Optimierung auf Mannigfaltigkeiten und Open-Source Software

Ronny Bergmann

Department of Mathematical Sciences (IMF),
Norwegian University of Science and Technology (NTNU), Trondheim, Norway.

Workshop
Mathematics and Industry

Berlin, 3. November 2023.

Norges teknisk-naturvitenskapelige universitet

Ronny Bergmann

- Studium & Promotion (2013, J. Prestin), Universität zu Lübeck.
- Habilitation (2018, G. Steidl), TU Kaiserslautern.
- Privatdozent (AG R. Herzog), TU Chemnitz.
- seit März 2021: Førsteamanuensis (Assoc. Prof./W2),
  IMF, NTNU, Trondheim, Norwegen.

Am Trondheimfjord.

Der Rayleigh-Quotient

Berechnung des kleinsten Eigenwertes λ einer sym. Matrix $A \in \mathbb{R}^{d\times d}$:

Minimiere den Rayleigh-Quotient

$$f(x) = \frac{x^\mathsf{T} A x}{\|x\|^2}, \qquad \text{für } x \neq 0$$

- Für jeden Eigenvektor v zum EW λ ist $f(v) = f(\alpha v)$, $\alpha > 0$, minimal
- keine isolierten Minima, das Newtonverfahren bspw. divergiert
- Eliminiere das Problem durch Betrachten des Rayleigh-Quotienten auf der Sphäre

$$f(p) = p^\mathsf{T} A p, \quad p \in \mathbb{S}^{d-1} \quad \text{mit} \quad \mathbb{S}^{d-1} := \{p \in \mathbb{R}^d \mid \|p\| = 1\}$$

- Mit Berücksichtigung der Geometrie:

unrestringierte Optimierungsverfahren

Eine Riemannsche Mannigfaltigkeit

Informell kann eine Mannigfaltigkeit definiert werden als eine Menge $\mathcal{M}$, zusammen mit einer „geeigneten" Menge an Karten, die Teilmengen von $\mathcal{M}$ mit offenen Teilmengen des $\mathbb{R}^d$ identifizieren sowie einem (in p) stetigen Skalarprodukt $(\cdot\,,\,\cdot)_p$ auf den Tangentialräumen $T_p\mathcal{M}$.

[Absil, Mahony, and Sepulchre 2008; Boumal 2023]

Herausforderungen.

- Tangentialvektoren in unterschiedlichen Räumen $X \in T_p\mathcal{M}$, $Y \in T_q\mathcal{M}$
- Verallgemeinern $\ell(t) = p + tX$: Exponentialabbildung $\exp_p X$
- Umkehrfunktion: Logarithmusabbildung $\log_p q$ nur lokal definiert
- … deren numerisch effiziente Realisierung

Optimierung auf Mannigfaltigkeiten

Aufgabe. Entwerfe einen Algorithmus zur numerischen Lösung von

$$\operatorname*{arg\,min}_{p\in\mathcal{M}} f(p)$$

mit

- einer hochdimensionalen Riemannschen Mannigfaltigkeit $\mathcal{M}$
- einer nicht-glatten (nicht-konvexen) Funktion $f\colon \mathcal{M} \to \mathbb{R}$

Beispiele.

- Rayleigh: $f_1(p) = p^{\mathrm{T}}Ap, \quad p \in \mathbb{S}^{d-1}$
- Verallgemeinerter Rayleigh: $f_2(X) = X^{\mathrm{T}}AX, \quad X \in \mathrm{St}(d,k)$ mit der Stiefel-Mannigfaltigkeit $\mathrm{St}(d,k) \coloneqq \{X \in \mathbb{R}^{n,k} \mid X^{\mathrm{T}}X = I\}$
- f_2 auf der Menge der k-dimensionalen Unterräume, der Grassmann-Mannigfaltigkeit $\mathrm{Gr}(d,k) \coloneqq \{\mathrm{span}(X) \mid X \in \mathrm{St}(d,k)\}$

Open-Source Software

Für einen neu entwickelten/untersuchtes numerisches Verfahren sollten

- die Implementierung des Algorithmus
- die numerischen Experimente

frei verfügbar bzw. **reproduzierbar** veröffentlicht sein.

Beispiel.
Für den Difference of Convex Algorithm (DCA) ist

[RB, Ferreira, Santos, and Souza 2023]

- der Algorithmus in Manopt.jl[1] implementiert und [RB 2022]
- jedes Experiment in ManoptExamples.jl[2] als Quarto-Notebook, das stets mit den gleichen Versionen der Julia-Pakete läuft.

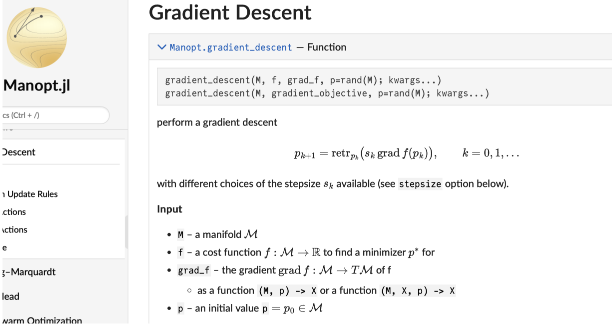
[1]siehe manoptjl.org, genauer manoptjl.org/stable/solvers/difference_of_convex/
[2]siehe juliamanifolds.github.io/ManoptExamples.jl/stable/

Dokumentation, Tests, Zitierbarkeit

Zur Verwendung des Codes sind außerdem wichtig

Dokumentation.

- der Eingabevariablen
- der optionalen Parameter
- Kurzbeschreibung
- der Rückgabe
- Science Communication



Manopt.jl

cs (Ctrl + /)

Descent

ı Update Rules

ctions

ıctions

e

ȝ-Marquardt

lead

warm Optimization

Gradient Descent

`Manopt.gradient_descent` — Function

```
gradient_descent(M, f, grad_f, p=rand(M); kwargs...)
gradient_descent(M, gradient_objective, p=rand(M); kwargs...)
```

perform a gradient descent

$$p_{k+1} = \operatorname{retr}_{p_k}\bigl(s_k \operatorname{grad} f(p_k)\bigr), \qquad k = 0, 1, \ldots$$

with different choices of the stepsize s_k available (see `stepsize` option below).

Input

- `M` – a manifold $\mathcal{M}$
- `f` – a cost function $f : \mathcal{M} \to \mathbb{R}$ to find a minimizer p^* for
- `grad_f` – the gradient $\operatorname{grad} f : \mathcal{M} \to T\mathcal{M}$ of f
  - as a function `(M, p) -> X` or a function `(M, X, p) -> X`
- `p` – an initial value `p` $= p_0 \in \mathcal{M}$

Tests. Automatisierte Tests, die sicherstellen, dass

- Jede Code-Zeile „wie erwartet" funktioniert (Code Coverage)
- Dies bei neuen Versionen weiterhin so ist (Continuous Integration, CI)

Zitierbarkeit. Jede Version ist exakt zitierbar (per DOI) dank Zenodo (CI)

Beispiel

Manopt.gradient_descent — Function

```
gradient_descent(M, f, grad_f, p=rand(M); kwargs...)
gradient_descent(M, gradient_objective, p=rand(M); kwargs...)
```

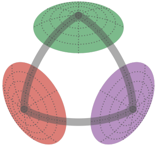
perform a gradient descent

```
using LinearAlgebra, Manifolds, Manopt, Random
# Setup: Erzeuge A, M und ein Startpunkt p0.
Random.seed!(42);    d = 500;    A = Symmetric(randn(d,d))
M = Sphere(d-1);    p0 = rand(M)

# grad_f: Tangentialrichtung d. steilsten Anstiegs.
f(M, p) = p'*A*p;    grad_f(M,p) = 2*A*p - 2*(p'*A*p)*p
q1 = gradient_descent(M, f, grad_f, p0)

# oder: Nutze (Euklidischen) Gradienten und teile dies Manopt mit.
∇f(Rn, p) = 2*A*p
q2 = gradient_descent(M, f, ∇f, p0; objective_type=:Euclidean)
```

Aktuelle Projekte



Algorithmen.

- (exact & linearized) Riemannian Chambolle-Pock Algorithm [RB, Herzog, Silva Louzeiro, Tenbrinck, and Vidal-Núñez 2021]
- Difference of Convex Algorithm [RB, Ferreira, Santos, and Souza 2023]

Open Source Software.

- ManifoldsBase.jl – ein Interface, um Mannigfaltigkeiten in Julia einheitlich zu definieren und generisch verwenden zu können.
- Manifolds.jl – eine Bibliothek von Mannigfaltigkeiten. Fokus: [Axen, Baran, RB, and Rzecki 2023]
  - Effizienz
  - Lesbarkeit des Codes / bei Verwenden des Codes
  - Dokumentation – inklusive Formeln & Literatur
- Manopt.jl – ein Interface für und Implementierung von Algorithmen zur Optimierung auf Mannigfaltigkeiten [RB 2022]

NTNU

Ausgewählte Referenzen

Axen, S. D., M. Baran, RB, and K. Rzecki (2023). "Manifolds.jl: An Extensible Julia Framework for Data Analysis on Manifolds". In: *ACM Transactions on Mathematical Software*. Accepted for pulication. DOI: 10.1145/3618296. arXiv: 2106.08777.

RB (2022). "Manopt.jl: Optimization on Manifolds in Julia". In: *Journal of Open Source Software* 7.70, p. 3866. DOI: 10.21105/joss.03866.

Boumal, N. (2023). *An introduction to optimization on smooth manifolds*. Cambridge University Press. URL: https://www.nicolasboumal.net/book.

ronnybergmann.net/talks/2023-Berlin-Manopt.pdf

Gløshaugen & Trondheimfjord im Herbst.