UNIVERSITY OF CAMBRIDGE

Choose your path wisely

Gradient descent in a Bregman distance framework

Martin Benning[1] Marta M. Betcke[2] Matthias J. Ehrhardt[1]
Carola-Bibiane Schönlieb[1]

[1]University of Cambridge, UK

[2]University College London, UK

INI-SLB Workshop on large scale optimisation, Schlumberger Cambridge Research, 28.09.2017

Outline

Outline

Introduction

Gradient descent

- We are interested in finding stationary points of smooth but not necessarily convex functionals $E : \mathcal{U} \to \mathbb{R}$ over Banach spaces $\mathcal{U}$.
- A standard numerical approach for finding these in finite dimensions or Hilbert spaces is gradient descent, i.e.

$$u^{k+1} = u^k - \tau^k \nabla E(u^k)\,,$$

  for all $k \in \mathbb{N}$ and some initial value u^0.
- How can this be extended to more general Banach spaces? And why would we be interested in such an extension?

Introduction

Gradient descent

- We are interested in finding stationary points of smooth but not necessarily convex functionals $E : \mathcal{U} \to \mathbb{R}$ over Banach spaces $\mathcal{U}$.
- A standard numerical approach for finding these in finite dimensions or Hilbert spaces is gradient descent, i.e.

$$u^{k+1} = u^k - \tau^k \nabla E(u^k)\,,$$

  for all $k \in \mathbb{N}$ and some initial value u^0.
- How can this be extended to more general Banach spaces? And why would we be interested in such an extension?

Introduction

Gradient descent

- We are interested in finding stationary points of smooth but not necessarily convex functionals $E : \mathcal{U} \to \mathbb{R}$ over Banach spaces $\mathcal{U}$.
- A standard numerical approach for finding these in finite dimensions or Hilbert spaces is gradient descent, i.e.

$$u^{k+1} = u^k - \tau^k \nabla E(u^k)\,,$$

  for all $k \in \mathbb{N}$ and some initial value u^0.
- How can this be extended to more general Banach spaces? And why would we be interested in such an extension?

Introduction - blind deconvolution

Imagine non-convex problems such as blind deconvolution, i.e.

$$(\hat{u}, \hat{h}) \in \underset{\substack{u \in \mathbb{R}^n \\ h \in \mathbb{R}^r}}{\arg\min} \left\{ \frac{1}{2} \|u * h - f\|_2^2 + \alpha \mathrm{TV}(u) + \chi_S(h) \right\}. \qquad (1)$$

Here $*$ denotes the discrete two-dimensional convolution, $\mathrm{TV}(u) := \| \|\nabla u\|_2 \|_1$ the total variation, and $\chi_S(h) = \begin{cases} 0 & h \in S \\ \infty & \text{else} \end{cases}$ the characteristic function over the simplex constraint

$$S = \left\{ h \in \mathbb{R}^r \;\middle|\; h_j \geq 0, \; \sum_j h_j = 1, \; \forall j \right\}.$$

Introduction - blind deconvolution

Imagine non-convex problems such as blind deconvolution, i.e.

$$(\hat{u}, \hat{h}) \in \underset{\substack{u \in \mathbb{R}^n \\ h \in \mathbb{R}^r}}{\arg\min} \left\{ \frac{1}{2} \|u * h - f\|_2^2 + \alpha \mathrm{TV}(u) + \chi_S(h) \right\} . \qquad (1)$$

Here $*$ denotes the discrete two-dimensional convolution, $\mathrm{TV}(u) := \| \|\nabla u\|_2 \|_1$ the total variation, and $\chi_S(h) = \begin{cases} 0 & h \in S \\ \infty & \text{else} \end{cases}$ the characteristic function over the simplex constraint

$$S = \left\{ h \in \mathbb{R}^r \;\middle|\; h_j \geq 0, \; \sum_j h_j = 1, \; \forall j \right\} .$$

Introduction - blind deconvolution

We discover several issues here:

- Problem (1) is non-smooth $\Rightarrow$ gradient descent not applicable
- Remedy: use forward backward splitting [Lions, Mercier, 1979]
- But: real issue is that total variation of blurred images is small

TV $= 22.46$ TV $= 6.03$ TV $= 12.04$

- How do we prevent an optimisation procedure from picking a blurry solution?

Introduction - blind deconvolution

We discover several issues here:

- Problem (1) is non-smooth $\Rightarrow$ gradient descent not applicable
- Remedy: use forward backward splitting [Lions, Mercier, 1979]
- But: real issue is that total variation of blurred images is small

TV $= 22.46$ TV $= 6.03$ TV $= 12.04$

- How do we prevent an optimisation procedure from picking a blurry solution?

Introduction - blind deconvolution

We discover several issues here:

- Problem (1) is non-smooth $\Rightarrow$ gradient descent not applicable
- Remedy: use forward backward splitting [Lions, Mercier, 1979]
- But: real issue is that total variation of blurred images is small



$\mathrm{TV} = 22.46$ $\mathrm{TV} = 6.03$ $\mathrm{TV} = 12.04$

- How do we prevent an optimisation procedure from picking a blurry solution?

Introduction - blind deconvolution

We discover several issues here:

- Problem (1) is non-smooth $\Rightarrow$ gradient descent not applicable
- Remedy: use forward backward splitting [Lions, Mercier, 1979]
- But: real issue is that total variation of blurred images is small

TV $= 22.46$

TV $= 6.03$

TV $= 12.04$

- How do we prevent an optimisation procedure from picking a blurry solution?

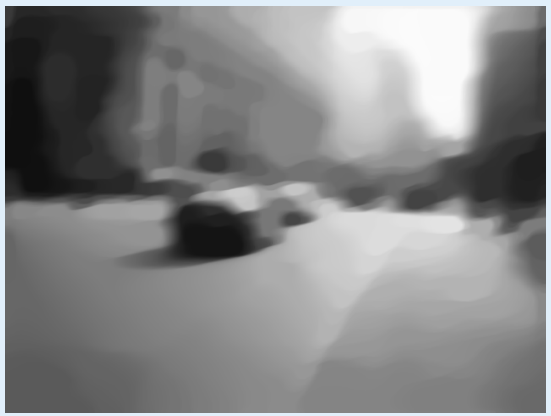
Introduction - blind deconvolution

We discover several issues here:

- Problem (1) is non-smooth $\Rightarrow$ gradient descent not applicable
- Remedy: use forward backward splitting [Lions, Mercier, 1979]
- But: real issue is that total variation of blurred images is small

TV $= 22.46$

TV $= 10.79$



TV $= 2.41$

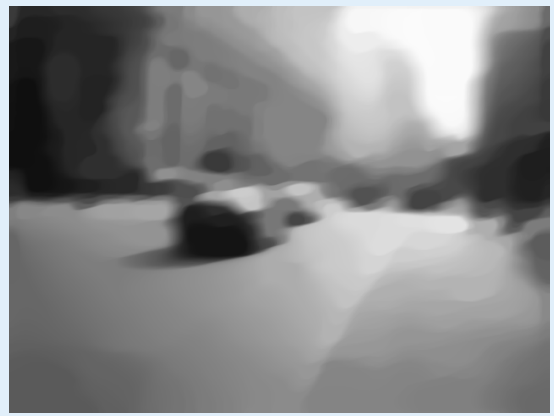
- How do we prevent an optimisation procedure from picking a blurry solution?

Introduction - blind deconvolution

We discover several issues here:

- Problem (1) is non-smooth $\Rightarrow$ gradient descent not applicable
- Remedy: use forward backward splitting [Lions, Mercier, 1979]
- But: real issue is that total variation of blurred images is small





TV $= 22.46$ TV $= 10.79$ TV $= 2.41$

- How do we prevent an optimisation procedure from picking a blurry solution?

Introduction - linearised Bregman iteration

Gradient descent cast as an optimisation problem

- Gradient descent can be cast as

$$u^{k+1} = \arg\min_u \left\{ \tau^k \langle \nabla E(u^k), u - u^k \rangle + \frac{1}{2} \|u - u^k\|^2 \right\},$$

where $\|\cdot\| = \sqrt{\langle \cdot, \cdot \rangle}$ is the norm of some Hilbert space.

Introduction - linearised Bregman iteration

Linearised Bregman iteration for non-convex optimisation

- Gradient descent can be cast as

$$u^{k+1} = \arg\min_u \left\{ \tau^k \langle \nabla E(u^k), u - u^k \rangle + D_J^{p^k}(u, u^k) \right\},$$
$$p^{k+1} = p^k - \tau^k \nabla E(u^k),$$

where $D_J^{p^k}(u^{k+1}, u^k) := J(u^{k+1}) - J(u^k) - \langle p^k, u^{k+1} - u^k \rangle$ is the (generalised) Bregman distance for $p^k \in \partial J(u^k)$.

Introduction - linearised Bregman iteration

Linearised Bregman iteration for non-convex optimisation

- Gradient descent can be cast as

$$u^{k+1} = \arg\min_u \left\{ \tau^k \langle \nabla E(u^k), u - u^k \rangle + D_J^{p^k}(u, u^k) \right\},$$
$$p^{k+1} = p^k - \tau^k \nabla E(u^k),$$

  where $D_J^{p^k}(u^{k+1}, u^k) := J(u^{k+1}) - J(u^k) - \langle p^k, u^{k+1} - u^k \rangle$ is the (generalised) Bregman distance for $p^k \in \partial J(u^k)$.
- Here $J : \mathcal{U} \to \mathbb{R} \cup \{\infty\}$ is proper, lower semi-continuous (l.s.c) and convex, and

$$\partial J(u) := \{p \in \mathcal{U}^* \mid D_J^p(v, u) \geq 0, \ \forall v \in \mathcal{U}\} \subset \mathcal{U}^* .$$

Introduction - blind deconvolution

We use the linearised Bregman iteration to (approximately) minimise

$$E(u,h) = \frac{1}{2}\|u * h - f\|_2^2 \,,$$

where $*$ denotes the two-dimensional, discrete convolution.

We make the following choice for J:

$$J(u,h) := \frac{1}{2}\|u\|_2^2 + \alpha \mathrm{TV}(u) + \sum_{j=1}^{r} h_j \log(h_j) - h_j + \chi_S(h)$$

with $\chi_S(h) = \begin{cases} 0 & h \in S \\ \infty & \text{else} \end{cases}$ and the simplex constraint set

$$S = \left\{ h \in \mathbb{R}^r \;\middle|\; h_j \geq 0,\ \sum_j h_j = 1,\ \forall j \right\} .$$

Introduction - blind deconvolution

We use the linearised Bregman iteration to (approximately) minimise

$$E(u,h) = \frac{1}{2}\|u * h - f\|_2^2\,,$$

where $*$ denotes the two-dimensional, discrete convolution.
We make the following choice for J:

$$J(u,h) := \frac{1}{2}\|u\|_2^2 + \alpha \mathrm{TV}(u) + \sum_{j=1}^{r} h_j \log(h_j) - h_j + \chi_S(h)$$

with $\chi_S(h) = \begin{cases} 0 & h \in S \\ \infty & \text{else} \end{cases}$ and the simplex constraint set

$$S = \left\{ h \in \mathbb{R}^r \;\middle|\; h_j \geq 0,\ \sum_j h_j = 1,\ \forall j \right\}.$$

Introduction - blind deconvolution

Note that the choice of J results in the following algorithm:

$$u^{k+1} = \arg\min_u \left\{ \frac{1}{2} \left\| u - (u^k - (u^k * h^k - f) \,\overline{*}\, h^k + \alpha q^k) \right\|_2^2 + \alpha \mathrm{TV}(u) \right\},$$

$$h_j^{k+1} = \frac{h_j^k \exp(-\tau^k (u^k \,\overline{*}\, (u^k * h^k - f))_j)}{\sum_{j=1}^r h_j^k \exp(-\tau^k (u^k \,\overline{*}\, (u^k * h^k - f))_j)} \quad \forall j \in \{1, \dots, r\}, \ (\dagger)$$

$$q^{k+1} = q^k - \frac{1}{\alpha} \left(u^{k+1} - u^k + (u^k * h^k - f) \,\overline{*}\, h^k \right), \quad q^k \in \partial \mathrm{TV}(u^k);$$

here $\overline{*}$ denotes the transpose convolution operation.

($\dagger$) is basically entropic descent, see Nemirovsky, Yudin, 1982 & Beck, Teboulle, Operations Research Letters, 2003

Introduction - blind deconvolution

Note that the choice of J results in the following algorithm:

$$u^{k+1} = \arg\min_u \left\{ \frac{1}{2} \left\| u - (u^k - (u^k * h^k - f) \,\overline{*}\, h^k + \alpha q^k) \right\|_2^2 + \alpha \mathrm{TV}(u) \right\},$$

$$h_j^{k+1} = \frac{h_j^k \exp(-\tau^k (u^k \,\overline{*}\, (u^k * h^k - f))_j)}{\sum_{j=1}^r h_j^k \exp(-\tau^k (u^k \,\overline{*}\, (u^k * h^k - f))_j)} \quad \forall j \in \{1, \dots, r\}, \ (\dagger)$$

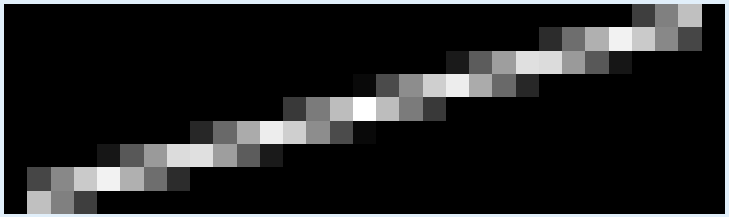
$$q^{k+1} = q^k - \frac{1}{\alpha} \left(u^{k+1} - u^k + (u^k * h^k - f) \,\overline{*}\, h^k \right), \quad q^k \in \partial \mathrm{TV}(u^k);$$

here $\overline{*}$ denotes the transpose convolution operation.

($\dagger$) is basically entropic descent, see Nemirovsky, Yudin, 1982 & Beck, Teboulle, Operations Research Letters, 2003

Introduction - blind deconvolution

We consider the same (noisy) motion blur example as before:



Reconstruction via linearised Bregman iteration

Introduction - linearised Bregman iteration

A non-exhaustive list of works on Bregman iteration:

- Censor, Zenios. Journal of Opt. Theory and App. 1992
- Teboulle. Mathematics of Operations Research 1992
- Eckstein. Mathematics of Operations Research 1993
- Kiwiel. SIAM Journal on Control and Optimization 1997
- Beck, Teboulle. Operations Research Letter 2003
- Osher, Burger, Goldfarb, Xu, Yin. Multiscale Mod. & Sim. 2005
- Yin, Osher, Goldfarb, Darbon. SIAM Journal on Imaging Sci. 2008
- Cai, Osher, Shen. Mathmatics of Computation 2009
- Bachmayr, Burger. Inverse Problems 2009
- Yin. SIAM Journal on Imaging Sciences 2010

Outline

The linearised Bregman iteration

Linearised Bregman iteration

$$u^{k+1} = \arg\min_{u \in \mathcal{U}} \left\{ \tau^k \langle \nabla E(u^k), u - u^k \rangle + D_J^{p^k}(u, u^k) \right\},$$
$$p^{k+1} = p^k - \tau^k \nabla E(u^k),$$

for $k \in \mathbb{N}$, and some $u^0 \in \mathcal{U}$ and $p^0 \in \partial J(u^0)$.

Note that this Algorithm can be simplified to

$$u^{k+1} = \arg\min_{u \in \mathcal{U}} \left\{ \tau^k \left\langle \nabla E(u^k) - \frac{1}{\tau^k} p^k, u - u^k \right\rangle + J(u) \right\},$$
$$p^{k+1} = p^k - \tau^k \nabla E(u^k).$$

The linearised Bregman iteration

Linearised Bregman iteration

$$u^{k+1} = \arg\min_{u \in \mathcal{U}} \left\{ \tau^k \langle \nabla E(u^k), u - u^k \rangle + D_J^{p^k}(u, u^k) \right\},$$
$$p^{k+1} = p^k - \tau^k \nabla E(u^k),$$

for $k \in \mathbb{N}$, and some $u^0 \in \mathcal{U}$ and $p^0 \in \partial J(u^0)$.

Note that this Algorithm can be simplified to

$$u^{k+1} = \arg\min_{u \in \mathcal{U}} \left\{ \tau^k \left\langle \nabla E(u^k) - \frac{1}{\tau^k} p^k, u - u^k \right\rangle + J(u) \right\},$$
$$p^{k+1} = p^k - \tau^k \nabla E(u^k).$$

The linearised Bregman iteration

Specialised linearised Bregman iteration (I)

Specific choice $J^k(u) = \frac{1}{2}\|u\|_2^2 + \tau^k \alpha R(u)$:

$$
\begin{aligned}
u^{k+1} &= \arg\min_{u \in \mathcal{U}} \left\{ \frac{1}{2}\|u - (u^k - \tau^k \nabla E(u^k))\|_2^2 + \tau^k \alpha D_R^{q^k}(u, u^k) \right\}, \\
&= \arg\min_{u \in \mathcal{U}} \left\{ \frac{1}{2}\|u - (u^k + \tau^k(\alpha q^k - \nabla E(u^k)))\|_2^2 + \tau^k \alpha R(u) \right\}, \\
&= (I + \alpha \tau^k \partial R)^{-1}(u^k + \tau^k(\alpha q^k - \nabla E(u^k)) \\
q^{k+1} &= q^k - \frac{1}{\tau^k \alpha}\left(u^{k+1} - u^k + \tau^k \nabla E(u^k)\right), \qquad q^k \in \partial R(u^k)
\end{aligned}
$$

The linearised Bregman iteration

Specialised linearised Bregman iteration (II)

More general specific choice $J^k(u) = H(u) + \tau^k \alpha R(u)$:

$$
\begin{aligned}
u^{k+1} &= \arg\min_{u \in \mathcal{U}} \left\{ H(u) - \langle u, \nabla H(u^k) - \tau^k \nabla E(u^k) \rangle + \tau^k \alpha D_R^{q^k}(u, u^k) \right\}, \\
&= \arg\min_{u \in \mathcal{U}} \left\{ H(u) - \langle u, \nabla H(u^k) + \tau^k (\alpha q^k - \nabla E(u^k)) \rangle + \tau^k \alpha R(u) \right\}, \\
&= (\nabla H + \alpha \tau^k \partial R)^{-1} (\nabla H^* (\nabla H(u^k) + \tau^k (\alpha q^k - \nabla E(u^k)))) \qquad (2a)
\end{aligned}
$$

$$
q^{k+1} = q^k - \frac{1}{\tau^k \alpha} \left(\nabla H(u^{k+1}) - \nabla H(u^k) + \tau^k \nabla E(u^k) \right), \; q^k \in \partial R(u^k) \qquad (2b)
$$

Assumptions on E, H and R

- $E : \mathcal{U} \to \mathbb{R}$ is a smooth functional that is bounded from below and has L-Lipschitz-continuous gradient, i.e.

$$\|\nabla E(u) - \nabla E(v)\|_{\mathcal{U}^*} \leq L\|u - v\|_{\mathcal{U}}\,, \qquad \forall u, v \in \mathcal{U}\,.$$

- $H : \mathcal{U} \to \mathbb{R}$ is a γ-strongly convex functional, i.e.

$$\gamma\|u - v\|_{\mathcal{U}}^2 \leq D_H^{\text{symm}}(u, v) := D_H(u, v) + D_H(v, u)\,, \quad \forall u, v \in \mathcal{U}\,,$$

  and has δ-Lipschitz gradient ∇H.
- The conditions on E and H imply the useful estimate

$$E(u) \leq E(v) + \langle \nabla E(v), u - v\rangle + \frac{L}{\gamma} D_H(u, v)\,. \tag{3}$$

- $R : \mathcal{U} \to \mathbb{R} \cup \{\infty\}$ is a proper, l.s.c. and convex functional.

Assumptions on E, H and R

- $E : \mathcal{U} \to \mathbb{R}$ is a smooth functional that is bounded from below and has L-Lipschitz-continuous gradient, i.e.

$$\|\nabla E(u) - \nabla E(v)\|_{\mathcal{U}^*} \leq L\|u - v\|_{\mathcal{U}}\,, \qquad \forall u, v \in \mathcal{U}\,.$$

- $H : \mathcal{U} \to \mathbb{R}$ is a γ-strongly convex functional, i.e.

$$\gamma\|u - v\|_{\mathcal{U}}^2 \leq D_H^{\text{symm}}(u, v) := D_H(u, v) + D_H(v, u)\,, \quad \forall u, v \in \mathcal{U}\,,$$

  and has δ-Lipschitz gradient ∇H.

- The conditions on E and H imply the useful estimate

$$E(u) \leq E(v) + \langle \nabla E(v), u - v\rangle + \frac{L}{\gamma} D_H(u, v)\,. \tag{3}$$

- $R : \mathcal{U} \to \mathbb{R} \cup \{\infty\}$ is a proper, l.s.c. and convex functional.

Assumptions on E, H and R

- $E : \mathcal{U} \to \mathbb{R}$ is a smooth functional that is bounded from below and has L-Lipschitz-continuous gradient, i.e.

$$\|\nabla E(u) - \nabla E(v)\|_{\mathcal{U}^*} \leq L\|u - v\|_{\mathcal{U}}\,, \qquad \forall u, v \in \mathcal{U}\ .$$

- $H : \mathcal{U} \to \mathbb{R}$ is a γ-strongly convex functional, i.e.

$$\gamma\|u - v\|_{\mathcal{U}}^2 \leq D_H^{\text{symm}}(u, v) := D_H(u, v) + D_H(v, u)\,, \quad \forall u, v \in \mathcal{U}\,,$$

  and has δ-Lipschitz gradient ∇H.
- The conditions on E and H imply the useful estimate

$$E(u) \leq E(v) + \langle \nabla E(v), u - v\rangle + \frac{L}{\gamma} D_H(u, v)\,. \tag{3}$$

- $R : \mathcal{U} \to \mathbb{R} \cup \{\infty\}$ is a proper, l.s.c. and convex functional.

Assumptions on E, H and R

- $E : \mathcal{U} \to \mathbb{R}$ is a smooth functional that is bounded from below and has L-Lipschitz-continuous gradient, i.e.

$$\|\nabla E(u) - \nabla E(v)\|_{\mathcal{U}^*} \leq L\|u - v\|_{\mathcal{U}}\,, \qquad \forall u, v \in \mathcal{U}\ .$$

- $H : \mathcal{U} \to \mathbb{R}$ is a γ-strongly convex functional, i.e.

$$\gamma\|u - v\|_{\mathcal{U}}^2 \leq D_H^{\text{symm}}(u, v) := D_H(u, v) + D_H(v, u)\,, \quad \forall u, v \in \mathcal{U}\,,$$

  and has δ-Lipschitz gradient ∇H.
- The conditions on E and H imply the useful estimate

$$E(u) \leq E(v) + \langle \nabla E(v), u - v\rangle + \frac{L}{\gamma} D_H(u, v)\,. \tag{3}$$

- $R : \mathcal{U} \to \mathbb{R} \cup \{\infty\}$ is a proper, l.s.c. and convex functional.

Assumptions for the linearised Bregman iteration

Surrogate objective

For the iterate $u^{k-1} \in \mathcal{U}$ of (2) and corresponding subgradient $q^{k-1} \in \partial R(u^{k-1})$ we define $E^k : \mathcal{U} \to \mathbb{R} \cup \{\infty\}$ as

$$E^k(u) := E(u) + \alpha D_R^{q^{k-1}}(u, u^{k-1}) \,. \tag{4}$$

Stepsize constraint

Choose $0 < \tau^k$ such that

$$D_H(u^{k+1}, u^k) \leq \frac{\gamma(1 - \tau^k \rho)}{\tau^k L} D_H^{\text{symm}}(u^{k+1}, u^k) \tag{5}$$

holds true for all $k \in \mathbb{N}$ and $0 < \delta < 1/\max_k \tau^k$.

Assumptions for the linearised Bregman iteration

Surrogate objective

For the iterate $u^{k-1} \in \mathcal{U}$ of (2) and corresponding subgradient $q^{k-1} \in \partial R(u^{k-1})$ we define $E^k : \mathcal{U} \to \mathbb{R} \cup \{\infty\}$ as

$$E^k(u) := E(u) + \alpha D_R^{q^{k-1}}(u, u^{k-1}) \,. \tag{4}$$

Stepsize constraint

Choose $0 < \tau^k$ such that

$$D_H(u^{k+1}, u^k) \leq \frac{\gamma(1 - \tau^k \rho)}{\tau^k L} D_H^{\text{symm}}(u^{k+1}, u^k) \tag{5}$$

holds true for all $k \in \mathbb{N}$ and $0 < \delta < 1/\max_k \tau^k$.

Assumptions for the linearised Bregman iteration

Surrogate objective

For the iterate $u^{k-1} \in \mathcal{U}$ of (2) and corresponding subgradient $q^{k-1} \in \partial R(u^{k-1})$ we define $E^k : \mathcal{U} \to \mathbb{R} \cup \{\infty\}$ as

$$E^k(u) := E(u) + \alpha D_R^{q^{k-1}}(u, u^{k-1}) \,. \tag{4}$$

Stepsize constraint

Choose $0 < \tau^k$ such that

$$\frac{1}{2}\|u^{k+1} - u^k\|_2^2 \leq \left(\frac{1 - \tau^k \rho}{\tau^k L}\right) \|u^{k+1} - u^k\|_2^2 \tag{5}$$

holds true for all $k \in \mathbb{N}$ and $0 < \delta < 1/\max_k \tau^k$.

Assumptions for the linearised Bregman iteration

Surrogate objective

For the iterate $u^{k-1} \in \mathcal{U}$ of (2) and corresponding subgradient $q^{k-1} \in \partial R(u^{k-1})$ we define $E^k : \mathcal{U} \to \mathbb{R} \cup \{\infty\}$ as

$$E^k(u) := E(u) + \alpha D_R^{q^{k-1}}(u, u^{k-1}) \,. \tag{4}$$

Stepsize constraint

Choose $0 < \tau^k$ such that

$$D_H(u^{k+1}, u^k) \leq \frac{\gamma(1 - \tau^k \rho)}{\tau^k L} D_H^{\text{symm}}(u^{k+1}, u^k) \tag{5}$$

holds true for all $k \in \mathbb{N}$ and $0 < \delta < 1/\max_k \tau^k$.

Properties of the linearised Bregman iteration

Sufficient decrease property

Let $0 < \tau^k < 1/\delta$ satisfy (5). Then the iterates of the linearised Bregman iteration (2) satisfy

$$\begin{aligned} & E^{k+1}(u^{k+1}) + \rho D_H^{\text{symm}}(u^{k+1}, u^k) \\ & \quad + \alpha \left(D_R^{q^{k+1}}(u^k, u^{k+1}) + D_R^{q^{k-1}}(u^k, u^{k-1}) \right) \\ & \leq E^k(u^k)\,. \end{aligned} \tag{6}$$

In addition, we observe

$$\lim_{k\to\infty} D_H^{\text{symm}}(u^{k+1}, u^k) = 0 \quad \text{and} \quad \lim_{k\to\infty} D_R^{\text{symm}}(u^{k+1}, u^k) = 0\,.$$

Properties of the linearised Bregman iteration

Sufficient decrease property

Let $0 < \tau^k < 1/\delta$ satisfy (5). Then the iterates of the linearised Bregman iteration (2) satisfy

$$\begin{aligned} & E^{k+1}(u^{k+1}) + \rho \|u^{k+1} - u^k\|_2^2 \\ & \quad + \alpha \left(D_R^{q^{k+1}}(u^k, u^{k+1}) + D_R^{q^{k-1}}(u^k, u^{k-1}) \right) \\ & \leq E^k(u^k)\,. \end{aligned} \tag{6}$$

In addition, we observe

$$\lim_{k\to\infty} \|u^{k+1} - u^k\|_2^2 = 0 \quad \text{and} \quad \lim_{k\to\infty} D_R^{\text{symm}}(u^{k+1}, u^k) = 0\,.$$

Properties of the linearised Bregman iteration

Sufficient decrease property

Let $0 < \tau^k < 1/\delta$ satisfy (5). Then the iterates of the linearised Bregman iteration (2) satisfy

$$\begin{aligned} & E^{k+1}(u^{k+1}) + \rho D_H^{\text{symm}}(u^{k+1}, u^k) \\ & \quad + \alpha \left(D_R^{q^{k+1}}(u^k, u^{k+1}) + D_R^{q^{k-1}}(u^k, u^{k-1}) \right) \\ & \leq E^k(u^k)\,. \end{aligned} \tag{6}$$

In addition, we observe

$$\lim_{k\to\infty} D_H^{\text{symm}}(u^{k+1}, u^k) = 0 \quad \text{and} \quad \lim_{k\to\infty} D_R^{\text{symm}}(u^{k+1}, u^k) = 0\,.$$

Properties of the linearised Bregman iteration

A subgradient lower bound of the iterates gap

Let the same assumptions hold true as in the previous theorem. Then the iterates (2) satisfy

$$\|\nabla E(u^{k+1}) + \alpha(q^{k+1} - q^k)\|_{\mathcal{U}^*} \leq \rho_2 \|u^{k+1} - u^k\|_{\mathcal{U}}\,, \tag{7}$$

for $q^k \in \partial R(u^k)$, $q^{k+1} \in \partial R(u^{k+1})$, $\rho_2 := \left(L + \delta/(\min_k \tau^k)\right)$ and $k \in \mathbb{N}$.

Proof: Use Lipschitz-continuity of ∇E, Lipschitz-continuity of ∇H and the triangle inequality for norms.

Properties of the linearised Bregman iteration

A subgradient lower bound of the iterates gap

Let the same assumptions hold true as in the previous theorem. Then the iterates (2) satisfy

$$\|\nabla E(u^{k+1}) + \alpha(q^{k+1} - q^k)\|_{\mathcal{U}^*} \leq \rho_2 \|u^{k+1} - u^k\|_{\mathcal{U}}\,, \tag{7}$$

for $q^k \in \partial R(u^k)$, $q^{k+1} \in \partial R(u^{k+1})$, $\rho_2 := \left(L + \delta/(\min_k \tau^k)\right)$ and $k \in \mathbb{N}$.

Proof: Use Lipschitz-continuity of ∇E, Lipschitz-continuity of ∇H and the triangle inequality for norms.

Outline

Convergence of the surrogate functional

Definition (Set of limiting points)

From now on we set $\mathcal{U} = \mathbb{R}^n$. The set of limiting points is defined as

$$\omega(u^0) := \left\{ \overline{u} \in \mathbb{R}^n \;\middle|\; \text{there exists an increasing sequence of integers } \{k_j\}_{j\in\mathbb{N}} \text{ such that } \lim_{j\to\infty} u^{k_j} = \overline{u} \right\}.$$

Lemma (Convergence of the surrogate functional)

Let $\overline{u} \in \omega(u^0)$. Then we obtain

$$\lim_{k\to\infty} E^k(u^k) = E(\overline{u}). \tag{8}$$

Convergence of the surrogate functional

Definition (Set of limiting points)

From now on we set $\mathcal{U} = \mathbb{R}^n$. The set of limiting points is defined as

$$\omega(u^0) := \left\{ \overline{u} \in \mathbb{R}^n \;\middle|\; \text{there exists an increasing sequence of integers } \{k_j\}_{j\in\mathbb{N}} \text{ such that } \lim_{j\to\infty} u^{k_j} = \overline{u} \right\}.$$

Lemma (Convergence of the surrogate functional)

Let $\overline{u} \in \omega(u^0)$. Then we obtain

$$\lim_{k\to\infty} E^k(u^k) = E(\overline{u}). \tag{8}$$

A global convergence result for Algorithm (2)

Theorem (Finite length property)

Let $\mathcal{U} = \mathbb{R}^n$ be finite dimensional, and let the sequences $\{u^k\}_{k\in\mathbb{N}}$ and $\{q^k\}_{k\in\mathbb{N}}$ generated by Algorithm (2) be bounded. Then, $\{u^k\}_{k\in\mathbb{N}}$

1. *has finite length, i.e.*

$$\sum_{k=0}^{\infty} \|u^{k+1} - u^k\|_{\mathcal{U}} < \infty \,. \tag{9}$$

2. *converges to a critical point $\hat{u}$ of E, i.e. $\nabla E(\hat{u}) = 0$, with $\hat{p} \in \partial J(\hat{u})$.*

Sketch of proof: Use the Kurdyka-Łojasiewicz property together with the previously proven properties.

A global convergence result for Algorithm (2)

Theorem (Finite length property)

Let $\mathcal{U} = \mathbb{R}^n$ be finite dimensional, and let the sequences $\{u^k\}_{k\in\mathbb{N}}$ and $\{q^k\}_{k\in\mathbb{N}}$ generated by Algorithm (2) be bounded. Then, $\{u^k\}_{k\in\mathbb{N}}$

1. *has finite length, i.e.*

$$\sum_{k=0}^{\infty} \|u^{k+1} - u^k\|_{\mathcal{U}} < \infty \,. \tag{9}$$

2. *converges to a critical point $\hat{u}$ of E, i.e. $\nabla E(\hat{u}) = 0$, with $\hat{p} \in \partial J(\hat{u})$.*

Sketch of proof: Use the Kurdyka-Łojasiewicz property together with the previously proven properties.

A global convergence result for Algorithm (2)

Theorem (Finite length property)

Let $\mathcal{U} = \mathbb{R}^n$ be finite dimensional, and let the sequences $\{u^k\}_{k\in\mathbb{N}}$ and $\{q^k\}_{k\in\mathbb{N}}$ generated by Algorithm (2) be bounded. Then, $\{u^k\}_{k\in\mathbb{N}}$

1. *has finite length, i.e.*

$$\sum_{k=0}^{\infty} \|u^{k+1} - u^k\|_{\mathcal{U}} < \infty \,.. \tag{9}$$

2. *converges to a critical point $\hat{u}$ of E, i.e. $\nabla E(\hat{u}) = 0$, with $\hat{p} \in \partial J(\hat{u})$.*

Sketch of proof: Use the Kurdyka-Łojasiewicz property together with the previously proven properties.

A global convergence result for Algorithm (2)

Theorem (Finite length property)

Let $\mathcal{U} = \mathbb{R}^n$ be finite dimensional, and let the sequences $\{u^k\}_{k\in\mathbb{N}}$ and $\{q^k\}_{k\in\mathbb{N}}$ generated by Algorithm (2) be bounded. Then, $\{u^k\}_{k\in\mathbb{N}}$

1. *has finite length, i.e.*

$$\sum_{k=0}^{\infty} \|u^{k+1} - u^k\|_{\mathcal{U}} < \infty \,. \tag{9}$$

2. *converges to a critical point $\hat{u}$ of E, i.e. $\nabla E(\hat{u}) = 0$, with $\hat{p} \in \partial J(\hat{u})$.*

Sketch of proof: Use the Kurdyka-Łojasiewicz property together with the previously proven properties.

A global convergence result for Algorithm (2)

If $\{q^k\}_{k\in\mathbb{N}}$ is unbounded we can find the following counter example for the previous theorem:

Example

- $E(u) := \frac{1}{2}(u+1)^2$
- $H(u) := \frac{1}{2}u^2$
- $R(u) := \chi_{\geq 0}(x) := \begin{cases} 0 & x \geq 0 \\ \infty & x < 0 \end{cases}$
- $u^0 > 0,\ q^0 = 0$

$$\Rightarrow \quad \lim_{k\to\infty} u^k = 0 \neq -1 = \hat{u} \quad \text{with } E'(\hat{u}) = 0$$

$$q^k \to -\infty$$

A global convergence result for Algorithm (2)

If $\{q^k\}_{k\in\mathbb{N}}$ is unbounded we can find the following counter example for the previous theorem:

Example

- $E(u) := \frac{1}{2}(u+1)^2$
- $H(u) := \frac{1}{2}u^2$
- $R(u) := \chi_{\geq 0}(x) := \begin{cases} 0 & x \geq 0 \\ \infty & x < 0 \end{cases}$
- $u^0 > 0,\ q^0 = 0$

$$\Rightarrow \quad \lim_{k\to\infty} u^k = 0 \neq -1 = \hat{u} \quad \text{with } E'(\hat{u}) = 0$$

$$q^k \to -\infty$$

A global convergence result for Algorithm (2)

If $\{q^k\}_{k\in\mathbb{N}}$ is unbounded we can find the following counter example for the previous theorem:

Example

- $E(u) := \frac{1}{2}(u+1)^2$
- $H(u) := \frac{1}{2}u^2$
- $R(u) := \chi_{\geq 0}(x) := \begin{cases} 0 & x \geq 0 \\ \infty & x < 0 \end{cases}$
- $u^0 > 0,\ q^0 = 0$

$$\Rightarrow \quad \lim_{k\to\infty} u^k = 0 \neq -1 = \hat{u} \quad \text{with } E'(\hat{u}) = 0$$

$$q^k \to -\infty$$

A global convergence result for Algorithm (2)

If $\{q^k\}_{k\in\mathbb{N}}$ is unbounded we can find the following counter example for the previous theorem:

Example

- $E(u) := \frac{1}{2}(u+1)^2$
- $H(u) := \frac{1}{2}u^2$
- $R(u) := \chi_{\geq 0}(x) := \begin{cases} 0 & x \geq 0 \\ \infty & x < 0 \end{cases}$
- $u^0 > 0,\ q^0 = 0$

$$\Rightarrow \quad \lim_{k\to\infty} u^k = 0 \neq -1 = \hat{u} \quad \text{with } E'(\hat{u}) = 0$$

$$q^k \to -\infty$$

A global convergence result for Algorithm (2)

If $\{q^k\}_{k\in\mathbb{N}}$ is unbounded we can find the following counter example for the previous theorem:

Example

- $E(u) := \frac{1}{2}(u+1)^2$
- $H(u) := \frac{1}{2}u^2$
- $R(u) := \chi_{\geq 0}(x) := \begin{cases} 0 & x \geq 0 \\ \infty & x < 0 \end{cases}$
- $u^0 > 0,\ q^0 = 0$

$$\Rightarrow \qquad \lim_{k\to\infty} u^k = 0 \neq -1 = \hat{u} \quad \text{with } E'(\hat{u}) = 0$$

$$q^k \to -\infty$$

Outline

Parallel Magnetic Resonance Imaging (MRI)

In parallel MRI one aims to minimise

$$E(u, b_1, \ldots, b_n) = \frac{1}{2} \sum_{j=1}^{n} \| (S(\mathcal{F}(K(u, b_1, \ldots, b_n))))_j - f_j \|_2^2 .$$

Here

- $\mathcal{F}$ is the (discrete) Fourier transform.
- S is a sub-sampling operator.
- K is the operator $K(u, b_1, \ldots, b_2) = (ub_1, ub_2, \ldots, ub_n)^T$, with
  - the spin-proton density u,
  - the coil sensitivities $b_1, b_2, \ldots, b_n$.
- sub-sampled k-space data $f_1, \ldots, f_n$.

Parallel Magnetic Resonance Imaging (MRI)

In parallel MRI one aims to minimise

$$E(u, b_1, \ldots, b_n) = \frac{1}{2} \sum_{j=1}^{n} \| (S(\mathcal{F}(K(u, b_1, \ldots, b_n))))_j - f_j \|_2^2 .$$

Here

- $\mathcal{F}$ is the (discrete) Fourier transform.
- S is a sub-sampling operator.
- K is the operator $K(u, b_1, \ldots, b_2) = (ub_1, ub_2, \ldots, ub_n)^T$, with
  - the spin-proton density u,
  - the coil sensitivities $b_1, b_2, \ldots, b_n$.
- sub-sampled k-space data $f_1, \ldots, f_n$.

Parallel Magnetic Resonance Imaging (MRI)

In parallel MRI one aims to minimise

$$E(u, b_1, \ldots, b_n) = \frac{1}{2} \sum_{j=1}^{n} \|(S(\mathcal{F}(K(u, b_1, \ldots, b_n))))_j - f_j\|_2^2 .$$

Here

- $\mathcal{F}$ is the (discrete) Fourier transform.
- S is a sub-sampling operator.
- K is the operator $K(u, b_1, \ldots, b_2) = (ub_1, ub_2, \ldots, ub_n)^T$, with
  - the spin-proton density u,
  - the coil sensitivities $b_1, b_2, \ldots, b_n$.
- sub-sampled k-space data $f_1, \ldots, f_n$.

Parallel Magnetic Resonance Imaging (MRI)

In parallel MRI one aims to minimise

$$E(u, b_1, \ldots, b_n) = \frac{1}{2} \sum_{j=1}^{n} \|(S(\mathcal{F}(K(u, b_1, \ldots, b_n))))_j - f_j\|_2^2 .$$

Here

- $\mathcal{F}$ is the (discrete) Fourier transform.
- S is a sub-sampling operator.
- K is the operator $K(u, b_1, \ldots, b_2) = (ub_1, ub_2, \ldots, ub_n)^T$, with
  - the spin-proton density u,
  - the coil sensitivities $b_1, b_2, \ldots, b_n$.
- sub-sampled k-space data $f_1, \ldots, f_n$.

Parallel Magnetic Resonance Imaging (MRI)

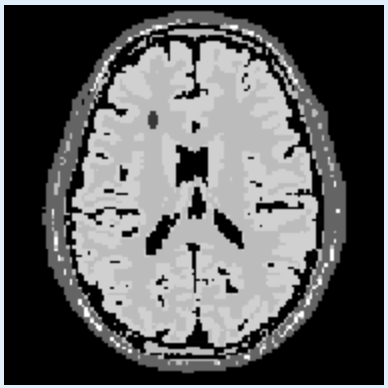
In parallel MRI one aims to minimise

$$E(u, b_1, \ldots, b_n) = \frac{1}{2} \sum_{j=1}^{n} \| (S(\mathcal{F}(K(u, b_1, \ldots, b_n))))_j - f_j \|_2^2 .$$

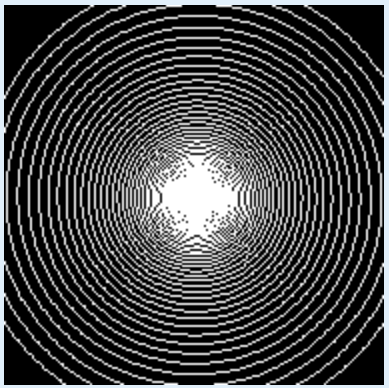
Here

- $\mathcal{F}$ is the (discrete) Fourier transform.
- S is a sub-sampling operator.
- K is the operator $K(u, b_1, \ldots, b_2) = (ub_1, ub_2, \ldots, ub_n)^T$, with
  - the spin-proton density u,
  - the coil sensitivities $b_1, b_2, \ldots, b_n$.
- sub-sampled k-space data $f_1, \ldots, f_n$.

Parallel MRI



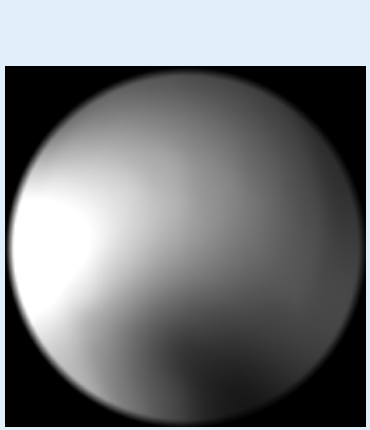
(a) Brain phantom



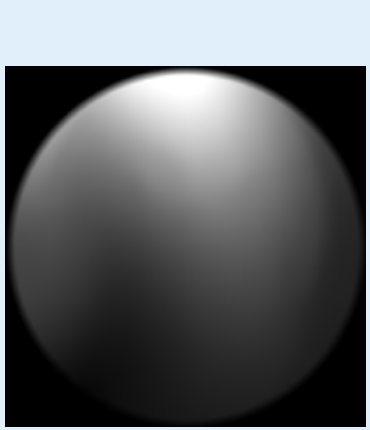
(b) 25% subsampling

Based on Berengere, Evans, Collins, NeuroImage 2006

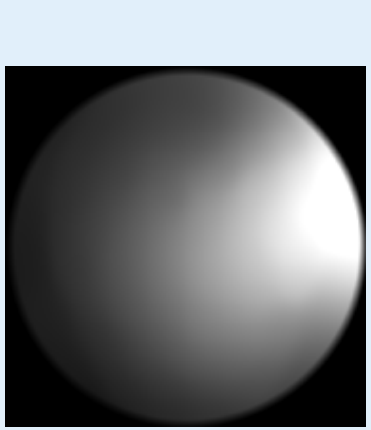
Coil sensitivities © Florian Knoll, NYU



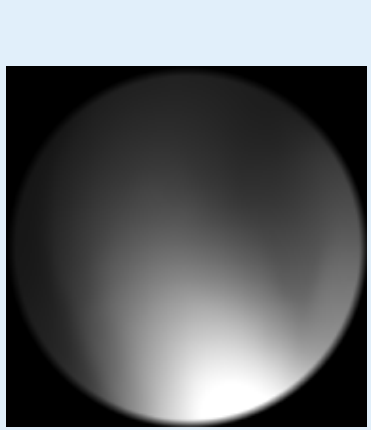
(c) 2nd coil



(d) 4th coil



(e) 6th coil



(f) 8th coil

Parallel MRI

Potential choice for H and R:

$$H(u, b_1, \dots, b_n) = H_0(u) + \sum_{k=1}^{n} H_k(b_k)\,,$$

$$R(u, b_1, \dots, b_n) = R_0(u) + \sum_{k=1}^{n} R_k(b_k)$$

with $$H_k(v) = \frac{1}{2}\|v\|_2^2 \qquad \forall k \in \{0, \dots, n\}\,,$$

and $$R_0(u) = \mathrm{TV}(u)\,,$$

$$R_k(b_k) = \|Cb_k\|_1 \qquad \forall k \in \{1, \dots, n\}\,.$$

Here C denotes the discrete two-dimensional cosine transform.

Parallel MRI

Classical gradient descent

Linearised Bregman iteration

Towards deep learning

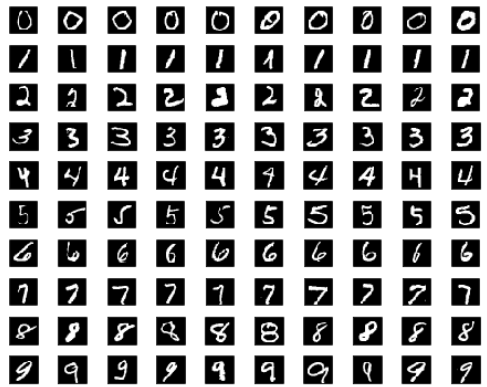
We want to train the two-layer neural network

$$K(U_1, U_2) := \min(\max(U_1 \max(U_2 D, 0), 0), 1)$$

where ρ_1 and ρ_2 are smooth approximations of $\min(\max(\cdot, 0), 1)$ and $\max(\cdot, 0)$, respectively.
Matrices $U_1 \in \mathbb{R}^{10\times 784}$ and $U_2 \in \mathbb{R}^{784\times 784}$ are the unknowns;
$D \in \mathbb{R}^{784\times 50000}$ is data matrix that contains 50000 training images of MNIST (LeCun et al. 2010).
We want to solve the inverse problem

$$K(U_1, U_2) = F\,,$$

where $F \in \{0, 1\}^{10\times 50000}$ is the labelling matrix.



Towards deep learning

We want to train the two-layer neural network

$$K(U_1, U_2) := \rho_1(U_1 \rho_2(U_2 D))$$

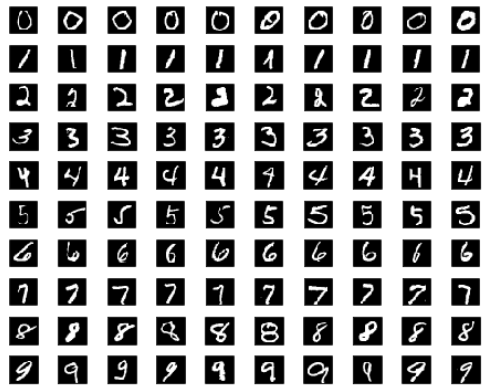
where ρ_1 and ρ_2 are smooth approximations of $\min(\max(\cdot, 0), 1)$ and $\max(\cdot, 0)$, respectively.

Matrices $U_1 \in \mathbb{R}^{10\times 784}$ and $U_2 \in \mathbb{R}^{784\times 784}$ are the unknowns; $D \in \mathbb{R}^{784\times 50000}$ is data matrix that contains 50000 training images of MNIST (LeCun et al. 2010).

We want to solve the inverse problem

$$K(U_1, U_2) = F\,,$$

where $F \in \{0, 1\}^{10\times 50000}$ is the labelling matrix.



Towards deep learning

We want to train the two-layer neural network

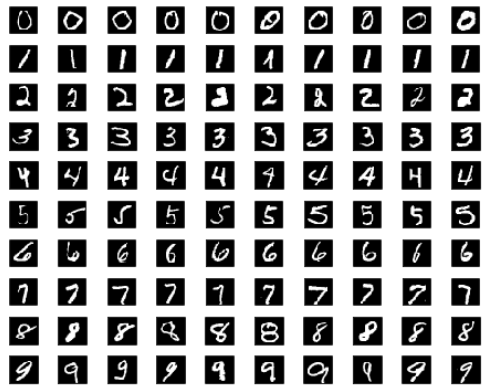
$$K(U_1, U_2) := \rho_1(U_1 \rho_2(U_2 D))$$

where ρ_1 and ρ_2 are smooth approximations of $\min(\max(\cdot, 0), 1)$ and $\max(\cdot, 0)$, respectively.
Matrices $U_1 \in \mathbb{R}^{10\times 784}$ and $U_2 \in \mathbb{R}^{784\times 784}$ are the unknowns;
$D \in \mathbb{R}^{784\times 50000}$ is data matrix that contains 50000 training images of MNIST (LeCun et al. 2010).

We want to solve the inverse problem

$$K(U_1, U_2) = F\,,$$

where $F \in \{0, 1\}^{10\times 50000}$ is the labelling matrix.



Towards deep learning

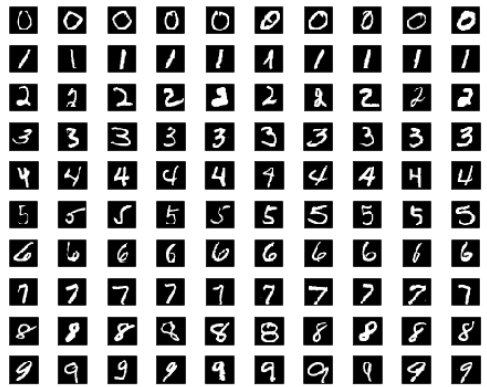
We want to train the two-layer neural network

$$K(U_1, U_2) := \rho_1(U_1 \rho_2(U_2 D))$$

where ρ_1 and ρ_2 are smooth approximations of $\min(\max(\cdot, 0), 1)$ and $\max(\cdot, 0)$, respectively.
Matrices $U_1 \in \mathbb{R}^{10\times 784}$ and $U_2 \in \mathbb{R}^{784\times 784}$ are the unknowns;
$D \in \mathbb{R}^{784\times 50000}$ is data matrix that contains 50000 training images of MNIST (LeCun et al. 2010).
We want to solve the inverse problem

$$K(U_1, U_2) = F\,,$$

where $F \in \{0, 1\}^{10\times 50000}$ is the labelling matrix.

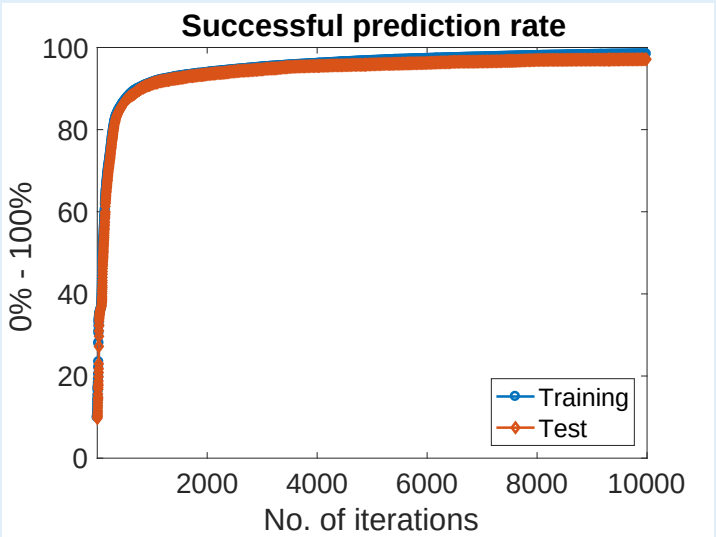


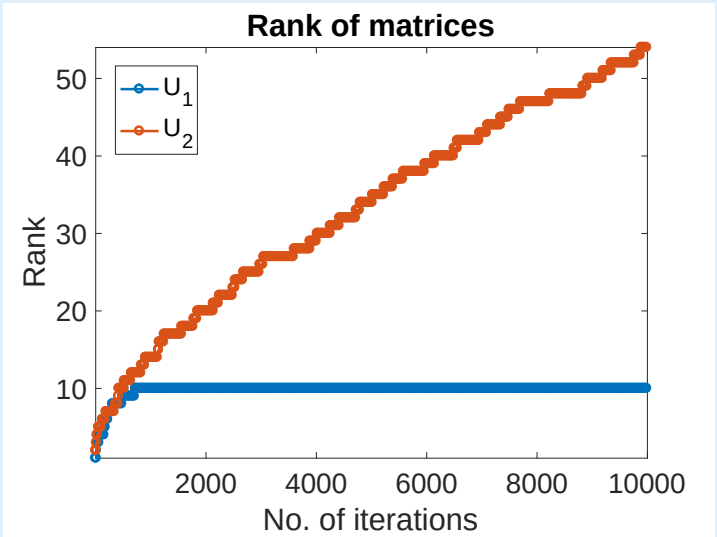
Towards deep learning

We compute U_1 and U_2 via (2) with $H(U_1, U_2) = \sum_{i=1}^{2} \|U_i\|_{\text{Fro}}^2$ and

$$R(U_1, U_2) := \sum_{i=1}^{2} \|U_i\|_* = \sum_{i=1}^{10} \sigma_{1,i} + \sum_{j=1}^{784} \sigma_{2,j} \,.$$

Here $\{\sigma_{1,i}\}_{i\in\{1,\ldots,10\}}$ and $\{\sigma_{2,j}\}_{j\in\{1,\ldots,784\}}$ denote the singular values of U_1 and U_2, respectively.





Outline

Conclusions & outlook

Conclusions

- We have generalised gradient descent to Banach spaces
- We have proven global convergence for this generalisation.
- We shown different solution paths for same initialisation.

Outlook

- Convergence analysis for unbounded $\{q^k\}_{k\in\mathbb{N}}$
- Variable splitting, BFGS-style updates, Newton-type methods
- Incremental or even stochastic versions

Preprint: `http://www.imi.kyushu-u.ac.jp/eng/files/imipublishattachment/file/math_58ec341a238fe.pdf`

Conclusions & outlook

Conclusions

- We have generalised gradient descent to Banach spaces
- We have proven global convergence for this generalisation.
- We shown different solution paths for same initialisation.

Outlook

- Convergence analysis for unbounded $\{q^k\}_{k\in\mathbb{N}}$
- Variable splitting, BFGS-style updates, Newton-type methods
- Incremental or even stochastic versions

Preprint: `http://www.imi.kyushu-u.ac.jp/eng/files/imipublishattachment/file/math_58ec341a238fe.pdf`

Conclusions & outlook

Conclusions

- We have generalised gradient descent to Banach spaces
- We have proven global convergence for this generalisation.
- We shown different solution paths for same initialisation.

Outlook

- Convergence analysis for unbounded $\{q^k\}_{k\in\mathbb{N}}$
- Variable splitting, BFGS-style updates, Newton-type methods
- Incremental or even stochastic versions

Preprint: `http://www.imi.kyushu-u.ac.jp/eng/files/imipublishattachment/file/math_58ec341a238fe.pdf`

Conclusions & outlook

Conclusions

- We have generalised gradient descent to Banach spaces
- We have proven global convergence for this generalisation.
- We shown different solution paths for same initialisation.

Outlook

- Convergence analysis for unbounded $\{q^k\}_{k\in\mathbb{N}}$
- Variable splitting, BFGS-style updates, Newton-type methods
- Incremental or even stochastic versions

Preprint: `http://www.imi.kyushu-u.ac.jp/eng/files/imipublishattachment/file/math_58ec341a238fe.pdf`

Conclusions & outlook

Conclusions

- We have generalised gradient descent to Banach spaces
- We have proven global convergence for this generalisation.
- We shown different solution paths for same initialisation.

Outlook

- Convergence analysis for unbounded $\{q^k\}_{k\in\mathbb{N}}$
- Variable splitting, BFGS-style updates, Newton-type methods
- Incremental or even stochastic versions

Preprint: `http://www.imi.kyushu-u.ac.jp/eng/files/imipublishattachment/file/math_58ec341a238fe.pdf`

Conclusions & outlook

Conclusions

- We have generalised gradient descent to Banach spaces
- We have proven global convergence for this generalisation.
- We shown different solution paths for same initialisation.

Outlook

- Convergence analysis for unbounded $\{q^k\}_{k\in\mathbb{N}}$
- Variable splitting, BFGS-style updates, Newton-type methods
- Incremental or even stochastic versions

Preprint: `http://www.imi.kyushu-u.ac.jp/eng/files/imipublishattachment/file/math_58ec341a238fe.pdf`

Thank you for your attention

Acknowledgements: LEVERHULME TRUST

Isaac Newton Trust

The Kurdyka-Łojasiewicz property

Definition (Kurdyka-Łojasiewicz (KL) property)

We assume for $\eta > 0$ that $\varphi : [0, \eta[\to \mathbb{R}_{\geq 0}$ is a concave function that is continuous at zero and satisfies $\varphi(0) = 0$, $\varphi \in C^1(]0, \eta[)$.

1. E fulfils the KL property at a point $\overline{u} \in \mathbb{R}^n$ if there exists $\eta \in]0, \infty[$, a neighbourhood U of $\overline{u}$ such that for all

$$u \in U \cap \{u \mid E(\overline{u}) < E(u) < E(\overline{u}) + \eta\}$$

we observe

$$\varphi'(E(u) - E(\overline{u}))\|\nabla E(u)\|_2 \geq 1\,. \tag{10}$$

2. If E satisfies the KL property for all arguments in $\mathbb{R}^n$, E is called a KL functional.

The Kurdyka-Łojasiewicz property

Definition (Kurdyka-Łojasiewicz (KL) property)

We assume for $\eta > 0$ that $\varphi : [0, \eta[\to \mathbb{R}_{\geq 0}$ is a concave function that is continuous at zero and satisfies $\varphi(0) = 0$, $\varphi \in C^1(]0, \eta[)$.

1. E fulfils the KL property at a point $\overline{u} \in \mathbb{R}^n$ if there exists $\eta \in]0, \infty[$, a neighbourhood U of $\overline{u}$ such that for all

$$u \in U \cap \{u \mid E(\overline{u}) < E(u) < E(\overline{u}) + \eta\}$$

we observe

$$\varphi'(E(u) - E(\overline{u}))\|\nabla E(u)\|_2 \geq 1\,. \tag{10}$$

2. If E satisfies the KL property for all arguments in $\mathbb{R}^n$, E is called a KL functional.

The Kurdyka-Łojasiewicz property

Example - Łojasiewicz 1965

Choose $\varphi(x) = \frac{1}{1-\theta}|x|^{1-\theta}$, for $\theta \in]0,1[$. Then (10) reads as

$$|E(u) - E(\overline{u})|^{-\theta} \|\nabla E(u)\|_2 \geq 1\,,$$

respectively

$$|E(u) - E(\overline{u})|^{\theta} \leq \|\nabla E(u)\|_2\,.$$

Thus, the norm of the gradient is a bound for the functional values.