Machine Learning 2

6. Sparse Linear Models

Lars Schmidt-Thieme

Information Systems and Machine Learning Lab (ISMLL)
Institute for Computer Science
University of Hildesheim, Germany

Syllabus

		A. Advanced Supervised Learning
Fri. 12.4.	(1)	A.1 Generalized Linear Models
Fri. 26.4.	(2)	A.2 Gaussian Processes
Fri. 3.5.	(3)	A.2b Gaussian Processes (ctd.)
Fri. 10.5.	(4)	A.3 Advanced Support Vector Machines
		B. Ensembles
Fri. 17.5.	(5)	B.1 Stacking
Fri. 24.5.	(6)	B.2 Boosting
Fri. 31.5.	(7)	B.3 Mixtures of Experts
Fri. 7.6.	(8)	(ctd.)
Fri. 14.6.	—	— Pentecoste Break —
		C. Sparse Models
Fri. 21.6.	(9)	C.1 Homotopy and Least Angle Regression
Fri. 28.6.	(10)	C.2 Proximal Gradients
Fri. 29.6.	(11)	C.3 Laplace Priors & C.4 Automatic Relevance Determination
		D. Complex Predictors
Fri. 6.7.	(12)	D.1 Latent Dirichlet Allocation (LDA)
Fri. 12.7.	(13)	Q & A

Outline

1. Homotopy Methods: Least Angle Regression

Interlude: A note on Model complexity

2. Proximal Gradient Methods

3. Laplace Priors (Bayesian Lasso)

Outline

1. Homotopy Methods: Least Angle Regression

Interlude: A note on Model complexity

2. Proximal Gradient Methods

3. Laplace Priors (Bayesian Lasso)

Sparse Models so far

- Variable subset selection
  - forward search, backward search
- L1 regularization / Lasso
  - Coordinate descent (shooting algorithm)

L1 Regularization

$$\begin{aligned} \text{min. } & f(\hat{\theta}) := \ell(y, \hat{y}(\hat{\theta}, X)) + \lambda ||\hat{\theta}||_1 \\ & \hat{\theta} \in \mathbb{R}^P \end{aligned}$$

is equivalent to

$$\begin{aligned} \text{min. } & f(\hat{\theta}) := \ell(y, \hat{y}(\hat{\theta}, X)) \\ & ||\hat{\theta}||_1 \leq B \\ & \hat{\theta} \in \mathbb{R}^P \end{aligned}$$

with

$$B := ||\hat{\theta}^*||_1$$

L1 Regularization / Equivalence

More generally, given

$$x^* := \arg\min_x f(x) + \lambda\, g(x), \quad \lambda \geq 0 \tag{1}$$

$$\tilde{x} := \arg\min_{x: g(x) \leq g(x^*)} f(x) \tag{2}$$

then

$$x^* = \tilde{x}$$

because

$$f(\tilde{x}) \underset{(2)}{\leq} f(x^*) \underset{(1)}{\leq} f(\tilde{x}) + \lambda\, (\underbrace{g(\tilde{x}) - g(x^*)}_{\leq 0}) \leq f(\tilde{x})$$

$$\leadsto \quad f(\tilde{x}) = f(x^*) \quad \leadsto \quad \tilde{x} = x^*$$

assuming x^* is unique.

Homotopy Methods

$$\text{min. } f(\hat{\theta}) := \ell(y, \hat{y}(\hat{\theta}, X)) \; + \lambda ||\hat{\theta}||_1$$

or equivalently

$$\text{min. } f(\hat{\theta}) := \ell(y, \hat{y}(\hat{\theta}, X))$$
$$||\hat{\theta}||_1 \leq B$$

- start with a solution for large $\lambda^{(0)}$ (or equiv. $B^{(0)} := 0$)
  - then $\hat{\theta}^{(0)} = 0$.
- stepwise decrease $\lambda^{(t)}$ (or equiv. increase $B^{(t)}$)
  - learn $\hat{\theta}^{(t)}$ starting from $\hat{\theta}^{(t-1)}$ (**warmstart**).

Homotopy Methods

For homotopy to work,

1. the parameters as function of λ

$$\hat{\theta}(\lambda) := \arg\min_{\hat{\theta}} \ell(y, \hat{y}(\hat{\theta}, X)) + \lambda ||\hat{\theta}||_1$$

   must be continuous, i.e.,
   - $\hat{y}$ must be continuous in $\hat{\theta}$ and
   - ℓ be continuous in $\hat{y}$.
2. the steps in $\lambda^{(t)}$ must be small enough.

Most simple model: linear regression

- model $\hat{y}(\hat{\theta}, X) := X\hat{\theta}$
- loss $\ell(y, \hat{y}) := ||y - \hat{y}||_2^2$

Advantage: can find optimal $\lambda^{(t)}$ sequence analytically! (actually $B^{(t)}$)

Least Angle Regression (LAR)

in step t:

1. choose the predictors with largest correlation with the residuum (**active predictors**):

$$C^{(t-1)} := X^T(y - \hat{y}^{(t-1)})$$

$$A^{(t)} := \arg\max_m |C_m^{(t-1)}| \qquad \text{(a set!)}$$

2. regress these predictors on the residuum:

$$X^{(t)} := X_{\cdot, A^{(t)}}$$

$$\hat{\gamma}^{(t)} := \arg\min_\gamma ||y - \hat{y}^{(t-1)} - X^{(t)}\gamma||_2$$

$$= (X^{(t)\,T} X^{(t)})^{-1} X^{(t)\,T}(y - \hat{y}^{(t-1)})$$

3. update parameters in this direction:

$$\hat{\beta}^{(t)} := \hat{\beta}^{(t-1)} + \alpha \Delta^{(t)} \hat{\gamma}^{(t)}$$

Note: $\Delta^{(t)}_{m_k,k} := 1$ for $A^{(t)} := \{m_1, m_2, \ldots, m_K\}$, $\Delta^{(t)}_{m,k} := 0$ otherwise.

Least Angle Regression (LAR): step length

Residuum correlations after the update

$$\begin{aligned} C^{(t)} =& X^T(y - \hat{y}^{(t)}) = X^T(y - X\hat{\beta}^{(t)}) = X^T(y - X(\hat{\beta}^{(t-1)} + \alpha\Delta^{(t)}\hat{\gamma}^{(t)})) \\ =& C^{(t-1)} - \alpha X^T X \Delta^{(t)}\hat{\gamma}^{(t)} \\ =& C^{(t-1)} - \alpha X^T X^{(t)}\hat{\gamma}^{(t)} \end{aligned}$$

are uniformly reduced for active predictors:

$$C^{(t)}|_{A^{(t)}} = C^{(t-1)}|_{A^{(t)}} - \alpha X^{(t)\,T} X^{(t)}\hat{\gamma}^{(t)} = (1-\alpha)C^{(t-1)}|_{A^{(t)}}$$

and may also change for non-active predictors:

$$C_m^{(t)} = C_m^{(t-1)} - \alpha X_{\cdot,m}^T X^{(t)}\hat{\gamma}^{(t)}$$

Note: Maybe a mistake somewhere here. Final formula for α differs from the one in the paper.

Least Angle Regression (LAR): step length (2/2)

Reduce until another predictor has same (max) residuum correlation:

$$C_m^{(t)} = C_m^{(t-1)} - \alpha X_{\cdot,m}^T X^{(t)} \hat{\gamma}^{(t)} \overset{!}{=} (1-\alpha) C_{\max}^{(t-1)}$$

$$\alpha = \frac{C_{\max}^{(t-1)} - C_m^{(t-1)}}{C_{\max}^{(t-1)} - X_{\cdot,m}^T X^{(t)} \hat{\gamma}^{(t)}}$$

or for negative correlations:

$$C_m^{(t)} = C_m^{(t-1)} - \alpha X_{\cdot,m}^T X^{(t)} \hat{\gamma}^{(t)} \overset{!}{=} -(1-\alpha) C_{\max}^{(t-1)}$$

$$\alpha = \frac{C_{\max}^{(t-1)} + C_m^{(t-1)}}{C_{\max}^{(t-1)} + X_{\cdot,m}^T X^{(t)} \hat{\gamma}^{(t)}}$$

yielding

$$\alpha := \text{minpos}\{ \frac{C_{\max}^{(t-1)} - C_m^{(t-1)}}{C_{\max}^{(t-1)} - X_{\cdot,m}^T X^{(t)} \hat{\gamma}^{(t)}}, \frac{C_{\max}^{(t-1)} + C_m^{(t-1)}}{C_{\max}^{(t-1)} + X_{\cdot,m}^T X^{(t)} \hat{\gamma}^{(t)}}$$

$$\mid m \in \{1, \ldots, M\} \setminus A^{(t)}\}, \quad \text{minpos}(X) := \min\{x \in X \mid x > 0\}$$

Example

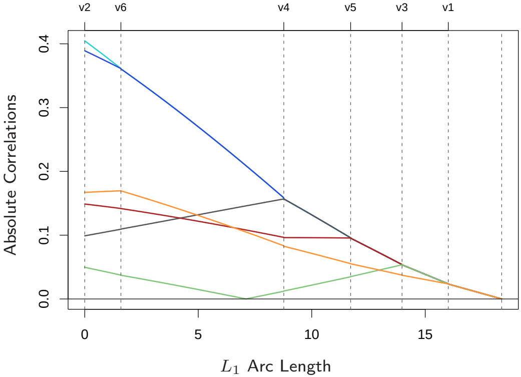
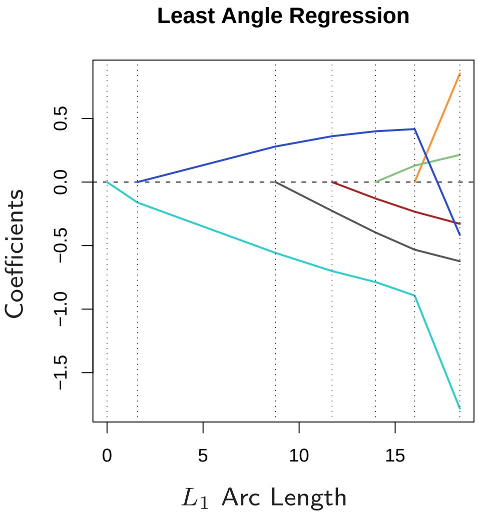


FIGURE 3.14. *Progression of the absolute correlations during each step of the LAR procedure, using a simulated data set with six predictors. The labels at the top of the plot indicate which variables enter the active set at each step. The step length are measured in units of L_1 arc length.*

[HTFF05, p. 75]

Example

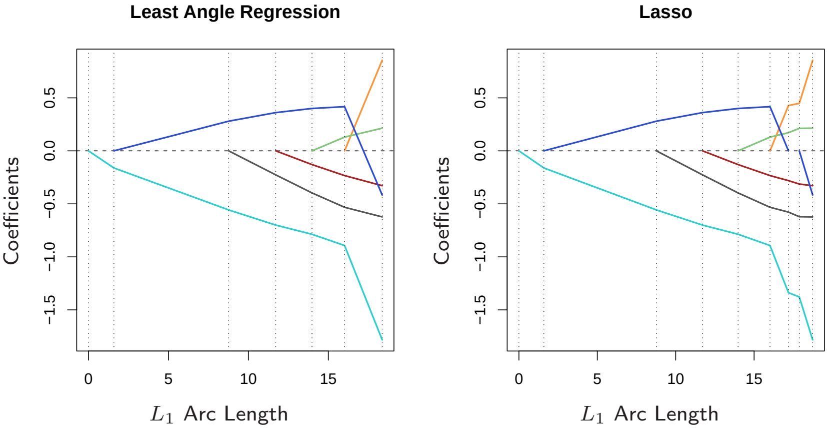


[HTFF05, p. 75]

Remarks

- algorithm can be used two ways:
  1. Estimate parameters for **all** λ (**regularization path**)
  2. Estimate parameters for **a specific** λ (Homotopy method)
     - start with large $\lambda^{(0)}$, stop once $\lambda^{(t)} < \lambda$ reached.
- not straightforward to extend from regression to GLMs
- LAR can be modified to solve the LASSO:
  - if the parameter $\beta_m^{(t)}$ for an active predictor m becomes 0 or changes sign, drop it from the active set.
- also called Least Angle Regression and Shrinkage (LARS)

Example



[HTFF05, p. 75]

Outline

1. Homotopy Methods: Least Angle Regression

Interlude: A note on Model complexity

2. Proximal Gradient Methods

3. Laplace Priors (Bayesian Lasso)

Model Complexity, Bias & Variance

Example (Linear models)

- $\hat{y}(x) = \beta_1 \cdot x$
- $\hat{y}(x) = (\beta_1 + \beta_2 + \ldots + \beta_K) \cdot x$

Both models have the same bias and variance! $\rightsquigarrow$ redundant parameters!

Model Complexity, Bias & Variance

Example (Linear models)

- $\hat{y}(x) = \beta_1 \cdot x$
- $\hat{y}(x) = (\beta_1 + \beta_2 + \ldots + \beta_K) \cdot x$

Both models have the same bias and variance! $\rightsquigarrow$ redundant parameters!

Example (1-parameter model)

- $\hat{y}(x) = \sin(\theta x)$

Can achieve 100% accuracy on any finite 1D binary classification dataset.
$\longrightarrow$ A single real number can store an infinite amount of information!

Model Complexity, Bias & Variance

Example (Linear models)

- $\hat{y}(x) = \beta_1 \cdot x$
- $\hat{y}(x) = (\beta_1 + \beta_2 + \ldots + \beta_K) \cdot x$

Both models have the same bias and variance! $\rightsquigarrow$ redundant parameters!

Example (1-parameter model)

- $\hat{y}(x) = \sin(\theta x)$

Can achieve 100% accuracy on any finite 1D binary classification dataset.
$\longrightarrow$ A single real number can store an infinite amount of information!

Example (Neural Network)

- Network 1: vanilla MLP
- Network 2: sparse Network with skip connections

Network 2 is more complex when both have same amount of parameters!

Measures of Model Complexity

- Parameter Counting
  - only really works when comparing models with the same architecture
  - even then not guaranteed to be useful

Measures of Model Complexity

- Parameter Counting
  - only really works when comparing models with the same architecture
  - even then not guaranteed to be useful
- Information Criteria (e.g. BIC, AIC)
  - Both very crude tools (lots of approximations used in derivation)
  - Both ignorant about the model architecture

Measures of Model Complexity

- Parameter Counting
  - only really works when comparing models with the same architecture
  - even then not guaranteed to be useful
- Information Criteria (e.g. BIC, AIC)
  - Both very crude tools (lots of approximations used in derivation)
  - Both ignorant about the model architecture
- VC-dimension
  - "What is size the the smallest binary classification problem that the model cannot solve."

Measures of Model Complexity

- Parameter Counting
  - only really works when comparing models with the same architecture
  - even then not guaranteed to be useful
- Information Criteria (e.g. BIC, AIC)
  - Both very crude tools (lots of approximations used in derivation)
  - Both ignorant about the model architecture
- VC-dimension
  - "What is size the the smallest binary classification problem that the model cannot solve."
- Rademacher Complexity
  - "How good can the model simulate noise."

Measures of Model Complexity

- Parameter Counting
  - only really works when comparing models with the same architecture
  - even then not guaranteed to be useful
- Information Criteria (e.g. BIC, AIC)
  - Both very crude tools (lots of approximations used in derivation)
  - Both ignorant about the model architecture
- VC-dimension
  - "What is size the the smallest binary classification problem that the model cannot solve."
- Rademacher Complexity
  - "How good can the model simulate noise."
- Kolmogorov Complexity & Minimum Description Length
  - "What is the minimal size of a program that implements the model."
  - **uncomputable!**

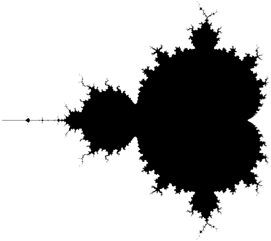
Kolmogorov Complexity - Mandelbrot Fractal

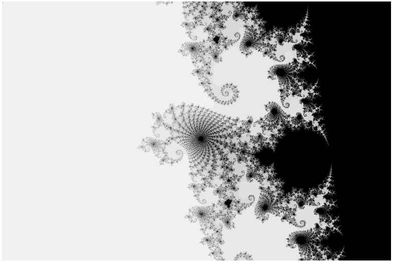
Generated by a simple formula:
Does the iteration

$$z_{k+1} = z_k^2 + c \quad z_0 = 0$$

diverge? (with $z, c \in \mathbb{C}$)

- Yes: c belongs to class 1 (white)
- No: c belongs to class 0 (black)





images: wikipedia.org

Kolmogorov Complexity - Mandelbrot Fractal

Generated by a simple formula:

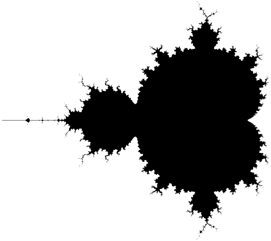
Does the iteration

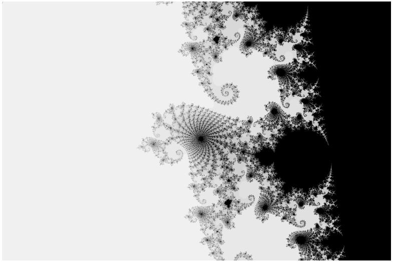
$$z_{k+1} = z_k^2 + c \quad z_0 = 0$$

diverge? (with $z, c \in \mathbb{C}$)

- Yes: c belongs to class 1 (white)
- No: c belongs to class 0 (black)

Very simple rules lead to incredible complexity.





images: wikipedia.org

Kolmogorov Complexity - Mandelbrot Fractal

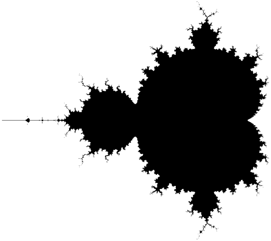
Generated by a simple formula:

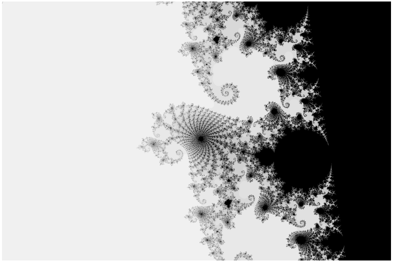
Does the iteration

$$z_{k+1} = z_k^2 + c \quad z_0 = 0$$

diverge? (with $z, c \in \mathbb{C}$)

- Yes: c belongs to class 1 (white)
- No: c belongs to class 0 (black)





Very simple rules lead to incredible complexity.

It would be very hard to reconstruct the rules, if we only know the image. In fact, in general it is impossible! $\leftrightsquigarrow$ uncomputability

images: wikipedia.org

Outline

1. Homotopy Methods: Least Angle Regression

Interlude: A note on Model complexity

2. Proximal Gradient Methods

3. Laplace Priors (Bayesian Lasso)

Regularized

We want to compute models

$$\theta^* = \arg\min_\theta \underbrace{L(\theta)}_{\text{Loss}} + \underbrace{R(\theta)}_{\text{Regularization}}$$

Even when R is not differentiable, e.g.

- $R(\theta) = \|\theta\|_1$ (L^1 regularization, LASSO)
- $R(\theta) = I_C(\theta) = \begin{cases} 0 : & \theta \in C \\ \infty : & \theta \notin C \end{cases}$ (hard constraint)

Regularized

We want to compute models

$$\theta^* = \arg\min_\theta \underbrace{L(\theta)}_{\text{Loss}} + \underbrace{R(\theta)}_{\text{Regularization}}$$

Even when R is not differentiable, e.g.

- $R(\theta) = \|\theta\|_1$ (L^1 regularization, LASSO)
- $R(\theta) = I_C(\theta) = \begin{cases} 0: & \theta \in C \\ \infty: & \theta \notin C \end{cases}$ (hard constraint)

Observation: For simple loss functions, we can sometimes compute θ^* analytically

$$\arg\min_\theta \frac{1}{2}\|\theta - y\|_2^2 + \lambda\|\theta\|_1 = \text{soft}(y, \lambda)$$

Proximal Problem

- find x with minimal f **in a vicinity of a given x^0**:

$$\text{prox}_f(x^0) := \arg\min_x f(x) + \frac{1}{2}||x - x^0||_2^2$$

Proximal Problem

- find x with minimal f **in a vicinity of a given x^0**:

$$\text{prox}_f(x^0) := \arg\min_x f(x) + \frac{1}{2}||x - x^0||_2^2$$

Can be solved analytically for some typical (possibly non-differentiable) regularization functions:

- $f := \lambda||x||_2^2$: $\qquad \text{prox}_f(x^0) = \frac{1}{2\lambda + 1} x^0$

Proximal Problem

- find x with minimal f **in a vicinity of a given** x^0:

$$\text{prox}_f(x^0) := \arg\min_x f(x) + \frac{1}{2}||x - x^0||_2^2$$

Can be solved analytically for some typical (possibly non-differentiable) regularization functions:

- $f := \lambda ||x||_2^2$: $\quad \text{prox}_f(x^0) = \frac{1}{2\lambda + 1} x^0$
- $f := \lambda ||x||_1$:

$$\text{prox}_f(x^0) = \text{soft}(x^0, \lambda) := (\text{soft}(x_n^0, \lambda))_{n=1,\ldots,N}$$

$$\text{soft}(z, \lambda) := \text{sign}(z)(|z| - \lambda)_0$$

Proximal Problem

- find x with minimal f **in a vicinity of a given x^0**:

$$\operatorname{prox}_f(x^0) := \arg\min_x f(x) + \frac{1}{2}||x - x^0||_2^2$$

Can be solved analytically for some typical (possibly non-differentiable) regularization functions:

- $f := \lambda||x||_2^2$: $\quad \operatorname{prox}_f(x^0) = \frac{1}{2\lambda + 1} x^0$
- $f := \lambda||x||_1$:

$$\operatorname{prox}_f(x^0) = \operatorname{soft}(x^0, \lambda) := (\operatorname{soft}(x_n^0, \lambda))_{n=1,\ldots,N}$$
$$\operatorname{soft}(z, \lambda) := \operatorname{sign}(z)(|z| - \lambda)_0$$

- $f := \lambda||x||_0$:

$$\operatorname{prox}_f(x^0) = \operatorname{hard}(x^0, \lambda) := (\operatorname{hard}(x_n^0, \lambda))_{n=1,\ldots,N},$$
$$\operatorname{hard}(z, \lambda) := \delta(|z| \geq \lambda)\, z$$

More Analytical Solutions for the Proximal Problem

- find x with minimal f **in a vicinity of a given x^0**:

$$\operatorname{prox}_f(x^0) := \arg\min_x f(x) + \frac{1}{2}||x - x^0||_2^2$$

$f := I_C$ for a **convex set** C and $I_C(x) := \begin{cases} 0, & \text{if } x \in C \\ \infty, & \text{else} \end{cases}$

$$\operatorname{prox}_f(x^0) = \arg\min_{x \in C} ||x - x^0||_2^2 =: \operatorname{proj}_C(x^0)$$

More Analytical Solutions for the Proximal Problem

- find x with minimal f **in a vicinity of a given x^0**:

$$\text{prox}_f(x^0) := \arg\min_x f(x) + \frac{1}{2}||x - x^0||_2^2$$

$$f := I_C \text{ for a } \textbf{convex set } C \text{ and } I_C(x) := \begin{cases} 0, & \text{if } x \in C \\ \infty, & \text{else} \end{cases}$$

$$\text{prox}_f(x^0) = \arg\min_{x \in C} ||x - x^0||_2^2 =: \text{proj}_C(x^0)$$

- **rectangles** / **box constraints** $C := [l_1, u_1] \times [l_2, u_2] \times \cdots \times [l_N, u_N]$:

$$\text{prox}_f(x^0) = \text{clip}(x^0, C) \quad \text{with } \text{clip}(x^0, C)_n := \min\{\max\{x_n^0, l_n\}, u_n\}$$

More Analytical Solutions for the Proximal Problem

- find x with minimal f **in a vicinity of a given x^0**:

$$\text{prox}_f(x^0) := \arg\min_x f(x) + \frac{1}{2}||x - x^0||_2^2$$

$$f := I_C \text{ for a } \textbf{convex set } C \text{ and } I_C(x) := \begin{cases} 0, & \text{if } x \in C \\ \infty, & \text{else} \end{cases}$$

$$\text{prox}_f(x^0) = \arg\min_{x \in C} ||x - x^0||_2^2 =: \text{proj}_C(x^0)$$

- **rectangles** / **box constraints** $C := [l_1, u_1] \times [l_2, u_2] \times \cdots \times [l_N, u_N]$:

$$\text{prox}_f(x^0) = \text{clip}(x^0, C) \quad \text{with } \text{clip}(x^0, C)_n := \min\{\max\{x_n^0, l_n\}, u_n\}$$

- **euclidean balls** $C := \{x \mid ||x||_2 \leq 1\}$:

$$\text{prox}_f(x^0) = \begin{cases} \frac{x^0}{||x^0||_2}, & \text{if } ||x^0||_2 > 1 \\ x^0, & \text{else} \end{cases}$$

More Analytical Solutions for the Proximal Problem

- find x with minimal f **in a vicinity of a given** x^0:

$$\text{prox}_f(x^0) := \arg\min_x f(x) + \frac{1}{2}||x - x^0||_2^2$$

$f := I_C$ for

- **L1 balls** $C := \{x \mid ||x||_1 \leq 1\}$:

$$\text{prox}_f(x^0) = \begin{cases} \text{soft}(x^0, \lambda), & \text{if } ||x^0||_1 > 1 \\ x^0, & \text{else} \end{cases}$$

$$\text{for } \lambda \text{ with } \sum_{n=1}^{N} (|x_n^0| - \lambda)_0 \overset{!}{=} 1$$

Generalized Gradient Descent

$$\min_x g(x) + h(x), \quad g, h \text{ convex}, g \text{ differentiable}$$

Generalized Gradient Descent:

$$x^{(t+1)} := \text{prox}_{\alpha^{(t)} h}(x^{(t)} - \alpha^{(t)} \nabla g(x^{(t)}))$$

$$\text{with } \text{prox}_f(x^0) := \arg\min_x f(x) + \frac{1}{2}||x - x^0||^2$$

- two-step approach:
  1. minimize component g via gradient descent
  2. minimize component h via prox operator
- requires control of step size $\alpha^{(t)}$
- generalizes gradient descent to objective functions with non-differentiable additive components
- convergence rate $O(1/t)$.

Application to Regularized Loss Minimization

$$\min \quad f(\theta) := \ell(\theta) + R(\theta)$$

- ℓ loss, convex and differentiable
  - e.g., RSS.
- R regularization, convex, but possibly not differentiable
  - e.g., $||\theta||_1$ or $I_C(\theta) := \begin{cases} 0, & \theta \in C \\ \infty, & \text{else} \end{cases}$

Application to Regularized Loss Minimization

Minimizing

$$\theta^{(t+1)} := \arg\min_\theta R(\theta) + \ell(\theta)$$

using a **Taylor expansion around previous estimate $\theta^{(t)}$**:

$$\ell(\theta) \approx \ell(\theta^{(t)}) + \nabla\ell(\theta^{(t)})^T(\theta - \theta^{(t)}) + \frac{1}{2}(\theta - \theta^{(t)})^T H (\theta - \theta^{(t)})$$

and **diagonal approximation of the Hessian** $H \approx \alpha^{(t)} I$

$$\approx \ell(\theta^{(t)}) + \nabla\ell(\theta^{(t)})^T(\theta - \theta^{(t)}) + \frac{1}{2}\alpha^{(t)}\|\theta - \theta^{(t)}\|_2^2$$

$$= \frac{1}{2}\alpha^{(t)}\|\theta - (\theta^{(t)} - \frac{1}{\alpha^{(t)}}\nabla\ell(\theta^{(t)}))\|_2^2 + r(\theta^{(t)})$$

yields a proximal problem

$$\theta^{(t+1)} \approx \arg\min_\theta \frac{1}{\alpha^{(t)}}R(\theta) + \frac{1}{2}\|\theta - (\theta^{(t)} - \tfrac{1}{\alpha^{(t)}}\nabla\ell(\theta^{(t)}))\|_2^2$$

$$= \mathrm{prox}_{\frac{1}{\alpha^{(t)}}R}\left(\theta^{(t)} - \frac{1}{\alpha^{(t)}}\nabla\ell(\theta^{(t)})\right)$$

Special Cases

$$\begin{aligned}\theta^{(t+1)} &:= \text{prox}_{\frac{1}{\alpha^{(t)}}R}(\theta^{(t)} - \frac{1}{\alpha^{(t)}}\nabla\ell(\theta^{(t)})) \\ &= \arg\min_{\theta} \frac{1}{\alpha^{(t)}}R(\theta) + \frac{1}{2}||\theta - (\theta^{(t)} - \frac{1}{\alpha^{(t)}}\nabla\ell(\theta^{(t)}))||_2^2\end{aligned}$$

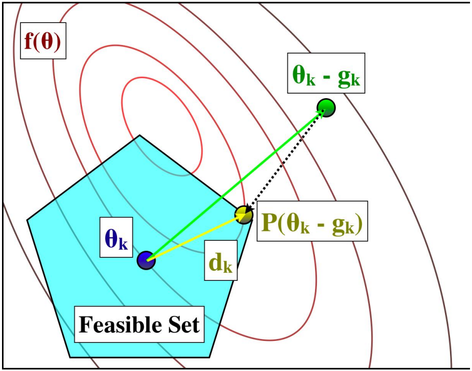
1. $R = 0$ yields **gradient descent**:

$$\theta^{(t+1)} = \theta^{(t)} - \frac{1}{\alpha^{(t)}}\nabla\ell(\theta^{(t)})$$

2. $R = I_C$ yields **projected gradient descent**:

$$\theta^{(t+1)} = \text{proj}_C(\theta^{(t)} - \frac{1}{\alpha^{(t)}}\nabla\ell(\theta^{(t)}))$$

Special Cases: Projected Gradient Descent



[Mur12, fig. 13.11]

Special Cases

$$
\begin{aligned}
\theta^{(t+1)} &:= \operatorname{prox}_{\frac{1}{\alpha^{(t)}}R}(\theta^{(t)} - \frac{1}{\alpha^{(t)}}\nabla\ell(\theta^{(t)})) \\
&= \arg\min_{\theta} \frac{1}{\alpha^{(t)}}R(\theta) + \frac{1}{2}||\theta - (\theta^{(t)} - \frac{1}{\alpha^{(t)}}\nabla\ell(\theta^{(t)}))||_2^2
\end{aligned}
$$

3. $R = \lambda||\theta||_1$ yields **iterative soft thresholding**:

$$
\theta^{(t+1)} = \operatorname{soft}(\theta^{(t)} - \frac{1}{\alpha^{(t)}}\nabla\ell(\theta^{(t)}), \frac{\lambda}{\alpha^{(t)}})
$$

Stepsizes $\alpha^{(t)}$

Taylor expansion of the Gradient:

$$\nabla\ell(\theta) \approx \nabla\ell(\theta^{(t)}) + \nabla^2\ell(\theta^{(t)})(\theta - \theta^{(t)}) \approx \nabla\ell(\theta^{(t)}) + \alpha^{(t)}(\theta - \theta^{(t)})$$
$$\implies \alpha^{(t)}(\theta^{(t)} - \theta^{(t-1)}) \approx \nabla\ell(\theta^{(t)}) - \nabla\ell(\theta^{(t-1)})$$

Idea:

$$\begin{aligned}\alpha^{(t)} &:= \arg\min_{\alpha} ||\alpha(\theta^{(t)} - \theta^{(t-1)}) - (\nabla\ell(\theta^{(t)}) - \nabla\ell(\theta^{(t-1)}))||_2^2 \\ &= \frac{(\theta^{(t)} - \theta^{(t-1)})^T(\nabla\ell(\theta^{(t)}) - \nabla\ell(\theta^{(t-1)}))}{(\theta^{(t)} - \theta^{(t-1)})^T(\theta^{(t)} - \theta^{(t-1)})}\end{aligned}$$

called **Barzilai-Borwein stepsize** or **spectral stepsize**.

- does not guarantee decreasing objective values.
- can be used with any gradient descent method.

Iterative Shrinkage and Thresholding Algorithm(ISTA)

- proximal gradient descent for L1 regularization
  - iterative soft thresholding
- Barzilai-Borwein stepsize
- in outer loop, homotopy on λ
  - i.e., gradually reducing $\lambda^{(t)}$ to λ

Note: This algorithm is called Sparse Reconstruction by Separable Approximation (SpaRSA) in the literature.

Algorithm

Algorithm 13.2: Iterative Shrinkage-Thresholding Algorithm (ISTA)

```
Input: $\mathbf{X} \in \mathbb{R}^{N \times D}$, $\mathbf{y} \in \mathbb{R}^N$, parameters $\lambda \geq 0$, $M \geq 1$, $0 < s < 1$ ;
Initialize $\boldsymbol{\theta}_0 = \mathbf{0}$, $\alpha = 1$, $\mathbf{r} = \mathbf{y}$, $\lambda_0 = \infty$;
repeat
    $\lambda_t = \max(s||\mathbf{X}^T\mathbf{r}||_\infty, \lambda)$ // Adapt the regularizer ;
    repeat
        $\mathbf{g} = \nabla L(\boldsymbol{\theta})$;
        $\mathbf{u} = \boldsymbol{\theta} - \frac{1}{\alpha}\mathbf{g}$;
        $\boldsymbol{\theta} = \text{soft}(\mathbf{u}, \frac{\lambda_t}{\alpha})$;
        Update $\alpha$ using BB stepsize in Equation 13.82 ;
    until $f(\boldsymbol{\theta})$ increased too much within the past $M$ steps;
    $\mathbf{r} = \mathbf{y} - \mathbf{X}\boldsymbol{\theta}$ // Update residual ;
until $\lambda_t = \lambda$;
```

[Mur12, p. 446]

Nesterov's Accelerated Generalized Gradient Descent

$$\min_x g(x) + h(x), \quad g, h \text{ convex}, g \text{ differentiable}$$

Generalized Gradient Descent:

$$x^{(t+1)} := \text{prox}_{\alpha^{(t)}h}(x^{(t)} + \frac{t-1}{t+2}(x^{(t)} - x^{(t-1)}) - \alpha^{(t)} \nabla g(x^{(t)}))$$

$$\text{with } \text{prox}_f(x^0) := \arg\min_x f(x) + \frac{1}{2}||x - x^0||^2$$

- added **momentum term**
- works also for vanilla gradient descent ($h = 0$)
- convergence rate $O(1/t^2)$!
- beware, there are at least 3 versions of **Nesterov's method**.

Fast Iterative Shrinkage and Thresholding Alg. (FISTA)

$$\theta^{(t+1)} := \text{prox}_{\frac{1}{\alpha^{(t)}}R}(\theta^{(t)} + \frac{t-1}{t+2}(\theta^{(t)} - \theta^{(t-1)}) - \frac{1}{\alpha^{(t)}}\nabla\ell(\theta^{(t)}))$$

for $R = \lambda||\theta||_1$ yields iterative soft thresholding:

$$\theta^{(t+1)} = \text{soft}(\theta^{(t)} + \frac{t-1}{t+2}(\theta^{(t)} - \theta^{(t-1)}) - \frac{1}{\alpha^{(t)}}\nabla\ell(\theta^{(t)}), \frac{\lambda}{\alpha^{(t)}})$$

using **Nesterov's Accelerated Generalized Gradient Descent**.

FISTA vs ISTA

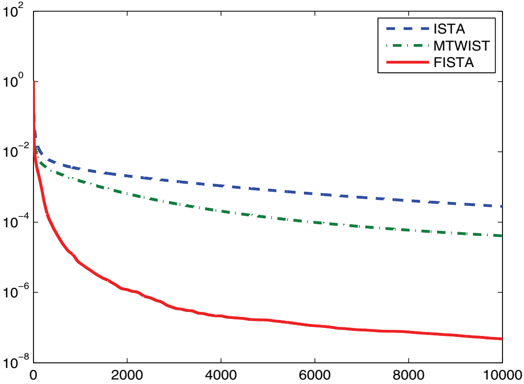


Figure 5. *Comparison of function value errors $F(\mathbf{x}_k) - F(\mathbf{x}^*)$ of ISTA, MTWIST, and FISTA.*

[BT09, p. 19]

Outline

1. Homotopy Methods: Least Angle Regression

Interlude: A note on Model complexity

2. Proximal Gradient Methods

3. Laplace Priors (Bayesian Lasso)

Laplace Priors correspond to L1 regularization

$$\hat{\beta} = \arg\min_{\beta} \underbrace{L(\beta)}_{\text{Loss}} + \lambda \underbrace{R(\beta)}_{\text{Regularization}}$$

$$\Big\downarrow \text{"Bayesianize"}$$

$$\hat{\beta} = \arg\max_{\beta} p(\beta \mid X, y)$$

Laplace Priors correspond to L1 regularization

$$\hat{\beta} = \arg\min_{\beta} \underbrace{L(\beta)}_{\text{Loss}} + \lambda \underbrace{R(\beta)}_{\text{Regularization}}$$

$$\Big\downarrow \text{"Bayesianize"}$$

$$\hat{\beta} = \arg\max_{\beta} p(\beta \mid X, y)$$

$$\underbrace{p(\beta \mid X, y)}_{\text{posterior}} \propto \underbrace{p(y \mid X, \beta)}_{\text{likelihood}} \cdot \underbrace{p(\beta)}_{\text{prior}}$$

Laplace Priors correspond to L1 regularization

$$\hat{\beta} = \arg\min_{\beta} \underbrace{L(\beta)}_{\text{Loss}} + \lambda \underbrace{R(\beta)}_{\text{Regularization}}$$

$$\Big\downarrow \text{"Bayesianize"}$$

$$\hat{\beta} = \arg\max_{\beta} p(\beta \mid X, y)$$

$$\underbrace{p(\beta \mid X, y)}_{\text{posterior}} \propto \underbrace{p(y \mid X, \beta)}_{\text{likelihood}} \cdot \underbrace{p(\beta)}_{\text{prior}}$$

- $p(y \mid X, \beta) = \mathcal{N}(y \mid X\beta, \sigma^2 I) \leftrightsquigarrow$ Bayesian Linear Regression
- $p(\beta) = \mathcal{N}(\beta \mid 0, \frac{1}{\lambda} I) \propto \exp(-\frac{1}{2}\lambda \|\beta\|_2^2) \leftrightsquigarrow$ Bayesian Ridge
- $p(\beta) = \text{Lap}(\beta \mid 0, \frac{1}{\lambda} I) \propto \exp(-\lambda \|\beta\|_1) \leftrightsquigarrow$ Bayesian Lasso
- Different priors correspond to different regularization!

Laplace Priors correspond to L1 regularization

Problem: Still not possible to find the MAP analytically.

Idea: Rewrite the Laplace as a **Gaussian-Scale-Mixture** with Exponential priors

$$\mathrm{Lap}(\beta_i \mid 0, \tfrac{1}{\lambda}) = \int \mathcal{N}(\beta_i \mid 0, \tau_i^2)\, \mathrm{Exp}(\tau_i^2 \mid \tfrac{1}{2}\lambda^2) \mathrm{d}\tau^2$$

i.e. each parameter is distributed as $\beta_i \sim \mathcal{N}(0, \tau_i^2)$ with $\tau_i^2 \sim \mathrm{Exp}(\frac{1}{2}\lambda^2)$

Laplace Priors correspond to L1 regularization

Problem: Still not possible to find the MAP analytically.

Idea: Rewrite the Laplace as a **Gaussian-Scale-Mixture** with Exponential priors

$$\mathrm{Lap}(\beta_i \mid 0, \tfrac{1}{\lambda}) = \int \mathcal{N}(\beta_i \mid 0, \tau_i^2)\,\mathrm{Exp}(\tau_i^2 \mid \tfrac{1}{2}\lambda^2)\mathrm{d}\tau^2$$

i.e. each parameter is distributed as $\beta_i \sim \mathcal{N}(0, \tau_i^2)$ with $\tau_i^2 \sim \mathrm{Exp}(\frac{1}{2}\lambda^2)$

Resulting posterior distribution:

$$p(\beta, \sigma^2 \mid X, y, \tau^2) \propto \underbrace{p(y \mid X, \beta, \sigma^2)}_{=\mathcal{N}(y|X\beta,\sigma^2 I)} \cdot \underbrace{p(\beta \mid \tau^2)}_{=\mathcal{N}(\beta|0,\mathrm{diag}(\tau^2))} \cdot \underbrace{p(\tau^2 \mid \gamma)}_{=\mathrm{Exp}(\tau^2|\frac{1}{2}\lambda^2)} \cdot \underbrace{p(\sigma^2)}_{=\mathrm{IG}(\sigma^2|a,b)}$$

Where $\mathrm{IG}(\sigma^2 \mid a, b)$ is an **Inverse-Gamma** prior on the variance, $(\tau_i)_{i=1\ldots M}$ are **latent variables** and λ is the user chosen regularization strength.

Laplace Priors correspond to L1 regularization

Problem: Still not possible to find the MAP analytically.

Idea: Rewrite the Laplace as a **Gaussian-Scale-Mixture** with Exponential priors

$$\text{Lap}(\beta_i \mid 0, \tfrac{1}{\lambda}) = \int \mathcal{N}(\beta_i \mid 0, \tau_i^2)\, \text{Exp}(\tau_i^2 \mid \tfrac{1}{2}\lambda^2) \mathrm{d}\tau^2$$

i.e. each parameter is distributed as $\beta_i \sim \mathcal{N}(0, \tau_i^2)$ with $\tau_i^2 \sim \text{Exp}(\frac{1}{2}\lambda^2)$

Resulting posterior distribution:

$$p(\beta, \sigma^2 \mid X, y, \tau^2) \propto \underbrace{p(y \mid X, \beta, \sigma^2)}_{=\mathcal{N}(y|X\beta,\sigma^2 I)} \cdot \underbrace{p(\beta \mid \tau^2)}_{=\mathcal{N}(\beta|0,\text{diag}(\tau^2))} \cdot \underbrace{p(\tau^2 \mid \gamma)}_{=\text{Exp}(\tau^2|\frac{1}{2}\lambda^2)} \cdot \underbrace{p(\sigma^2)}_{=\text{IG}(\sigma^2|a,b)}$$

Where $\text{IG}(\sigma^2 \mid a, b)$ is an **Inverse-Gamma** prior on the variance, $(\tau_i)_{i=1\ldots M}$ are **latent variables** and λ is the user chosen regularization strength.

$\boxed{p \text{ is now smooth in all parameters! We can apply EM-algorithm!}}$

Laplace Priors correspond to L1 regularization

L2 regularization:

$$f(\beta) := ||y - X\beta||_2^2 + \lambda||\beta||_2^2$$

Gaussian priors:

$$\begin{aligned} p(y_n \mid x_n, \beta, \sigma^2) &:= \mathcal{N}(y_n \mid x_n^T\beta, \sigma^2) \\ p(\beta) &:= \mathcal{N}(\beta \mid 0, \tfrac{1}{\lambda}I) \\ &= (2\pi\lambda)^{-M/2}\, e^{-\frac{1}{2}\lambda||\beta||_2^2} \end{aligned}$$

using negative loglikelihood as objective function:

$$f(\beta) := -\log p(y \mid X, \beta)$$

Laplace Priors correspond to L1 regularization

L2 regularization:

$$f(\beta) := ||y - X\beta||_2^2 + \lambda||\beta||_2^2$$

Gaussian priors:

$$\begin{aligned} p(y_n \mid x_n, \beta, \sigma^2) &:= \mathcal{N}(y_n \mid x_n^T\beta, \sigma^2) \\ p(\beta) &:= \mathcal{N}(\beta \mid 0, \tfrac{1}{\lambda} I) \\ &= (2\pi\lambda)^{-M/2}\, e^{-\frac{1}{2}\lambda||\beta||_2^2} \end{aligned}$$

L1 regularization:

$$f(\beta) := ||y - X\beta||_2^2 + \lambda||\beta||_1$$

Laplace priors:

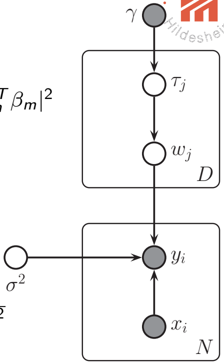
$$\begin{aligned} p(y_n \mid x_n, \beta, \sigma^2) &:= \mathcal{N}(y_n \mid x_n^T\beta, \sigma^2) \\ p(\beta_m) &:= \text{Lap}(\beta_m \mid 0, \tfrac{1}{\lambda}) \\ &= \tfrac{1}{2}\lambda e^{-\lambda|\beta_m|} \end{aligned}$$

using negative loglikelihood as objective function:

$$f(\beta) := -\log p(y \mid X, \beta)$$

Laplace Prior as Gaussian Scale Mixture

$$
\begin{aligned}
p(y_n \mid x_n, \beta, \sigma^2) &:= \mathcal{N}(y_n \mid x_n^T \beta, \sigma^2) && = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2\sigma^2}|y_n - x_n^T \beta_m|^2} \\
p(\beta_m \mid \tau_m^2) &:= \mathcal{N}(\beta_m \mid 0, \tau_m^2) && = \frac{1}{\sqrt{2\pi\tau_m^2}} e^{-\frac{1}{2\tau_m^2}|\beta_m|^2} \\
p(\tau_m^2) &:= \operatorname{Exp}(\tau_m^2 \mid \frac{1}{2}\lambda^2) && = \frac{1}{2}\lambda^2 e^{-\frac{1}{2}\lambda^2 \tau_m^2} \\
p(\sigma^2) &:= \operatorname{IG}(\sigma^2 \mid a, b) && = \frac{b^a}{\Gamma(a)} \sigma^{-2(1+a)} e^{-\frac{b}{\sigma^2}}
\end{aligned}
$$



Note: $\Lambda := \operatorname{diag}(\tau_1^2, \tau_2^2, \ldots, \tau_M^2)$

[Mur12, p. 446]

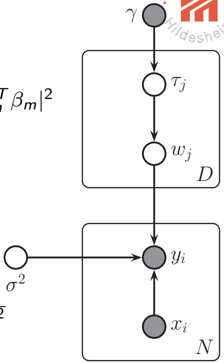
Laplace Prior as Gaussian Scale Mixture

$$p(y_n \mid x_n, \beta, \sigma^2) := \mathcal{N}(y_n \mid x_n^T \beta, \sigma^2) \quad = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2\sigma^2}|y_n - x_n^T \beta_m|^2}$$

$$p(\beta_m \mid \tau_m^2) := \mathcal{N}(\beta_m \mid 0, \tau_m^2) \quad = \frac{1}{\sqrt{2\pi\tau_m^2}} e^{-\frac{1}{2\tau_m^2}|\beta_m|^2}$$

$$p(\tau_m^2) := \mathrm{Exp}(\tau_m^2 \mid \frac{1}{2}\lambda^2) \quad = \frac{1}{2}\lambda^2 e^{-\frac{1}{2}\lambda^2\tau_m^2}$$

$$p(\sigma^2) := \mathrm{IG}(\sigma^2 \mid a, b) \quad = \frac{b^a}{\Gamma(a)} \sigma^{-2(1+a)} e^{-\frac{b}{\sigma^2}}$$



Negative-Log-Likelihood:

$$\ell(\beta, \sigma^2 \mid X, y, \tau^2) = \frac{1}{2} N \log \sigma^2 + \frac{1}{2\sigma^2}||y - X\beta||_2^2$$

$$+ \sum_{m=1}^{M} \log \tau_m^2 + \frac{1}{2}\beta^T \Lambda^{-1} \beta + \frac{1}{2}\lambda^2 \sum_{m=1}^{M} \tau_m^2 + (1+a)\log\sigma^2 + \frac{b}{\sigma^2}$$

Note: $\Lambda := \mathrm{diag}(\tau_1^2, \tau_2^2, \ldots, \tau_M^2)$

[Mur12, p. 446]

E-step for τ^2

We need to compute the expectation of

$$p(\tau^2 \mid X, y, \beta, \sigma^2) \propto p(\beta \mid \tau^2) p(\tau^2)$$

where $p(\beta_i \mid \tau_i^2) = \mathcal{N}(\beta_i \mid 0, \tau_i^2)$ and $p(\tau_i^2) = \text{Exp}(\tau_i^2 \mid \frac{1}{2}\lambda^2)$

E-step for τ^2

We need to compute the expectation of

$$p(\tau^2 \mid X, y, \beta, \sigma^2) \propto p(\beta \mid \tau^2) p(\tau^2)$$

where $p(\beta_i \mid \tau_i^2) = \mathcal{N}(\beta_i \mid 0, \tau_i^2)$ and $p(\tau_i^2) = \text{Exp}(\tau_i^2 \mid \frac{1}{2}\lambda^2)$

It turns out simpler to estimate $\frac{1}{\tau^2}$: One can show that (tutorial)

$$\frac{1}{\tau^2} \mid \beta \sim \text{InvGauss}(\sqrt{\frac{\lambda^2}{\beta^2}}, \lambda^2)$$

Where the **Inverse Gaussian distribution** is given by

$$\text{InvGauss}(x \mid \mu, \nu) = \sqrt{\frac{\nu}{2\pi x^3}} e^{-\frac{\nu}{2\mu^2 x}(x-\mu)^2}$$

with mean $\mathbb{E}[x] = \mu$ and variance $\text{Var}[x] = \mu^3/\nu \implies \boxed{\mathbb{E}\left[\frac{1}{\tau_i^2}\right] = \frac{\lambda}{|\beta_i|}}$

E-step for σ^2

We need to compute the expectation of

$$p(\sigma^2 \mid X, y, \beta, \tau^2) \propto p(y \mid X, \beta, \sigma^2) p(\sigma^2)$$

where $p(y \mid X, \beta, \sigma^2) = \mathcal{N}(y \mid X\beta, \sigma^2 I)$ and $p(\sigma^2) = \mathrm{IG}(\sigma^2 \mid a, b)$

E-step for σ^2

We need to compute the expectation of

$$p(\sigma^2 \mid X, y, \beta, \tau^2) \propto p(y \mid X, \beta, \sigma^2) p(\sigma^2)$$

where $p(y \mid X, \beta, \sigma^2) = \mathcal{N}(y \mid X\beta, \sigma^2 I)$ and $p(\sigma^2) = \mathrm{IG}(\sigma^2 \mid a, b)$
One can show that (tutorial)

$$p(\sigma^2 \mid X, y, \beta, \tau^2) = \mathrm{IG}(\sigma^2, a', b')$$

with $a' = a + \frac{1}{2}N$ and $b' = b + \frac{1}{2}\|y - X\beta\|_2^2$.
Here the **Inverse Gamma** distribution is given by

$$p(x \mid a, b) = \frac{b^a}{\Gamma(a)} x^{-a-1} e^{-\frac{\beta}{x}}$$

Note that if $X \sim \Gamma(a, b) \iff X^{-1} \sim \mathrm{IG}(a, b)$, so $\boxed{\mathbb{E}[\frac{1}{\sigma^2}] = \frac{a'}{b'}}$

Remark on Conjugate Prior

Note that the posterior of σ^2 is again an Inverse Gamma distribution!

$$\underbrace{p(\sigma^2 \mid X, y, \beta)}_{=\mathrm{IG}(a',b')} \propto \underbrace{p(y \mid X, \beta, \sigma^2)}_{\mathcal{N}(\mu,\nu)} \; \underbrace{p(\sigma^2)}_{=\mathrm{IG}(a,b)}$$

This is because the IG is a **conjugate prior** to the normal distribution. Conjugate priors let you interpret how the data changes the believe about the parameters. $\longrightarrow$ Main reason for choosing this prior!

Remark on Conjugate Prior

Note that the posterior of σ^2 is again an Inverse Gamma distribution!

$$\underbrace{p(\sigma^2 \mid X, y, \beta)}_{=\mathrm{IG}(a', b')} \propto \underbrace{p(y \mid X, \beta, \sigma^2)}_{\mathcal{N}(\mu, \nu)} \underbrace{p(\sigma^2)}_{=\mathrm{IG}(a, b)}$$

This is because the IG is a **conjugate prior** to the normal distribution. Conjugate priors let you interpret how the data changes the believe about the parameters. $\longrightarrow$ Main reason for choosing this prior!

Remark: inverse distributions

Note that the Inverse Gamma distribution is called Inverse Gamma because

$$X \sim \Gamma(a, b) \iff X^{-1} \sim \mathrm{IG}(a, b) \tag{1}$$

However, despite the name, the same is **not true** for the Inverse Gaussian!

M-step for β

We need to compute

$$\hat{\beta} = \arg\min_{\beta} \ell(\beta, \sigma^2, \tau^2) = \arg\min_{\beta} \frac{1}{2\sigma^2}\|y - X\beta\|_2^2 + \frac{1}{2}\beta^T \Lambda^{-1} \beta$$

where we dropped all terms independent of β. Then

$$\nabla_\beta \ell = 0 \iff (\frac{1}{\sigma^2} X^T X + \Lambda^{-1})\hat{\beta} = \frac{1}{\sigma^2} X^T y$$

So $\boxed{\hat{\beta} = (X^T X + (\frac{1}{\sigma^2}\Lambda)^{-1})^{-1} X^T y}$ which is a ridge regression objective!

EM summary

1. Expectation of τ^2:

$$p(\frac{1}{\tau_i^2} \mid \beta) = \text{Inv-Gauss}(\sqrt{\frac{\lambda^2}{\beta_i^2}}, \lambda^2)$$

$$\mathbb{E}\left[\frac{1}{\tau_i^2}\right] = \frac{\lambda}{|\beta_i|}$$

2. Expectation of σ^2:

$$p(\sigma^2 \mid X, y, \beta) = \text{IG}(\sigma^2 \mid a', b')$$

$$\mathbb{E}[\frac{1}{\sigma^2}] = \frac{a'}{b'}$$

3. Maximization w.r.t. β:

$$\ell(\beta) = \frac{1}{2\sigma^2}||y - X\beta||_2^2 + \frac{1}{2}\beta^T\Lambda^{-1}\beta$$

$$\hat{\beta} = (X^TX + (\frac{1}{\sigma^2}\Lambda)^{-1})^{-1}X^Ty$$

Why Laplace Prior?

- Bayesian Lasso
  - provides posterior distribution, not just point estimates
- Can be generalized to other models / losses
- Motivates to experiment with other types of priors, too
- Less scalable than the other methods, though.

Further Readings

- L1 regularization: [Mur12, chapter 13.3–5], [HTFF05, chapter 3.4, 3.8, 4.4.4], [Bis06, chapter 3.1.4].
  - LAR, LARS: [HTFF05, chapter 3.4.4], [Mur12, chapter 13.4.2],
- Non-convex regularizers: [Mur12, chapter 13.6].
- Automatic Relevance Determination (ARD): [Mur12, chapter 13.7], [HTFF05, chapter 11.9.1], [Bis06, chapter 7.2.2].
- Sparse Coding: [Mur12, chapter 13.8].
- Multivariate Laplace Distribution: [EKL06]

References

Christopher M. Bishop.
Pattern recognition and machine learning, volume 1.
springer New York, 2006.

Amir Beck and Marc Teboulle.
A fast iterative shrinkage-thresholding algorithm for linear inverse problems.
SIAM Journal on Imaging Sciences, 2(1):183–202, 2009.

Torbjørn Eltoft, Taesu Kim, and Te-Won Lee.
On the multivariate laplace distribution.
IEEE Signal Processing Letters, 13(5):300–303, 2006.

Trevor Hastie, Robert Tibshirani, Jerome Friedman, and James Franklin.
The elements of statistical learning: data mining, inference and prediction, volume 27.
Springer, 2005.

Kevin P. Murphy.
Machine learning: a probabilistic perspective.
The MIT Press, 2012.