Globally Convergent Type-I Anderson Acceleration for Non-Smooth Fixed-Point Iterations

Theo Koppenhöfer

Lund

April 11, 2023

Table of contents

The problem setting

Problem (find fixed point)

Find a fixed point $x \in \mathbb{R}^n$ *of* $f : \mathbb{R}^n \to \mathbb{R}^n$, *i.e.* $x = f(x)$.

or equivalently

Problem (find zero)

Find a zero $x \in \mathbb{R}^n$ *of* $g = \mathrm{Id} - f$, *i.e.* $0 = g(x)$.

We also assume

- f is nonexpansive, i.e. $\|f(x) - f(y)\| \leq \|x - y\|$
- n is large $\to$matrix-free
- ∇f is unknown $\to$no Newton
- cost of evaluation of f is high $\to$no line search
- noisy problem $\to$no finite difference derivatives

Fixed point iteration

To keep things simple we try

Algorithm 1: Fixed point iteration (original)

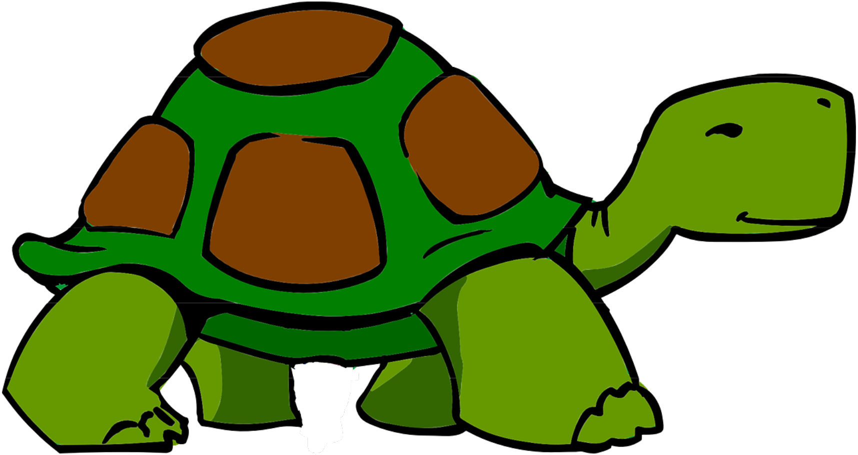
Input : Initial value $x_0 \in \mathbb{R}^n$ and function $f\colon \mathbb{R}^n \to \mathbb{R}^n$.

for $k = 0, 1, \dots$ **do**

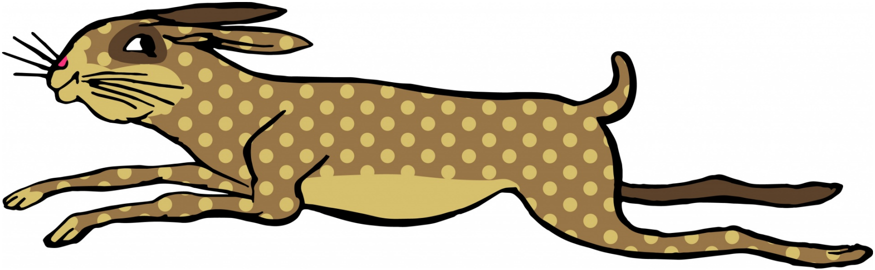
 Set $x_{k+1} = f(x_k)$.

end

This works, but ...



We want to be like...



General AA

We may as well use the information gained from previous evaluations. If we form a weighted average we get

---

Algorithm 2: General AA (Anderson Acceleration)

Input : $x_0 \in \mathbb{R}^n$ and $f : \mathbb{R}^n \to \mathbb{R}^n$.

for $k = 0, 1, \ldots$ **do**

  Set $f_k = f(x_k)$.

  Choose $\alpha = \alpha^k \in \mathbb{R}^k$ such that $\sum_i \alpha_i = 1$.

  Set $x_{k+1} = \sum_i \alpha_i f_i$.

end

AA-II

Since finding a fixed point of f is equivalent to finding a zero of $g = \mathrm{Id} - f$ we have the ansatz

Algorithm 3: AA-II

Input : $x_0 \in \mathbb{R}^n$ and $f\colon \mathbb{R}^n \to \mathbb{R}^n$.

for $k = 0, 1, \dots$ **do**

  Set $f_k = f(x_k)$.

  Set $g_k = x_k - f_k$.

  Choose $\alpha \in \mathbb{R}^k$ such that $\sum_i \alpha_i = 1$ and such that α minimises $\|\sum_i \alpha_i g_i\|_2$.

  Set $x_{k+1} = \sum_i \alpha_i f_i$.

end

Rewriting AA-II

Setting

$$\alpha = \begin{bmatrix} \gamma_0 \\ \gamma_1 - \gamma_0 \\ \vdots \\ \gamma_k - \gamma_{k-1} \\ 1 - \gamma_k \end{bmatrix} \text{ and } Y_k = \begin{bmatrix} g_1 - g_0 & \cdots & g_k - g_{k-1} \end{bmatrix} \in \mathbb{R}^{n \times k}$$

one obtains the least squares problem

$$\min_{\substack{\alpha \in \mathbb{R}^{k+1} \\ \sum_i \alpha_i = 1}} \left\| \sum_i \alpha_i g_i \right\| = \min_{\gamma \in \mathbb{R}^k} \| g_k - Y_k \gamma \|$$

which is solved by

$$\gamma = \gamma^k = \left(Y_k^\top Y_k \right)^{-1} Y_k^\top g_k \,.$$

If we now set

$$S_k = \begin{bmatrix} x_1 - x_0 & \cdots & x_k - x_{k-1} \end{bmatrix} \in \mathbb{R}^{n\times k}$$

we see that

$$\begin{aligned} S_k - Y_k &= \begin{bmatrix} x_1 - x_0 - g_0 + g_1 & \cdots & x_k - x_{k-1} - g_k + g_{k-1} \end{bmatrix} \\ &= \begin{bmatrix} f_1 - f_0 & \cdots & f_k - f_{k-1} \end{bmatrix} \end{aligned}$$

and hence

$$
\begin{aligned}
x_{k+1} &= \sum_i \alpha_i f(x_i) \\
&= f_k - (S_k - Y_k)\gamma \\
&\qquad f_k = x_k - g_k \text{ and } \gamma = \left(Y_k^\top Y_k\right)^{-1} Y_k^\top \\
&= x_k - \underbrace{\left(\mathrm{Id} + (S_k - Y_k)\left(Y_k^\top Y_k\right)^{-1} Y_k^\top\right)}_{=H_k} g_k \\
&= x_k - H_k g_k \,.
\end{aligned}
$$

We thus have the reformulation

Algorithm 4: AA-II (reformulated)

Input : $x_0 \in \mathbb{R}^n$ and $f\colon \mathbb{R}^n \to \mathbb{R}^n$.

Set $x_1 = f(x_0)$.

for $k = 0, 1, \ldots$ **do**

Set $g_k = g(x_k)$.

Construct S_k from $x_0, \ldots, x_k$ and Y_k from $g_0, \ldots, g_k$.

Set $H_k = \mathrm{Id} + (S_k - Y_k)\left(Y_k^\top Y_k\right)^{-1} Y_k^\top$.

Set $x_{k+1} = x_k - H_k g_k$.

end

AA-I

This is the form of a quasi-Newton-like method so one could expect H_k to be an approximate inverse of $\nabla f(x_k)$. Indeed

Proposition (Approximate inverse Jacobian)

H_k minimises $\|H_k - \mathrm{Id}\|_F$ under the multisecant condition $H_k S_k = Y_k$.

From Broydens method we know that it is a good idea to approximate the Jacobian rather than its inverse.

Definition (Approximate Jacobian)

Let B_k be minimiser of $\|B_k - \mathrm{Id}\|_F$ under the condition $B_k Y_k = S_k$.

Analogously to AA-II we have

$$B_k = \mathrm{Id} + (Y_k - S_k)\left(S_k^\top S_k\right)^{-1} S_k^\top .$$

This yields the AA-I algorithm

Algorithm 5: AA-I

Input : $x_0 \in \mathbb{R}^n$ and $f\colon \mathbb{R}^n \to \mathbb{R}^n$.

Set $x_1 = f(x_0)$
for $k = 0, 1, \dots$ **do**
  Set $g_k = g(x_k)$.
  Construct S_k from $x_0, \dots, x_k$ and Y_k from $g_0, \dots, g_k$.
  Set $B_k = \mathrm{Id} + (Y_k - S_k)\left(S_k^\top S_k\right)^{-1} S_k^\top$.
  Set $H_k = B_k^{-1}$.
  Set $x_{k+1} = x_k - H_k g_k$.
end

Luckily for us we can save some computations by using the rank-1 update formula

Proposition (Rank-1 update for B_k)

We have

$$B_{k+1} = B_k + \frac{(y_k - B_k s_k)\hat{s}_k^\top}{\hat{s}_k^\top s_k}$$

where $y_k = g_{k+1} - g_k$, $B_0 = \mathrm{Id}$ *and*

$$\hat{s}_k = s_k - \sum_{j=0}^{k-1} \frac{\hat{s}_k^\top s_k}{\|\hat{s}_k\|^2}\hat{s}_k$$

is the Gram-Schmidt orthogonalisation of $s_k = x_{k+1} - x_k$.

From the Sherman-Morrison formula it then follows that

Proposition (Rank-1 update for H_k)

We have

$$H_{k+1} = H_k + \frac{(s_k - H_k y_k)\hat{s}_k^\top H_k}{\hat{s}_k^\top H_k y_k}$$

where $y_k = g_{k+1} - g_k$, $H_0 = \mathrm{Id}$ *and*

$$\hat{s}_k = s_k - \sum_{j=0}^{k-1} \frac{\hat{s}_k^\top s_k}{\|\hat{s}_k\|^2} \hat{s}_k$$

is the Gram-Schmidt orthogonalisation of $s_k = x_{k+1} - x_k$.

Taking everything together we obtain

Algorithm 6: AA-I (rank-1 update)

Input : $x_0 \in \mathbb{R}^n$ and $f\colon \mathbb{R}^n \to \mathbb{R}^n$.

Set $H_0 = \mathrm{Id}$ and $x_1 = f(x_0)$.

for $k = 0, 1, \ldots$ **do**

  Set $g_k = g(x_k)$.

  Set $s_{k-1} = x_k - x_{k-1}$, $y_{k-1} = g_k - g_{k-1}$ and $\hat{s}_{k-1} = s_{k-1} - \sum_{i=0}^{k-2} \frac{\hat{s}_i^\top s_{k-1}}{\|\hat{s}_i\|^2} s_i$.

  Set $H_k = H_{k-1} + \frac{(s_{k-1} - H_{k-1} y_{k-1}) s_{k-1}^\top H_{k-1}}{\hat{s}_{k-1}^\top H_{k-1} y_{k-1}}$.

  Set $x_{k+1} = x_k - H_k g_k$.

end

Powell-type regularisation

Note that B_k may be singular. To fix this set

$$\tilde{y}_k = \theta_k y_k + (1 - \theta_k) B_k s_k$$

or equivalently

$$\tilde{y}_k = \theta_k y_k + (1 - \theta_k) B_k s_k$$

where

$$\theta_k = \phi_{\bar{\theta}} \left(\eta_k \right)$$

with

$$\phi_{\bar{\theta}} (\eta) = \begin{cases} \frac{1 - \operatorname{sgn}(\eta)\bar{\theta}}{1 - \eta} & \text{if } |\eta| < \bar{\theta} \\ 1 & \text{else} \end{cases} \quad \text{and} \quad \eta_k = \frac{\hat{s}_k^\top H_k y_k}{\|\hat{s}_k\|^2}$$

One can obtain

Lemma (Powell-type regularisation)

Let $s_k \in \mathbb{R}^n$, $B_0 = \mathrm{Id}$, *and inductively*

$$B_{k+1} = B_k + \frac{(\tilde{y}_k - B_k s_k)\hat{s}_k^\top}{\hat{s}_k^\top s_k}$$

with $\hat{s}_k$ *and* $\tilde{y}_k$ *defined as before. If this is well-defined then* $|\det(B_k)| \geq \theta^k > 0$ *and* B_k *is invertible.*

Proof.

See [1, Lemma 2]. □

Algorithm 7: AA-I with Powell-like-regularisation

s **Input :** $x^0 \in \mathbb{R}^n$, $f : \mathbb{R}^n \to \mathbb{R}^n$ and $\bar{\theta} \in (0,1)$.

Set $H_0 = \mathrm{Id}$ and $x_1 = f(x_0)$.

for $k = 0, 1, \ldots$ **do**

Set $g_k = g(x_k)$, $s_{k-1} = x_k - x_{k-1}$ and $y_{k-1} = g_k - g_{k-1}$.

Set $\hat{s}_{k-1} = s_{k-1} - \sum_{i=0}^{k-2} \frac{\hat{s}_i^\top s_{k-1}}{\|\hat{s}_i\|^2} s_i$.

Set $\eta_{k-1} = \frac{\hat{s}_{k-1}^\top H_{k-1} y_{k-1}}{\|\hat{s}_{k-1}\|^2}$, $\theta_{k-1} = \phi_{\bar{\theta}}(\eta_{k-1})$ and $\tilde{y}_{k-1} = \theta_{k-1} y_{k-1} - (1-\theta_{k-1}) g_{k-1}$.

Set $H_k = H_{k-1} + \frac{(s_{k-1} - H_{k-1}\tilde{y}_{k-1})}{\hat{s}_{k-1}^\top H_{k-1}\tilde{y}_{k-1}}$ and $x_{k+1} = x_k - H_k g_k$.

end

Restarting iteration

Note that

$$B_{k+1} = B_k + \frac{(\tilde{y}_k - B_k s_k)\hat{s}_k^\top}{\hat{s}_k^\top s_k}$$

is ill-defined iff $\|\hat{s}_k\|^2 = \hat{s}_k^\top s_k = 0$, i.e. $\hat{s}_k = 0$. This occurs in algorithm 7 for $k > n$ as then $\hat{s}_k = 0$ by linear dependence. If we restart the algorithm with x_k as the new starting point if $k = m+1$ for some $m \in \mathbb{N}$ or $\|\hat{s}_{k-1}\| < \tau \|s_{k-1}\|$ for some $\tau \in (0,1)$ then

$$g_k \neq 0 \implies s_k = -B_k g_k \neq 0 \implies \hat{s}_k \neq 0\,.$$

Algorithm 8: AA-I with Powell-like-regularisation and Restarting

Input : $x^0 \in \mathbb{R}^n$, $f : \mathbb{R}^n \to \mathbb{R}^n$, $\boldsymbol{m} \in \mathbb{N}$ and $\bar{\theta}, \boldsymbol{\tau} \in (0,1)$

Set $H_0 = \mathrm{Id}$, $x_1 = f(x_0)$ and $\boldsymbol{m_0 = 0}$.

for $k = 0, 1, \dots$ **do**

Set $g_k = g(x_k)$, $m_k = m_{k-1} + 1$, $s_{k-1} = x_k - x_{k-1}$ and $y_{k-1} = g_k - g_{k-1}$.

Set $\hat{s}_{k-1} = s_{k-1} - \sum_{i=k-m_k}^{k-2} \frac{\hat{s}_i^\top s_{k-1}}{\|\hat{s}_i\|^2} s_i$.

if $m_k = m + 1$ *or* $\|\hat{s}_{k-1}\| < \tau \|s_{k-1}\|$ **then**

Set $m_k = 0$, $\hat{s}_{k-1} = s_{k-1}$ and $H_{k-1} = \mathrm{Id}$.

end

Set $\eta_{k-1} = \frac{\hat{s}_{k-1}^\top H_{k-1} y_{k-1}}{\|\hat{s}_{k-1}\|^2}$, $\theta_{k-1} = \phi_{\bar{\theta}}(\eta_{k-1})$ and $\tilde{y}_{k-1} = \theta_{k-1} y_{k-1} - (1 - \theta_{k-1}) g_{k-1}$.

Set $H_k = H_{k-1} + \frac{(s_{k-1} - H_{k-1}\tilde{y}_{k-1})}{\hat{s}_{k-1}^\top H_{k-1}\tilde{y}_{k-1}}$ and $x_{k+1} = x_k - H_k g_k$.

end

Lemma (Restarting iteration)

If we additionally choose m_k by the rule above we have

$$\|B_k\| \leq 3\left(\frac{1+\bar{\theta}+\tau}{\tau}\right)^m - 2\,.$$

Proof.

See [1, Lemma 3]. $\square$

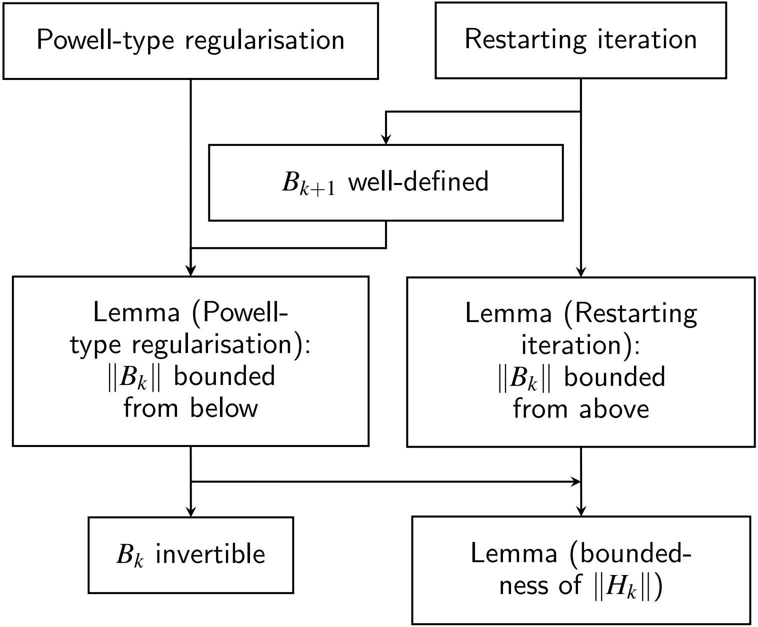
Lemma (bound on $\|H_k\|_2$)

In algorithm 8 we have that

$$\|H_k\|_2 \le \frac{1}{\bar{\theta}^m}\left(3\left(\frac{1+\bar{\theta}+\tau}{\tau}\right)^m - 2\right)^{n-1}.$$

Proof.

This follows from Lemma (Restarting iteration) and Lemma (Powell-type regularisation). $\square$


Powell-type regularisation
Restarting iteration
B_{k+1} well-defined
Lemma (Powell-type regularisation): $\|B_k\|$ bounded from below
Lemma (Restarting iteration): $\|B_k\|$ bounded from above
B_k invertible
Lemma (boundedness of $\|H_k\|$)

Safeguarding steps

To guarantee the decrease in $\|g_k\|$ one can interleave the AA-I steps with Krasnosel'skii-Mann steps which are given by

$$x_{k+1} = (1-\alpha)x_k + \alpha f(x_k)$$

for some fixed $\alpha \in (0,1)$.

Algorithm 9: AA-I with Powell-like-regularisation, Restarting and Safeguarding

Input : $x^0 \in \mathbb{R}^n$, $f\colon \mathbb{R}^n \to \mathbb{R}^n$, $m \in \mathbb{N}$, $\bar{\theta}, \tau, \alpha \in (0,1)$ and safe-guarding constants $D, \varepsilon > 0$

Set $H_0 = \mathrm{Id}$, $x_1 = \tilde{x}_1 = f(x_0)$, $m_0 = n_{AA} = 0$ and $\bar{U} = \|g_0\|_2$.

for $k = 0, 1, \dots$ **do**

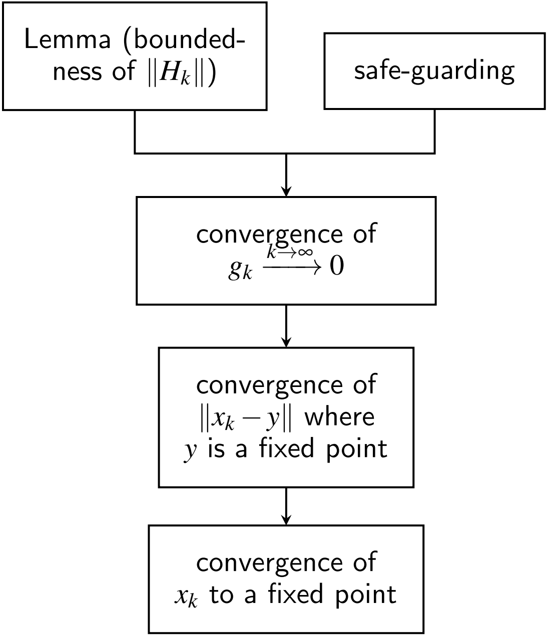
- Set $g_k = g(x_k)$, $m_k = m_{k-1} + 1$, $s_{k-1} = \tilde{x}_k - x_{k-1}$ and $y_{k-1} = g(\tilde{x}_k) - g_{k-1}$.
- Set $\hat{s}_{k-1} = s_{k-1} - \sum_{i=k-m_k}^{k-2} \frac{\hat{s}_i^\top s_{k-1}}{\|\hat{s}_i\|^2} s_i$.
- **if** $m_k = m+1$ *or* $\|\hat{s}_{k-1}\| < \tau \|s_{k-1}\|$ **then**
  - Set $m_k = 0$, $\hat{s}_{k-1} = s_{k-1}$ and $H_{k-1} = \mathrm{Id}$.
- **end**
- Set $\eta_{k-1} = \frac{\hat{s}_{k-1}^\top H_{k-1} y_{k-1}}{\|\hat{s}_{k-1}\|^2}$, $\theta_{k-1} = \phi_{\bar{\theta}}(\eta_{k-1})$ and $\tilde{y}_{k-1} = \theta_{k-1} y_{k-1} - (1 - \theta_{k-1}) g_{k-1}$.
- Set $H_k = H_{k-1} + \frac{(s_{k-1} - H_{k-1}\tilde{y}_{k-1})}{\hat{s}_{k-1}^\top H_{k-1} \tilde{y}_{k-1}}$ and $\tilde{x}_{k+1} = x_k - H_k g_k$.
- **if** $\|g_k\| \le D\bar{U}(n_{AA}+1)^{-(1+\varepsilon)}$ **then**
  - Set $x_{k+1} = \tilde{x}_{k+1}$ and $n_{AA} = n_{AA} + 1$.
- **else**
  - Set $x_{k+1} = (1-\alpha)x_k + \alpha f(x_k)$
- **end**

end

Convergence result

Theorem (Convergence)

Let x_k be generated by algorithm 9 then $x_k \xrightarrow{k\to\infty} x$ and $f(x) = x$ is a fixed point.


Lemma (boundedness of $\|H_k\|$)
safe-guarding
convergence of $g_k \xrightarrow{k\to\infty} 0$
convergence of $\|x_k - y\|$ where y is a fixed point
convergence of x_k to a fixed point

Regularised logistic regression

We take $x \in \mathbb{R}^{2000\times 500}$, $y \in \mathbb{R}^{2000}$ from the UCI Madelon dataset [2].
The aim is to minimise

$$F(\theta) = \frac{1}{2000}\sum_i \log(1 + \sum_j y_i x_{ij}\theta_j) + \frac{\lambda}{2}\|\theta\|^2$$

with gradient descent, i.e.

$$f\colon \mathbb{R}^{500} \to \mathbb{R}^{500}, \quad \theta \mapsto \theta - \alpha\nabla F(\theta)$$

for some α.

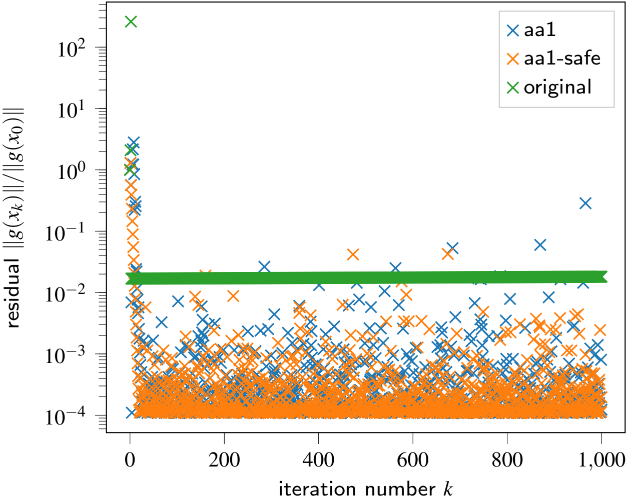


Figure: Residual norms for the logistic regression problem.

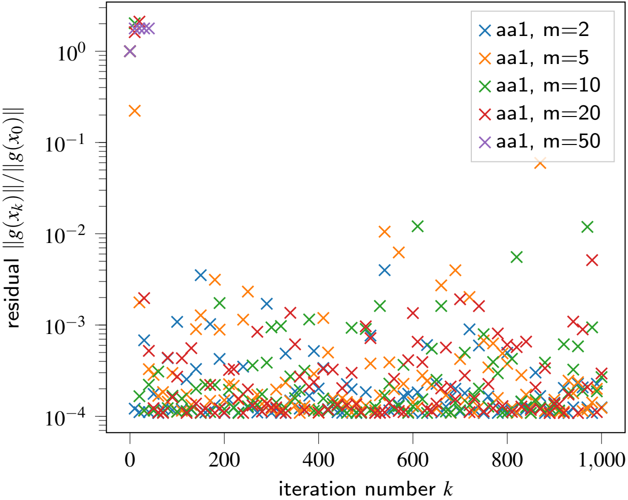


Figure: Residual norms for the logistic regression problem.

Facility location

The aim is to minimise

$$F\colon \mathbb{R}^{300} \to \mathbb{R}, \quad y \mapsto \sum_{i=1}^{500} \|y - c_i\|$$

for $c_i \in \mathbb{R}^{300}$ with sparsity 0.01. This can lead to the formulation

$$\tilde{f}\colon \mathbb{R}^{500\times 300} \to \mathbb{R}^{500\times 300}, \quad z \mapsto \Big(z_i + 2\ \langle x \rangle\ -\ x_i\ - \langle z \rangle\Big)_i$$

with

$$\langle x \rangle\ = \frac{1}{500}\sum_i x_i \qquad x_i\ =\ \operatorname{prox}_{\|\cdot\|}\ (z_i + c_i) - c_i$$

and

$$\operatorname{prox}_{\|\cdot\|}\ (v) = \left(1 - \frac{1}{\|v\|}\right)_+ v\,.$$

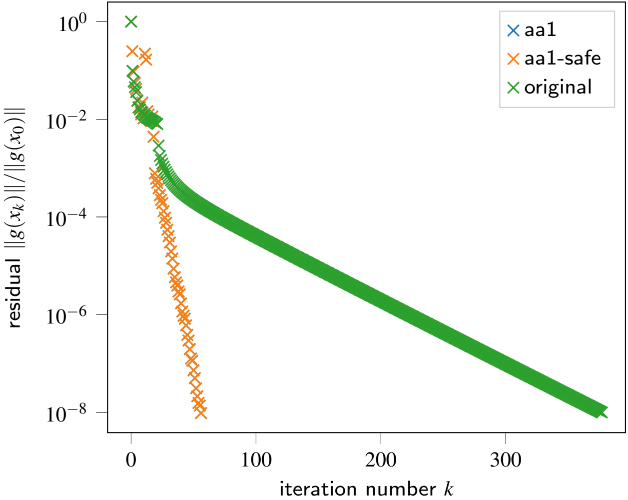


Figure: Residual norms for the facility location problem.

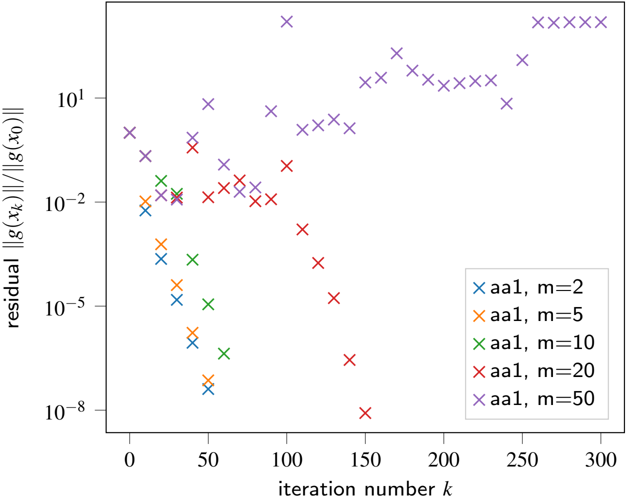


Figure: Residual norms for the facility location problem.

Elastic net regression

Our aim is to minimise

$$F\colon \mathbb{R}^{1000} \to \mathbb{R}, \quad x \mapsto \frac{1}{2}\|Ax-b\|^2 + \mu\left(\frac{1}{4}\|x\|^2 + \frac{1}{2}\|x\|_1\right)$$

with $A \in \mathbb{R}^{500\times 1000}$, $b \in \mathbb{R}^{500}$ and some $\mu \in \mathbb{R}$. From the Iterative Shrinkage-Thresholding Algorithm one obtains

$$f\colon \mathbb{R}^{1000} \to \mathbb{R}^{1000}, \quad x \mapsto S_{\alpha\mu/2}\left(x - \alpha\left(A^\top(Ax-b) + \frac{\mu}{2}x\right)\right)$$

with shrinkage operator

$$S_\kappa(x) = \left(\operatorname{sgn}(x_i)(|x_i| - \kappa)_+\right)_i$$

and some $\alpha \in \mathbb{R}$.

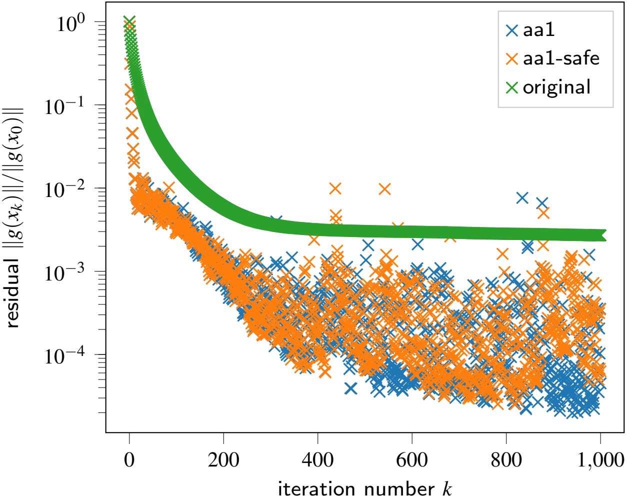


Figure: Residual norms for the elastic net regression problem.

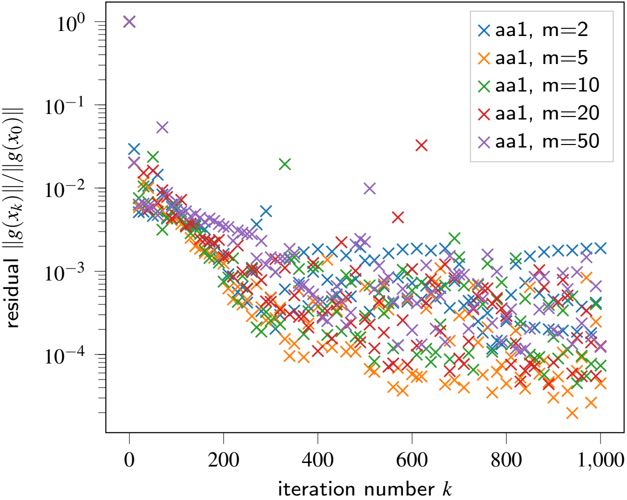


Figure: Residual norms for the elastic net regression problem.

Markov decision process

Our aim is to find a fixed point of the Bellman operator

$$f\colon \mathbb{R}^{1000} \to \mathbb{R}^{1000}, \quad x \mapsto \left(\max_a R(s,a) + \gamma \sum_{s'} P(s,a,s') x_{s'} \right)_s$$

with some $R \in \mathbb{R}^{300\times 200}$, $P \in \mathbb{R}^{300\times 200\times 300}$, $\gamma \in \mathbb{R}$.

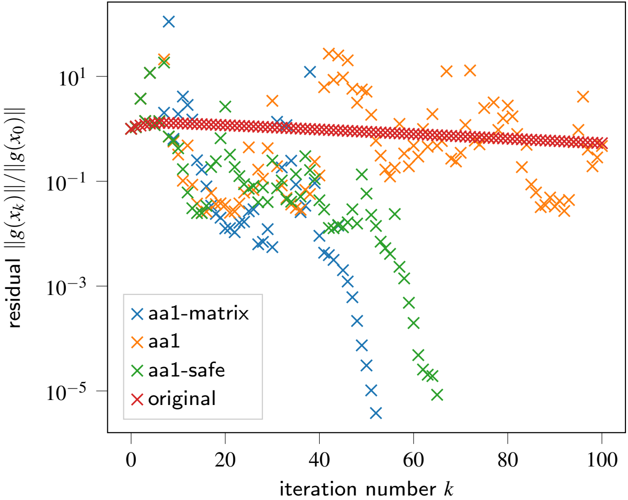


Figure: Residual norms for the elastic net regression problem.

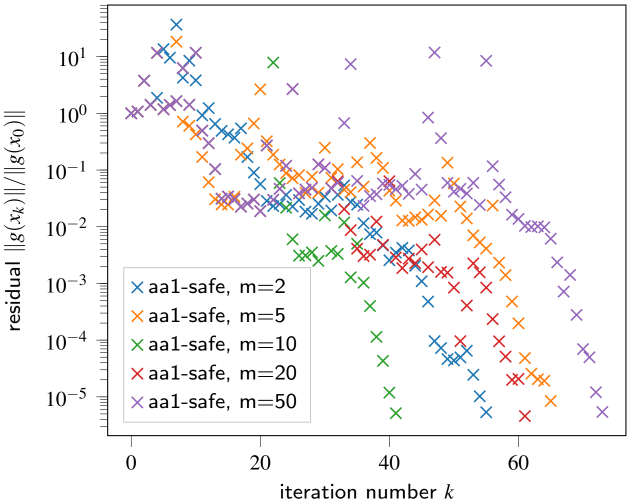


Figure: Residual norms for the elastic net regression problem.

Summary

- aim is to find a fixed point of f where
  - the dimension is large
  - f is expensive to evaluate, noisy and the gradient is a mystery
- 3 modifications to the AA-I algorithm yield well-definedness and convergence for non-expansive problems
  - Powell-type regularisation
  - Restarting iteration
  - Safeguarding steps
-

Sources I

[1] J. Zhang, B. O'Donoghue, and S. Boyd, "Globally convergent type-I Anderson acceleration for nonsmooth fixed-point iterations," *SIAM J. Optim.*, vol. 30, no. 4, pp. 3170–3197, 2020, ISSN: 1052-6234. DOI: 10.1137/18M1232772. [Online]. Available: https://doi-org.ludwig.lub.lu.se/10.1137/18M1232772.

[2] I. Guyon. (2004), Madelon data set, [Online]. Available: https://archive.ics.uci.edu/ml/datasets/Madelon.

[3] H.-r. Fang and Y. Saad, "Two classes of multisecant methods for nonlinear acceleration," *Numer. Linear Algebra Appl.*, vol. 16, no. 3, pp. 197–221, 2009, ISSN: 1070-5325. DOI: 10.1002/nla.617. [Online]. Available: https://doi-org.ludwig.lub.lu.se/10.1002/nla.617.

Thank you for your attention.