Regularized Nonlinear Acceleration.

Alexandre d’Aspremont,

CNRS & D.I. ENS.

with Damien Scieur & Francis Bach.

Support from ERC SIPA and ITN MacSeNet.
Introduction

Generic convex optimization problem

\[
\min_{x \in \mathbb{R}^n} f(x)
\]
Introduction

Algorithms produce a sequence of iterates.

We only keep the last (or best) one...
Aitken’s $\Delta^2$ [Aitken, 1927]. Given a sequence $\{s_k\}_{k=1}^\infty \in \mathbb{R}^N$ with limit $s_*$, and suppose

$$s_{k+1} - s_* = a(s_k - s_*), \quad \text{for } k = 1, \ldots$$

We can compute $a$ using

$$s_{k+1} - s_k = a(s_k - s_{k-1}) \quad \Rightarrow \quad a = \frac{s_{k+1} - s_k}{s_k - s_{k-1}}$$

and get the limit $s^*$ by solving

$$s_{k+1} - s^* = \frac{s_{k+1} - s_k}{s_k - s_{k-1}}(s_k - s^*)$$

which yields

$$s^* = \frac{s_{k-1}s_{k+1} - s_k^2}{s_{k+1} - 2s_k + s_{k-1}}$$

This is Aitken’s $\Delta^2$ and allows us to compute $s_*$ from $\{s_{k+1}, s_k, s_{k-1}\}$.
Aitken’s $\Delta^2$ [Aitken, 1927], again. Given a sequence $\{s_k\}_{k=1,\ldots} \in \mathbb{R}^N$ with limit $s_*$, and suppose that for $k = 1, \ldots$,

$$a_0 (s_k - s_*) + a_1 (s_{k+1} - s_*) = 0 \quad \text{and} \quad a_0 + a_1 = 1 \quad (\text{normalization})$$

We have

$$\left(a_0 + a_1\right) s_* = a_0 s_{k-1} + a_1 s_k$$

$$0 = a_0 (s_k - s_{k-1}) + a_1 (s_{k+1} - s_k)$$

We get $s^*$ using

$$\begin{bmatrix} 0 & s_{k+1} - s_k & s_k - s_{k-1} \\ -1 & s_k & s_{k-1} \\ 0 & 1 & 1 \end{bmatrix} \cdot \begin{bmatrix} s^* \\ a_1 \\ a_0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad \Leftrightarrow \quad s^* = \begin{bmatrix} s_{k+1} - s_k & s_k - s_{k-1} \\ s_k & s_{k-1} \\ s_{k+1} - s_k & s_k - s_{k-1} \end{bmatrix}$$

Same formula as before, but generalizes to **higher dimensions**.

Alex d’Aspremont
Turing Institute, May 2017. 5/33
Convergence acceleration. Consider

\[ s_k = \sum_{i=0}^{k} \frac{(-1)^i}{(2i + 1)} \xrightarrow{k \to \infty} \frac{\pi}{4} = 0.785398 \ldots \]

we have

\[
\begin{array}{cccc}
  k & \frac{(-1)^k}{(2k+1)} & \sum_{i=0}^{k} \frac{(-1)^i}{(2i+1)} & \Delta^2 \\
  0 & 1 & 1.0000 & - \\
  1 & -0.33333 & 0.66667 & - \\
  2 & 0.2 & 0.86667 & 0.79167 \\
  3 & -0.14286 & 0.72381 & 0.78333 \\
  4 & 0.11111 & 0.83492 & 0.78631 \\
  5 & -0.090909 & 0.74401 & 0.78492 \\
  6 & 0.076923 & 0.82093 & 0.78568 \\
  7 & -0.066667 & 0.75427 & 0.78522 \\
  8 & 0.058824 & 0.81309 & 0.78552 \\
  9 & -0.052632 & 0.76046 & 0.78531 \\
\end{array}
\]
Convergence acceleration.

- Similar results apply to sequences satisfying
  \[
  \sum_{i=0}^{k} a_i (s_{n+i} - s_*) = 0
  \]
  using Aitken’s ideas recursively.

- This produces **Wynn’s ε-algorithm** [Wynn, 1956].

- See [Brezinski, 1977] for a survey on acceleration, extrapolation.

- Directly related to the Levinson-Durbin algo on AR processes.

- **Vector case:** focus on **Minimal Polynomial Extrapolation** [Sidi et al., 1986].

Overall: a simple **postprocessing** step.
Outline

- Introduction
- **Minimal Polynomial Extrapolation**
- Regularized MPE
- Numerical results
Minimal Polynomial Extrapolation

**Quadratic example.** Minimize

\[ f(x) = \frac{1}{2} \| Bx - b \|_2^2 \]

using the basic gradient algorithm, with

\[ x_{k+1} := x_k - \frac{1}{L} (B^T B x_k - b). \]

we get

\[ x_{k+1} - x^* := \left( I - \frac{1}{L} B^T B \right) (x_k - x^*) \]

since \( B^T B x^* = b \).

This means \( x_{k+1} - x^* \) follows a **vector autoregressive process.**
Minimal Polynomial Extrapolation

We have

\[ \sum_{i=0}^{k} c_i (x_i - x^*) = \sum_{i=1}^{k} c_i A^i (x_0 - x^*) \]

and setting \( 1^T c = 1 \), yields

\[ \left( \sum_{i=0}^{k} c_i x_i \right) - x^* = p(A)(x_0 - x^*), \quad \text{where } p(v) = \sum_{i=1}^{k} c_i v^i \]

- Setting \( c \) such that \( p(A)(x_0 - x^*) = 0 \), we would have

\[ x^* = \sum_{i=0}^{k} c_i x_i \]

- Get the limit by averaging iterates (using weights depending on \( x_k \)).
- We typically do not observe \( A \) (or \( x^* \)).
- How do we extract \( c \) from the iterates \( x_k \)?
We have

\[ x_k - x_{k-1} = (x_k - x^*) - (x_{k-1} - x^*) \]

\[ = (A - I)A^{k-1}(x_0 - x^*) \]

hence if \( p(A) = 0 \), we must have

\[ \sum_{i=1}^{k} c_i (x_i - x_{i-1}) = (A - I)p(A)(x_0 - x^*) = 0 \]

so if \((A - I)\) is nonsingular, the coefficient vector \( c \) solves the linear system

\[
\begin{cases} 
\sum_{i=1}^{k} c_i (x_i - x_{i-1}) = 0 \\
\sum_{i=1}^{k} c_i = 1
\end{cases}
\]

and \( p(\cdot) \) is the minimal polynomial of \( A \) w.r.t. \((x_0 - x^*)\).
Approximate Minimal Polynomial Extrapolation

Approximate MPE.

- For $k$ smaller than the degree of the minimal polynomial, we find $c$ that minimizes the residual

$$\| (A - I)p(A)(x_0 - x^*) \|_2 = \left\| \sum_{i=1}^{k} c_i (x_i - x_{i-1}) \right\|_2$$

- Setting $U \in \mathbb{R}^{n \times k+1}$, with $U_i = x_{i+1} - x_i$, this means solving

$$c^* \triangleq \arg\min_{1^T c = 1} \| U c \|_2 \quad \text{(AMPE)}$$

in the variable $c \in \mathbb{R}^{k+1}$.

- Also known as Eddy-Mešina method [Mešina, 1977, Eddy, 1979] or Reduced Rank Extrapolation with arbitrary $k$ (see [Smith et al., 1987, §10]).
**Uniform Bound**

**Chebyshev polynomials.** Crude bound on $\|Uc^*\|_2$ using Chebyshev polynomials, to bound error as a function of $k$, with

$$\left\| \sum_{i=0}^{k} c_i^* x_i - x^* \right\|_2 = \left\| (I - A)^{-1} \sum_{i=0}^{k} c_i^* U_i \right\|_2$$

$$\leq \left\| (I - A)^{-1} \right\|_2 \left\| p(A)(x_1 - x_0) \right\|_2$$

We have

$$\left\| p(A)(x_1 - x_0) \right\|_2 \leq \|p(A)\|_2 \left\| (x_1 - x_0) \right\|_2$$

$$= \max_{i=1,...,n} |p(\lambda_i)| \left\| (x_1 - x_0) \right\|_2$$

where $0 \leq \lambda_i \leq \sigma$ are the eigenvalues of $A$. It suffices to find $p(\cdot) \in \mathbb{R}_k[x]$ solving

$$\inf_{\{p(\cdot) \in \mathbb{R}_k[x] : p(1) = 1\}} \sup_{v \in [0, \sigma]} |p(v)|$$

Explicit solution using modified **Chebyshev polynomials**.

Alex d’Aspremont

Turing Institute, May 2017. 13/33
Chebyshev polynomials $T_3(x, \sigma)$ and $T_5(x, \sigma)$ for $x \in [0, 1]$ and $\sigma = 0.85$.

The maximum value of $T_k$ on $[0, \sigma]$ decreases geometrically fast when $k$ grows.
Approximate Minimal Polynomial Extrapolation

Proposition [Scieur, d’Aspremont, and Bach, 2016]

**AMPE convergence.** Let $A$ be symmetric, $0 \preceq A \preceq \sigma I$ with $\sigma < 1$ and $c^*$ be the solution of (AMPE). Then

$$
\left\| \sum_{i=0}^{k} c_i^* x_i - x^* \right\|_2 \leq \kappa(A - I) \frac{2\zeta^k}{1 + \zeta^{2k}} \| x_0 - x^* \|_2
$$

(1)

where $\kappa(A - I)$ is the condition number of the matrix $A - I$ and $\zeta$ is given by

$$
\zeta = \frac{1 - \sqrt{1 - \sigma}}{1 + \sqrt{1 - \sigma}} < \sigma,
$$

(2)

Typically, $\sigma = 1 - \mu/L$ (gradient method) so the convergence rate is

$$
\left\| \sum_{i=0}^{k} c_i^* x_i - x^* \right\|_2 \leq \kappa(A - I) \left( \frac{1 - \sqrt{\mu/L}}{1 + \sqrt{\mu/L}} \right)^k \| x_0 - x^* \|_2
$$
**Approximate Minimal Polynomial Extrapolation**

**AMPE versus Nesterov, conjugate gradient.**

- Key difference with conjugate gradient: we do not observe $A$.

- Chebyshev polynomials satisfy a two-step recurrence. For quadratic minimization using the gradient method:

\[
\begin{aligned}
  z_{k-1} &= y_{k-1} - \frac{1}{L} (By_{k-1} - b) \\
  y_k &= \frac{\alpha_{k-1}}{\alpha_k} \left( \frac{2z_{k-1}}{\sigma} - y_{k-1} \right) - \frac{\alpha_{k-2}}{\alpha_k} y_{k-2}
\end{aligned}
\]

where $\alpha_k = \frac{2-\sigma}{\sigma} \alpha_{k-1} - \alpha_{k-2}$

- Nesterov’s acceleration recursively computes a similar polynomial with

\[
\begin{aligned}
  z_{k-1} &= y_{k-1} - \frac{1}{L} (By_{k-1} - b) \\
  y_k &= z_{k-1} + \beta_k (z_{k-1} - z_{k-2}),
\end{aligned}
\]

see also [Hardt, 2013].
Approximate Minimal Polynomial Extrapolation

Accelerating optimization algorithms. For gradient descent, we have

\[ \tilde{x}_{k+1} := \tilde{x}_k - \frac{1}{L} \nabla f(\tilde{x}_k) \]

This means

\[ \tilde{x}_{k+1} - x^* := A(\tilde{x}_k - x^*) + O(\|\tilde{x}_k - x^*\|_2^2) \]

where

\[ A = I - \frac{1}{L} \nabla^2 f(x^*), \]

meaning that \( \|A\|_2 \leq 1 - \frac{\mu}{L} \), whenever \( \mu I \preceq \nabla^2 f(x) \preceq LI \).

Approximation error is a sum of three terms

\[ \left\| \sum_{i=0}^{k} \tilde{c}_i \tilde{x}_i - x^* \right\|_2 \leq \left\| \sum_{i=0}^{k} c_i x_i - x^* \right\|_2 + \left\| \sum_{i=0}^{k} (\tilde{c}_i - c_i) x_i \right\|_2 + \left\| \sum_{i=0}^{k} \tilde{c}_i (\tilde{x}_i - x_i) \right\|_2 \]

**Stability** is key here.
The iterations span a Krylov subspace

\[ K_k = \text{span} \{ U_0, AU_0, \ldots, A^{k-1}U_0 \} \]

so the matrix \( U \) in AMPE is a **Krylov matrix**.

Similar to **Hankel or Toeplitz** case. \( U^T U \) has a condition number typically growing exponentially with dimension [Tyrtyshnikov, 1994].

In fact, the Hankel, Toeplitz and Krylov problems are directly connected, hence the link with Levinson-Durbin [Heinig and Rost, 2011].

For generic optimization problems, eigenvalues are perturbed by deviations from the linear model, which can make the situation even worse.

Be wise, regularize . . .
Outline

- Introduction
- Minimal Polynomial Extrapolation
- Regularized MPE
- Numerical results
Regularized Minimal Polynomial Extrapolation

**Regularized AMPE.** Add a regularization term to AMPE.

- Regularized formulation of problem (AMPE),

\[
\begin{align*}
\text{minimize} & \quad c^T(U^TU + \lambda I)c \\
\text{subject to} & \quad 1^Tc = 1
\end{align*}
\]  

(RMPE)

- Solution given by a linear system of size \(k + 1\).

\[
c^*_\lambda = \frac{(U^TU + \lambda I)^{-1}1}{1^T(U^TU + \lambda I)^{-1}1}
\]  

(3)
Regularized Minimal Polynomial Extrapolation

Regularized AMPE.

Proposition [Scieur et al., 2016]

**Stability** Let $c^*_\lambda$ be the solution of problem (RMPE). Then the solution of problem (RMPE) for the perturbed matrix $\tilde{U} = U + E$ is given by $c^*_\lambda + \Delta c_\lambda$ where

$$\|\Delta c_\lambda\|_2 \leq \frac{\|P\|_2}{\lambda} \|c^*_\lambda\|_2$$

with $P = \tilde{U}^T \tilde{U} - U^T U$ the perturbation matrix.
Regularized Minimal Polynomial Extrapolation

**RMPE algorithm.**

**Input:** Sequence \( \{x_0, x_1, \ldots, x_{k+1}\} \), parameter \( \lambda > 0 \)

1. Form \( U = [x_1 - x_0, \ldots, x_{k+1} - x_k] \)
2. Solve the linear system \((U^T U + \lambda I) z = 1\)
3. Set \( c = z / (z^T 1) \)

**Output:** Return \( \sum_{i=0}^{k} c_i x_i \), approximating the optimum \( x^* \)
**Regularized AMPE.** Define

\[
S(k, \alpha) \triangleq \min_{\{q \in \mathbb{R}_k[x]: q(1)=1\}} \left\{ \max_{x \in [0, \sigma]} ((1 - x)q(x))^2 + \alpha \|q\|_2^2 \right\},
\]

**Proposition [Scieur et al., 2016]**

**Error bounds** Let matrices \( X = [x_0, x_1, ..., x_k], \; \tilde{X} = [x_0, \tilde{x}_1, ..., \tilde{x}_k] \) and scalar \( \kappa = \|(A - I)^{-1}\|_2 \). Suppose \( \tilde{c}_\lambda^* \) solves problem (RMPE) and assume \( A = g'(x^*) \) symmetric with \( 0 \leq A \leq \sigma I \) where \( \sigma < 1 \). Let us write the perturbation matrices \( P = \tilde{U}^T \tilde{U} - U^T U \) and \( E = (X - \tilde{X}) \). Then

\[
\|\tilde{X} \tilde{c}_\lambda^* - x^*\|_2 \leq C(E, P, \lambda) \; S(k, \lambda/\|x_0 - x^*\|_2^2)^{1/2} \; \|x_0 - x^*\|_2
\]

where

\[
C(E, P, \lambda) = \left( \kappa^2 + \frac{1}{\lambda} \left( 1 + \frac{\|P\|_2}{\lambda} \right)^2 \left( \|E\|_2 + \kappa \frac{\|P\|_2}{2\sqrt{\lambda}} \right)^2 \right)^{1/2}
\]
On the gradient method. Setting for instance $L = 100$, $\mu = 10$, $M = 10^{-1}$, $\|x_0 - x^*\|_2 = 10^{-4}$ and finally $\lambda = \|P\|_2$.

Left: Relative value for the regularization parameter $\lambda$ used in the theoretical bound. Right: Convergence speedup relative to the gradient method, for Nesterov’s accelerated method and the theoretical RMPE.
Proposition [Scieur et al., 2016]

**Asymptotic acceleration**  Using the gradient method with stepsize in $]0, \frac{2}{\mu}[$ on a $L$-smooth, $\mu$-strongly convex function $f$ with Lipschitz-continuous Hessian of constant $M$.

$$
\| \hat{X} \tilde{c}_\lambda^* - x^* \|_2 \leq \kappa \left( 1 + \frac{(1 + \frac{1}{\beta})^2}{4\beta^2} \right)^{1/2} \frac{2\zeta^k}{1 + \zeta^{2k}} \| x_0 - x^* \|
$$

with

$$
\zeta = \frac{1 - \sqrt{\mu/L}}{1 + \sqrt{\mu/L}}
$$

for $\| x_0 - x^* \|$ small enough, where $\lambda = \beta \| P \|_2$ and $\kappa = \frac{L}{\mu}$ is the condition number of the function $f(x)$.

We (asymptotically) recover the accelerated rate in [Nesterov, 1983].
Regularized Minimal Polynomial Extrapolation

Complexity, online mode.

- **Cholesky updates.** Given the Cholesky factorization $LL^T = \tilde{U}^T \tilde{U} + \lambda I$ and a new vector $u_+$,

$$L_+L_+^T = \begin{bmatrix} L & 0 \\ a^T & b \end{bmatrix} \begin{bmatrix} L^T & a \\ 0 & b \end{bmatrix} = \begin{bmatrix} \tilde{U}^T \tilde{U} + \lambda I & \tilde{U}^T u_+ \\ (\tilde{U}^T u_+)^T & u_+^T u_+ + \lambda \end{bmatrix}.$$ 

the solutions $a$ and $b$ are

$$a = L^{-1} \tilde{U}^T u_+, \quad b = a^T a + \lambda.$$

- The complexity of an update at iteration $i$ is $O(in + i^2)$, so the overall complexity after $k$ iterations is

$$O(nk^2 + k^3)$$

In the experiments that follow, $k$ is typically 5...
Smooth functions. Suppose $f$ is not strongly convex.

- The function
  \[
  \min_{x \in \mathbb{R}^n} f_\varepsilon(x) \triangleq f(x) + \frac{\varepsilon}{2D^2} \|x\|_2^2
  \]
  has a Lipschitz continuous gradient with parameter $L + \varepsilon/D^2$ and is strongly convex with parameter $\varepsilon/D^2$.

- Accelerated algorithm converge with a linear rate, with a bound equivalent to
  \[
  \sqrt{1 + \frac{LD^2}{\varepsilon}},
  \]
  which matches the optimal complexity bound for smooth functions.

Handling the strongly convex case, allows us to produce bounds in the smooth case, on paper...
Outline

- Introduction
- Minimal Polynomial Extrapolation
- Regularized MPE
- **Numerical results**
Logistic regression with $\ell_2$ regularization, on Madelon Dataset (500 features, 2000 data points), solved using several algorithms. The penalty parameter has been set to $10^2$ in order to have a condition number equal to $1.2 \times 10^9$. 
Logistic regression on *Sido0 Dataset* (4932 features, 12678 data points). Penalty parameter $\tau = 10^2$, so the condition number is equal to $1.5 \times 10^5$. 
Logistic regression on *Madelon UCI Dataset*, solved using the gradient method, Nesterov’s method and AMPE (i.e. RMPE with $\lambda = 0$). The condition number is equal to $1.2 \times 10^9$. We see that without regularization, AMPE becomes unstable as $\|\tilde{U}^T \tilde{U}^{-1}\|_2$ gets too large.
Conclusion

**Postprocessing works.**

- Simple *postprocessing* step.
- Marginal complexity, can be performed in parallel.
- Significant convergence speedup over optimal methods.
- Adaptive. Does not need knowledge of smoothness parameters.

Work in progress. . .

- Extrapolating accelerated methods.
- Constrained problems.
- Better handling of smooth functions.
- . . .
Open problems

- **Regularization.** How do we account for the fact that we are estimating the limit of a VAR sequence with a fixed point?

- The VAR matrix $A$ is formed implicitly, but we have some information on its spectrum through smoothness.

- Explicit bounds on the **regularized Chebyshev problem,**

\[
S(k, \alpha) \triangleq \min_{\{q \in \mathbb{R}_k[x] : q(1) = 1\}} \left\{ \max_{x \in [0, \sigma]} ((1 - x)q(x))^2 + \alpha \|q\|_2^2 \right\}.
\]

References


Peter Wynn. On a device for computing the $\epsilon m(Sn)$ transformation. *Mathematical Tables and Other Aids to Computation*, 10(54):91–96, 1956.