Covariant LEAst-square Re-fitting for image restoration

N. Papadakis
Joint work with C.-A. Deledalle, J. Salmon & S. Vaiter

1 CNRS, Institut de Mathématiques de Bordeaux
2 University California San Diego
3 Université de Montpellier, IMAG
4 CNRS, Institut de Mathématiques de Bourgogne

Variational methods and optimization in imaging

February 5th 2019
Introduction to Re-fitting

Invariant LEAst square Re-fitting

Covariant LEAst-square Re-fitting

Practical considerations and experiments

Conclusions
1. Introduction to Re-fitting

What is Re-fitting?

True piecewise constant signal $x_0$
What is Re-fitting?

Observations \( y = x_0 + w \)
1. Introduction to Re-fitting

What is Re-fitting?

Total Variation (TV) restoration \( \hat{x}(y) = \arg\min_x \frac{1}{2}||y - x||^2 + \lambda||\nabla x||_1 \)
What is Re-fitting?

Total Variation (TV) restoration \( \hat{x}(y) = \arg\min_x \frac{1}{2}||y - x||^2 + \lambda ||\nabla x||_1 \)

- Problem: \( \hat{x}(y) \) is biased [Strong and Chan 1996], i.e., \( \mathbb{E}\hat{x}(y) \neq x_0 \).
1. Introduction to Re-fitting

What is Re-fitting?

Problem: \( \hat{x}(y) \) is biased [Strong and Chan 1996], i.e., \( \mathbb{E}\hat{x}(y) \neq x_0 \).

A solution: Compute \( R_{\hat{x}}(y) \) as the mean of \( y \) on each piece of \( \hat{x}(y) \).
1. Introduction to Re-fitting

Which bias reduction with Re-fitting?

Noise free image

Noisy data

Anisotropic TV

Error
1. Introduction to Re-fitting

Which bias reduction with Re-fitting?

Noise free image

Noisy data

Re-fitting

Error
Re-fitting arbitrary models

Can we generalize this approach to other estimators than TV?

- General setting
  \[ \hat{x}(y) = \arg \min_x F(x, y) + \lambda G(x) \]

- Data fidelity w.r.t observations \( y \): \( F \) convex
- Prior model on the solution: \( G \) convex
- Regularization parameter: \( \lambda > 0 \)

For inverse problems

- \( F(x, y) = F(\Phi x - y) \)
- \( \Phi \) is a linear operator: convolution, mask...
1. Introduction to Re-fitting

Related works


![Graph](image)

Apply the model to the residual $y - \Phi \hat{x}(y)$
1. Introduction to Re-fitting

Related works


Apply the model to the residual $y - \Phi \hat{x}(y)$

Iterative process

- $x_0 = \arg\min_x F(\Phi x - y) + \lambda G(x)$
- $x_k = x_{k-1} + \arg\min_x F(\Phi x - (y - \Phi x_{k-1})) + \lambda_k G(x)$
1. Introduction to Re-fitting

Related works


Apply the model to the residual $y - \Phi \hat{x}(y)$

Iterative process

- $x_0 = \arg\min_x F(\Phi x - y) + \lambda G(x)$
- $x_k = x_{k-1} + \arg\min_x F(\Phi x - (y - \Phi x_{k-1})) + \lambda_k G(x)$

Limitations

- Support not preserved
- How many iterations?
- Varying parameters $\lambda_k$
1. Introduction to Re-fitting

Objectives

Automatic re-fitting process

$$\hat{x}(y) = \arg\min_x F(x, y) + \lambda G(x)$$

- Keep structures and regularity present in the biased solution
- Correct the model bias
- No additional parameter

Bias-variance trade-off

- Bias reduction is not always favorable in terms of MSE
- Re-fitting re-injects part of the variance

Handle existing “black box” algorithms

- Non-Local Means [Buades et al. 2005]
- BM3D [Dabov et al. 2007], DDID [Knaus & Zwicker 2013]
- Trained Deep Networks
1. Introduction to Re-fitting

Objectives

Automatic re-fitting process

\[ \hat{x}(y) = \underset{x}{\arg\min} F(x, y) + \lambda G(x) \]

- Keep structures and regularity present in the biased solution
- Correct the model bias
- No additional parameter

Handle existing “black box” algorithms \( \hat{x}(y) \)

- Non-Local Means [Buades et al. 2005]
- BM3D [Dabov et al. 2007], DDID [Knaus & Zwicker 2013]
- Trained Deep Networks
1. Introduction to Re-fitting

Objectives

Automatic re-fitting process

\[ \hat{x}(y) = \arg\min_x F(x, y) + \lambda G(x) \]

- Keep structures and regularity present in the biased solution
- Correct the model bias
- No additional parameter

Handle existing “black box” algorithms \( \hat{x}(y) \)

- Non-Local Means [Buades et al. 2005]
- BM3D [Dabov et al. 2007], DDID [Knaus & Zwicker 2013]
- Trained Deep Networks

Bias-variance trade-off

- Bias reduction is not always favorable in terms of MSE
- Re-fitting re-injects part of the variance
1. Introduction to Re-fitting

Objectives

Automatic re-fitting process

\[ \hat{x}(y) = \arg\min_x F(x, y) + \lambda G(x) \]

- Keep structures and regularity present in the biased solution
- Correct the model bias
- No additional parameter

Handle existing “black box” algorithms \( \hat{x}(y) \)

- Non-Local Means [Buades et al. 2005]
- BM3D [Dabov et al. 2007], DDID [Knaus & Zwicker 2013]
- Trained Deep Networks

Bias-variance trade-off

- Bias reduction is not always favorable in terms of MSE
- Re-fitting re-injects part of the variance

\[ \text{MSE is not expected to be reduced with re-fitting} \]
Introduction to Re-fitting

Invariant LEAst square Re-fitting

Covariant LEAst-square Re-fitting

Practical considerations and experiments

Conclusions
2. Invariant LEAst square Re-fitting

Formalizing Re-fitting for TV

- **Linear inverse problem:**
  \[
  y = \Phi x_0 + w, \quad y \in \mathbb{R}^p, \quad x_0 \in \mathbb{R}^n, \quad \Phi \in \mathbb{R}^{p \times n}, \quad \mathbb{E}[w] = 0_p
  \]
  - observation
  - signal of interest
  - linear operator
  - white noise

- **TV regularization:**
  \[
  \hat{x}(y) = \arg\min_x \frac{1}{2} ||\Phi x - y||^2 + \lambda ||\nabla x||_1
  \]

- **Re-fitting TV (constrained least-square [Efron et al. 2004, Lederer 2013]):**
  \[
  \tilde{x}(y) \in \arg\min_{x \in \mathcal{M}_{\hat{x}}(y)} ||\Phi x - y||^2
  \]
  with \(\mathcal{M}_{\tilde{x}}(y)\) the **model subspace:**
  \[
  \mathcal{M}_{\tilde{x}}(y) = \{ x \in \mathbb{R}^n \mid \forall i, (\nabla \hat{x}(y))_i = 0 \Rightarrow (\nabla x)_i = 0 \}
  \]

- **Set of signals with same jumps (co-support)**
Generalizing TV Re-fitting procedure

- Reformulating without the notion of jumps
- Understanding what is captured by $M_{\hat{x}}(y)$

**Idea:** $M_{\hat{x}}(y)$ captures linear invariances of $\hat{x}(y)$ w.r.t small perturbations on $y$
Invariant Re-fitting [Deledalle, P. & Salmon 2015]

- Piece-wise affine mapping $y \mapsto \hat{x}(y)$

$$\hat{x}(y) = \arg\min_x F(\Phi x, y) + \lambda G(x)$$
2. Invariant LEAst square Re-fitting

**Invariant Re-fitting** [Deledalle, P. & Salmon 2015]

- Piece-wise affine mapping $y \mapsto \hat{x}(y)$

\[
\hat{x}(y) = \arg\min_x F(\Phi x, y) + \lambda G(x)
\]

- Jacobian of the estimator:

\[
\left( J_{\hat{x}}(y) \right)_{ij} = \frac{\partial \hat{x}(y)}{\partial y_j}
\]

- Model subspace:

\[
M_{\hat{x}}(y) = \hat{x}(y) + \text{Im} \left[ J_{\hat{x}}(y) \right]
\]

- Invariant Least square Re-fitting:

\[
R_{\text{inv}} \hat{x}(y) = \arg\min_{x \in M_{\hat{x}}(y)} \| \Phi x - y \|_2^2
\]
**Invariant Re-fitting** [Deledalle, P. & Salmon 2015]

- Piece-wise affine mapping $y \mapsto \hat{x}(y)$

$$
\hat{x}(y) = \arg\min_x F(\Phi x, y) + \lambda G(x)
$$
2. Invariant LEAst square Re-fitting

**Invariant Re-fitting** [Deledalle, P. & Salmon 2015]

- Piece-wise affine mapping $y \mapsto \hat{x}(y)$
  $$\hat{x}(y) = \arg\min_{x} F(\Phi x, y) + \lambda G(x)$$

- Jacobian of the estimator:
  $$(J\hat{x}(y))_{ij} = \frac{\partial \hat{x}(y)_i}{\partial y_j}$$
Invariant Re-fitting [Deledalle, P. & Salmon 2015]

- Piece-wise affine mapping $y \mapsto \hat{x}(y)$
  \[ \hat{x}(y) = \arg\min_x F(\Phi x, y) + \lambda G(x) \]

- Jacobian of the estimator:
  \[ (J_{\hat{x}(y)})_{ij} = \frac{\partial \hat{x}(y)_i}{\partial y_j} \]

- Model subspace:
  \[ \mathcal{M}_{\hat{x}}(y) = \hat{x}(y) + \text{Im}[J_{\hat{x}(y)}] \]

Tangent space of the mapping
2. Invariant LEAst square Re-fitting

Invariant Re-fitting [Deledalle, P. & Salmon 2015]

- Piece-wise affine mapping $y \mapsto \hat{x}(y)$
  $$\hat{x}(y) = \arg\min_x F(\Phi x, y) + \lambda G(x)$$

- Jacobian of the estimator:
  $$\left( J\hat{x}(y) \right)_{ij} = \frac{\partial \hat{x}(y)_i}{\partial y_j}$$

- Model subspace:
  $$\mathcal{M}_{\hat{x}}(y) = \hat{x}(y) + \text{Im}[J\hat{x}(y)]$$

- Invariant Least square Re-fitting:
  $$\mathcal{R}_{\hat{x}}^{\text{inv}}(y) = \arg\min_{x \in \mathcal{M}_{\hat{x}}(y)} \| \Phi x - y \|^2$$

Tangent space of the mapping
2. Invariant LEASt square Re-fitting

**Practical Re-fitting for $\ell_1$ analysis minimization**

\[
\hat{x}(y) = \arg\min_x \frac{1}{2}||\Phi x - y||^2 + \lambda||\Gamma x||_1
\]  

(1)

Remark: $\Gamma = \operatorname{Id}$ is the LASSO, $\Gamma = [\nabla_x, \nabla_y]^\top$ is the Anisotropic TV.

**Numerical stability issue**

- Piecewise constant solution [Strong & Chan 2003, Caselles et al., 2009]
- Re-fitting $\hat{x}(y)$ requires the support: $\hat{I} = \{i \in [1, m], \text{s.t. } |\Gamma \hat{x}|_i > 0\}$
- But in practice, $\hat{x}$ is only approximated through a converging sequence $\hat{x}^k$
- Unfortunately, $\hat{x}^k \approx \hat{x} \not\Rightarrow \hat{I}^k \approx \hat{I}$
- Illustration for Anisotropic TV denoising ($\Phi = \operatorname{Id}$):
Practical Re-fitting for $\ell_1$ analysis minimization

$$\hat{x}(y) = \arg\min_x \frac{1}{2} ||\Phi x - y||^2 + \lambda ||\Gamma x||_1$$  \hspace{1cm} (1)

Remark: $\Gamma = \text{Id}$ is the LASSO, $\Gamma = [\nabla_x, \nabla_y]^\top$ is the Anisotropic TV.

Proposed approach

- Provided $\ker \Phi \cap \ker \Gamma = \{0\}$, one has for (1) [Vaiter et al. 2016]:
  $$R_{\hat{x}}^{\text{inv}}(y) = J_y[y] \quad \text{with} \quad J_y = \frac{\partial \hat{x}(y)}{\partial y} \bigg|_y$$

- Interpretation: $R_{\hat{x}}^{\text{inv}}(y)$ is the derivative of $\hat{x}(y)$ in the direction of $y$

- Algorithm: Compute $\tilde{x}^k$ by chain rule as the derivative of $\hat{x}^k(y)$ in the direction of $y$

- Question: Does $\tilde{x}^k$ converge towards $R_{\hat{x}}^{\text{inv}}(y)$?
2. Invariant LEAst square Re-fitting

Practical Re-fitting for $\ell_1$ analysis minimization

$$\hat{x}(y) = \underset{x}{\operatorname{argmin}} \frac{1}{2}||\Phi x - y||^2 + \lambda||\Gamma x||_1$$

(1)

Remark: $\Gamma = \text{Id}$ is the LASSO, $\Gamma = [\nabla_x, \nabla_y]^\top$ is the Anisotropic TV.

Proposed approach

- Provided $\text{Ker } \Phi \cap \text{Ker } \Gamma = \{0\}$, one has for (1) [Vaiter et al. 2016]:

$$R_{\hat{x}}^{\text{inv}}(y) = J_y[y] \quad \text{with} \quad J_y = \frac{\partial \hat{x}(y)}{\partial y} \bigg|_y$$

- Interpretation: $R_{\hat{x}}^{\text{inv}}(y)$ is the derivative of $\hat{x}(y)$ in the direction of $y$

- Algorithm: Compute $\tilde{x}^k$ by chain rule as the derivative of $\hat{x}^k(y)$ in the direction of $y$

- Question: Does $\tilde{x}^k$ converge towards $R_{\hat{x}}^{\text{inv}}(y)$?

Yes, at least for the ADMM or the Chambolle-Pock sequences
2. Invariant LEAst square Re-fitting

**Implementation for Anisotropic TV**

\[
\hat{x}(y) = \arg\min_x \frac{1}{2} ||y - x||^2 + \lambda ||\Gamma x||_1
\]

- Primal-dual implementation [Chambolle and Pock 2011]:

\[
\begin{align*}
    z^{k+1} &= \text{Proj}_{B_\lambda} \left( z^k + \sigma \Gamma x^k \right) \\
    x^{k+1} &= x^k + \tau \left( y - \Gamma^\top \left( z^{k+1} \right) \right) / (1 + \tau)
\end{align*}
\]

- Projection: \( \text{Proj}_{B_\lambda}(z)_i = \begin{cases} 
    z_i & \text{if } |z_i| \leq \lambda \\
    \lambda \text{sign}(z_i) & \text{otherwise}
\end{cases} \)

Complexity: twice that of the Chambolle-Pock algorithm.
## Implementation for Anisotropic TV

\[
\hat{x}(y) = \arg\min_x \frac{1}{2}\|y - x\|^2 + \lambda\|\Gamma x\|_1
\]

- **Primal-dual implementation** [Chambolle and Pock 2011]:

\[
\begin{align*}
\tilde{z}^{k+1} &= \text{Proj}_{B_\lambda}(z^k + \sigma \Gamma x^k) \\
\tilde{x}^{k+1} &= \text{Proj}_{B_\lambda(z^k + \sigma \Gamma x^k)}(\frac{x^k + \tau(y - \Gamma^\top(\tilde{z}^{k+1}))}{1+\tau}) \\
x^{k+1} &= \frac{x^k + \tau(y - \Gamma^\top(\tilde{z}^{k+1}))}{1+\tau}
\end{align*}
\]

- **Projection:**

\[
\text{Proj}_{B_\lambda}(z)_i = \begin{cases} 
  z_i & \text{if } |z_i| \leq \lambda \\
  \lambda \text{sign}(z_i) & \text{otherwise}
\end{cases}
\]

\[
P_z = \begin{cases} 
  \text{Id} & \text{if } |z_i| \leq \lambda + \beta \\
  0 & \text{otherwise}
\end{cases}
\]
2. Invariant LEAst square Re-fitting

**Implementation for Anisotropic TV**

\[ \hat{x}(y) = \arg\min_x \frac{1}{2} ||y - x||^2 + \lambda ||\Gamma x||_1 \]

- **Primal-dual implementation** [Chambolle and Pock 2011]:

\[
\begin{cases}
\tilde{z}^{k+1} = \text{Proj}_{B_\lambda} (z^k + \sigma \Gamma x^k) \\
\tilde{x}^{k+1} = \text{Proj}_{z^{k+1} + \sigma \Gamma x^k} (z^k + \sigma \Gamma \tilde{x}^k) \\
x^{k+1} = \frac{x^k + \tau (y - \Gamma^\top (z^{k+1}))}{1+\tau} \\
\tilde{x}^{k+1} = \frac{\tilde{x}^k + \tau (y - \Gamma^\top (\tilde{z}^{k+1}))}{1+\tau}
\end{cases}
\]

- **Projection:**

\[
\text{Proj}_{B_\lambda} (z)_i = \begin{cases} 
z_i & \text{if } |z_i| \leq \lambda \\
\lambda \text{sign}(z_i) & \text{otherwise}
\end{cases}
\]

\[
P_z = \begin{cases} 
\text{Id} & \text{if } |z_i| \leq \lambda + \beta \\
0 & \text{otherwise}
\end{cases}
\]

**Theorem:** The sequence \( \tilde{x}^k \) converges to the re-fitting \( R_{\hat{x}}^{\text{inv}}(y) \) of \( \hat{x}(y) \), \( \forall \beta > 0 \) s.t. \( \beta < \sigma |\Gamma \hat{x}(y)|_i, \forall i \in I \)
2. Invariant LEAst square Re-fitting

**Implementation for Anisotropic TV**

\[ \hat{x}(y) = \arg\min_x \frac{1}{2}||y - x||^2 + \lambda||\Gamma x||_1 \]

- **Primal-dual implementation** [Chambolle and Pock 2011]:

\[
\begin{align*}
\hat{z}^{k+1} &= \text{Proj}_{B_{\lambda}} (z^k + \sigma \Gamma x^k) \\
\hat{x}^{k+1} &= \frac{x^k + \tau (y - \Gamma^\top (\hat{z}^{k+1}))}{1 + \tau} \\
\tilde{x}^{k+1} &= \frac{\tilde{x}^k + \tau (y - \Gamma^\top (\hat{z}^{k+1}))}{1 + \tau}
\end{align*}
\]

- **Projection:**

\[
\text{Proj}_{B_{\lambda}}(z)_i = \begin{cases} 
  z_i & \text{if } |z_i| \leq \lambda \\
  \lambda \text{sign}(z_i) & \text{otherwise}
\end{cases}
\]

\[
P_z = \begin{cases} 
  \text{Id} & \text{if } |z_i| \leq \lambda + \beta \\
  0 & \text{otherwise}
\end{cases}
\]

**Theorem:** The sequence \(\tilde{x}^k\) converges to the re-fitting \(R^{\text{inv}}_{\hat{x}}(y)\) of \(\hat{x}(y)\), \(\forall \beta > 0\) s.t. \(\beta < \sigma |\Gamma \hat{x}(y)|_i, \forall i \in I\)

**Complexity:** twice that of the Chambolle-Pock algorithm.
Anisotropic TV: illustration

\[ y \]

\[ \hat{x}(y) \]

\[ R_{\hat{x}}^{\text{inv}}(y) \]
2. Invariant LEAst square Re-fitting

Anisotropic TV: Bias-variance trade-off

PSNR: 22.45, SSIM: 0.416

PSNR: 24.80, SSIM: 0.545

PSNR: 27.00, SSIM: 0.807

Regularization parameter $\lambda$

0.2 0.4 0.6 0.8 1 1.2 1.4 1.6 1.8 2

Quadratic cost

80
100
120
140
160
180
200
220
240

Original $\hat{x}(y)$

Re-fitted $\hat{R}(y)$

Optimum original

Optimum re-fitted

Sub-optimum original

Sub-optimum re-fitted

PSNR: 28.89, SSIM: 0.809

PSNR: 28.28, SSIM: 0.823

Anisotropic TV

CLEAR
2. Invariant LEAst square Re-fitting

Anisotropic TV: Bias-variance trade-off

Anisotropic TV

CLEAR

N. Papadakis

CLEAR 15 / 41
2. Invariant LEAst square Re-fitting

**Anisotropic TV: Bias-variance trade-off**

<table>
<thead>
<tr>
<th>Quadratic cost</th>
<th>Regularization parameter $\lambda$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$10^0$</td>
<td>1 2 3 4 5 6 7 8 9 10</td>
</tr>
<tr>
<td>$10^1$</td>
<td></td>
</tr>
<tr>
<td>$10^2$</td>
<td></td>
</tr>
</tbody>
</table>

**PSNR:**
- Anisotropic TV: 22.84, SSIM: 0.312
- CLEAR: 23.96, SSIM: 0.694
- Anisotropic TV: 35.99, SSIM: 0.938
- CLEAR: 38.22, SSIM: 0.935
- Anisotropic TV: 46.42, SSIM: 0.986

N. Papadakis
CLEAR 15 / 41
Example: Anisotropic TGV

\[ x_0 \]

\[ \hat{x}(y) \]

\[ R_{\hat{x}}^{\text{inv}}(y) \]
Example: Isotropic TV

- Restoration model for image $y$

$$\hat{x}(y) = \arg\min_x \frac{1}{2} ||y - x||^2 + \lambda ||\nabla x||_{1,2}$$

- Model subspace of isotropic TV is the same than anisotropic TV:

  signals whose gradients share their support with $\nabla \hat{x}(y)$
Example: Isotropic TV

- Restoration model for image $y$

$$\hat{x}(y) = \arg\min_x \frac{1}{2} ||y - x||^2 + \lambda ||\nabla x||_{1,2}$$

- Model subspace of isotropic TV is the same than anisotropic TV:

  signals whose gradients share their support with $\nabla \hat{x}(y)$

Non sparse support: noise is re-injected

Illustration done with an ugly standard (i.e. non Condat and non Chambolle-Pock) discretization of isotropic TV
Limitations

Model subspace
- Only captures linear invariances w.r.t. small perturbations of $y$

Jacobian matrix
- Captures desirable covariant relationships between the entries of $y$ and the entries of $\hat{x}(y)$ that should be preserved [Deledalle, P., Salmon and Vaiter, 2017, 2019]
Introduction to Re-fitting

Invariant LEAst square Re-fitting

**Covariant LEAst-square Re-fitting**

Practical considerations and experiments

Conclusions
3. Covariant LEAst-square Re-fitting

**Least-square Re-fitting**

**General problem**

\[ \hat{x}(y) = \arg\min_x F(\Phi x, y) + \lambda G(x) \]

- \(\Phi\) linear operator, \(F\) and \(G\) convex
3. Covariant LEAst-square Re-fitting

**Least-square Re-fitting**

**General problem**

\[
\hat{x}(y) = \arg\min_x F(\Phi x, y) + \lambda G(x)
\]

- \(\Phi\) linear operator, \(F\) and \(G\) convex

**Desirable properties of Re-fitting operator**

1. \(h \in \mathcal{H}_{\hat{x}}\) iff \(h \in \mathcal{M}_{\hat{x}}(y)\)
2. Affine map: \(h(y) = Ay + b\)
3. Preservation of covariants: \(J_h(y) = \rho J_{\hat{x}}(y)\)
4. Coherence: \(h(\Phi \hat{x}(y)) = \hat{x}(y)\)
3. Covariant LEAest-square Re-fitting

Least-square Re-fitting

General problem

\[ \hat{x}(y) = \arg\min_x F(\Phi x, y) + \lambda G(x) \]

- $\Phi$ linear operator, $F$ and $G$ convex

Desirable properties of Re-fitting operator

1. $h \in \mathcal{H}_{\hat{x}}$ iff $h \in \mathcal{M}_{\hat{x}}(y)$
2. Affine map: $h(y) = Ay + b$
3. Preservation of covariants: $J_h(y) = \rho J_{\hat{x}}(y)$
4. Coherence: $h(\Phi \hat{x}(y)) = \hat{x}(y)$

Covariant LEAest-square Re-fitting

\[ \mathcal{R}_{\hat{x}}^{\text{cov}}(y) = \arg\min_{x \in \mathcal{H}_{\hat{x}}} \frac{1}{2} ||\Phi x - y||^2 \]
Covariant LEAst-square Re-fitting

Proposition

The covariant Re-fitting has an explicit formulation

\[ \mathcal{R}_{\hat{x}}^{\text{cov}}(y) = \hat{x}(y) + \rho J(y - \Phi \hat{x}(y)) = \arg\min_{x \in \mathcal{H}_{\hat{x}}} \frac{1}{2} \| \Phi x - y \|^2 \]

where for \( \delta = y - \Phi \hat{x}(y) \):

\[ \rho = \begin{cases} 
\frac{\langle \Phi J \delta, \delta \rangle}{\| \Phi J \delta \|^2} & \text{if } \Phi J \delta \neq 0 \\
1 & \text{otherwise}
\end{cases} \]
3. Covariant LEAst-square Re-fitting

**Covariant LEAst-square Re-fitting**

**Proposition**

The covariant Re-fitting has an explicit formulation

$$R_{\hat{x}}^{\text{cov}}(y) = \hat{x}(y) + \rho J(y - \Phi \hat{x}(y)) = \arg\min_{x \in \mathcal{H}_{\hat{x}}} \frac{1}{2} \|\Phi x - y\|^2$$

where for $\delta = y - \Phi \hat{x}(y)$:

$$\rho = \begin{cases} 
\frac{\langle \Phi J \delta, \delta \rangle}{\|\Phi J \delta\|^2} & \text{if } \Phi J \delta \neq 0 \\
1 & \text{otherwise}
\end{cases}$$

**Properties**

- If $\Phi J$ is an orthogonal projector, $\rho = 1$ and $\Phi R_{\hat{x}}^{\text{cov}}(y) = \Phi R_{\hat{x}}^{\text{inv}}(y)$
Covariant LEAst-square Re-fitting

Proposition

The covariant Re-fitting has an explicit formulation

\[ R_{\hat{x}}^{\text{cov}}(y) = \hat{x}(y) + \rho J(y - \Phi \hat{x}(y)) = \arg\min_{x \in \mathcal{H}_{\hat{x}}} \frac{1}{2} \| \Phi x - y \|^2 \]

where for \( \delta = y - \Phi \hat{x}(y) \):

\[ \rho = \begin{cases} 
\frac{\langle \Phi J \delta, \delta \rangle}{\| \Phi J \delta \|^2} & \text{if } \Phi J \delta \neq 0 \\
1 & \text{otherwise}
\end{cases} \]

Properties

- If \( \Phi J \) is an orthogonal projector, \( \rho = 1 \) and \( \Phi R_{\hat{x}}^{\text{cov}}(y) = \Phi R_{\hat{x}}^{\text{inv}}(y) \)
- If \( F \) convex, \( G \) convex and 1-homogenous and
  \[ \hat{x}(y) = \arg\min_{x} F(\Phi x - y) + G(x), \]

then \( J\Phi \hat{x}(y) = \hat{x}(y) \) a.e. so that:

\[ R_{\hat{x}}^{\text{cov}}(y) = (1 - \rho)\hat{x}(y) + \rho Jy \]
Covariant LEAst-square Re-fitting

Proposition

The covariant Re-fitting has an explicit formulation

\[ \hat{R}_x^{cov}(y) = \hat{x}(y) + \rho J(y - \Phi \hat{x}(y)) = \arg\min_{x \in H_x} \frac{1}{2} \| \Phi x - y \|^2 \]

where for \( \delta = y - \Phi \hat{x}(y) \):

\[ \rho = \begin{cases} \frac{\langle \Phi J \delta, \delta \rangle}{\| \Phi J \delta \|^2} & \text{if } \Phi J \delta \neq 0 \\ 1 & \text{otherwise} \end{cases} \]

Properties

- If \( \Phi J \) is an orthogonal projector, \( \rho = 1 \) and \( \Phi \hat{R}_x^{cov}(y) = \Phi \hat{R}_x^{inv}(y) \)
- If \( F \) convex, \( G \) convex and 1-homogenous and

\[ \hat{x}(y) = \arg\min_x F(\Phi x - y) + G(x), \]

then \( J \Phi \hat{x}(y) = \hat{x}(y) \) a.e. so that:

\[ \hat{R}_x^{cov}(y) = (1 - \rho) \hat{x}(y) + \rho J y \]
3. Covariant LEAst-square Re-fitting

**Statistical interpretation**

**Theorem (Bias reduction)**

*If $\Phi J$ is an orthogonal projector or $\rho$ satisfies technical conditions*

\[
\|\Phi (\mathbb{E}[\hat{D}(Y)] - x_0)\|_2 \leq \|\Phi (\mathbb{E}[\hat{T}(Y)] - x_0)\|_2
\]
Example: Isotropic TV

Noise free

Noisy data $y$

$\hat{x}(y)$

$\mathcal{R}^{\text{inv}}(y)$

$\mathcal{R}^{\text{cov}}(y)$
3. Covariant LEAst-square Re-fitting

Why not iterating as Boosting approaches?

- Differentiable estimator w.r.t $y$:

\[ \tilde{x}_0 = \hat{x}(y) = \arg\min_x \|x - y\|^2 + \lambda \|\nabla x\|_{1,2} \]
3. Covariant LEAst-square Re-fitting

Why not iterating as Boosting approaches?

- Differentiable estimator w.r.t $y$:
  \[
  \tilde{x}_0 = \hat{x}(y) = \arg\min_x \|x-y\|^2 + \lambda \|\nabla x\|_{1,2}
  \]

- Bregman iterations:
  \[
  \tilde{x}_{k+1} = \arg\min_x \|x-y\|^2 + \lambda D\|\nabla.\|_{1,2}(x, \tilde{x}_k)
  \]
3. Covariant LEAst-square Re-fitting

Why not iterating as Boosting approaches?

- Differentiable estimator w.r.t $y$:
  \[
  \tilde{x}_0 = \hat{x}(y) = \arg\min_x \|x - y\|^2 + \lambda \|\nabla x\|_{1,2}
  \]

- Bregman iterations:
  \[
  \tilde{x}_{k+1} = \arg\min_x \|x - y\|^2 + \lambda D\|\nabla .\|_{1,2}(x, \tilde{x}_k)
  \]
Why not iterating as Boosting approaches?

- Differentiable estimator w.r.t. $y$:
  \[
  \tilde{x}_0 = \hat{x}(y) = \arg\min_x \|x - y\|^2 + \lambda \|\nabla x\|_{1,2}
  \]

- Bregman iterations:
  \[
  \tilde{x}_{k+1} = \arg\min_x \|x - y\|^2 + \lambda D \|\nabla .\|_{1,2}(x, \tilde{x}_k)
  \]
Why not iterating as Boosting approaches?

- **Differentiable estimator w.r.t \( y \):**

\[
\tilde{x}_0 = \hat{x}(y) = \arg\min_x \|x - y\|^2 + \lambda \|\nabla x\|_{1,2}
\]

- **Bregman iterations:**

\[
\tilde{x}_{k+1} = \arg\min_x \|x - y\|^2 + \lambda D \|\nabla_{x, \tilde{x}_k}\|_{1,2}(x, \tilde{x}_k)
\]

- **Convergence:**

\[
\tilde{x}_k \to y
\]
3. Covariant LEAst-square Re-fitting

Why not iterating as Boosting approaches?

- Differentiable estimator w.r.t $y$:
  \[ \tilde{x}_0 = \hat{x}(y) = \arg\min_x \|x - y\|^2 + \lambda \|\nabla x\|_{1,2} \]

- Covariant iterations:
  \[ \tilde{x}^{k+1}(y) = \tilde{x}^k(y) + \rho J(y - \Phi \tilde{x}^k(y)) \]
3. Covariant LEAst-square Re-fitting

Why not iterating as Boosting approaches?

- Differentiable estimator w.r.t $y$:

$$\tilde{x}_0 = \hat{x}(y) = \arg\min_x \|x - y\|^2 + \lambda \|\nabla x\|_{1,2}$$

- Covariant iterations:

$$\tilde{x}^{k+1}(y) = \tilde{x}^k(y) + \rho J(y - \Phi \tilde{x}^k(y))$$
3. Covariant LEAst-square Re-fitting

Why not iterating as Boosting approaches?

- Differentiable estimator w.r.t $y$:

$$\tilde{x}_0 = \hat{x}(y) = \arg\min_x \|x-y\|^2 + \lambda \|\nabla x\|_{1,2}$$

- Covariant iterations:

$$\tilde{x}_{k+1}(y) = \tilde{x}_k(y) + \rho J(y - \Phi \tilde{x}_k(y))$$

- Convergence:

$$\tilde{x}^k(y) \rightarrow R_{\hat{x}}^{\text{inv}}(y)$$
Introduction to Re-fitting

Invariant LEAst square Re-fitting

Covariant LEAst-square Re-fitting

Practical considerations and experiments

Conclusions
4. Practical considerations and experiments

How computing the covariant Re-fitting?

- Explicit expression:
  \[ R_{x_0}^{\text{cov}}(y) = \hat{x}(y) + \rho J \delta \]
  
  with \( J = \frac{\partial \hat{x}(y)}{\partial y} \), \( \delta = y - \Phi \hat{x}(y) \) and
  
  \[ \rho = \begin{cases} 
  \frac{\langle \Phi J \delta, \delta \rangle}{||\Phi J \delta||^2} & \text{if } \Phi J \delta \neq 0 \\
  1 & \text{otherwise} 
  \end{cases} \]

Main issue

- Being able to compute \( J \delta \)
Algorithmic differentiation

- Iterative algorithm to obtain $\hat{x}(y)$:

$$x^{k+1} = \psi(x^k, y)$$

- Differentiation in the direction $\delta$:

\[
\begin{align*}
    x^{k+1} &= \psi(x^k, y) \\
    \tilde{x}^{k+1} &= \partial_1 \psi(x^k, y)\tilde{x}^k + \partial_2 \psi(x^k, y)\delta
\end{align*}
\]

- $J_{x^k}(y)\delta = \tilde{x}^k$

- Joint estimation of $x^k$ and $J_{x^k}(y)\delta$

- Double the computational cost
4. Practical considerations and experiments

Application of the Jacobian matrix to a vector

Finite difference based differentiation

- \( \hat{x}(y) \) can be any black box algorithm
- Directional derivative w.r.t to direction \( \delta \):

\[
J_{\hat{x}}(y)\delta \approx \frac{\hat{x}(y + \varepsilon \delta) - \hat{x}(y)}{\varepsilon}
\]

- Need to apply twice the algorithm
4. Practical considerations and experiments

Computation of the Re-fitting

Covariant LEAst-square Re-fitting

\[ \mathcal{R}_{\hat{x}}^{\text{cov}}(y) = \hat{x}(y) + \rho J\delta, \quad \text{with} \quad \delta = y - \Phi \hat{x}(y) \quad \text{and} \quad \rho = \frac{\langle J\delta, \delta \rangle}{\|J\delta\|^2_2} \]

Two-steps with any denoising algorithm

1. Apply algorithm to \( y \) to get \( \hat{x}(y) \) and set \( \delta = y - \Phi \hat{x}(y) \)
2. Compute \( J\delta \) (with algorithmic or finite difference based differentiation)
4. Practical considerations and experiments

Computing the Re-fitting

Covariant LEAst-square Re-fitting

\[ R_{\hat{x}}^{cov}(y) = \hat{x}(y) + \rho J \delta, \quad \text{with} \ \delta = y - \Phi \hat{x}(y) \quad \text{and} \quad \rho = \frac{\langle J \delta, \delta \rangle}{\|J \delta\|_2^2} \]

Two-steps with any denoising algorithm

1. Apply algorithm to \( y \) to get \( \hat{x}(y) \) and set \( \delta = y - \Phi \hat{x}(y) \)
2. Compute \( J \delta \) (with algorithmic or finite difference based differentiation)

One-step for absolutely 1-homogeneous regularizer

Re-fitting simplifies to

\[ R_{\hat{x}}^{cov}(y) = (1 - \rho) \hat{x}(y) + \rho J y \]

1. Estimate jointly \( \hat{x}(y) \) and \( J y \) with the differentiated algorithm
1 step Implementation: Anisotropic TV

- Model with 1-homogeneous regularizer:

\[
\hat{x}(y) = \arg\min_x \frac{1}{2}||y - x||^2 + \lambda||\nabla x||_1
\]

- Primal-dual implementation [Chambolle and Pock 2011]:

\[
\begin{align*}
    z^{k+1} &= \text{Proj}_{B_\lambda}(z^k + \sigma \nabla x^k) \\
    \tilde{z}^{k+1} &= P_{z^k + \sigma \nabla x^k}(\tilde{z}^k + \sigma \nabla \tilde{x}^k) \\
    x^{k+1} &= x^k + \tau(y + \text{div}(z^{k+1})) \\
    \tilde{x}^{k+1} &= \frac{\tilde{x}^k + \tau(y + \text{div}(\tilde{z}^{k+1}))}{1 + \tau}
\end{align*}
\]

- Projection:

\[
\text{Proj}_{B_\lambda}(z)^i = \begin{cases} 
    z_i & \text{if } |z_i| \leq \lambda \\
    \lambda \text{sign}(z_i) & \text{otherwise}
\end{cases}
\]

\[
P_z = \begin{cases} 
    \text{Id} & \text{if } |z_i| \leq \lambda + \beta \\
    0 & \text{otherwise}
\end{cases}
\]

- \(x^k \rightarrow \hat{x}(y)\) and \(\tilde{x}^k = J_{x^k} y \rightarrow J_{\tilde{x}} y\)

- \(J\) is an orthogonal projector: \(R_{\hat{x}}^{\text{cov}}(y) = R_{\tilde{x}}^{\text{inv}}(y) = J_{\tilde{x}} y\)
1 step Implementation: Isotropic TV

- Model with 1-homogeneous regularizer:

$$\hat{x}(y) = \arg\min_x \frac{1}{2}||y - x||^2 + \lambda||\nabla x||_{1,2}$$

- Primal-dual implementation [Chambolle and Pock 2011]:

$$\begin{cases} 
    z^{k+1} &= \text{Proj}_{B\lambda} \left( z^k + \sigma \nabla x^k \right) \\
    x^{k+1} &= x^k + \tau \left( y + \text{div}(z^{k+1}) \right) / (1 + \tau)
\end{cases}$$

- Projection:

$$\text{Proj}_{B\lambda}(z)_i = \begin{cases} 
    z_i & \text{if } ||z_i||_2 \leq \lambda \\
    \lambda \frac{z_i}{||z_i||_2} & \text{otherwise}
\end{cases}$$
4. Practical considerations and experiments

1 step Implementation: Isotropic TV

- Model with 1-homogeneous regularizer:

\[
\hat{x}(y) = \arg\min_x \frac{1}{2} \|y - x\|^2 + \lambda \|\nabla x\|_{1,2}
\]

- Primal-dual implementation [Chambolle and Pock 2011]:

\[
\begin{align*}
    z^{k+1} &= \text{Proj}_{B\lambda} (z^k + \sigma \nabla x^k) \\
    \tilde{z}^{k+1} &= P_{z^k + \sigma \nabla x^k} (\tilde{z}^k + \sigma \nabla \tilde{x}^k) \\
    x^{k+1} &= x^k + \tau (y + \text{div}(z^{k+1})) \\
    \tilde{x}^{k+1} &= \tilde{x}^k + \tau (y + \text{div}(\tilde{z}^{k+1}))
\end{align*}
\]

- Projection:

\[
\text{Proj}_{B\lambda} (z)_i = \begin{cases} 
    z_i & \text{if } \|z_i\|_2 \leq \lambda \\
    \frac{\lambda}{\|z_i\|_2} z_i & \text{otherwise}
\end{cases}
\]

\[
P_z = \begin{cases} 
    \text{Id} & \text{if } \|z_i\| \leq \lambda + \beta \\
    \frac{\lambda}{\|z_i\|_2} \left( \text{Id} - \frac{z_i z_i^\top}{\|z_i\|_2^2} \right) & \text{otherwise}
\end{cases}
\]

- \(x^k \rightarrow \hat{x}(y)\) and \(\tilde{x}^k = J_{x^k} y \rightarrow \tilde{x}\)
4. Practical considerations and experiments

1 step Implementation: Isotropic TV

- Model with 1-homogeneous regularizer:

\[
\hat{x}(y) = \arg\min_x \frac{1}{2} ||y - x||^2 + \lambda ||\nabla x||_{1,2}
\]

- Primal-dual implementation [Chambolle and Pock 2011]:

\[
\begin{align*}
    z^{k+1} &= \text{Proj}_{B\lambda} \left( z^k + \sigma \nabla x^k \right) \\
    \tilde{z}^{k+1} &= P_{\tilde{z}^k + \sigma \nabla x^k} \left( \tilde{z}^k + \sigma \nabla \hat{x}^k \right) \\
    x^{k+1} &= \frac{x^k + \tau \left( y + \text{div}(z^{k+1}) \right)}{1 + \tau} \\
    \tilde{x}^{k+1} &= \frac{\tilde{x}^k + \tau \left( y + \text{div}(\tilde{z}^{k+1}) \right)}{1 + \tau}
\end{align*}
\]

- Projection:

\[
\text{Proj}_{B\lambda}(z)_i = \begin{cases} 
    z_i & \text{if } ||z_i||_2 \leq \lambda \\
    \lambda \frac{z_i}{||z_i||_2} & \text{otherwise}
\end{cases}
\]

\[
P_z = \begin{cases} 
    \text{Id} & \text{if } ||z_i|| \leq \lambda + \beta \\
    \frac{\lambda}{||z_i||_2} \left( \text{Id} - \frac{z_i z_i^\top}{||z_i||_2^2} \right) & \text{otherwise}
\end{cases}
\]

- \( x^k \rightarrow \hat{x}(y) \) and \( \tilde{x}^k = J_{x^k} y \rightarrow \tilde{x} \)

- Covariant re-fitting: \( R^\text{cov}_{\hat{x}}(y) = (1 - \rho)\hat{x} + \rho \tilde{x} \), with \( \rho = \frac{\langle \tilde{x} - \hat{x}, y - \hat{x} \rangle}{||\tilde{x} - \hat{x}||_2^2} \)
4. Practical considerations and experiments

Inpainting with Isotropic TV

\[ y \]
\[ \hat{x}(y) \]
\[ R_{\hat{x}}^{\text{cov}}(y) \]

\[ x_0 \]
\[ \| \hat{x}(y) - x_0 \|_2 \]
\[ \| R_{\hat{x}}^{\text{cov}}(y) - x_0 \|_2 \]

Attenuated structures
Residual lost structures
4. Practical considerations and experiments

Extension to chrominance [Pierre, Aujol, Deledalle, P., 2017]

Noise free image  Noisy image  Denoised image
4. Practical considerations and experiments

**Extension to chrominance**  
[Pierre, Aujol, Deledalle, P., 2017]

Noise free image  
Noisy image  
Re-fitting
4. Practical considerations and experiments

2 steps Implementation: Non-Local Means

- Model without 1-homogeneous regularizer:

\[
\hat{x}(y)_i = \frac{\sum_j w^y_{i,j} y_j}{\sum_j w^y_{i,j}}, \quad w^y_{i,j} = \exp\left(-\frac{\|P_i y - P_j y\|_2^2}{h^2}\right)
\]

- Differentiate NLM code

- Algorithm:

  1. Run NLM code \(\hat{x}(y)\) and set \(\delta = y - \hat{x}(y)\)
  2. Run differentiated NLM code in the direction \(\delta\) to get \(J\delta\)
  3. Set \(\rho = \frac{\langle J\delta, \delta \rangle}{||J\delta||_2^2}\)
  4. Covariant re-fitting: \(R^\text{cov}_{\hat{x}}(y) = \hat{x}(y) + \rho J\delta\)
4. Practical considerations and experiments

Non-Local Means

\[ y \]

\[ \hat{x}(y) \]

\[ R_{\hat{x}}^{\text{cov}}(y) \]

\[ x_0 \]

\[ \|\hat{x}(y) - x_0\|_2 \]

\[ \|R_{\hat{x}}^{\text{cov}}(y) - x_0\|_2 \]
Non-Local Means: Bias-variance trade-off

**PSNR:** 22.18, **SSIM:** 0.397

**PSNR:** 25.07, **SSIM:** 0.564

**PSNR:** 26.62, **SSIM:** 0.724

**PSNR:** 30.12, **SSIM:** 0.815

**PSNR:** 29.20, **SSIM:** 0.823
Bias-variance trade-off: Non-Local Means

<table>
<thead>
<tr>
<th>Filtering parameter $h$</th>
<th>Quadratic cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>120</td>
</tr>
<tr>
<td>2</td>
<td>140</td>
</tr>
<tr>
<td>3</td>
<td>160</td>
</tr>
<tr>
<td>4</td>
<td>180</td>
</tr>
<tr>
<td>5</td>
<td>200</td>
</tr>
<tr>
<td>6</td>
<td>220</td>
</tr>
</tbody>
</table>

Non-Local Means

CLEAR

PSNR: 22.18, SSIM: 0.847
PSNR: 24.68, SSIM: 0.914
PSNR: 24.64, SSIM: 0.910
PSNR: 27.04, SSIM: 0.946
PSNR: 27.29, SSIM: 0.955
4. Practical considerations and experiments

Bias-variance trade-off: Non-Local Means

![Image of fingerprint comparison]

- Non-Local Means: PSNR = 27.29, SSIM = 0.955
- CLEAR: PSNR = 27.04, SSIM = 0.946

![Graph showing quadratic cost vs. filtering parameter h]

- Quadratic cost: PSNR = 27.04, SSIM = 0.946
- Filtering parameter h: 1 to 6

N. Papadakis

CLEAR 35 / 41
2 steps Implementation for Black Box algorithm

- Denoising algorithm: \( y \mapsto \hat{x}(y) \)
- Re-fitting with finite difference:
  1. \( \delta = y - \hat{x}(y) \)
  2. \( J\delta = \frac{\hat{x}(y+\varepsilon\delta)-\hat{x}(y)}{\varepsilon} \)
  3. \( \rho = \frac{\langle J\delta,\delta \rangle}{\|J\delta\|^2} \)
  4. \( R_{\hat{x}}^{\text{cov}}(y) = \hat{x}(y) + \rho J\delta \)
4. Practical considerations and experiments

**BM3D** [Dabov et al. 2007, Lebrun 2012]

- **PSNR:** 22.17, **SSIM:** 0.528
- **PSNR:** 25.00, **SSIM:** 0.7523
- **PSNR:** 26.92, **SSIM:** 0.861

![Output of BM3D and CLEAR algorithms](image)

**BM3D**

**CLEAR**
4. Practical considerations and experiments

**DDID [Knaus & Zwicker 2013]**

- **PSNR:** 22.16, **SSIM:** 0.452
- **PSNR:** 26.33, **SSIM:** 0.716
- **PSNR:** 26.60, **SSIM:** 0.721

![Graph showing quadratic cost against filtering parameter log γ]

- **PSNR:** 31.02, **SSIM:** 0.858
- **PSNR:** 29.91, **SSIM:** 0.845

**DDID**

**CLEAR**
4. Practical considerations and experiments

**DnCNN** [Zhang et al., 2017]

Residual Network **learning noise** to remove

![Noise level 25](image)

![DnCNN](image)
4. Practical considerations and experiments

DnCNN [Zhang et al., 2017]

Residual Network learning noise to remove

Noise level 25

CLEAR
DnCNN [Zhang et al., 2017]

Residual Network learning noise to remove

Noise level 50  DnCNN
4. Practical considerations and experiments

DnCNN [Zhang et al., 2017]

Residual Network learning noise to remove

Noise level 50

CLEAR
4. Practical considerations and experiments

**DnCNN [Zhang et al., 2017]**

Residual Network learning noise to remove

Noise level 150

DnCNN
4. Practical considerations and experiments

DnCNN [Zhang et al., 2017]

Residual Network learning noise to remove

Noise level 150

CLEAR
4. Practical considerations and experiments

**DnCNN** [Zhang et al., 2017]

Residual Network learning noise to remove

- Noise level 150
- CLEAR

- **No interesting structural information to recover from noise model**
FFDNet  [Zhang et al., 2018]
Network learning denoised image for Gaussian noise of variance $[0; 75]$
FFDNet [Zhang et al., 2018]

Network learning denoised image for Gaussian noise of variance \([0; 75]\)
FFDNet  [Zhang et al., 2018]

Network learning denoised image for Gaussian noise of variance $[0; 75]$
FFDNet [Zhang et al., 2018]  
Network learning denoised image for Gaussian noise of variance $[0; 75]$
FFDNet [Zhang et al., 2018]

Network learning denoised image for Gaussian noise of variance $[0; 75]$. 

Noise level 150

FFDNet
FFDNet [Zhang et al., 2018]

Network learning denoised image for Gaussian noise of variance [0; 75]
Introduction to Re-fitting

Invariant LEAst square Re-fitting

Covariant LEAst-square Re-fitting

Practical considerations and experiments

Conclusions
Conclusions

Covariant LEAst-square Re-fitting

- Correct part of the bias of restoration models
- No additional parameter
- Stability for a larger range of parameters
- Double the computational cost
5. Conclusions

Conclusions

Covariant LEAst-square Re-fitting

- Correct part of the bias of restoration models
- No additional parameter
- Stability for a larger range of parameters
- Double the computational cost

When using re-fitting?

- Differentiable estimators: no algorithm with quantization
- Regularization prior adapted to data
- Respect data range: oceanography, radiotherapy...
Main related references


Differentiable estimator w.r.t $y$:

$$\hat{x}(y) = \arg\min_x \|x - y\|^2 + \lambda \|
abla x\|_{1,2}$$
Sign in TV models [Brinkmann et al. 2016]

- Differentiable estimator w.r.t $y$:
  \[
  \hat{x}(y) = \arg\min_x \|x - y\|^2 + \lambda \|\nabla x\|_{1,2}
  \]

- Orientation preservation:
  \[
  \mathcal{O}_{\hat{x}(y)} = \{x | (\nabla x)_i = \alpha_i (\nabla \hat{x})_i, \forall i \in \mathcal{I}\}
  \]
Sign in TV models [Brinkmann et al. 2016]

- Differentiable estimator w.r.t $y$:

$$\hat{x}(y) = \arg\min_x \|x - y\|^2 + \lambda \|\nabla x\|_{1,2}$$

- Orientation preservation:

$$\mathcal{O}_{\hat{x}(y)} = \{x| (\nabla x)_i = \alpha_i (\nabla \hat{x})_i, \forall i \in I\}$$

- Infimal Convolution of Bregman divergences

$$\hat{x}^\text{ICB} = \arg\min_{x \in \mathcal{O}_{\hat{x}(y)}} \|x - y\|^2$$
Sign in TV models [Brinkmann et al. 2016]

- Differentiable estimator w.r.t $y$:
  \[
  \hat{x}(y) = \arg\min_x \|x - y\|^2 + \lambda \|\nabla x\|_{1,2}
  \]

- Direction preservation:
  \[
  \mathcal{D}_{\hat{x}(y)} = \{ x \mid (\nabla x)_i = \alpha_i (\nabla \hat{x})_i, \alpha_i \geq 0, \forall i \in I \} 
  \]
Sign in TV models [Brinkmann et al. 2016]

- Differentiable estimator w.r.t $y$:

$$\hat{x}(y) = \arg\min_x \|x - y\|^2 + \lambda \|
abla x\|_{1,2}$$

- Direction preservation:

$$\mathcal{D}_{\hat{x}}(y) = \{ x | (\nabla x)_i = \alpha_i (\nabla \hat{x})_i, \alpha_i \geq 0, \forall i \in \mathcal{I} \}$$

- Bregman divergence

$$\tilde{x}^B = \arg\min_{x \in \mathcal{D}_{\hat{x}}(y)} \|x - y\|^2$$
Sign influence in Re-fitting

\[ \hat{x}(y) \]

Anisotropic TV \[ \hat{x}(y) \]

\[ \mathcal{R}_{\hat{x}}^{\text{inv}} = \mathcal{R}_{\hat{x}}^{\text{cov}} = \hat{x}^{\text{ICB}} \]

Orientation

[Brinkmann et al.]

Direction

[Brinkmann et al.]
New Re-fitting models

- Covariant Re-fitting:

\[
R_{\hat{x}}^{\text{cov}}(y) = \arg\min_{x \in \mathcal{H}_{\hat{x}}} \frac{1}{2}||\Phi x - y||^2
\]

- Apply linear Jacobian: orientation penalization

\[
J(y) = \arg\min_{x \in \mathcal{M}} \hat{x}(y) = \arg\min_{x \in \mathcal{M}} \frac{1}{2}||\Phi x - y||^2 + \frac{2}{\lambda}||\nabla \hat{x}||^2
\]

- New Re-fitting penalizing direction changes

\[
R_{\hat{x}}^{\text{inv}}(y) = \arg\min_{x \in \mathcal{M}} \hat{x}(y) = \arg\min_{x \in \mathcal{M}} \frac{1}{2}||\Phi x - y||^2 + F(\nabla x, \nabla \hat{x}(y))
\]
New Re-fitting models

- **Covariant Re-fitting:**

\[
\mathcal{R}_{\hat{x}}^{\text{cov}}(y) = \arg\min_{x \in \mathcal{H}_{\hat{x}}} \frac{1}{2}||\Phi x - y||^2 = \hat{x}(y) + \rho J(y - \Phi \hat{x}(y))
\]
New Re-fitting models

- Covariant Re-fitting:

\[
\mathcal{R}^{\text{cov}}_{\hat{x}}(y) = \arg\min_{x \in \mathcal{H}_{\hat{x}}} \frac{1}{2} \| \Phi x - y \|^2 = \hat{x}(y) + \rho J(y - \Phi \hat{x}(y))
\]

- Apply linear Jacobian: orientation penalization

\[
J(y) = \arg\min_{x \in \mathcal{M}_{\hat{x}}(y)} \frac{1}{2} \| \Phi x - y \|^2 + \left( \frac{\lambda}{2\|\nabla \hat{x}\|} \left\| \nabla x - \langle \nabla \hat{x}, \nabla x \rangle \frac{\nabla \hat{x}}{\|\nabla \hat{x}\|^2} \right\|^2 \right)
\]

\[= 0, \forall x \in \mathcal{O}_{\hat{x}}(y)\]
New Re-fitting models

- **Covariant Re-fitting:**

\[
\mathcal{R}_{\hat{x}}^{\text{cov}}(y) = \arg\min_{x \in \mathcal{H}_{\hat{x}}} \frac{1}{2} \| \Phi x - y \|^2 = \hat{x}(y) + \rho J(y - \Phi \hat{x}(y))
\]

- **Apply linear Jacobian: orientation penalization**

\[
J(y) = \arg\min_{x \in \mathcal{M}_{\hat{x}}(y)} \frac{1}{2} \| \Phi x - y \|^2
\]

\[
+ \left( \frac{\lambda}{2 \| \nabla \hat{x} \|} \left\| \nabla x - \langle \nabla \hat{x}, \nabla x \rangle \frac{\nabla \hat{x}}{\| \nabla \hat{x} \|^2} \right\|^2 \right)
\]

\[
= 0, \forall x \in \mathcal{O}_{\hat{x}}(y)
\]

- **J(y): Compromise between \( \tilde{x}^{\text{ICB}} \) and \( \mathcal{R}_{\hat{x}}^{\text{inv}} \)**
New Re-fitting models

- Covariant Re-fitting:

\[ R_{\hat{x}}^{\text{cov}}(y) = \arg\min_{x \in \mathcal{H}_{\hat{x}}} \frac{1}{2} \| \Phi x - y \|^2 = \hat{x}(y) + \rho J(y - \Phi \hat{x}(y)) \]

- Apply linear Jacobian: orientation penalization

\[ J(y) = \arg\min_{x \in \mathcal{M}_{\hat{x}}(y)} \frac{1}{2} \| \Phi x - y \|^2 + \left( \frac{\lambda}{2 \| \nabla \hat{x} \|} \| \nabla x - \langle \nabla \hat{x}, \nabla x \rangle \frac{\nabla \hat{x}}{\| \nabla \hat{x} \|} \| \|^2 \right) \]

\[ = 0, \forall x \in \mathcal{O}_{\hat{x}}(y) \]

- \( J(y) \): Compromise between \( \tilde{x}^{\text{ICB}} \) and \( R_{\hat{x}}^{\text{inv}} \)

- New Re-fitting penalizing direction changes

\[ R_{\hat{x}}(y) = \arg\min_{x \in \mathcal{M}_{\hat{x}}(y)} \frac{1}{2} \| \Phi x - y \|^2 + F(\nabla x, \nabla \hat{x}(y)) \]

\[ = 0, \forall x \in \mathcal{D}_{\hat{x}}(y) \]
Comparison of Re-fitting approaches

- $y$
- $Tv$ iso
- Bregman iteration
  - [Osher et al.]
- Orientation
  - [Brinkmann et al.]
- Direction
  - [Brinkmann et al.]
- Covariant
- New model

N. Papadakis
Comparison of Re-fitting approaches

Comparison of Re-fitting approaches

\( y \) \hspace{1cm} \text{Tv iso} \hspace{1cm} \text{Bregman iteration} \hspace{1cm} \text{Orientation} \\
[Osher et al.] \hspace{1cm} \text{[Brinkmann et al.]} \\
[Brinkmann et al.] \\

Direction \hspace{1cm} \text{Covariant} \hspace{1cm} \text{New model}