Maths of the DEMReg approach

Iain Hannah

July 29, 2022

More details see Hannah & Kontar (2012, 2013); Kontar et al. (2004); Hansen (1992).

Regularization to solve DEM problem

The observed data g_i per channel/filter i is related to the underlying differential emission measure DEM $\xi(T_j) = n^2 dh/dT$ [cm^{-5} K^{-1}] and the temperature response K_{ij} as:

$$g_i = \int K_{ij}\, \xi(T_j)\, dT. \tag{1}$$

This can also be written as a series of linear equations

$$g_i = K_{ij}\xi_j\ , \tag{2}$$

where we have $i = 1, ..., M$ and $j = 1, ..., N$ and typically $M < N$ (i.e. have 6 SDO/AIA channels and if we wanted to recover a DEM with 30 temperature bins, so $M = 6$ and $N = 30$). To solve this and recover the DEM we are really trying to find a minimum to the following problem

$$\|\mathbf{K}\boldsymbol{\xi} - \mathbf{g}\|^2 = \text{min}, \tag{3}$$

with $\|\mathbf{x}\|^{\mathbf{2}} = \mathbf{x}^{\mathbf{T}}\mathbf{x} = \mathbf{\Sigma}_{\mathbf{i=1}}^{\mathbf{M}}\mathbf{x}_{\mathbf{i}}^{\mathbf{2}}$ which is the L2 (or Euclidean) norm.

Eqn 1 is an ill-posed inversion problem and so the addition of linear constraints can be used to recover a solution without the amplification of noise, and a zero-order regularization is often used which selects the smallest norm solution out of the infinitely many possible solutions. So instead of solving Eqn 3 we actually want to solve:

$$\|\mathbf{K}\boldsymbol{\xi} - \mathbf{g}\|^2 = \min \text{ subject to } \|\mathbf{L}(\boldsymbol{\xi} - \boldsymbol{\xi}_\mathbf{0})\|^2 \leq \text{const} \tag{4}$$

which can be solved using Lagrangian multipliers, i.e.

$$\|\mathbf{K}\boldsymbol{\xi} - \mathbf{g}\|^2 + \lambda\, \|\mathbf{L}(\boldsymbol{\xi} - \boldsymbol{\xi}_\mathbf{0})\|^2 = \min \tag{5}$$

where $\mathbf{L}$ is the "constraint matrix", λ is the "regularization parameter" and $\boldsymbol{\xi}_\mathbf{0}$ is an optional guess solution.

GSVD to solve linear regularization

A solution to Eqn 5 can be found via GSVD as a function of λ (Hansen 1992). Here the GSVD of $\mathbf{K} \in \mathbb{R}^{\mathbf{M}\times\mathbf{N}}$ and $\mathbf{L} \in \mathbb{R}^{\mathbf{N}\times\mathbf{N}}$ produces a set of singular values α_k, β_k (with $\phi_k = \alpha_k/\beta_k$) and singular vectors $\mathbf{u}_k, \mathbf{v}_k, \mathbf{w}_k$, with $k = 1, ..., N$ which satisfy $\alpha_k^2 + \beta_k^2 = 1$, $\mathbf{U^T K W} = \mathrm{diag}(\boldsymbol{\alpha})$ and $\mathbf{V^T L W} = \mathrm{diag}(\boldsymbol{\beta})$. Note that because in our case $M \leq N$, only the first M elements of $\mathbf{u_k}$ and α_k are non-zero.

The solution is then, as given in Eqn 18 of Hansen (1992)

$$\boldsymbol{\xi}_\lambda = \sum_{k=1}^{M} \left(f_k \frac{\mathbf{g}\cdot\mathbf{u}_k}{\alpha_k} + (1-f_k)\mathbf{w}_k^{-1}\boldsymbol{\xi}_0 \right) \mathbf{w}_k, \tag{6}$$

where the filter factors f_k are given by

$$f_k = \frac{\phi_k^2}{\phi_k^2+\lambda} = \frac{\alpha_k^2}{\alpha_k^2+\lambda\beta_k^2} \quad \text{and} \quad 1-f_k = \frac{\lambda\beta_k^2}{\alpha_k^2+\lambda\beta_k^2}. \tag{7}$$

In the code Eqn 6 is implemented as

$$\boldsymbol{\xi}_\lambda = \sum_{k=1}^{M} \frac{(\alpha_k(\mathbf{g}\cdot\mathbf{u}_k) + \lambda\beta_k^2\mathbf{w}_k^{-1}\boldsymbol{\xi}_0)\mathbf{w}_k}{\alpha_k^2+\lambda\beta_k^2}. \tag{8}$$

This is how it is done in the original idl code[1] that accompanied Hannah & Kontar (2012). In the map versions (Hannah & Kontar 2013) of the idl[2] and python[3] code no guess solution is used and $\boldsymbol{\xi}_\lambda$ is calculated in a slightly different manner, with the regularized inversion of $\mathbf{K}$, $\mathbf{R}_\lambda \simeq \mathbf{K}^\dagger$ being directly calculated as

$$\mathbf{R}_\lambda = \sum_{k=1}^{M} \frac{\alpha_k\left(\mathbf{u}_k\mathbf{w}_k\right)}{\alpha_k^2+\lambda\beta_k^2}. \tag{9}$$

The DEM solution is then found via $\boldsymbol{\xi}_\lambda = \mathbf{R}_\lambda\mathbf{g}$. It is done this way as $\mathbf{R}_\lambda$ is used for estimating the temperature resolution[4] of the DEM solution, so does not need to be computed separately in the map versions where computational speed is more desirable.

Note that:

- Hannah & Kontar (2012) Eqn 6, $\boldsymbol{\xi}_0$ term not quite correct but correct in code.
- Kontar et al. (2004) Eqn 24, no initial guess but is in the code[5].
- Hansen (1992) Eqn 15, λ^2 is the regularization parameter, instead of λ here.

Now to get a useful solution we need to choose a suitable $\mathbf{L}$ and λ.

[1] idl_org/dem_inv_reg_solution.pro
[2] idl/demmap_pos.pro
[3] python/demmap_pos.py
[4] idl_org/dem_inv_reg_resolution.pro
[5] ssw/packages/xray/idl/inversion/inv_reg_solution.pro

Choosing constraint matrix $\mathbf{L}$

Generally we are considering zero order regularization, which means that $\mathbf{L} \propto \mathbf{I}$ and so should have no non-diagonal terms. When setting the diagonal terms of $\mathbf{L}$ they are usually normalised by some values, typically either the temperature bin width (i.e. $\mathbf{L}_{jj} = 1/d \log T_j$) or, where appropriate, some expected form of the final solution (this is not the same as the guess solution $\boldsymbol{\xi}_0$). For instance the minimum of the EM loci curves $\boldsymbol{\xi}_{mlc}$, taken as a smoothed and normalised version of the lowest of all the emission measures found at each temperature assuming an isothermal solution (i.e. $\mathbf{L}_{jj}$ is the minimum of g_i/K_{ij} over all i).

Choosing regularization parameter λ

The regularization parameter λ can be found based on the $\boldsymbol{\xi}_\lambda$ in Eqn 6, that matches some additional conditions. The standard one is from Morozov's discrepancy principle, namely

$$\frac{1}{M} \left\| \mathbf{K}\boldsymbol{\xi}_\lambda - \mathbf{g} \right\|^2 = \rho \tag{10}$$

where ρ is the regularization tweak parameter, which is effectively controlling the χ^2 of the solution in data space. So by default are aiming for $\rho = 1$. Given that the real observations have an associated uncertainty δg_i then we could expect that the best value of λ is such that

$$\left\| \mathbf{K}\boldsymbol{\xi}_\lambda - \mathbf{g} \right\|^2 = \rho \left\| \delta \mathbf{g} \right\|^2 . \tag{11}$$

So the optimal λ should satisfy

$$\left\| \mathbf{K}\boldsymbol{\xi}_\lambda - \mathbf{g} \right\|^2 - \rho \left\| \delta \mathbf{g} \right\|^2 = \min. \tag{12}$$

From Hansen (1992) Eqn 20, the term $\left\| \mathbf{K}\boldsymbol{\xi} - \mathbf{g} \right\|^2$ can be directly calculated from the GSVD products, namely

$$\left\| \mathbf{K}\boldsymbol{\xi}_\lambda - \mathbf{g} \right\|^2 = \sum_{k=1}^{M} (1 - f_k)^2 \left(\mathbf{g} \cdot \mathbf{u}_k - \alpha_k \mathbf{w}_k^{-1} \boldsymbol{\xi}_0 \right)^2 . \tag{13}$$

In the code Eqn 13 is implemented as

$$\left\| \mathbf{K}\boldsymbol{\xi}_\lambda - \mathbf{g} \right\|^2 = \sum_{k=1}^{M} \left(\frac{\lambda \beta_k^2 \left(\mathbf{g} \cdot \mathbf{u}_k - \alpha_k \mathbf{w}_k^{-1} \boldsymbol{\xi}_0 \right)}{\alpha_k^2 + \lambda \beta_k^2} \right)^2 . \tag{14}$$

This is how it is done in the original idl code[6]. In the map versions the idl[7] and python[8] code Eqn 14 should have used just the $(\mathbf{g} \cdot \mathbf{u}_k)$ in the numerator brackets as no guess solution, i.e. $\boldsymbol{\xi}_0$. However $(\mathbf{g} \cdot \mathbf{u}_k - \alpha_k)$ was used, which might have affected the determination of the optimal λ, but crucially the $\boldsymbol{\xi}_\lambda$ calculation in all version was correct. This has now been corrected in the map versions of the code.

[6] idl_org/dem_inv_reg_parameter.pro
[7] idl/dem_inv_reg_parameter_map.pro
[8] python/dem_reg_map.py

In all versions of the code Eqn 14 is calculated for a range of λ parameters (the range determined by a scaling of the min and max of the ϕ_k values) and then the left hand side of Eqn 12 is calculated for each, and the λ giving the smallest solution is selected.

Using Eqn 12 should provide the optimal mathematical solution but not necessarily the most physical one as it can return $\boldsymbol{\xi}_\lambda$ with some negative terms. So an additional "positivity" constraint can be invoked in which the chosen λ needs to satisfy Eqn 12 and $\boldsymbol{\xi}_\lambda > 0$. In the original idl code[9] this is achieved by calculating both Eqn 14 and Eqn 8 for a range of λ and choosing the one matches the two criteria. For the map code (idl[10] and python[11]) Eqn 14 was found for a smaller number of λ samples and if $\boldsymbol{\xi}_\lambda < 0$, ρ was increased iteratively and Eqn 14 recalculated until $\boldsymbol{\xi}_\lambda > 0$ or a max number of iterations was reached. It was done this way as it was faster than calculating both Eqn 14 and Eqn 8 for a larger sample of λ as done originally. Potentially could speed up original approach by finding a subset of λ using one criteria and then only do second criteria calculation on this smaller sample, i.e. find the λ that give $\boldsymbol{\xi}_\lambda > 0$ and then do Eqn 14 on that subset.

Errors on the DEM solution

More details to come, see Hannah & Kontar (2012, 2013) and the code in the meantime.

The vertical errors on the DEM solution are just the linear propagation of the uncertainties on the input data.

The horizontal error on the DEM solution are the "temperature resolution" and found from measuring the spread of non-diagonal terms in $\mathbf{KR}_\lambda$, since the better the solution $\boldsymbol{\xi}_\lambda = \mathbf{R}_\lambda \mathrm{g}$ then the more $\mathbf{KR}_\lambda \approx \mathbf{I}$.

References

Hannah, I. G., & Kontar, E. P. 2012, A&A, 539, A146, doi: 10.1051/0004-6361/201117576

—. 2013, A&A, 553, A10, doi: 10.1051/0004-6361/201219727

Hansen, P. C. 1992, Inverse Problems, 8, 849, doi: 10.1088/0266-5611/8/6/005

Kontar, E. P., Piana, M., Massone, A. M., Emslie, A. G., & Brown, J. C. 2004, Sol. Phys., 225, 293, doi: 10.1007/s11207-004-4140-x

[9] idl_org/dem_inv_reg_parameter_pos.pro

[10] idl/demmap_pos.pro

[11] python/demmap_pos.py