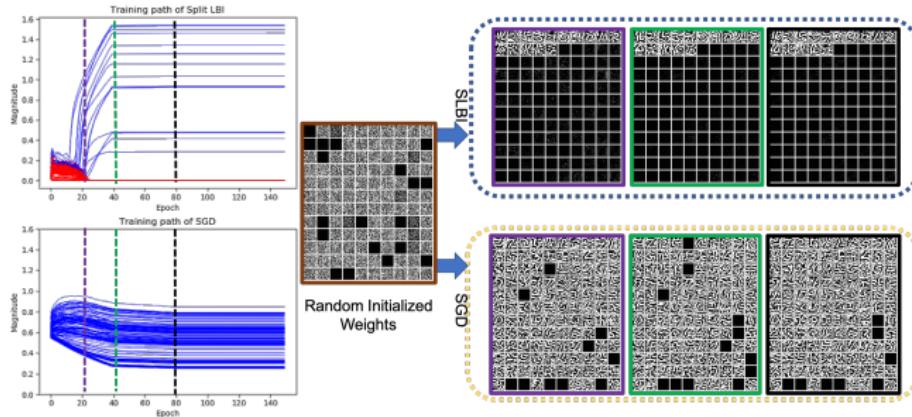


Figure: Visualization of solution path and filter patterns in the third convolutional layer (i.e., conv.c5) of LetNet-5, trained on MNIST. It shows that our SplitLBI enjoys a sparse selection of filters without sacrificing accuracy. From Fu et al. DensiLBI, ICML 2020, <https://github.com/DensiLBI2020/DensiLBI>

Non-semantic Features Learned on ImageNet

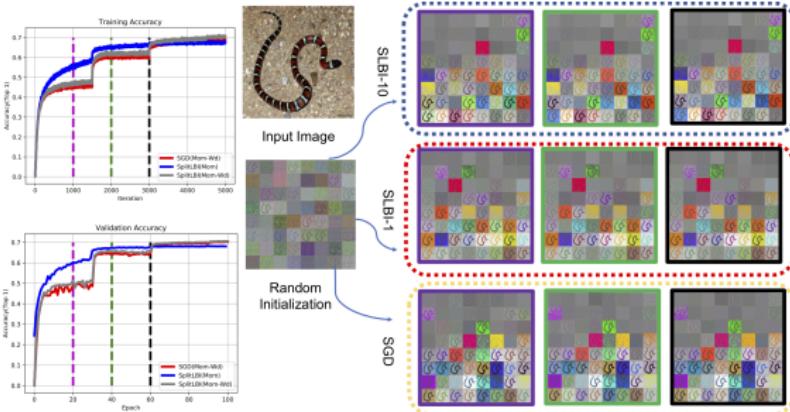


Figure: Visualization of the first convolutional layer filters of ResNet-18 trained on ImageNet-2012, where texture features are more important than colour ones. Given the input image and initial weights visualized in the middle, filter response gradients at 20 (purple), 40 (green), and 60 (black) epochs are visualized. SGD with Momentum (Mom) and Weight Decay (WD), is compared with SLBI.

How does it work?

A story behind these algorithms is as follows:

- The simple iterative algorithm shadows a particular kind of dynamics: **differential inclusions of inverse scale spaces**, as special cases of **Mirror Descent**, where important features are learned fast
- Simple discretized algorithm, amenable for parallel implementation
- Under nearly the same condition as LASSO, it reaches **model selection consistency with early stopping**
- but may incur **less bias** than LASSO
- Equipped with **variable splitting**, it **weakens** the conditions of generalized LASSO in feature selection

Sparse Linear Regression

Assume that $\beta^* \in \mathbb{R}^p$ is sparse and unknown. Consider recovering β^* from n linear measurements

$$y = X\beta^* + \epsilon, \quad y \in \mathbb{R}^n$$

where $\epsilon \sim \mathcal{N}(0, \sigma^2)$ is **noise**.

- **Basic Sparsity:** $S := \text{supp}(\beta^*)$ ($s = |S|$) and T be its complement.
 - X_S (X_T) be the columns of X with indices restricted on S (T)
 - X is n -by- p , with $p \gg n \geq s$.
- Generalized **Structural/Transformational Sparsity:** $\gamma^* = D\beta^*$ is sparse, where D is a linear transform (wavelet, gradient, etc.), $S = \text{supp}(\gamma^*)$
- *How to recover β^* (or γ^*) sparsity pattern (**sparsistency**) and estimate values with variations (**consistency**)?*

Best Possible in Basic Setting: The Oracle Estimator

Had God revealed S to us, the *oracle estimator* was the subset least square solution (MLE) with $\tilde{\beta}_T^* = 0$ and

$$\tilde{\beta}_S^* = \beta_S^* + \frac{1}{n} \Sigma_n^{-1} X_S^T \epsilon, \quad \text{where } \Sigma_n = \frac{1}{n} X_S^T X_S \quad (2)$$

“Oracle properties”

- **Model selection consistency:** $\text{supp}(\tilde{\beta}^*) = S$;
- **Normality:** $\tilde{\beta}_S^* \sim \mathcal{N}(\beta^*, \frac{\sigma^2}{n} \Sigma_n^{-1})$.

So $\tilde{\beta}^*$ is **unbiased**, i.e. $\mathbb{E}[\tilde{\beta}^*] = \beta^*$.

Recall LASSO

LASSO:

$$\min_{\beta} \|\beta\|_1 + \frac{t}{2n} \|y - X\beta\|_2^2.$$

optimality condition:

$$\frac{\rho_t}{t} = \frac{1}{n} X^T (y - X\beta_t), \quad (3a)$$

$$\rho_t \in \partial \|\beta_t\|_1, \quad (3b)$$

where $\lambda = 1/t$ is often used in literature.

- Chen-Donoho-Saunders'1996 (BPDN)
- Tibshirani'1996 (LASSO)

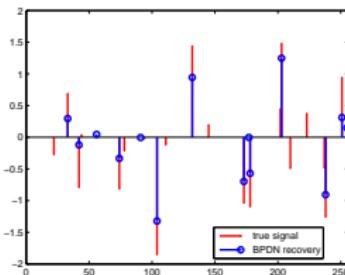
The Bias of LASSO

LASSO is **biased**, i.e. $\mathbb{E}(\hat{\beta}) \neq \beta^*$

- e.g. $X = Id$, $n = p = 1$, LASSO is soft-thresholding

$$\hat{\beta}_\tau = \begin{cases} 0, & \text{if } \tau < 1/\tilde{\beta}^*; \\ \tilde{\beta}^* - \frac{1}{\tau}, & \text{otherwise,} \end{cases}$$

- e.g. $n = 100$, $p = 256$, $X_{ij} \sim \mathcal{N}(0, 1)$, $\epsilon_i \sim \mathcal{N}(0, 0.1)$



True vs LASSO (t hand-tuned,
courtesy of Wotao Yin)

LASSO Estimator is Biased at Path Consistency

Even when the following **path consistency** (conditions given by [Zhao-Yu'06](#), [Zou'06](#), [Yuan-Lin'07](#), [Wainwright'09](#), etc.) is reached at τ_n :

$$\exists \tau_n \in (0, \infty) \text{ s.t. } \text{supp}(\hat{\beta}_{\tau_n}) = S,$$

LASSO estimate is biased away from the oracle estimator

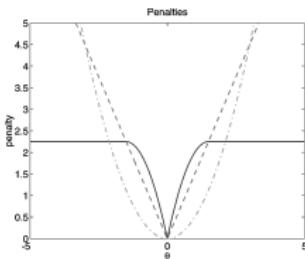
$$(\hat{\beta}_{\tau_n})_S = \tilde{\beta}_S^* - \frac{1}{\tau_n} \Sigma_{n,S}^{-1} \text{sign}(\beta_S^*), \quad \tau_n > 0.$$

How to remove the bias and return the Oracle Estimator?

Nonconvex Regularization?

- To reduce bias, **non-convex** regularization was proposed ([Fan-Li's SCAD](#), [Zhang's MPLUS](#), [Zou's Adaptive LASSO](#), l_q ($q < 1$), etc.)

$$\min_{\beta} \sum_i p(|\beta_i|) + \frac{t}{2n} \|y - X\beta\|_2^2.$$



- Yet it is generally hard to locate the **global optimizer**
- Any other simple scheme?*

New Idea

- LASSO:

$$\min_{\beta} \|\beta\|_1 + \frac{t}{2n} \|y - X\beta\|_2^2.$$

New Idea

- LASSO:

$$\min_{\beta} \|\beta\|_1 + \frac{t}{2n} \|y - X\beta\|_2^2.$$

- KKT optimality condition:

$$\Rightarrow \rho_t = \frac{1}{n} X^T (y - X\beta_t) t$$

New Idea

- LASSO:

$$\min_{\beta} \|\beta\|_1 + \frac{t}{2n} \|y - X\beta\|_2^2.$$

- KKT optimality condition:

$$\Rightarrow \rho_t = \frac{1}{n} X^T (y - X\beta_t) t$$

- Taking derivative (assuming differentiability) w.r.t. t

$$\Rightarrow \dot{\rho}_t = \frac{1}{n} X^T (y - X(\dot{\beta}_t t + \beta_t)), \quad \rho_t \in \partial \|\beta_t\|_1$$

New Idea

- LASSO:

$$\min_{\beta} \|\beta\|_1 + \frac{t}{2n} \|y - X\beta\|_2^2.$$

- KKT optimality condition:

$$\Rightarrow \rho_t = \frac{1}{n} X^T (y - X\beta_t)t$$

- Taking derivative (assuming differentiability) w.r.t. t

$$\Rightarrow \dot{\rho}_t = \frac{1}{n} X^T (y - X(\dot{\beta}_t t + \beta_t)), \quad \rho_t \in \partial \|\beta_t\|_1$$

- Assuming sign-consistency in a neighborhood of τ_n ,

for $i \in S$, $\rho_{\tau_n}(i) = \text{sign}(\beta^*(i)) \in \pm 1 \Rightarrow \dot{\rho}_{\tau_n}(i) = 0$,

$$\Rightarrow \dot{\beta}_{\tau_n} \tau_n + \beta_{\tau_n} = \tilde{\beta}^*$$

New Idea

- LASSO:

$$\min_{\beta} \|\beta\|_1 + \frac{t}{2n} \|y - X\beta\|_2^2.$$

- KKT optimality condition:

$$\Rightarrow \rho_t = \frac{1}{n} X^T (y - X\beta_t)t$$

- Taking derivative (assuming differentiability) w.r.t. t

$$\Rightarrow \dot{\rho}_t = \frac{1}{n} X^T (y - X(\dot{\beta}_t t + \beta_t)), \quad \rho_t \in \partial \|\beta_t\|_1$$

- Assuming sign-consistency in a neighborhood of τ_n ,

for $i \in S$, $\rho_{\tau_n}(i) = \text{sign}(\beta^*(i)) \in \pm 1 \Rightarrow \dot{\rho}_{\tau_n}(i) = 0$,

$$\Rightarrow \dot{\beta}_{\tau_n} \tau_n + \beta_{\tau_n} = \tilde{\beta}^*$$

- Equivalently, the blue part removes bias of LASSO automatically

$$\beta_{\tau_n}^{lasso} = \tilde{\beta}^* - \frac{1}{\tau_n} \Sigma_n^{-1} \text{sign}(\beta^*) \Rightarrow \dot{\beta}_{\tau_n}^{lasso} \tau_n + \beta_{\tau_n}^{lasso} = \tilde{\beta}^* (\text{oracle})!$$

Differential Inclusion: Inverse Scaled Spaces (ISS)

Differential inclusion replacing $\dot{\beta}_{\tau_n}^{lasso} \tau_n + \beta_{\tau_n}^{lasso}$ by β_t

$$\dot{\rho}_t = \frac{1}{n} X^T (y - X\beta_t), \quad (4a)$$

$$\rho_t \in \partial \|\beta_t\|_1. \quad (4b)$$

starting at $t = 0$ and $\rho(0) = \beta(0) = \mathbf{0}$.

- Replace ρ/t in LASSO KKT by $d\rho/dt$

$$\frac{\rho_t}{t} = \frac{1}{n} X^T (y - X\beta_t)$$

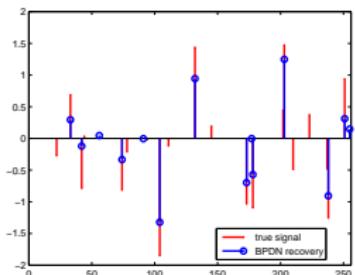
- Burger-Gilboa-Osher-Xu'06 (in image recovery it recovers the objects in an inverse-scale order as t increases (larger objects appear in β_t first))

Examples

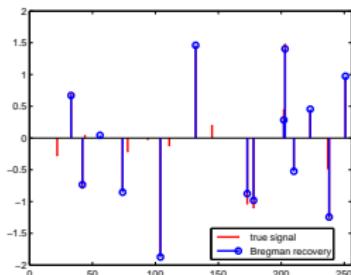
- e.g. $X = Id$, $n = p = 1$, hard-thresholding

$$\beta_\tau = \begin{cases} 0, & \text{if } \tau < 1/(\tilde{\beta}^*); \\ \tilde{\beta}^*, & \text{otherwise,} \end{cases}$$

- the same example shown before (figures by courtesy of Wotao Yin)



True vs LASSO



True vs ISS

Solution Path: Sequential Restricted Maximum Likelihood Estimate

- ρ_t is piece-wise linear in t ,

$$\rho_t = \rho_{t_k} + \frac{t - t_k}{n} X^T (y - X\beta_{t_k}), \quad t \in [t_k, t_{k+1})$$

where $t_{k+1} = \sup\{t > t_k : \rho_{t_k} + \frac{t-t_k}{n} X^T (y - X\beta_{t_k}) \in \partial \|\beta_{t_k}\|_1\}$

- β_t is piece-wise constant in t : $\beta_t = \beta_{t_k}$ for $t \in [t_k, t_{k+1})$ and $\beta_{t_{k+1}}$ is the sequential restricted Maximum Likelihood Estimate by solving nonnegative least square (Burger et al.'13; Osher et al.'16)

$$\begin{aligned} \beta_{t_{k+1}} &= \arg \min_{\beta} \|y - X\beta\|_2^2 \\ \text{subject to } & (\rho_{t_{k+1}})_i \beta_i \geq 0 \quad \forall i \in S_{k+1}, \\ & \beta_j = 0 \quad \forall j \in T_{k+1}. \end{aligned} \tag{5}$$

- Note: Sign consistency $\rho_t = \text{sign}(\beta^*) \Rightarrow \beta_t = \tilde{\beta}^*$ the oracle estimator

Example: Regularization Paths of LASSO vs. ISS

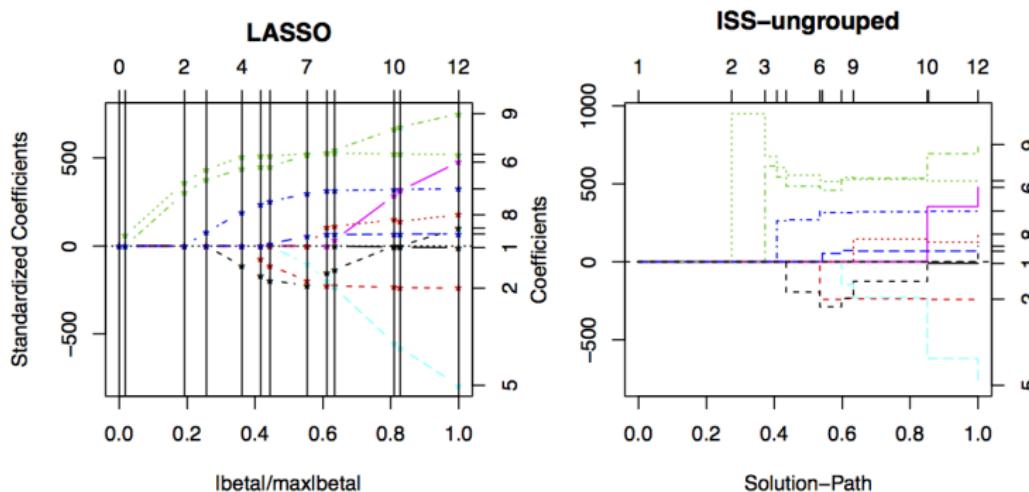


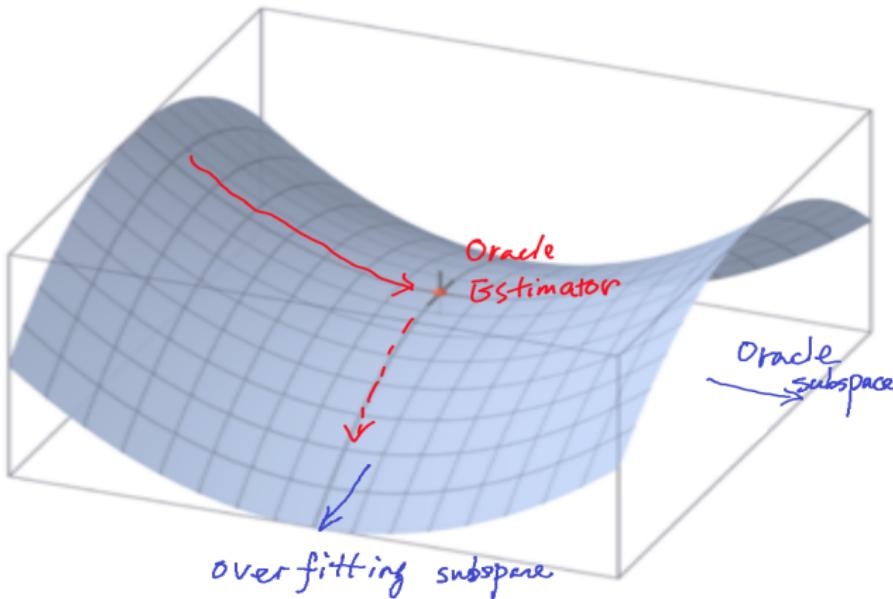
Figure: Diabetes data ([Efron et al.'04](#)) and regularization paths are different, yet bearing similarities on the order of parameters being nonzero

How does it work? A Path Consistency Theory

Our aim is to show that under nearly the **same** conditions for sign-consistency of LASSO, there exists points on their paths $(\beta(t), \rho(t))_{t \geq 0}$, which are

- **sparse**
- **sign-consistent** (the same sparsity pattern of nonzeros as true signal)
- **the oracle estimator** which is unbiased, better than the LASSO estimate.
- **Early stopping** regularization is necessary to prevent overfitting noise!

Intuition



History: two traditions of regularizations

- Penalty functions
 - ℓ_2 : Ridge regression/Tikhonov regularization: $\frac{1}{n} \sum_{i=1}^n \ell(y_i, x_i^T \beta) + \lambda \|\beta\|_2^2$
 - ℓ_1 (sparse): Basis Pursuit/LASSO (ISTA): $\frac{1}{n} \sum_{i=1}^n \ell(y_i, x_i^T \beta) + \lambda \|\beta\|_1^2$
- Early stopping of dynamic regularization paths
 - ℓ_2 -equivalent: Landweber iterations/gradient descent/ ℓ_2 -Boost

$$\frac{d\beta_t}{dt} = -\frac{1}{n} \sum_{i=1}^n \nabla_\beta \ell(y_i, x_i^T \beta), \quad \beta_t = \nabla \left\{ \frac{1}{2} \|\beta_t\|^2 \right\}$$

- ℓ_1 (sparse)-equiv.: Orthogonal Matching Pursuit, **Linearized Bregman Iteration** (sparse Mirror Descent) (not ISTA! – later)

$$\frac{d\rho_t}{dt} = -\frac{1}{n} \sum_{i=1}^n \nabla_\beta \ell(y_i, x_i^T \beta), \quad \rho_t \in \partial \|\beta_t\|_1$$

Assumptions

(A1) **Restricted Strongly Convex**: $\exists \gamma \in (0, 1]$,

$$\frac{1}{n} X_S^T X_S \geq \gamma I$$

(A2) **Incoherence/Irrepresentable Condition**: $\exists \eta \in (0, 1)$,

$$\left\| \frac{1}{n} X_T^T X_S^\dagger \right\|_\infty = \left\| \frac{1}{n} X_T^T X_S \left(\frac{1}{n} X_S^T X_S \right)^{-1} \right\|_\infty \leq 1 - \eta$$

- "Irrepresentable" means that one can not represent (regress) column vectors in X_T by covariates in X_S .
- The incoherence/irrepresentable condition is used independently in [Tropp'04](#), [Yuan-Lin'05](#), [Zhao-Yu'06](#), and [Zou'06](#), [Wainwright'09](#), etc.

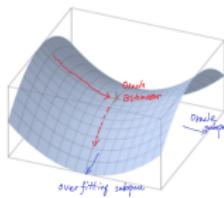
Understanding the Dynamics

ISS as **restricted gradient descent**:

$$\dot{\beta}_t = -\nabla L(\beta_t) = \frac{1}{n} X^T (y - X\beta_t), \quad \rho_t \in \partial \|\beta_t\|_1$$

such that

- **incoherence condition** and **strong signals** ensure it firstly evolves on index set S to reduce the loss
- **strongly convex** in subspace restricted on index set $S \Rightarrow$ fast decay in loss
- **early stopping** after all strong signals are detected, before picking up the noise



Path Consistency

Theorem (Osher-Ruan-Xiong-Y.-Yin'2016)

Assume (A1) and (A2). Define an early stopping time

$$\bar{\tau} := \frac{\eta}{2\sigma} \sqrt{\frac{n}{\log p}} \left(\max_{j \in T} \|X_j\| \right)^{-1},$$

and the smallest magnitude $\beta_{\min}^* = \min(|\beta_i^*| : i \in S)$. Then

- **No-false-positive:** for all $t \leq \bar{\tau}$, the path has no-false-positive with high probability, $\text{supp}(\beta(t)) \subseteq S$;
- **Consistency:** moreover if the signal is strong enough such that

$$\beta_{\min}^* \geq \left(\frac{4\sigma}{\gamma^{1/2}} \vee \frac{8\sigma(2 + \log s)(\max_{j \in T} \|X_j\|)}{\gamma\eta} \right) \sqrt{\frac{\log p}{n}},$$

there is $\tau \leq \bar{\tau}$ such that solution path $\beta(t) = \tilde{\beta}^*$ for every $t \in [\tau, \bar{\tau}]$.

Note: equivalent to LASSO with $\lambda^* = 1/\bar{\tau}$ (Wainwright'09) up to $\log s$.

Large scale algorithm: Linearized Bregman Iteration

Damped Dynamics: continuous solution path

$$\dot{\rho}_t + \frac{1}{\kappa} \dot{\beta}_t = \frac{1}{n} X^T (y - X\beta_t), \quad \rho_t \in \partial \|\beta_t\|_1. \quad (6)$$

Linearized Bregman Iteration as forward Euler discretization proposed even earlier than ISS dynamics (Osher-Burger-Goldfarb-Xu-Yin'05, Yin-Osher-Goldfarb-Darbon'08): for $\rho_k \in \partial \|\beta_k\|_1$,

$$\rho_{k+1} + \frac{1}{\kappa} \beta_{k+1} = \rho_k + \frac{1}{\kappa} \beta_k + \frac{\alpha_k}{n} X^T (y - X\beta_k), \quad (7)$$

where

- Damping factor: $\kappa > 0$
- Step size: $\alpha_k > 0$ s.t. $\alpha_k \kappa \|\Sigma_n\| \leq 2$
- Moreau Decomposition: $z_k := \rho_k + \frac{1}{\kappa} \beta_k \Leftrightarrow \beta_k = \kappa \cdot \text{Shrink}(z_k, 1)$

Comparison with ISTA

Linearized Bregman (LB) iteration:

$$z_{t+1} = z_t - \alpha_t X^T (\kappa X \textcolor{red}{Shrink}(z_t, 1) - y)$$

which is not **ISTA**:

$$z_{t+1} = \textcolor{red}{Shrink}(z_t - \alpha_t X^T (Xz_t - y), \lambda).$$

Comparison:

- **ISTA:**

- as $t \rightarrow \infty$ solves **LASSO**: $\frac{1}{n} \|y - X\beta\|_2^2 + \lambda \|\beta\|_1$
- **parallel run** ISTA with $\{\lambda_k\}$ for LASSO regularization paths

- **LB: a single run** generates the whole regularization path at same cost of ISTA-LASSO estimator for a fixed regularization

LBI generates regularization paths

$n = 200$, $p = 100$, $S = \{1, \dots, 30\}$, $x_i \sim N(0, \Sigma_p)$ ($\sigma_{ij} = 1/(3p)$ for $i \neq j$ and 1 otherwise)

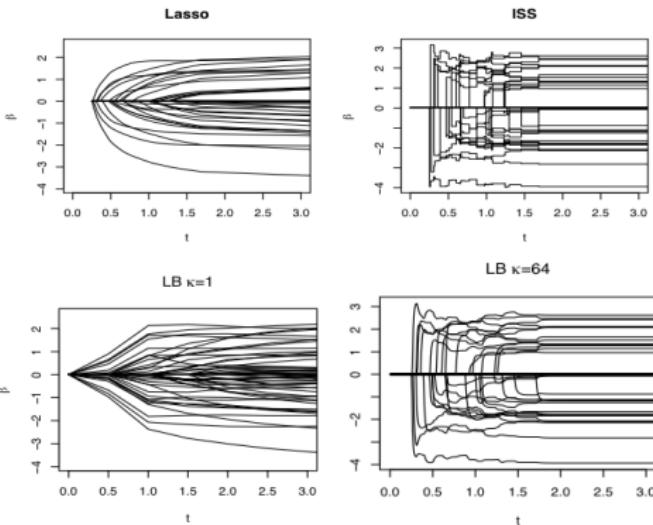
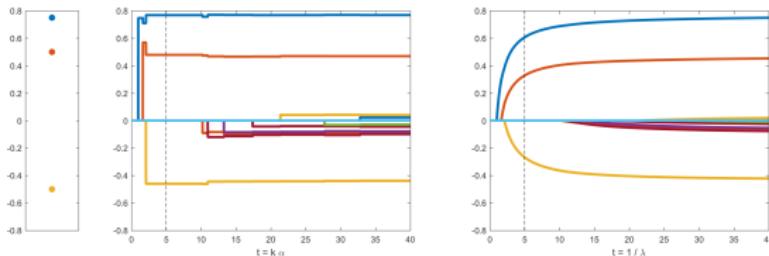


Figure: As $\kappa \rightarrow \infty$, LB paths have a limit as piecewise-constant ISS path

Accuracy: LB may be less biased than LASSO



- Left shows (the magnitudes of) nonzero entries of β^* .
- Middle shows the regularization path of LB.
- Right shows the regularization path of LASSO vs. $t = 1/\lambda$.

Path Consistency in Discrete Setting

Theorem (Osher-Ruan-Xiong-Y.-Yin'2016)

Assume that κ is large enough and α is small enough, with $\kappa\alpha\|X_S^*X_S\| < 2$,

$$\bar{\tau} := \frac{(1 - B/\kappa\eta)\eta}{2\sigma} \sqrt{\frac{n}{\log p}} \left(\max_{j \in T} \|X_j\| \right)^{-1}$$

$$\beta_{\max}^* + 2\sigma \sqrt{\frac{\log p}{\gamma n}} + \frac{\|X\beta^*\|_2 + 2s\sqrt{\log n}}{n\sqrt{\gamma}} \triangleq B \leq \kappa\eta,$$

then all the results for ISS can be extended to the discrete algorithm.

Note: it recovers the previous theorem as $\kappa \rightarrow \infty$ and $\alpha \rightarrow 0$, so LB can be less biased than LASSO.

General Loss and Regularizer

$$\dot{\eta}_t = -\frac{\kappa_0}{n} \sum_{i=1}^n \nabla_{\eta} \ell(x_i, \theta_t, \eta_t) \quad (8a)$$

$$\dot{\rho}_t + \frac{\dot{\theta}_t}{\kappa_1} = -\frac{1}{n} \sum_{i=1}^n \nabla_{\theta} \ell(x_i, \theta_t, \eta_t) \quad (8b)$$

$$\rho_t \in \partial \|\theta_t\|_* \quad (8c)$$

where

- $\ell(x_i, \theta)$ is a loss function: negative logarithmic likelihood, non-convex loss (neural networks), etc.
- $\|\theta_t\|_*$ is the Minkowski-functional (gauge) of dictionary convex hulls:

$$\|\theta\|_* := \inf\{\lambda \geq 0 : \theta \in \lambda K\}, \quad K \text{ is a symmetric convex hull of } \{a_i\}$$

- it can be generalized to non-convex regularizers

Linearized Bregman Iteration Algorithms

Differential inclusion (8) admits the following Euler Forward discretization

$$\eta_{t+1} = \eta_t - \frac{\alpha_k \kappa_0}{n} \sum_{i=1}^n \nabla_{\eta} \ell(x_i, \theta_t, \eta_t) \quad (9a)$$

$$z_{t+1} = z_t - \frac{\alpha_k}{n} \sum_{i=1}^n \nabla_{\theta} \ell(x_i, \theta_t, \eta_t) \quad (9b)$$

$$\theta_{t+1} = \kappa_1 \cdot \text{prox}_{\|\cdot\|_*}(z_{t+1}) \quad (9c)$$

where (9c) is given by Moreau Decomposition with

$$\text{prox}_{\|\cdot\|_*}(z_t) = \arg \min_x \frac{1}{2} \|x - z_t\|^2 + \|x\|_*,$$

and

- $\alpha_k > 0$ is step-size while $\alpha_k \kappa_i \|\nabla_{\theta}^2 \hat{\mathbb{E}} \ell(x, \theta)\| < 2$
- as simple as ISTA, easy to parallel implementation

More reference

- **Logistic Regression:** loss – conditional likelihood, regularizer – ℓ_1
([Shi-Yin-Osher-Sajda'10, Huang-Yao'18](#))
- **Graphical Models** (Gaussian/Ising/Potts Model): loss – likelihood, composite conditional likelihood, regularizer – ℓ_1 and group ℓ_1
([Huang-Yao'18](#))
- **Fused LASSO/TV**: split Bregman with composite ℓ_2 loss and ℓ_1 gauge
([Osher-Burger-Goldfarb-Xu-Yin'06, Burger-Gilboa-Osher-Xu'06, Yin-Osher-Goldfarb-Darbon'08, Huang-Sun-Xiong-Yao'16](#))
- **Matrix Completion/Regression:** gauge – the matrix nuclear norm
([Cai-Candès-Shen'10](#))

Structural or Transformational Sparsity

Structural/Transformational Sparse Regression:

$$y = X\beta^* + \epsilon, \quad (10a)$$

$$\gamma^* = D\beta^*, \quad (10b)$$

where

$$S = \text{supp}(\gamma^*), \quad s := |S| \ll p.$$

Split LB vs. Generalized LASSO

Structural Sparse Regression:

- **Generalized LASSO** (genlasso):

$$\arg \min_{\beta} \left(\frac{1}{2n} \|y - X\beta\|_2^2 + \lambda \|\mathcal{D}\beta\|_1 \right). \quad (11)$$

- **Split LBI**: Loss that splits prediction vs. sparsity control

$$\ell(\beta, \gamma) := \frac{1}{2n} \|y - X\beta\|_2^2 + \frac{1}{2\nu} \|\gamma - \mathcal{D}\beta\|_2^2 \quad (\nu > 0). \quad (12)$$

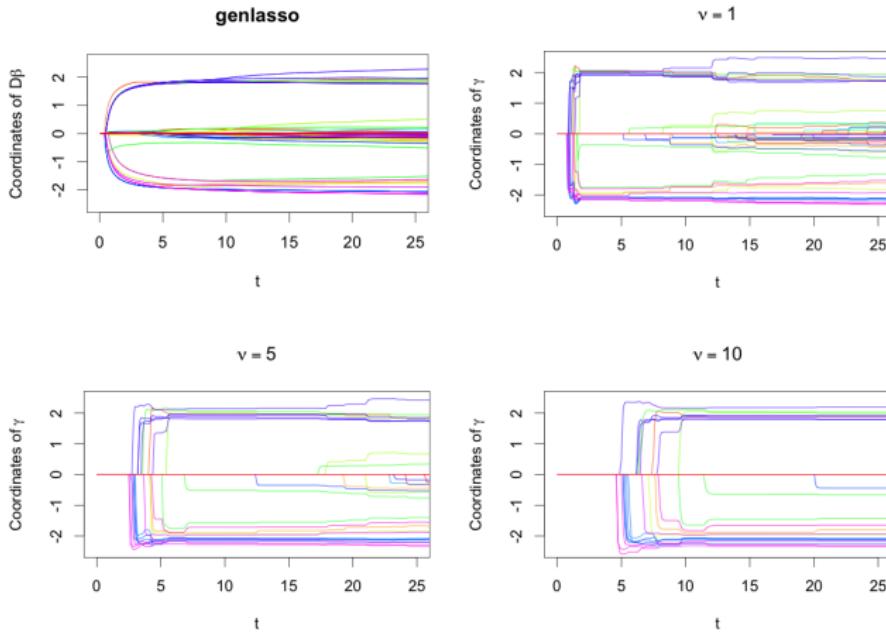
Algorithm [Huang-Sun-Xiong-Y. 2016] :

$$\beta_{k+1} = \beta_k - \kappa \alpha \nabla_{\beta} \ell(\beta_k, \gamma_k), \quad (13a)$$

$$z_{k+1} = z_k - \alpha \nabla_{\gamma} \ell(\beta_k, \gamma_k), \quad (13b)$$

$$\gamma_{k+1} = \kappa \cdot \text{prox}_{\|\cdot\|_1}(z_{k+1}), \quad (13c)$$

Split LBI vs. Generalized LASSO paths



Split LB may beat Generalized LASSO in Model Selection

| genlasso | Split LBI | | | genlasso | Split LBI | | |
|------------------|------------------|------------------|--------------------------------|------------------|------------------|------------------|--------------------------------|
| | $\nu = 1$ | $\nu = 5$ | $\nu = 10$ | | $\nu = 1$ | $\nu = 5$ | $\nu = 10$ |
| .9426 (.0390) | .9845 (.0185) | .9969 (.0065) | .9982 (.0043) | .9705 (.0212) | .9955 (.0056) | .9996 (.0014) | .9998 (.0009) |

- Example: $n = p = 50$, $X \in \mathbb{R}^{n \times p}$ with $X_j \sim N(0, I_p)$, $\epsilon \sim N(0, I_n)$
- (Left) $D = I$ (LASSO vs. Split LB)
- (Right) 1-D fused (generalized) LASSO vs. [Split LB](#) (next page).
- In terms of Area Under the ROC Curve (AUC), LB has less false discoveries than genlasso
- Why? Split LB may need **weaker** irrepresentable conditions than generalized LASSO...

Structural Sparsity Assumptions

- Define $\Sigma(\nu) := (I - D(\nu X^* X + D^T D)^\dagger D^T)/\nu$.
- **Assumption 1:** Restricted Strong Convexity (RSC).

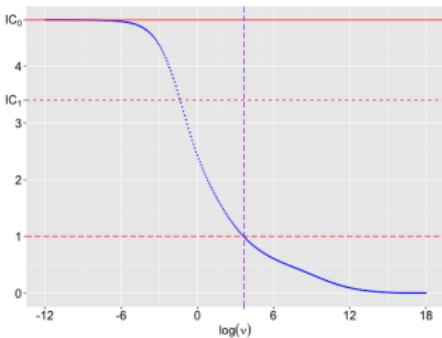
$$\Sigma_{S,S}(\nu) \succeq \lambda \cdot I. \quad (14)$$

- **Assumption 2:** Irrepresentable Condition (IRR).

$$\text{IRR}(\nu) := \|\Sigma_{S^c, S}(\nu) \cdot \Sigma_{S,S}^{-1}(\nu)\|_\infty \leq 1 - \eta. \quad (15)$$

- $\nu \rightarrow 0$: RSC and IRR above reduce to the **necessary and sufficient** for consistency of genlasso (Vaiter'13, LeeSunTay'13).
- $\nu \neq 0$: by allowing variable splitting in proximity, IRR above can be **weaker** than literature, bringing **better** variable selection consistency than genlasso (observed before)!

Split LB improves Irrepresentable Condition (Huang-Sun-Xiong-Y.'16)



Theorem (Huang-Sun-Xiong-Y.'2016)

- $IC_0 \geq IC_1$.
- $IRR(\nu) \rightarrow IC_0 (\nu \rightarrow 0)$.
- $IRR(\nu) \rightarrow C (\nu \rightarrow \infty)$ with $C = 0 \iff \ker(X) \subseteq \ker(D_S)$.

Remark: Identifiable Conditions (IC)

- Let the columns of W form an orthogonal basis of $\ker(D_{S^c})$.

$$\Omega^S := \left(D_{S^c}^\dagger \right)^T \left(X^* X W \left(W^T X^* X W \right)^\dagger W^T - I \right) D_S^T, \quad (16)$$

$$\text{IC}_0 := \left\| \Omega^S \right\|_\infty, \quad \text{IC}_1 := \min_{u \in \ker(D_{S^c})} \left\| \Omega^S \text{sign}(D_S \beta^*) - u \right\|_\infty. \quad (17)$$

- The sign consistency of genlasso has been proved, under $\text{IC}_1 < 1$ [Vaiter et al. 2013].
- The sign consistency of Split LBI is proved under $\text{IRR}(\nu) < 1$ [Huang-Sun-Xiong-Y.'2016].
- As $\text{IRR}(\nu) < \text{IC}_1$ when ν grows, our IRR is easier to meet.

Consistency

Theorem (Huang-Sun-Xiong-Y.'2016)

Under RSC and IRR, with large κ and small δ , there exists K such that with high probability, the following properties hold.

- *No-false-positive property:* γ_k ($k \leq K$) has no false-positive, i.e.
 $\text{supp}(\gamma_k) \subseteq S = \text{supp}(\gamma^*)$.
- *Sign consistency of γ_K :* If $\gamma_{\min}^* := \min(|\gamma_j^*| : j \in S)$ (the minimal signal) is not weak, then $\text{supp}(\gamma_K) = \text{supp}(\gamma^*)$.
- *ℓ_2 consistency of γ_K :* $\|\gamma_K - \gamma^*\|_2 \leq C_1 \sqrt{s \log m/n}$.
- *ℓ_2 “consistency” of β_K :* $\|\beta_K - \beta^*\|_2 \leq C_2 \sqrt{s \log m/n} + C_3 \nu$.

- Issues due to variable splitting (despite benefit on IRR):
 - $D\beta_K$ does not follow the sparsity pattern of $\gamma^* = D\beta^*$.
 - β_K incurs an additional loss $C_3 \nu$ ($\nu \sim \sqrt{s \log m/n}$ minimax optimal).

Consistency

Theorem (Huang-Sun-Xiong-Y.'2016)

Define

$$\tilde{\beta}_k := \text{Proj}_{\ker(D_{S_k^c})}(\beta_k) \quad (S_k = \text{supp}(\gamma_k)) \quad (18)$$

Under RSC and IRR, with large κ and small δ , there exists K such that with high probability, the following properties hold, if γ_{\min}^* is not weak.

- *Sign consistency of $D\tilde{\beta}_K$* : $\text{supp}(D\tilde{\beta}_K) = \text{supp}(D\beta^*)$.
- *ℓ_2 consistency of $\tilde{\beta}_K$* : $\left\| \tilde{\beta}_K - \beta^* \right\|_2 \leq C_4 \sqrt{s \log m/n}$.

Application: Partial Order of Basketball Teams

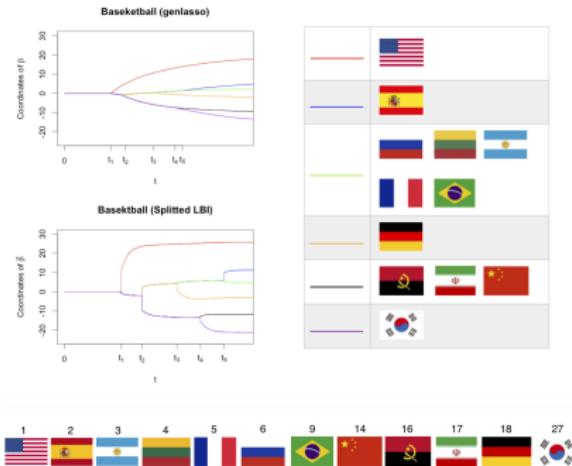


Figure: Partial order ranking for basketball teams. Top left shows $\{\beta_\lambda\}$ ($t = 1/\lambda$) by genlasso and $\tilde{\beta}_k$ ($t = k\alpha$) by Split LBI. Top right shows the same grouping result just passing t_5 . Bottom is the FIBA ranking of all teams.

Application: Alzheimer's Disease Detection

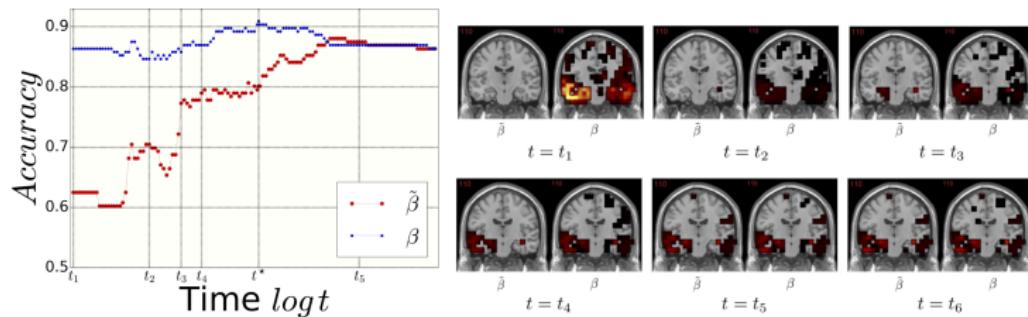


Figure: [Sun-Hu-Y.-Wang'17] A split of prediction (β) vs. interpretability ($\tilde{\beta}$): $\tilde{\beta}$ corresponds to the degenerate voxels interpretable for AD, while β additionally leverages the procedure bias to improve the prediction

Application: Sparse Subnets in Early Stopping (Lottery Tickets)

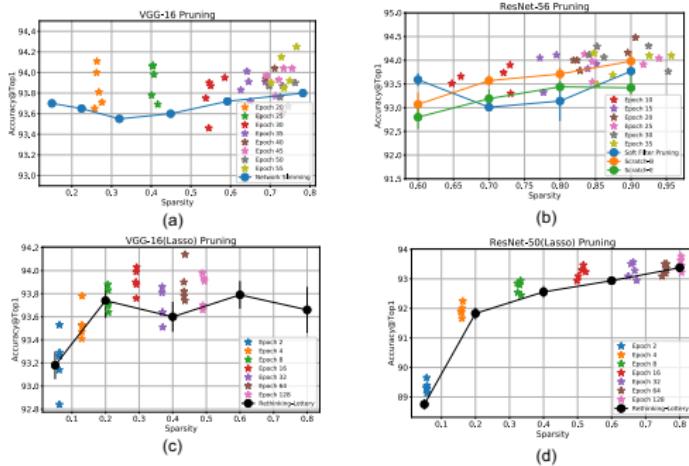


Figure: SplitLBI with early stopping finds sparse subnets whose test accuracies (stars) after retrain are comparable or even better than the baselines (Network Slimming, Soft-Filter Pruning, Scratch-B, Scratch-E, and “Rethinking-Lottery” as reported in Rethink the Value of Pruning. Sparse filters of VGG-16 and ResNet-56 are show in (a) and (b), while sparse weights of VGG-16 and ResNet-50 are shown in (c) and (d).

Application: Deep Learning

- A neural network of I -layer is defined as

$$\Phi_W(x) = \sigma_I \left(W^I \sigma_{I-1} \left(W^{I-1} \cdots \sigma_1 \left(W^1 x \right) \right) \right) \quad (19)$$

where $W = \{W^i\}_{i=1}^I$, and σ_i is the nonlinear activation function of the i -th layer.

- Loss functions on training samples

$$\widehat{\mathcal{L}}_n(W) = \frac{1}{n} \sum_{i=1}^n \ell(y_i, \Phi_W(x_i)). \quad (20)$$

- Augmented loss function

$$\bar{\mathcal{L}}(W, \Gamma) = \widehat{\mathcal{L}}_n(W) + \frac{1}{2\nu} \|W - \Gamma\|_F^2, \quad (21)$$

with $\nu > 0$ controlling the gap admitted between W and Γ .

DassiLBI

- DassiLBI (**D**eep **S**tructurally **S**plitting **L**inearized **Bregman **I**terations):**

$$W_{k+1} = W_k - \kappa \alpha_k \cdot \nabla_W \bar{\mathcal{L}}(W_k, \Gamma_k), \quad (22a)$$

$$V_{k+1} = V_k - \alpha_k \cdot \nabla_V \bar{\mathcal{L}}(W_k, \Gamma_k), \quad (22b)$$

$$\Gamma_{k+1} = \kappa \cdot \text{Prox}_{\Omega_\lambda}(V_{k+1}), \quad (22c)$$

where $V_0 = \Gamma_0 = 0$, W_0 can be small random numbers such as Gaussian initialization.

- The proximal map above controls the sparsity of Γ ,

$$\text{Prox}_{\Omega_\lambda}(V) = \arg \min_{\Gamma} \left\{ \frac{1}{2} \|\Gamma - V\|_2^2 + \Omega_\lambda(\Gamma) \right\}, \quad (23)$$

Structural Sparsity for Convolutional Networks

Group Lasso:

$$\Omega_1(\Gamma) = \sum_g \|\Gamma^g\|_2,$$

where $\|\Gamma^g\|_2 = \sqrt{\sum_{i=1}^{|\Gamma^g|} (\Gamma_i^g)^2}$. Thus the proximal map reduces

$$\Gamma^g = \kappa \cdot \max(0, 1 - 1/\|V^g\|_2) V^g.$$

- For a convolutional filter selection, $\Gamma^g = \Gamma^g(c_{in}, c_{out}, \text{size})$ denote the convolutional filters with kernel size `size`, input channels c_{in} and output channels c_{out} , set $g = c_{out}$.
- For weight selection, g is an element that reduces to the Lasso penalty.

Controlling the False Discovery Rates via Knockoffs

- The early stopping rule $\bar{\tau}$ in theory (for power) is unknown in applications;
- Knockoff ([Barber-Candès \(2015\)](#)) gives a data adaptive early stopping rule with FDR control:

$$\text{FDR} = \mathbb{E} \left[\frac{\#\text{false discoveris}}{1 \vee \#\text{discoveries}} \right].$$

- The method makes a fake Knockoff feature \tilde{X} that serve as the control group:

$$\tilde{X}^T \tilde{X} = X^T X, \quad X^T \tilde{X} = X^T X - \text{diag}(s),$$

where s is some proper non-negative vector. The Knockoff features mimic the original feature X , but decoupled with the original feature.

Knockoff Methods

Knockoff is extended to:

- 1 Group sparse and multi-task regression model (Dai-Barber, ICML 2016).
- 2 Huber's robust regression with LBI (Xu et al. ICML 2016).
- 3 High dimensional setting (Barber-Candes 2019).
- 4 Model-X Knockoff for random design (Candes et al. 2016).
- 5 Deep Knockoff for nonparametric random designs (Romano et al. 2019).
- 6 Recently, Split Knockoffs for structural sparsity ([Cao-Sun-Y. 2021](#)).

Summary

We have seen:

- The limit of Linearized Bregman iteration follows **differential inclusions of inverse scale space**, where significant features emerge earlier on solution paths
- It renders the **unbiased Oracle Estimator** under sign-consistency
- Sign consistency under nearly the **same** condition as LASSO
 - Restricted Strongly Convex + Irrepresentable Condition
- **Split** extension: sign consistency under a **weaker** condition than generalized LASSO
 - under a provably weaker Irrepresentable Condition
- **Early stopping** regularization is exploited against overfitting under noise

A Renaissance of Boosting as restricted gradient descent ...

Some Reference

- Osher, Ruan, Xiong, Yao, and Yin, "Sparse Recovery via Differential Equations", *Applied and Computational Harmonic Analysis*, 2016
- Xiong, Ruan, and Yao, "A Tutorial on Libra: R package for Linearized Bregman Algorithms in High Dimensional Statistics", *Handbook of Big Data Analytics*, Eds. by Wolfgang Karl Härdle, Henry Horng-Shing Lu, and Xiaotong Shen, Springer, 2017. <https://arxiv.org/abs/1604.05910>
- Xu, Xiong, Cao, and Yao, "False Discovery Rate Control and Statistical Quality Assessment of Annotators in Crowdsourced Ranking", *ICML 2016*, arXiv:1604.05910
- Huang, Sun, Xiong, and Yao, "Split LBI: an iterative regularization path with structural sparsity", *NIPS 2016*, <https://github.com/yuany-pku/split-lbi>
- Sun, Hu, Wang, and Yao, "GSplit LBI: taming the procedure bias in neuroimaging for disease prediction", *MICCAI 2017*
- Huang and Yao, "A Unified Dynamic Approach to Sparse Model Selection", *AISTATS 2018*
- Huang, Sun, Xiong, and Yao, "Boosting with Structural Sparsity: A Differential Inclusion Approach", *Applied and Computational Harmonic Analysis*, 48(1):1-45, 2020, arXiv: 1704.04833
- Qianqian Xu, Xinwei Sun, Zhiyong Yang, Qingming Huang, and Yuan Yao. "iSplit LBI: Individualized Partial Ranking with Ties via Split LBI". *NeurIPS 2019*, arXiv:1910.05905
- Yanwei Fu, Chen Liu, Donghao Li, Xinwei Sun, Jinshan Zeng, and Yuan Yao. DessiLBI: Exploring Structural Sparsity of Deep Neural Networks via Differential Inclusion Paths. *ICML 2020*, arXiv:2007.02010.
- Yanwei Fu, Chen Liu, Donghao Li, Zuyuan Zhong, Xinwei Sun, Jinshan Zeng, and Yuan Yao. Exploring Structural Sparsity of Deep Networks via Inverse Scale Spaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2022.
- Yang Cao, Xinwei Sun, Yuan Yao. "Controlling the False Discovery Rate in Structural Sparsity: Split Knockoffs". arXiv:2103.16159.
- R package: <http://cran.r-project.org/web/packages/Libra/index.html>
- Pytorch package for deep learning: <https://github.com/DessiLBI2020/DessiLBI>