2nd Int. Workshop on New Computational Methods for Inverse Problems

ENS Cachan, May 15, 2012

Proximal Splitting Derivatives for Risk Estimation

Application to image processing

Charles Deledalle, Samuel Vaiter, Gabriel Peyré, Jalal Fadili and Charles Dossal

CEREMADE, CNRS-Paris Dauphine

15 mai 2012

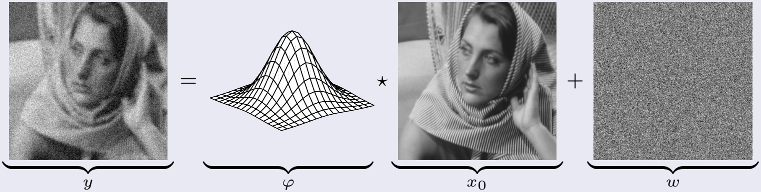
Motivations

Goal : recover an image $x_0 \in \mathbb{R}^N$ from its low-dimensionnal noisy observation $y \in \mathbb{R}^P$

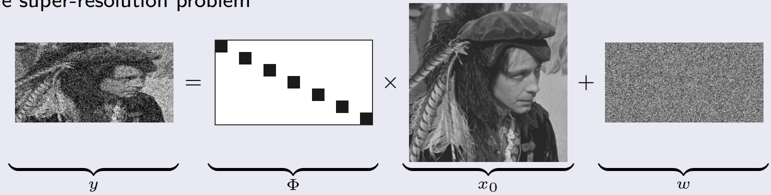
Linear inverse problem

We consider $y = \Phi x_0 + w$ with $\Phi : \mathbb{R}^N \to \mathbb{R}^P$ and $w \sim \mathcal{N}(0, \sigma^2 \mathrm{Id}_P)$, e.g.:

- the deconvolution problem



- or, the super-resolution problem



Recover x_0 from y is an ill-posed inverse problem

Motivations

Goal : recover an image $x_0 \in \mathbb{R}^N$ from its low-dimensionnal noisy observation $y \in \mathbb{R}^P$

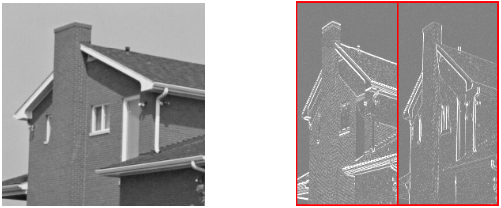
Convex regularization of the ill-posed inverse problem

- Forward model: $y = \Phi x_0 + w$
- Inverse model: $x_\theta(y) \in \underset{x}{\text{argmin}} \underbrace{F(x, y)}_{\text{data fidelity}} + \underbrace{G_\theta(x)}_{\text{regularization}} \neq \emptyset$ (Variational or MAP)

F a proper lsc convex function, e.g., $F(x, y) = \frac{1}{2}\|y - \Phi x\|^2$

G_θ a **parametric** proper lsc convex function

ex: Total-Variation $G_\theta(x) = \lambda \|\nabla x\|$ where $\|\nabla x\| = \sum_k \|(\nabla x)_k\|$ $\theta = \{\lambda > 0\}$



(a) Image x (b) Gradient ∇x

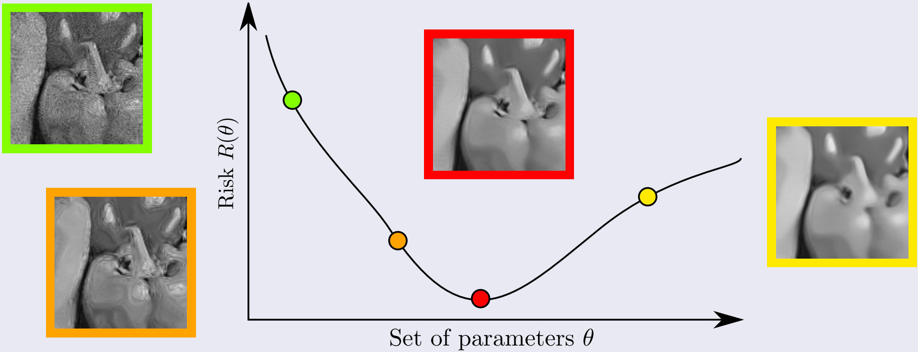
How to select the optimal set of parameters θ?

Motivations

Goal : recover an image $x_0 \in \mathbb{R}^N$ from its low-dimensionnal noisy observation $y \in \mathbb{R}^P$

Parameter selection

Given a family of estimators $x_\theta(y)$ of x_0, find the best set of parameters θ



Goal: minimize the risk $R(\theta) = \|x_\theta(y) - x_0\|^2$

Difficulty: $R(\theta)$ is unknown since x_0 unknown

Mean: **$R(\theta)$ can be "approached" if one knows the divergence $\mathrm{div}_y\, x_\theta(y)$**

Outline

Outline

1. Unbiased Risk Estimation
2. Generalized Forward Backward and Derivatives
3. Numerical Examples

Unbiased Risk Estimation

- Forward model: $y = \Phi x_0 + w, \quad w \sim \mathcal{N}(0, \sigma^2 \mathrm{Id}_P)$
- Goal: Unbiasedly estimate the risk associated to

$$x_\theta(y) \in \underset{x}{\operatorname{argmin}}\ F(x, y) + G_\theta(x)$$

Ideally $\mathbb{E}_y \|x_\theta(y) - x_0\|^2$.

Estimates must depend solely on y

Definition (Generalized Stein's Unbiased Risk Estimator (GSURE))

Let $x_\theta(y)$ an estimator of x_0. GSURE is defined as:

$$\mathrm{GSURE}(x_\theta, y) = \|\Phi^*(\Phi\Phi^*)^+ y - \Phi x_\theta(y)\|^2 - \sigma^2 \operatorname{tr}((\Phi\Phi^*)^+) + 2\sigma^2 \operatorname{div}_y((\Phi\Phi^*)^+ \Phi x_\theta(y)).$$

Theorem ([Stein, 1981, Eldar, 2009])

Assume $y \mapsto \Phi x_\theta(y)$ is weakly differentiable. Then

$$\mathbb{E}_w \mathrm{GSURE}(x_\theta, y) = \mathbb{E}_w \|\Pi x_\theta(y) - \Pi x_0\|^2$$

where $\Pi = \Phi^*(\Phi\Phi^*)^+\Phi$ is the projection on $\mathrm{Ker}(\Phi)^\perp$.

How to estimate the divergence term $\operatorname{div}_y((\Phi\Phi^*)^+ \Phi x_\theta(y))$?

Generalized SURE

GSURE based on the divergence term $\operatorname{div}_y((\Phi\Phi^*)^+\Phi x_\theta(y))$?

Implementation [Vonesch et al., 2008]

- Use the Jacobian trace formula of the divergence

$$\operatorname{div}_y((\Phi\Phi^*)^+\Phi x_\theta(y)) = \operatorname{tr}(\underbrace{(\Phi\Phi^*)^+\partial_y\Phi x_\theta(y)}_{J(y)})$$

- In practice, the Jacobian $J(y) \in \mathbb{R}^{P\times P}$ cannot be stored in memory
- Use the trace estimator of $A \in \mathbb{R}^{P\times P}$

$$\operatorname{tr} A = \mathbb{E}_\delta \langle A\delta, \delta\rangle \quad \text{where} \quad \delta \sim \mathcal{N}(0, \operatorname{Id}_P)$$

- Finally, we have the approximation

$$\operatorname{div}_y((\Phi\Phi^*)^+\Phi x_\theta(y)) \approx \frac{1}{k}\sum_{i=1}^{k}\langle J(y)[\delta_i],\, \delta_i\rangle$$

  where δ_i are k realizations of δ

- Compute $J(y)[\delta_i] \in \mathbb{R}^P$ as the action of $J(y)$ on $\delta_i \in \mathbb{R}^P$
- P sufficiently large $\Rightarrow$ good approximation even for small k (e.g., $k = 1$)

Next: **How to evaluate $J(y)[\delta_i]$ when $x_\theta(y)$ is given by a proximal splitting algorithm?**

Generalized SURE

GSURE based on the divergence term $\mathrm{div}_y((\Phi\Phi^*)^+\Phi x_\theta(y))$?

Implementation [Vonesch et al., 2008]

- Use the Jacobian trace formula of the divergence

$$\mathrm{div}_y((\Phi\Phi^*)^+\Phi x_\theta(y)) = \mathrm{tr}(\underbrace{(\Phi\Phi^*)^+\partial_y\Phi x_\theta(y)}_{J(y)})$$

- In practice, the Jacobian $J(y) \in \mathbb{R}^{P\times P}$ cannot be stored in memory
- Use the trace estimator of $A \in \mathbb{R}^{P\times P}$

$$\mathrm{tr}\, A = \mathbb{E}_\delta \langle A\delta, \delta\rangle \quad \text{where} \quad \delta \sim \mathcal{N}(0, \mathrm{Id}_P)$$

- Finally, we have the approximation

$$\mathrm{div}_y((\Phi\Phi^*)^+\Phi x_\theta(y)) \approx \frac{1}{k}\sum_{i=1}^{k}\langle J(y)[\delta_i],\, \delta_i\rangle$$

  where δ_i are k realizations of δ
- Compute $J(y)[\delta_i] \in \mathbb{R}^P$ as the action of $J(y)$ on $\delta_i \in \mathbb{R}^P$
- P sufficiently large $\Rightarrow$ good approximation even for small k (e.g., $k = 1$)

Next: **How to evaluate $J(y)[\delta_i]$ when $x_\theta(y)$ is given by a proximal splitting algorithm?**

Note: In the following, the dependency with θ will be dropped for simplicity

Outline

1. Unbiased Risk Estimation
2. Generalized Forward Backward and Derivatives
3. Numerical Examples

Forward Backward (FB)

Solve: $x(y) \in \underset{x}{\operatorname{argmin}}\ F(x,y) + G(x)$

where $x \mapsto F(x,y)$ — C^1 with L-Lipschitz gradient

$x \mapsto G(x)$ — simple

Simple function: A lsc proper convex function G is simple if the following has a closed-form expression

$$\operatorname{Prox}_{\gamma G}(x,y) = \underset{z}{\operatorname{argmin}}\ \frac{1}{2}\|x-z\|^2 + \gamma G(z), \quad \forall \gamma > 0$$

Iterative scheme: $x^{(\ell+1)}(y) = \operatorname{Prox}_{\lambda\tau G}(x^{(\ell)} - \tau \nabla_1 F(x^{(\ell)}, y))$

Forward Backward (FB)

Solve: $x(y) \in \underset{x}{\operatorname{argmin}}\ F(x,y) + G(x)$

where $x \mapsto F(x,y)$ C^1 with L-Lipschitz gradient
$x \mapsto G(x)$ simple

Simple function: A lsc proper convex function G is simple if the following has a closed-form expression

$$\operatorname{Prox}_{\gamma G}(x,y) = \underset{z}{\operatorname{argmin}}\ \frac{1}{2}\|x - z\|^2 + \gamma G(z), \quad \forall \gamma > 0$$

Iterative scheme: $x^{(\ell+1)}(y) = \operatorname{Prox}_{\lambda\tau G}(x^{(\ell)} - \tau \nabla_1 F(x^{(\ell)}, y))$

Example (ℓ_1 sparse regularization)

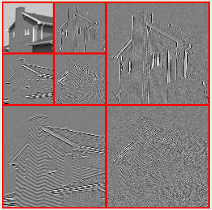
Solve: $x(y) \in \underset{x}{\operatorname{argmin}}\ \underbrace{\frac{1}{2}\|\Phi\Psi x - y\|^2}_{F(x,y)} + \underbrace{\lambda\|x\|_1}_{G(x)}$

where Ψ is, e.g., an orthogonal wavelet transform

Use: $\nabla_1 F(x,y) = \Psi^*\Phi^*(\Phi\Psi x - y)$,
$\operatorname{Prox}_{\tau G_i}(x) = T_{\lambda\tau}(x)$

where $T_{\lambda\tau}(x)$ is the component-wise soft-thresholding

$$T_\rho(x)_i = \max(0, 1 - \rho/\|x_i\|)x_i$$



(a) Wavelet coefficients

Generalized Forward Backward and Derivatives

Generalized Forward Backward (GFB) [Raguet et al., 2011]

Solve: $x(y) \in \underset{x}{\operatorname{argmin}}\ F(x, y) + G(x)$ where $G(x) = \sum_{i=1}^{Q} G_i(x)$.

where
$x \mapsto F(x, y)$ C^1 with L-Lipschitz gradient
$x \mapsto G_i(x)$ simple

G does not have to be simple!

Generalized Forward Backward (GFB) [Raguet et al., 2011]

Solve: $x(y) \in \underset{x}{\operatorname{argmin}}\ F(x,y) + G(x)$ where $G(x) = \sum_{i=1}^{Q} G_i(x)$.

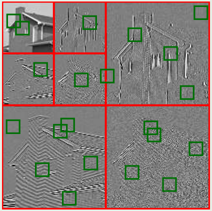
where $x \mapsto F(x,y)$ C^1 with L-Lipschitz gradient
$x \mapsto G_i(x)$ simple

G does not have to be simple!

Example (Block sparsity)

Solve: $x(y) \in \underset{x}{\operatorname{argmin}}\ \underbrace{\frac{1}{2}\|\Phi\Psi x - y\|^2}_{F(x,y)} + \underbrace{\lambda\|\mathcal{B}x\|}_{G(x)}$ where $\|\mathcal{B}x\| = \sum_k \|(\mathcal{B}x)_k\|$

and $\mathcal{B}$ extracts all blocks of size B (G is not simple)



(a) Blocks

Generalized Forward Backward and Derivatives

Generalized Forward Backward (GFB) [Raguet et al., 2011]

Solve: $x(y) \in \underset{x}{\operatorname{argmin}}\ F(x,y) + G(x)$ where $G(x) = \sum_{i=1}^{Q} G_i(x)$.

where $x \mapsto F(x,y)$ — C^1 with L-Lipschitz gradient
$x \mapsto G_i(x)$ — simple

G does not have to be simple!

Example (Block sparsity)

Solve: $x(y) \in \underset{x}{\operatorname{argmin}}\ \underbrace{\frac{1}{2}\|\Phi\Psi x - y\|^2}_{F(x,y)} + \underbrace{\lambda\|\boldsymbol{\mathcal{B}} x\|}_{G(x)}$ where $\|\boldsymbol{\mathcal{B}} x\| = \sum_k \|(\boldsymbol{\mathcal{B}} x)_k\|$

and $\boldsymbol{\mathcal{B}}$ extracts all blocks of size B (G is not simple)

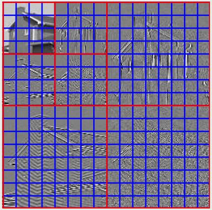
Recast: $x(y) \in \underset{x}{\operatorname{argmin}}\ \underbrace{\frac{1}{2}\|\Phi\Psi x - y\|^2}_{F(x,y)} + \sum_i \underbrace{\lambda\|\boldsymbol{\mathcal{B}}_i x\|}_{G_i(x)}$

where $\boldsymbol{\mathcal{B}}_i$ a partition of non-overlapping blocks

Note: $\nabla_1 F(x,y) = \Psi^* \Phi^* (\Phi\Psi x - y)$,
$\operatorname{Prox}_{\tau G_i}(x) = \boldsymbol{\mathcal{B}}_i^* T_{\lambda\tau}(\boldsymbol{\mathcal{B}}_i x)$ (G_i is simple)

where $T_\rho(b)$ for $b \in \mathbb{R}^B$ is the block-wise soft-thresholding

$$T_\rho(b)_i = \max(0, 1 - \rho/\|b_i\|) b_i$$



(a) Non-overlapping blocks

Generalized Forward Backward (GFB) [Raguet et al., 2011]

Solve: $x(y) \in \underset{x}{\operatorname{argmin}}\ F(x,y) + G(x)$ where $G(x) = \sum_{i=1}^{Q} G_i(x)$.

where $x \mapsto F(x,y)$ C^1 with L-Lipschitz gradient
$x \mapsto G_i(x)$ simple

G does not have to be simple!

Example (Block sparsity)

Solve: $x(y) \in \underset{x}{\operatorname{argmin}}\ \underbrace{\frac{1}{2}\|\Phi\Psi x - y\|^2}_{F(x,y)} + \underbrace{\lambda\|\mathcal{B}x\|}_{G(x)}$ where $\|\mathcal{B}x\| = \sum_k \|(\mathcal{B}x)_k\|$

and $\mathcal{B}$ extracts all blocks of size B (G is not simple)

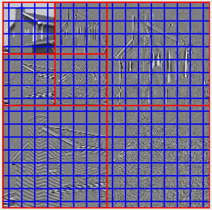
Recast: $x(y) \in \underset{x}{\operatorname{argmin}}\ \underbrace{\frac{1}{2}\|\Phi\Psi x - y\|^2}_{F(x,y)} + \sum_i \underbrace{\lambda\|\mathcal{B}_i x\|}_{G_i(x)}$

where $\mathcal{B}_i$ a partition of non-overlapping blocks

Note: $\nabla_1 F(x,y) = \Psi^*\Phi^*(\Phi\Psi x - y)$,
$\operatorname{Prox}_{\tau G_i}(x) = \mathcal{B}_i^* T_{\lambda\tau}(\mathcal{B}_i x)$ (G_i is simple)

where $T_\rho(b)$ for $b \in \mathbb{R}^B$ is the block-wise soft-thresholding

$$T_\rho(b)_i = \max(0, 1 - \rho/\|b_i\|) b_i$$



(a) Non-overlapping blocks

Generalized Forward Backward (GFB) [Raguet et al., 2011]

Solve: $x(y) \in \underset{x}{\operatorname{argmin}}\ F(x,y) + G(x)$ where $G(x) = \sum_{i=1}^{Q} G_i(x)$.

where $x \mapsto F(x,y)$ C^1 with L-Lipschitz gradient
$x \mapsto G_i(x)$ simple

G does not have to be simple!

GFB Scheme and Derivatives

The following sequence converges to $x(y)$

$$x^{(\ell+1)} = \frac{1}{Q} \sum_{i=1}^{Q} z_i^{(\ell+1)}$$

$$z_i^{(\ell+1)} = z_i^{(\ell)} - x^{(\ell)} + \operatorname{Prox}_{n\gamma G_i}(u^{(\ell)})$$

$$u^{(\ell)} = 2x^{(\ell)} - z_i^{(\ell)} - \gamma \nabla_1 F(x^{(\ell)}, y)$$

Generalized Forward Backward and Derivatives

Generalized Forward Backward (GFB) [Raguet et al., 2011]

Solve: $x(y) \in \underset{x}{\operatorname{argmin}}\ F(x, y) + G(x)$ where $G(x) = \sum_{i=1}^{Q} G_i(x)$.

where
$x \mapsto F(x, y)$ — C^1 with L-Lipschitz gradient
$x \mapsto G_i(x)$ — simple

G does not have to be simple!

GFB Scheme and Derivatives

The following sequence converges to $x(y)$

$$x^{(\ell+1)} = \frac{1}{Q} \sum_{i=1}^{Q} z_i^{(\ell+1)}$$

$$z_i^{(\ell+1)} = z_i^{(\ell)} - x^{(\ell)} + \mathrm{Prox}_{n\gamma G_i}(u^{(\ell)})$$

$$u^{(\ell)} = 2x^{(\ell)} - z_i^{(\ell)} - \gamma \nabla_1 F(x^{(\ell)}, y)$$

Computation of GSURE associated to $x^{(\ell)}(y)$ depends on $\xi^{(\ell)} = \partial x^{(\ell)}(y)[\delta]$

Generalized Forward Backward and Derivatives

Generalized Forward Backward (GFB) [Raguet et al., 2011]

Solve: $x(y) \in \underset{x}{\operatorname{argmin}}\ F(x,y) + G(x)$ where $G(x) = \sum_{i=1}^{Q} G_i(x)$.

where $x \mapsto F(x,y)$ C^1 with L-Lipschitz gradient
$x \mapsto G_i(x)$ simple

G does not have to be simple!

GFB Scheme and Derivatives

The following sequence converges to $x(y)$

$$
\begin{aligned}
x^{(\ell+1)} &= \frac{1}{Q}\sum_{i=1}^{Q} z_i^{(\ell+1)} \\
z_i^{(\ell+1)} &= z_i^{(\ell)} - x^{(\ell)} + \operatorname{Prox}_{n\gamma G_i}(u^{(\ell)}) \\
u^{(\ell)} &= 2x^{(\ell)} - z_i^{(\ell)} - \gamma \nabla_1 F(x^{(\ell)}, y)
\end{aligned}
$$

Apply the chain rule

$$
\begin{aligned}
\xi^{(\ell+1)} &= \frac{1}{Q}\sum_{i=1}^{Q} \zeta_i^{(\ell+1)} \\
\zeta_i^{(\ell+1)} &= \zeta_i^{(\ell)} - \xi^{(\ell)} + \mathcal{G}_i^{(\ell)}(\Xi^{(\ell)}) \\
\Xi^{(\ell)} &= 2\xi^{(\ell)} - \zeta_i^{(\ell)} - \gamma(\mathcal{F}_1^{(\ell)}(\xi^{(\ell)}) + \mathcal{F}_2^{(\ell)}(\delta))
\end{aligned}
$$

where

$$
\begin{aligned}
\zeta_i^{(\ell)} &= \partial z_i^{(\ell)}(y)[\delta] \quad \text{and} \quad \mathcal{G}_i^{(\ell)} = \partial \operatorname{Prox}_{n\gamma G_i}(u^{(\ell)}) \\
\Xi^{(\ell)} &= \partial u^{(\ell)}(y)[\delta] \quad \text{and} \quad \mathcal{F}_k^{(\ell)} = \partial_k \nabla_1 F(x^{(\ell)}, y)
\end{aligned}
$$

Computation of GSURE associated to $x^{(\ell)}(y)$ depends on $\xi^{(\ell)} = \partial x^{(\ell)}(y)[\delta]$

Generalized Forward Backward (GFB) [Raguet et al., 2011]

Solve: $x(y) \in \underset{x}{\operatorname{argmin}}\ F(x, y) + G(x)$ where $G(x) = \sum_{i=1}^{Q} G_i(x)$.

where $x \mapsto F(x, y)$ — C^1 with L-Lipschitz gradient
$x \mapsto G_i(x)$ — simple

G does not have to be simple!

Example (Block sparsity)

- Recall that the gradient and proximal operators are

$$\nabla_1 F(x, y) = \Psi^* \Phi^* (\Phi \Psi x - y),$$
$$\mathrm{Prox}_{\tau G_i}(x) = \boldsymbol{\mathcal{B}}_i{}^* T_{\lambda\tau}(\boldsymbol{\mathcal{B}}_i x)$$

- Their derivatives

$$\partial_1 \nabla_1 F(x, y)[\delta_x] = \Psi^* \Phi^* \Phi \Psi \delta_x$$
$$\partial_2 \nabla_1 F(x, y)[\delta_y] = -\Psi^* \Phi^* \delta_y$$
$$\partial\, \mathrm{Prox}_{\tau G_i}(x)[\delta_x] = \boldsymbol{\mathcal{B}}_i{}^* \partial T_{\lambda\tau}(\boldsymbol{\mathcal{B}}_i \delta_x)$$

where $\partial T_\rho(b)$ for $b \in \mathbb{R}^B$ and $\delta_b \in \mathbb{R}^B$ is

$$\partial T_\rho(b)[\delta_b]_i = \begin{cases} 0 & \text{if } \|b_i\| \leqslant \rho \\ \delta_{b,i} - \frac{\rho}{\|b_i\|} P_{b_i}(\delta_{b,i}) & \text{otherwise} \end{cases}$$

where P_α is the orthogonal projector on $\alpha^\perp$ for $\alpha \in \mathbb{R}^B$

Other schemes

We have considered most known proximal splitting schemes:

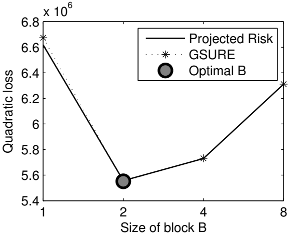
- Primal: Forward-Backward and Douglas-Rachford are encompassed in GFB
- Dual: ADMM
- Primal-dual: Chambolle-Pock algorithm

Summary

1. Choose a proximal splitting scheme
2. For a given y and parameter θ, run the algorithm
    - Compute iterates $x_\theta^{(\ell)}(y)$
    - Compute derivatives applied to k standard iid Gaussian vectors δ_i
3. Compute $\mathrm{GSURE}(\Phi x_\theta^{(\ell)}, y)$ by empirical average
4. Repeat 2-3 and choose θ that minimizes GSURE

Outline

Numerical Examples

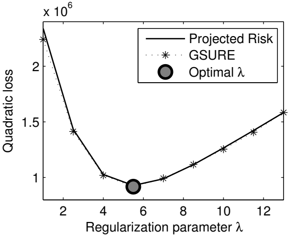


(b)

(c) $\Phi x_0(y)$

(d) $x_B(y)$ at the optimal B

Figure: Φ random CS matrix ($P/N = 0.5$). $G(x) = \lambda \|\mathcal{B}x\|$. Optimization of the block size B.

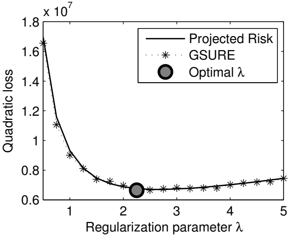


(a)

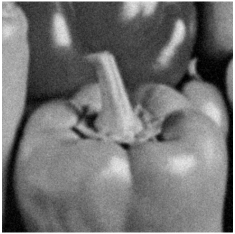
(b) y

(c) $x_\lambda(y)$ at the optimal λ

Figure: Φ sub-sampling matrix ($P/N = 0.5$). $G(x) = \lambda \|\nabla x\|$. Optimization of λ.



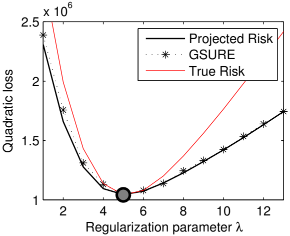
(a)



(b) y

(c) $x_\lambda(y)$ at the optimal λ

Figure: Φ Gaussian convolution ($P = N$, width 2px). $G(x) = \lambda \|\nabla x\|$. Optimization of λ.



(a)

(b) $\Phi x_0(y)$

(c) $x_\lambda(y)$ at the optimal λ

Figure: Φ random CS matrix ($P/N = 0.5$). $G(x) = \lambda \|\nabla x\|$. Optimization of λ

Conclusion

Risk estimation for linear inverse problems

- Solver: Iterative proximal splitting algorithms
- Derivative: Use the chain rule to derive the sequence of iterates
- Risk: The derivatives provide you the GSURE
- Exhaustive search: Evaluate for different parameters and select the optimal one

Future work

- Optimize jointly several parameters
- Avoid exhaustive search

Thanks for your attention

[Eldar, 2009] Eldar, Y. C. (2009).
Generalized SURE for exponential families: Applications to regularization.
IEEE Transactions on Signal Processing, 57(2):471–481.

[Raguet et al., 2011] Raguet, H., Fadili, J., and Peyré, G. (2011).
Generalized forward-backward splitting.
Technical report, Preprint Hal-??

[Stein, 1981] Stein, C. (1981).
Estimation of the mean of a multivariate normal distribution.
The Annals of Statistics, 9(6):1135–1151.

[Vonesch et al., 2008] Vonesch, C., Ramani, S., and Unser, M. (2008).
Recursive risk estimation for non-linear image deconvolution with a wavelet-domain sparsity constraint.
In *ICIP*, pages 665–668. IEEE.