



**Dipartimento di Matematica e Informatica
Corso di Laurea Triennale in Informatica**

TESI DI LAUREA

**Metodi di ottimizzazione su
varietà differenziabili per la media
di Fréchet su disco di Poincaré**

**Relatore:
Prof. Bruno Iannazzo**

**Candidato:
Luca Moroni**

**Matricola:
311279**

Anno Accademico 2020/2021

Alla mia famiglia ed alla loro estrema pazienza.

Contenuti

1	Introduzione	3
2	Metodi di ottimizzazione su R^n e Varieta differenziabili	4
2.1	Condizioni necessarie per l'ottimalità	4
2.2	Il caso della convessità	5
2.3	Condizioni Sufficienti per L'ottimalità	6
2.4	Discesa del gradiente in R^n	6
2.4.1	Selezionare la direzione di discesa	7
2.4.2	Selezione del passo	8
2.5	Discesa del gradiente su varietà	10
2.5.1	Retrazioni	10
2.5.2	Metodi <i>line-search</i>	11
2.5.3	Convergenza	12
3	Disco di poincare e Iperboloide	13
3.1	Iperboloide	13
3.1.1	Metrica	14
3.1.2	Distanza	14
3.1.3	Mappa Esponenziale	14
3.1.4	Gradiente della funzione distanza	14
3.2	Disco di Poincaré	14
3.2.1	Metrica	15
3.2.2	Distanza	15
3.2.3	Mappa Esponenziale	15
3.2.4	Gradiente della funzione distanza	16
3.3	Iperboloide come modello conforme	16
3.4	Media di Fréchet e problema del centroide	18
4	Metodi utilizzati con pseudocodice ed esperimenti	19
4.1	Algoritmi e Pseudocodice	19
4.1.1	Passo fisso	19
4.1.2	Armijo	20
4.1.3	Barzilai-Borwein	20
4.2	Esperimenti	22
5	Appendice	30
5.1	Varietà	30
5.1.1	Vettori Tangenti	30
5.1.2	Metriche distanze e e gradienti Riemanniani	31
5.2	Topologia	32
5.3	Definizioni Utili	32
5.3.1	Teorema di Taylor	33

1 Introduzione

Le applicazioni pratiche nelle quali è necessario effettuare ottimizzazione di funzioni non lineari, ovvero trovarne un valore di minimo, nei giorni d'oggi sono molteplici. Praticamente i metodi di ottimizzazione non lineare sono alla base del funzionamento di tecnologie utilizzatissime, tra cui le reti neurali, il cui apprendimento pone le fondamenta sulla solida teoria matematica dei metodi numerici per l'ottimizzazione. Nella sua espressione più generale, ovvero quando la funzione da ottimizzare non ha come dominio un sottoinsieme dello spazio euclideo ma una varietà differenziabile, le applicazioni possibili in numero sono nettamente superiori, possiamo pensare ad esempio al calcolo di autospazi di una matrice, che rappresenta un problema molto comune e necessario da risolvere nelle applicazioni dell'algebra lineare, inoltre, possiamo pensare al problema della decomposizione SVD il quale può essere modellizzato come un problema di ottimizzazione su varietà, un altro problema è l'analisi delle componenti indipendenti, Independent Component Analysis (ICA) conosciuto come separazione delle sorgenti e modellizzabile anch'esso come un problema di ottimizzazione su varietà, ha ricevuto un forte interesse nella ricerca di questi ultimi anni dato dall'ampia applicazione nell'ambito biomedico. Infine, l'applicazione su cui si basa la tesi, il calcolo della centroide di p -punti definiti in una varietà riemanniana.

Nella presente trattazione verrà, perciò, presentato il problema della ricerca del minimo della funzione **somma delle distanze al quadrato**, quindi del calcolo del centroide di p -punti, sulla varietà riemanniana **disco di Poincaré**, verrà confrontata la risoluzione di tale problema con il corrispettivo problema trasferito nella varietà conforme **iperboloide**, tale sperimentazione prende spunto dagli esperimenti effettuati in [1], nel quale si confrontano tali varietà per il medesimo problema, nella presente però utilizzeremo vari algoritmi, non solo quello a passo fisso, ed inoltre gli algoritmi di ottimizzazione sono stati implementati utilizzando la mappa esponenziale per entrambe le varietà. Capiremo dunque se, tali modelli conformi sono equivalenti, non solo da un punto di vista geometrico ma anche sotto un aspetto computazionale tale per cui effettuare ottimizzazione. Inoltre tale trattazione vuole far comprendere la possibilità, in problemi di ottimizzazione su varietà, di intercambiare modelli conformi. Gli esperimenti trattati vorrebbero, infine, dare l'incipit per continuare con altre sperimentazioni analoghe andando, magari, a trattare anche il modello del **semipiano di Poincaré**, del quale non siamo riusciti a trovare materiale sufficiente per poter implementare gli algoritmi di ottimizzazione considerati.

2 Metodi di ottimizzazione su R^n e Varietà differenziabili

Nella presente trattazione i vettori verranno indicati tramite le ultime lettere minuscole dell'alfabeto x, v, w , che assumeremo essere vettori colonna, il trasposto di un vettore verrà identificato tramite il carattere apice ($'$) perciò per definire il prodotto scalare euclideo di due vettori u, v scriveremo $u'v$, mentre se si intende un prodotto scalare differente da quello euclideo, come è il prodotto scalare riemanniano, lo indicheremo con $\langle \cdot, \cdot \rangle$.

In tale sezione andremo ad esplicitare le metodologie generali per trovare un minimo di una funzione f definita su una varietà differenziabile M a valori reali (notiamo che R^n è un caso particolare di varietà differenziabile). Cercheremo di trovare quei punti che garantiscono l'ottimalità di una funzione, ovvero punti di massimo e di minimo, che descriveremo a breve.

Sia $f : R^n \rightarrow R$ non vincolata. Un vettore x^* è un minimo locale non vincolato per la funzione f se, informalmente, f assume un valore in x^* non più grande dei valori che assume in un intorno di x^* stesso. Più formalmente x^* è minimo locale per f se esiste $\epsilon > 0$ tale che

$$f(x^*) \leq f, \text{ per ogni } x \text{ con } \|x - x^*\| < \epsilon$$

$\|x\|$ rappresenta la norma euclidea del vettore x .

Un vettore x^* è un minimo globale non vincolato rispetto ad f se la funzione assume un valore in x^* non più grande dei valori che assume in ogni altro vettore nel proprio dominio, più formalmente x^* è tale che

$$f(x^*) \leq f, \text{ per ogni } x \in R^n$$

2.1 Condizioni necessarie per l'ottimalità

Se la funzione di costo f è differenziabile, possiamo utilizzare il gradiente e l'espansione in serie di Taylor per comparare il costo di un vettore con il costo di un suo intorno. Ci aspettiamo perciò che se x^* è un minimo non vincolato allora la variazione della funzione al primo ordine per una piccola variazione Δx è non negativa

$$\nabla f(x^*)' \Delta x \geq 0.$$

Perciò valendo sia per Δx positivi che negativi abbiamo la seguente condizione

necessaria,

$$\nabla f(x^*)' = 0.$$

Ci aspettiamo inoltre che anche la variazione della funzione al secondo ordine per una piccola variazione di Δx è non negativa,

$$\nabla f(x^*)' \Delta x + (1/2) \Delta x' \nabla^2 f(x^*) \Delta x \geq 0$$

dato che la precedente osservazione ha imposto $\nabla f(x^*)' \Delta x = 0$ otteniamo che

$$\Delta x' \nabla^2 f(x^*) \Delta x \geq 0$$

e che perciò

$$\nabla^2 f(x^*) \text{ è una matrice semidefinita positiva.}$$

Diamo il seguente risultato che sintetizza quanto detto fin'ora.

Proposizione 1 (Condizioni necessarie di ottimalità) *Sia x^* un minimo locale non vincolato di $f : R^n \rightarrow R$, assumiamo f differenziabile con continuità in un insieme aperto S contenente x^* . Allora*

$$\nabla f(x^*) = 0$$

Se f è due volte differenziabile con continuità in S , allora

$$\nabla^2 f(x^*) \text{ è una matrice semidefinita positiva.}$$

2.2 Il caso della convessità

Se la funzione f è convessa su un insieme convesso non ci sono distinzioni tra un minimo locale ed un minimo globale, tutti i punti di minimo locale sono anche punti di minimo globale. Perciò un fatto importante è che la condizione $\nabla f(x^*) = 0$ è sufficiente per l'ottimalità, da cui deriva il seguente risultato.

Proposizione 2 (Funzioni Convesse) *Sia $f : R^n \rightarrow R$ una funzione convessa su un insieme convesso X .*

- *Un minimo locale di f su X è anche un minimo globale su X . Se inoltre f è strettamente convessa, allora esiste al più un minimo per f .*

- Se f è convessa e l'insieme X è aperto, allora $\nabla f(x^*) = 0$ è una condizione necessaria e sufficiente per il vettore $x^* \in X$ per essere minimo globale di f su X .

2.3 Condizioni Sufficienti per L'ottimalità

Non è difficile trovare esempi in cui le condizioni di ottimalità necessarie definite nella sezione precedente [$\nabla f(x^*) = 0$ e $\nabla^2 f(x^*) \geq 0$] portino ad identificare punti che non sono di minimo locale come ad esempio punti di sella o punti di massimo.

Supponiamo di avere un vettore x^* che soddisfa le seguenti condizioni

- $\nabla f(x^*) = 0$.
- $\nabla^2 f(x^*)$ è una matrice definita positiva.

Allora abbiamo che per ogni $\Delta x \neq 0$ con $\Delta x \in R^n \setminus 0$

$$\Delta x' \nabla^2 f(x^*) \Delta x > 0$$

Ciò implica che in x^* la variazione di f al secondo ordine data da un piccolo spostamento Δx è positiva, perciò f tende ad incrementare nell'intorno di x^* e ciò implica che le due condizioni di cui sopra sono necessarie e sufficienti per l'ottimalità locale di x^* .

Proposizione 3 (Condizioni di ottimalità sufficienti del secondo ordine) Sia $f : S \subset R^n \rightarrow R$ derivabile due volte con continuità. Si supponga che esista $x^* \in S$ che soddisfi le condizioni

$$\nabla f(x^*) = 0 \quad e \quad \nabla^2 f(x^*) \text{ è una matrice definita positiva.}$$

Allora, x^* è un minimo locale di f . In particolare esistono $\gamma > 0$ e $\epsilon > 0$ tali che

$$f(x) \geq f(x^*) + \frac{\gamma}{2} \|x - x^*\|^2, \quad \text{per ogni } x \quad \text{con } \|x - x^*\| < \epsilon$$

2.4 Discesa del gradiente in R^n

Andiamo ora a definire nel dettaglio le principali metodologie computazionali di ottimizzazione non lineare su R^n . Tali metodologie saranno trattate con un occhio alle possibili applicazioni.

Consideriamo il problema di trovare il minimo non vincolato di una funzione $f : R^n \rightarrow R$. Nella maggior parte di casi non è possibile trovare una soluzione analitica, perciò l'approccio adottato è quello di applicare un algoritmo iterativo chiamato *iterative descent* che opera come segue: prendiamo un vettore iniziale x^0

e successivamente generiamo una sequenza x^1, x^2, \dots tale che f decresce ad ogni iterazione $f(x^{k+1}) < f(x^k)$ e sperabilmente converga un punto di minimo.

I principali algoritmi di discesa sono metodi del gradiente poichè sono basati sul gradiente della funzione f .

2.4.1 Selezionare la direzione di discesa

Sia $v \in R^n \setminus 0$, v^\perp è un iperpiano, contenente i punti tali che $v'd = 0$, divide lo spazio in due componenti connesse:

- $v'd > 0$ (semipiano).
- $v'd < 0$ (semipiano).

Se $v = \nabla f(x)$ ogni d tale che $v'd < 0$ è una direzione di decrescita, come mostrato nel seguente teorema.

Teorema 1 Sia $f \in C^1(\Omega)$, con Ω aperto di R^n e sia $x \in \Omega$ e $\nabla f(x) \neq 0$ per ogni $d \in R^n$ tale che $\nabla f(x)'d < 0$ ed esiste $\alpha^0 > 0$ tale che $f(x + \alpha d) < f(x)$ con $\alpha \in (0, \alpha^0]$

Detto ciò possiamo dare una classe di algoritmi con $x^0 \in \Omega$ e dove $d_k \in R^n$ è detta direzione ed α_k è detto passo, definiti secondo la seguente formula,

$$x^{k+1} = x^k + \alpha^k d^k \quad k = 0, 1, \dots,$$

Dove se $\nabla f(x^k) \neq 0$ la direzione d^k è scelta in modo tale che

$$\nabla f(x^k)'d^k < 0$$

Ci sono varie possibilità nella scelta di d^k e di α^k .

Molti metodi di gradiente sono definiti nella forma,

$$x^{k+1} = x^k - \alpha^k D^k \nabla f(x^k),$$

dove D^k è una matrice definita positiva e con $d^k = -D^k \nabla f(x^k)$ allora è immediato verificare che $\nabla f(x^k)'d^k < 0$.

A seconda della scelta di D^k abbiamo differenti metodi applicabili.

- **Steepest Descent:** $D^k = I$, $k = 0, 1, \dots$
- **Metodi di Newton:** $D^k = (\nabla^2 f(x^k))^{-1}$, $k = 0, 1, \dots$, $\nabla^2 f(x^k)$ deve essere definita positiva, proprietà sempre verificata se f è convessa.
- **Steepest Descent Riscalata:** D^k matrice diagonale.

- **Metodi di Schamansky:** $D^k = (\nabla^2 f(x^0))^{-1}$.
- **Metodi di quasi Newton:** $D^k = H(x^k) \approx \nabla^2 f(x^k)$.

Nonostante i metodi di Newton convergono molto velocemente, non sono utilizzati nella pratica dal momento in cui il calcolo della matrice hessiana è computazionalmente oneroso, perciò sono stati introdotti i metodi di quasi Newton, nel qual caso la matrice D^k è scelta in modo tale che la direzione $d^k = -D^k \nabla f(x^k)$ tenda ad approssimare la direzione del metodo di Newton, e D_k approssima la matrice hessiana. L'idea fondamentale riguardo questa metodologia è che tramite due iterate successive $k, k+1$ ed i valori $x^k, x^{k+1}, \nabla f(x^k), \nabla f(x^{k+1})$ noi possiamo avere accesso a delle informazioni riguardo la matrice hessiana, in particolare abbiamo che

$$\nabla f(x^{k+1}) - \nabla f(x^k) \approx H(x^{k+1})(x^{k+1} - x^k).$$

Definiti $p^k = x^{k+1} - x^k$ e $q^k = \nabla f(x^{k+1}) - \nabla f(x^k)$ allora D^{k+1} deve soddisfare l'uguaglianza $D^{k+1}p^k = q^k$, questo prende il nome di equazione delle secanti.

2.4.2 Selezione del passo

Ci sono numerose metodologie per la scelta del passo α^k nei metodi del gradiente, ne elenchiamo alcune.

Ricerca lineare esatta:

scegliamo α^k tale che minimizza la funzione costo f lungo la direzione d^k , perciò,

$$f(x^k + \alpha^k d^k) = \min_{\alpha \geq 0} f(x^k + \alpha d^k).$$

Ricerca lineare esatta limitata:

Questa è una versione modifica della metodologia precedente più semplice da implementare in vari casi. definito un scalare $s > 0$ α^k è scelto in modo tale che minimizza il costo di f nell'intervallo $[0, s]$,

$$f(x^k + \alpha^k d^k) = \min_{\alpha \in [0, s]} f(x^k + \alpha d^k).$$

Le due metodologie appena presentate sono solitamente implementate con l'aiuto delle tecniche risolutive dell'ottimizzazione in una variabile, che è un problema certamente più facile da risolvere in modo analitico.

Metodo di Armijo:

Scegliere α^k in modo che garantisca una decrescita sufficiente a garantire la convergenza del metodo sotto opportune ipotesi. Il metodo di armijo è stato definito

come segue. Siano fissati λ, σ e γ con $0 < \sigma < 1$, e $0 < \gamma < 1$ e sia $\alpha^k = \sigma^{m_k} \lambda$, dove m_k è il primo intero non negativo per cui vale

$$f(x^k) - f(x^k + \sigma^m \lambda d^k) \geq -\gamma \sigma^m \lambda \nabla f(x^k)' d^k.$$

Solitamente si scelgono i valori come segue, $\gamma \in [10^{-5}, 10^{-1}]$, il fattore di riduzione $\sigma \in [1/10, 1/2]$, possiamo scegliere $s = 1$ e moltiplicare la direzione d^k per uno scalare.

Per questa metodologia diamo anche il seguente teorema che rappresenta un risultato importante.

Definizione 1 Una successione di direzioni $\{d^k\}_k$ è detta *limitata* se vale la seguente,

$$\|d^k\| < M \text{ per ogni } k \text{ con } M \in \mathbb{R}^+$$

Definizione 2 Una successione di direzioni $\{x^k\}_k$, $\{d^k\}_k$ è detta **gradient related** se per ogni sotto-successione di $\{x^k\}_k$ che converge ad un punto non stazionario $\{d^k\}_k$ è limitata ed inoltre vale,

$$\limsup_{k \in K} \nabla f(x)' d^k < 0.$$

Ora diamo il principale risultato che garantisce la convergenza globale per metodi di discesa che utilizzano la tecnica di Armijo.

Teorema 2 Sia $\{x^k\}_k$ una sequenza infinita (non definitivamente uguale al suo limite) generata dal metodo $x^{k+1} = x^k + \alpha^k d^k$, dove α^k è generato con la regola di armijo e d^k è **gradient related**. Allora ogni punto limite è stazionario.

Metodo a passo costante:

Si definisce un passo costante $s > 0$ e si fissa $\alpha^k = s$, $k = 0, 1, \dots$

Il passo costante è una metodologia veramente semplice da implementare ma si deve fare attenzione nella scelta del passo, se si sceglie un valore di s troppo grande allora il metodo divergerà, contrariamente se il passo s venisse scelto troppo piccolo l'ordine di convergenza risulterebbe troppo lento.

Metodo di diminuzione del passo:

Scegliamo il passo in modo tale che $\alpha^k \rightarrow 0$. Una problematica legata a questa metodologia è che il passo può diventare troppo piccolo per cui non può essere garantita la convergenza, per questa ragione viene richiesto che.

$$\sum_{n=1}^{\infty} \alpha^k = \infty.$$

2.5 Discesa del gradiente su varietà

Ora che abbiamo un contesto teorico e una consapevolezza di cosa voglia dire effettuare ottimizzazione su uno spazio euclideo n -dimensionale (R^n), possiamo approcciare le metodologie computazionali per trovare un minimo non vincolato di una funzione $f : M \rightarrow R$ analoghe a quella definite nella sezione precedente. Prima di andare nel vivo del discorso dobbiamo però munirci degli strumenti necessari, richiamiamo la parte dedicata alle varietà differenziabili presente nell'appendice e aggiungiamo quanto segue.

2.5.1 Retrazioni

Su varietà differenziabili la nozione di muoversi nella direzione del vettore tangente rimanendo nella varietà stessa è generalizzato dal concetto di **Retrazione**, concettualmente una retrazione R su x , denominata come R_x , è una mappa che va da $T_x M$ (spazio tangente in x rispetto a M) in M con una condizione di rigidità locale che preserva il gradiente in x .

Definizione 3 (Retrazione) Una retrazione in una varietà M è una mappa liscia R che ha come dominio l'insieme dei $T_x M \quad \forall x \in M$ in M con le seguenti proprietà. Sia R_x la restrizione di R a $T_x M$.

- $R_x(0_x) = x$, dove 0_x denota l'elemento zero (origine) di $T_x M$.
- Con l'identificazione canonica $T_{0_x} T_x M \simeq T_x M$, R_x soddisfa $DR_x(0_x) = id_{T_x M}$,
dove $id_{T_x M}$ denota la mappa identità su $T_x M$.

Ci riferiamo alla condizione $DR_x(0_x) = id_{T_x M}$ come condizione di *rigidità locale*. Equivalentemente, per ogni vettore tangente ξ in $T_x M$, la curva $\gamma_\xi : t \mapsto R_x(t\xi)$ soddisfa $\gamma_\xi(0)' = \xi$.

Oltre a trasformare elementi di $T_x M$ in elementi di M , un secondo importante scopo della retrazione R_x è quello di trasformare una funzione costo ($f : M \rightarrow R$) definita in un intorno di $x \in M$ in una funzione costo definita sullo spazio vettoriale $T_x M$. Nello specifico, data una funzione reale f su una varietà M su cui è definita una retrazione R , abbiamo che $\hat{f} = f \circ R$ denota il *pullback* di f attraverso R . Per $x \in M$, abbiamo che

$$\hat{f}_x = f \circ R_x$$

denota la restrizione di \hat{f} su $T_x M$. Si noti che \hat{f}_x è una funzione reale su uno spazio vettoriale. Dalla condizione di rigidità locale abbiamo che $D\hat{f}_x(0_x) = Df(x)$. Se M è dotato di una metrica Riemanniana abbiamo

$$\text{grad} \hat{f}_x(0_x) = \text{grad} f(x).$$

Un caso particolare di retrazione è la **mappa esponenziale** definita su $p \in M$ e $v \in T_p M$, che ha proprietà di retrazione, rappresenta la curva geodetica calcolata nel punto 1 originaria in p con derivata in 0 punto pari a v .

In alcuni casi può essere necessario dover effettuare operazioni tra vettori appartenenti a spazi tangente differenti, come ad esempio nel caso dei metodi di ottimizzazione quasi Newton. Nell'ottimizzazione basata su retrazioni lo strumento standard per trasportare vettori appartenenti ad uno spazio tangente in un altro spazio tangente è il *trasportovettoriale*. Informalmente, dati due vettori tangenti ad x , chiamati $v_x, w_x \in T_x M$, un trasporto vettoriale \mathcal{T} associato ad una retrazione R , è una mappa liscia che genera un vettore appartenente ad $T_{R_x(w_x)} M$. Per semplicità utilizzeremo la notazione $\mathcal{T}_{x \rightarrow y}(v)$.

Un caso speciale di trasporto vettoriale è il *trasporto parallelo* che può essere interpretato come un trasporto vettoriale la cui retrazione associata è la mappa esponenziale.

2.5.2 Metodi *line-search*

I metodi di ricerca in linea (definiti *line-search*) su varietà si basano sulla formula di aggiornamento

$$x^{k+1} = R_{x^k}(t^k \eta^k),$$

dove η^k è in $T_{x^k} M$ e t^k è uno scalare. Una volta scelta la retrazione R ci rimane da decidere la direzione η^k e la lunghezza del passo t^k . Per ottenere la convergenza del metodo delle restrizioni sulla scelta di questi due parametri devono essere fatte.

Definizione 4 (sequenza gradient related su varietà) Data una funzione di costo f su una varietà riemanniana M , una sequenza $\{\eta^k\}$, $\eta^k \in T_{x^k} M$ è **gradient related** se per ogni sotto-sequenza $\{x^k\}_{k \in K}$ di $\{x^k\}$ che converge ad un punto non stazionario di f , la corrispondente sotto-sequenza $\{\eta^k\}_{k \in K}$ è limitata e soddisfa

$$\limsup_{k \rightarrow \infty, k \in K} \langle \text{grad } f(x^k), \eta^k \rangle < 0$$

Definizione 5 (Punto di Armijo su varietà) Data una funzione f su una varietà riemanniana M dotata di una retrazione R , un punto $x \in M$, un vettore tangente $\eta \in T_x M$, e degli scalari $\bar{\alpha} > 0, \beta, \sigma \in (0, 1)$, un **punto di Armijo su varietà** è $\eta^A = t^A \eta = \beta^m \bar{\alpha} \eta$, dove m è il più piccolo intero non negativo tale che $f(x) - f(R_x(\beta^m \bar{\alpha} \eta)) \geq -\sigma \langle \text{grad } f(x), \beta^m \bar{\alpha} \eta \rangle_x$. Il valore reale t^A è il **Passo di Armijo**.

Andiamo ora a definire un algoritmo generico per la discesa di gradiente su varietà (**Accelerated Line Search**), tale algoritmo rappresenta una generalità di metodologie di ottimizzazione delineando però, nel contempo, delle restrizioni

fondamentali che garantiscono la convergenza della sequenza di punti generati.

Algorithm 1: Accelerated Line Search (ALS)

Input: Manifold M ; funzione costo f differenziabile definita su M ;
 retraction R ; scalari $\bar{\alpha} > 0, c, \beta, \sigma \in (0, 1)$; initial iterate $x^0 \in M$

Output: Sequenza $\{x^k\}$

for $k = 0, 1, 2, \dots$ **do**

Prendere $\eta^k \in T_x M$ tale per cui la sequenza $\{\eta^i\}_{i=0,1,\dots}$ sia *gradient related*.

Selezionare x^{k+1} tale che

$$f(x^k) - f(x^{k+1}) \geq c(f(x^k) - f(R_{x^k}(t^{k^A} \eta^k))) \quad (1)$$

Dove t^{k^A} è il passo definito dalla regola di Armijo su varietà definita poco sopra.

end

Dallo pseudocodice appena esplicitato è chiaro che tale algoritmo può essere visto come una generalizzazione della scelta del passo di Armijo descritto nella sotto-sezione 2.2, difatti scegliere $x^{k+1} = R_{x^k}(t^{k^A} \eta^k)$ soddisfa (1). Inoltre la condizione rilassata (1) ci permette un ampio spazio di manovra nella scelta di x^{k+1} garantendoci la convergenza ad un punto stazionario, come enunceremo formalmente, e ciò può portare alla definizione di algoritmi altamente efficienti.

2.5.3 Convergenza

Il concetto di convergenza può essere ridefinito e generalizzato su varietà. Una sequenza infinita $\{x^k\}_{k=0,1,\dots}$ di punti su una varietà M è detta essere convergente se esiste una carta (U, ψ) di M , un punto $x^* \in M$ e $K > 0$ tali che x^k è in U per ogni $k \geq K$ e tale che la sequenza $\{\psi(x^k)\}_{k=K,K+1,\dots}$ converge a $\psi(x^*)$ (in tal caso applichiamo il concetto di convergenza di R^n , avendo ψ come codominio uno spazio euclideo). Il punto $\psi^{-1}(\lim_{k \rightarrow \infty} \psi(x^k))$ è chiamato il *limite* della sequenza $\{x^k\}_{k=0,1,\dots}$. Tutte le sequenze convergenti di una varietà di *Hausdorff* hanno uno ed un solo punto limite.

Diamo ora un importante risultato rispetto alla convergenza dell'algoritmo *ALS*, tale enunciato deriva direttamente dal risultato di convergenza definito nella sezione 2.2 rispetto al passo di Armijo in R^n , in tal caso però viene data una generalizzazione di quest'ultimo essendo, come già detto, *ALS* una generalizzazione del passo di Armijo e lavorando non più su R^n ma su varietà differenziabili.

Teorema 3 *Sia $\{x^k\}$ una sequenza infinita di iterate generate da ALS. Allora ogni punto di accumulazione di $\{x^k\}$ è un punto stazionario della funzione costo f*

Inoltre assumendo compattezza della varietà M possiamo enunciare il seguente

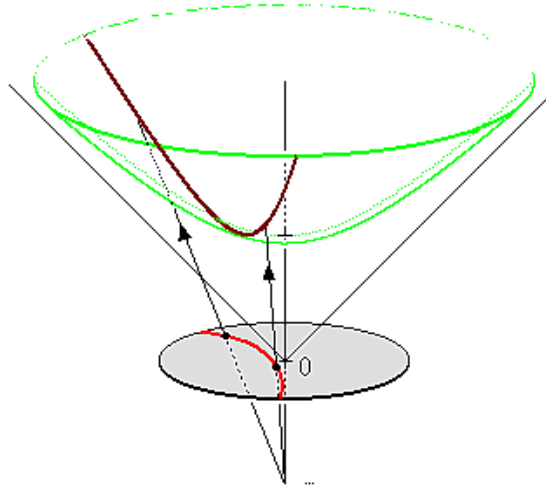


Figure 1: Rappresentazione della proiezione stereografica tra disco e iperboloide

Corollario 1 Sia $\{x^k\}$ una sequenza finita di iterate generata da ALS. Assumiamo che l'insieme di livello $L = \{x \in M : f(x) \leq f(x^0)\}$ è compatto allora $\lim_{k \rightarrow \infty} \|\text{grad } f(x^k)\| = 0$.

3 Disco di poicare e Iperboloide

Lo spazio iperbolico può essere definito in vari modi equivalenti, noi descriveremo l'iperboloide ed il disco di poicare, ciascuno di questi è un modello dello spazio iperbolico, nessuno dei due è prevalente in letteratura.

3.1 Iperboloide

Possiamo definire H^n come il luogo dei punti di norma -1 in R^{n+1} , dotato dell'usuale prodotto di minkowski ($\langle u, v \rangle_{n+1} = \sum_{i=1}^n u_i v_i - u_{n+1} v_{n+1}$; $u, v \in R^{n+1}$), questo luogo di punti ha in realta due componenti connesse, noi ne scegliamo una. Diamo la definizione di tale modello dello spazio iperbolico.

Definizione 6 (Iperboloide) Il modello dell'iperboloide è definito nel modo seguente.

$$H^n = \{x \in R^{n+1} | \langle x, x \rangle_{n+1} = -1; \quad x^{n+1} > 0\}.$$

Lo spazio tangente in un punto $p \in H^n$ è $T_p H^n = \{x \in R^{n+1} | \langle p, x \rangle_{n+1} = 0\}$.

Possiamo asserire che l'iperboloide è una **varietà riemanniana**. Andiamo ora ad esplicitare gli strumenti necessari per effettuare ottimizzazione su questa varietà, le formule riportate di seguito, tra cui la derivata della funzione distanza, sono state riprese da [1]

3.1.1 Metrica

La metrica, definita nel *Tangent Bundle* dell'iperboloide, è la metrica indotta dal prodotto di Minkowski già esplicitato.

3.1.2 Distanza

La distanza tra due punti dell'iperboloide rappresenta la lunghezza della geodetica che unisce i due punti, è definita come segue.

$$d_{H^n}(u, v) = \operatorname{arccosh}(-\langle u, v \rangle_{n:1}); \quad u, v \in H^n.$$

3.1.3 Mappa Esponenziale

La mappa esponenziale definita su $p \in H^n$ e $v \in T_p H^n$. E' definita come segue.

$$\operatorname{Exp}_p(v) = \cosh(\|v\|_{n:1})p + \sinh(\|v\|_{n:1})\frac{v}{\|v\|_{n:1}}.$$

3.1.4 Gradiente della funzione distanza

La definizione dell'iperboloide e dello spazio di minkowski nel quale è costruito ci aiuta nel calcolo del gradiente della funzione distanza.

Data una funzione costo E qualsiasi possiamo calcolarne il gradiente in un punto $p \in H^n$, anch'esso arbitrario, passando dapprima per il gradiente definito nell'ambiente, in tal caso lo spazio di minkowski, ricordiamo che quest'ultimo è così definito.

$$\nabla_p^{R^{n:1}} E = \left(\frac{\partial E}{\partial x_1} \Big|_p, \dots, \frac{\partial E}{\partial x_n} \Big|_p, -\frac{\partial E}{\partial x_{n+1}} \Big|_p \right).$$

Una volta calcolato il gradiente nell'ambiente procediamo a proiettarlo nello spazio tangente del punto $p \in H^n$ come segue.

$$\nabla_p^{H^n} E = \nabla_p^{R^{n:1}} E + \langle p, \nabla_p^{R^{n:1}} E \rangle_{n:1} p.$$

Relativamente al problema della media di fréchet, che andremo ad esplicitare, definiamo la seguente formula la quale rappresenta il gradiente su spazio di minkowski della funzione distanza.

$$\nabla_u^{R^{n:1}} d_{H^n}(u, v) = -(\langle u, v \rangle_{n:1}^2 - 1)^{-1/2} v.$$

3.2 Disco di Poincaré

Andiamo ora a definire l'altro modello dello spazio iperbolico definito disco di Poicaré, definito come l'insieme dei punti di norma strettamente minore di 1 in

R^n .

Di seguito ne riportiamo una definizione formale.

Definizione 7 (Disco di Poincaré) *Il modello del disco di Poincaré è definito nel modo seguente*

$$D^n = \{x \in R^n \mid \|x\| < 1\}$$

Essendo D^n un sottoinsieme di R^n lo spazio tangente in ogni punto di D^n è R^n . Possiamo asserire che il disco di Poincaré è una **varietà riemanniana**. Come fatto per l'iperboloide andiamo ad esplicitare i vari strumenti necessari per effettuare ottimizzazione. Le formule riportate sono state riprese da [5], [2], mentre per il calcolo del gradiente della funzione distanza abbiamo fatto riferimento a [3].

3.2.1 Metrica

La metrica, definita nello spazio tangente del disco di Poincaré $T_p D^n$ rispetto ad un punto $p \in D^n$, è la seguente.

$$g^D = \lambda_p^2 g^E; \text{ con } \lambda_p = \frac{2}{1 - \|p\|^2} \text{ e con } g^E \text{ la metrica euclidea.}$$

3.2.2 Distanza

La distanza tra due punti del disco di Poincaré rappresenta la lunghezza della geodetica che unisce i due punti, è definita come segue.

$$d_{D^n}(u, v) = \operatorname{arccosh}\left(1 + \frac{\|u - v\|^2}{(1 - \|u\|^2)(1 - \|v\|^2)}\right)$$

3.2.3 Mappa Esponenziale

La mappa esponenziale definita su $p \in D^n$ e $v \in T_p D^n$. E' definita come segue.

$$\begin{aligned} \operatorname{Exp}_p(v) = & \frac{\lambda_p(\cosh(\lambda_p \|v\|) + \langle p, \frac{v}{\|v\|} \rangle \sinh(\lambda_p \|v\|))p}{1 + (\lambda_p - 1)\cosh(\lambda_p \|v\|) + \lambda_p \langle p, \frac{v}{\|v\|} \rangle \sinh(\lambda_p \|v\|)} + \\ & \frac{\frac{1}{\|v\|} \sinh(\lambda_p \|v\|)v}{1 + (\lambda_p - 1)\cosh(\lambda_p \|v\|) + \lambda_p \langle p, \frac{v}{\|v\|} \rangle \sinh(\lambda_p \|v\|)} \end{aligned}$$

3.2.4 Gradiente della funzione distanza

Dati $u, v \in D^n$ la formula del gradiente riemanniano della funzione distanza è dato da.

$$\nabla_u^E d(u, v) = \frac{4}{b\sqrt{c^2 - 1}} \left(\frac{(\|v\|^2 - 2\langle u, v \rangle + 1)v}{a^2} - \frac{v}{a} \right)$$

con $a = 1 - \|u\|^2$, $b = 1 - \|v\|^2$, $c = 1 + \frac{2}{ab} \|u - v\|^2$

Sappiamo che per ricavare il gradiente riemanniano dal gradiente euclideo è sufficiente riscalare ∇^E per l'inverso della metrica riemanniana, perciò,

$$\nabla_u^{D^n} d(u, v) = \frac{\nabla_u^E d(u, v)}{\lambda_u^2}.$$

3.3 Iperboloide come modello conforme

Come inizialmente accennato le due varietà trattate fin'ora sono due modelli conformi dello spazio iperbolico ovvero, esiste un diffeomorfismo invertibile conforme (mantiene gli angoli ma non le lunghezze) che porta da una varietà all'altra. Il diffeomorfismo in questione è la proiezione stereografica ρ definita come segue.

$$\begin{aligned} \rho : H^n &\rightarrow D^n \\ \rho(x) &= \frac{1}{x_{n+1} + 1} (x_1, \dots, x_n) ; \text{ con } x \in H^n \\ \rho^{-1} : D^n &\rightarrow H^n \\ \rho^{-1}(y) &= \frac{2}{1 - r} (y_1, \dots, \frac{1+r}{2}) ; \text{ con } r = \|y\|^2 \text{ e } y \in D^n \end{aligned}$$

A questo punto facciamo vedere che vale la seguente uguaglianza, ci sarà utile nella sezione successiva.

$$d_{D^n}(a, b) = d_{H^n}(\rho^{-1}(a), \rho^{-1}(b))$$

Daremo due risultati, dimostreremo che l'uguaglianza è valida, dando dapprima una dimostrazione per $n = 2$, e poi la generalizzeremo per n qualsiasi.

Caso $n = 2$, siano $a, b \in D^2$.

$$a = (a_1, a_2)$$

$$b = (b_1, b_2)$$

$$\rho^{-1}(a) = \frac{2}{1-r_a}(a_1, a_2, \frac{1+r_a}{2})$$

$$\rho^{-1}(b) = \frac{2}{1-r_b}(b_1, b_2, \frac{1+r_b}{2})$$

$$\text{con } r_a = (a_1^2 + a_2^2) \text{ e } r_b = (b_1^2 + b_2^2)$$

$$\begin{aligned} d_{H^n}(\rho^{-1}(a), \rho^{-1}(b)) &= \\ \text{arccosh}(-\langle (\frac{2a_1}{1-r_a}, \frac{2a_1}{1-r_a}, \frac{1+r_a}{2}), (\frac{2b_1}{1-r_b}, \frac{2b_1}{1-r_b}, \frac{1+r_b}{2}) \rangle_{2:1}) &= \\ \text{arccosh}(-(\frac{4a_1b_1}{(1-r_a)(1-r_b)} + \frac{4a_2b_2}{(1-r_a)(1-r_b)} - \frac{(1+r_a)(1+r_b)}{(1-r_a)(1-r_b)}) &= \\ \text{arccosh}(\frac{(1+r_a)(1+r_b) - 4a_1b_1 - 4a_2b_2}{(1-r_a)(1-r_b)}) &= \\ \text{arccosh}(\frac{1+r_a+r_b+r_ar_b - 4a_1b_1 - 4a_2b_2}{1-r_a-r_b+r_ar_b}) &= \\ \text{arccosh}(\frac{1-r_a-r_b+2r_a+2r_b+r_ar_b - 4a_1b_1 - 4a_2b_2}{1-r_a-r_b+r_ar_b}) &= \\ \text{arccosh}(1 + \frac{2r_a+2r_b-4a_1b_1-4a_2b_2}{(1-r_a)(1-r_b)}) &= \\ \text{arccosh}(1 + 2\frac{r_a+r_b-2a_1b_1-2a_2b_2}{(1-r_a)(1-r_b)}) &= \\ \text{arccosh}(1 + 2\frac{\|a-b\|^2}{(1-\|a\|^2)(1-\|b\|^2)}) &= \\ d_{D^n}(a, b). \end{aligned}$$

Facciamo vedere ora che l'uguaglianza vale anche nel caso in cui n è generico, siano $a, b \in D^n$. Procediamo facendo vedere che i due termini dell'uguaglianza rappresentano la stessa quantità.

$$d_D(a, b) = \text{arccosh}(1 + 2\frac{\|a+b\|^2}{(1-\|a\|^2)(1-\|b\|^2)})$$

$$d_H(\rho^{-1}(a), \rho^{-1}(b)) = \text{arccosh}(-\langle \rho^{-1}(a), \rho^{-1}(b) \rangle_{n:1})$$

Facciamo vedere ora, che gli argomenti della funzione arccosh, nelle due equazioni riportate, rappresentano in realtà lo stesso valore.

$$\begin{aligned}
& -\langle \rho^{-1}(a), \rho^{-1}(b) \rangle_{n:1} = \\
& -\left\langle \frac{2}{1 - \|a\|^2} \left[\frac{a}{\frac{1+\|a\|^2}{2}} \right], \frac{2}{1 - \|b\|^2} \left[\frac{b}{\frac{1+\|b\|^2}{2}} \right] \right\rangle_{n:1} = \\
& \frac{4}{(1 - \|a\|^2)(1 - \|b\|^2)} \left(a'b - \frac{(1 + \|a\|^2)(1 + \|b\|^2)}{4} \right) = \\
& \frac{-4a'b + 1 + \|a\|^2 + \|b\|^2 + \|a\|^2\|b\|^2}{(1 - \|a\|^2)(1 - \|b\|^2)} = \\
& \frac{-4a'b + 1 + 2\|a\|^2 + 2\|b\|^2 - \|a\|^2 - \|b\|^2 + \|a\|^2\|b\|^2}{(1 - \|a\|^2)(1 - \|b\|^2)} = \\
& 1 + 2 \frac{\|a - b\|^2}{(1 - \|a\|^2)(1 - \|b\|^2)}
\end{aligned}$$

3.4 Media di Fréchet e problema del centroide

In tale sezione formalizzeremo il concetto di **media di Fréchet**, daremo inoltre un'importante congettura. Partiamo dapprima dalla seguente definizione.

Definizione 8 (Media di Fréchet) *Data la funzione **somma delle distanze al quadrato** di p -punti, così definita.*

$$f(\Theta) = \frac{1}{p} \sum_{i=1}^p d^2(\Theta, x^{(i)}).$$

La media di Fréchet è l'argomento Θ che minimizza tale funzione.

Il problema di trovare la media di Fréchet (centroide) di p -punti consiste nel trovare il punto che minimizza la funzione **somma delle distanze al quadrato** definita a partire dai punti in questione, il cui minimo esiste ed è unico. Tale problema può essere risolto direttamente in uno spazio euclideo il minimo è dato dalla media aritmetica dei p -punti. Mentre in uno spazio iperbolico, come nel nostro caso, la soluzione non è diretta e perciò si applicano le metodologie di ottimizzazione su varietà differenziabili trattate nella sezione 2.5.

Per effettuare ottimizzazione abbiamo bisogno di calcolare il gradiente della funzione somma delle distanze al quadrato, avendo definito nella sezione precedente il gradiente della funzione distanza su disco e su iperboloide, possiamo calcolare il gradiente della funzione come segue,

$$\nabla f(\Theta) = \frac{2}{p} \sum_{i=1}^p d(\Theta, x^{(i)}) \cdot \nabla d(\Theta, x^{(i)}).$$

Diamo ora una congettura che possiamo affermare essere corretta in funzione

dell'uguaglianza $d_{D^n}(a, b) = d_{H^n}(\rho^{-1}(a), \rho^{-1}(b))$ che abbiamo verificato nella sezione precedente.

Dati $a^{(i)}, \Theta \in D^n$; con $i = 1 \dots p$ dall'uguaglianza precedente è facile notare che $\frac{1}{p} \sum_{i=1}^p d_{D^n}^2(\Theta, a^{(i)}) = \frac{1}{p} \sum_{i=1}^p d_{H^n}^2(\rho^{-1}(\Theta), \rho^{-1}(a^{(i)}))$, perciò essendo ρ un diffeomorfismo se $\exists \Theta \in D^n$ per cui $\frac{1}{p} \sum_{i=1}^p d_{D^n}^2(\Theta, a^{(i)})$ è minima allora esiste $\exists \Psi \in H^n$ per cui è minima $\frac{1}{p} \sum_{i=1}^p d_{H^n}^2(\Psi, \rho^{-1}(a^{(i)}))$. Essendo inoltre $f(x) = \hat{f} = f(\rho^{-1}(x))$ allora cercare il minimo di f equivale a cercare il minimo di $\hat{f} \circ \rho^{-1}$.

In funzione di tale congettura possiamo trasportare il problema definito su disco di Poincaré nella varietà Iperboloide e viceversa.

4 Metodi utilizzati con pseudocodice ed esperimenti

In questa sezione andremo a definire lo pseudocodice degli algoritmi di ottimizzazione che abbiamo utilizzato per confrontare il calcolo del minimo della funzione per il calcolo della media di Fréchet sui modelli dello spazio iperbolico descritti nella sezione precedente, ovvero disco di Poincaré ed iperboloide, sfrutteremo il fatto che questi due modelli sono tra di loro conformi, ovvero che esiste una mappa conforme che li lega.

4.1 Algoritmi e Pseudocodice

Descriviamo tre algoritmi di ottimizzazione, di cui due sono algoritmi di prim'ordine **passo fisso** e **armijo**, ovvero che utilizzano informazioni solamente sul gradiente nel punto per decidere il passo e la direzione dell'iterazione, mentre il terzo, **Barzilai-Borwein**, è un algoritmo di quasi Newton e perciò ha accesso ad informazioni di approssimazione al secondo ordine, sfruttando l'equazione delle secanti. Tutti gli algoritmi si basano sul calcolo del gradiente della funzione da minimizzare.

4.1.1 Passo fisso

L'algoritmo a Passo fisso è un algoritmo del primo ordine molto elementare, la direzione scelta è l'opposta della direzione del gradiente, tale algoritmo implementato su una varietà differenziabile necessita solamente del gradiente riemanniano della funzione da minimizzare, di una retrazione e di un parametro α che denota il passo fisso applicato ad ogni iterazione.

Algorithm 2: Fixed lenght step size Algorithm

Input: Manifold M ; funzione costo f differenziabile definita su M ;
retraction R ; scalare $\alpha > 0$; initial iterate $x^0 \in M$
Output: Sequenza $\{x^k\}$
for $k = 0, 1, 2, \dots$ **do**
 $d_k \leftarrow -\alpha g^k$; con g^k gradiente della funzione al passo k .
 $x^{k+1} \leftarrow R_{x^k}(d_k)$
end

4.1.2 Armijo

L'algoritmo di Armijo è anche esso un algoritmo del primo ordine, la direzione scelta è l'opposta della direzione del gradiente, per la selezione del passo si basa sul calcolo del punto di armijo riemanniano, descritto nella sezione 2.5.2, per poter essere eseguito necessita, anche esso, del gradiente riemanniano della funzione da minimizzare, di una retrazione e di tre parametri $\gamma \in (0, 1)$, $\beta \in (0, 1)$ e $\lambda > 0$. Solitamente il parametro λ deve essere calcolato ad ogni iterazione, necessario per effettuare il giusto rescaling di d_k , per semplicità di sperimentazione abbiamo deciso di assumere tale parametro fisso. L'implementazione riemanniana di tale algoritmo fa riferimento alle formule in [4].

Algorithm 3: Armijo Algorithm

Input: Manifold M ; funzione costo f differenziabile definita su M ;
retrazione R ; scalari $\gamma \in (0, 1)$, $\sigma \in (0, 1)$, $\lambda > 0$; initial iterate $x^0 \in M$
Output: Sequenza $\{x^k\}$
for $k = 0, 1, 2, \dots$ **do**
 Trovare il più piccolo $h = 0, 1, \dots$, tale per cui

$$f(R_{x^k}(\sigma^h \lambda d_k)) \leq f(x^k) + \gamma \sigma^h \lambda \langle g^k, d_k \rangle_{x^k};$$

 con g^k gradiente della funzione al passo k .
 $x^{k+1} \leftarrow R_{x^k}(-\sigma^h \lambda g^k)$.
end

4.1.3 Barzilai-Borwein

L'algoritmo di Barzilai-Borwein è un metodo quasi Newton e si basa sulla soluzione, per $k \geq 1$, del problema dei minimi quadrati, che ci permette di avere ricavare informazioni sulla matrice hessiana, quindi del secondo ordine. Tale algoritmo è stato implementato in riferimento a [4].

$$\min_t \|s_k t - y_k\|_2,$$

con $s_k = x^{k+1} - x^k$ ed $y_k = \nabla f(x^{k+1}) - \nabla f(x^k)$, il quale, assumendo che $x^{k+1} \neq x^k$ ha un'unica soluzione $t = \frac{s'_k y_k}{s'_k s_k}$. Quando $s'_k y_k > 0$ il passo viene scelto come.

$$a_{k+1}^{BB} = \frac{s'_k s_k}{s'_k y_k}.$$

Per tradurre l'algoritmo dalla sua definizione su spazio euclideo in una versione compatibile con ottimizzazione su varietà differenziabili dobbiamo andare a ridefinire s_k ed y_k . Al passo $k+1$, l'hessiana è una mappa lineare da $T_{x^{k+1}}M$ a $T_{x^{k+1}}M$ perciò considereremo il vettore $-\alpha^k g^k \in T_{x^k}M$ e lo trasporteremo in $T_{x^{k+1}}M$, perciò

$$s_k = \mathcal{T}_{x^k \rightarrow x^{k+1}}(-\alpha^k g^k).$$

Per ottenere y_k dobbiamo sottrarre due gradienti appartenenti a due differenti spazi tangente, perciò

$$y_k = g^{k+1} - \mathcal{T}_{x^k \rightarrow x^{k+1}}(g^k).$$

Mentre per quanto riguarda il prodotto scalare utilizzeremo quello riemanniano, quindi

$$a_{k+1}^{BB} = \frac{\langle s_k, s_k \rangle_{x^{k+1}}}{\langle s_k, y_k \rangle_{x^{k+1}}}.$$

Algorithm 4: Barzilai-Borwein Algorithm

Input: Manifold M ; funzione costo f differenziabile definita su M ;
 retrazione R trasporto vettoriale \mathcal{T} ; scalari $\alpha_{max} > \alpha_{min} > 0$,
 $\alpha_0^{BB} \in [\alpha_{min}, \alpha_{max}]$; punto di partenza $x^0 \in M$

Output: Sequenza $\{x^k\}$

$g^0 \leftarrow \nabla^{(R)} f(x^0).$

for $k = 0, 1, 2, \dots$ **do**

$x^{k+1} \leftarrow R_{x^k}(-a_k^{BB} g^k).$
 $f_{k+1} \leftarrow f(x^{k+1}), g^{k+1} \leftarrow \nabla^{(R)} f(x^{k+1}).$
 $s_k \leftarrow \mathcal{T}_{x^k \rightarrow x^{k+1}}(-\alpha_{k+1}^{BB} g^k).$
 $y_k \leftarrow g^{k+1} - \mathcal{T}_{x^k \rightarrow x^{k+1}}(g^k).$
 $\tau_{k+1} \leftarrow \frac{\langle s_k, s_k \rangle_{x^{k+1}}}{\langle s_k, y_k \rangle_{x^{k+1}}}.$

$\begin{cases} \alpha_{k+1}^{BB} = \min\{\alpha_{max}, \max\{\tau_{k+1}, \alpha_{min}\}\} & \text{if } \langle s_k, y_k \rangle_{x^{k+1}} > 0. \\ \alpha_{k+1}^{BB} = \alpha_{max} & \text{otherwise} \end{cases}$

end

Nella nostra implementazione dell'algoritmo **Riemannian Barzilai-Borwein** abbiamo deciso di non applicare valori di riscaldamento, tramite ricerca lineare esatta o non, al valore α_k^{BB} , per motivazioni sperimentali, sarebbe stato computazionalmente più oneroso, alcuni esperimenti avrebbero richiesto più tempo per essere eseguiti, considerando inoltre che l'implementazione riportata si comporta mediamente bene nella pratica.

4.2 Esperimenti

Gli esperimenti effettuati ci permetteranno di confrontare il problema di ottimizzazione della funzione **somma delle distanze al quadrato** implementato sulla varietà disco di Poincaré e l'equivalente istanza trasportata su iperboloide, inoltre tali esperimenti saranno utili per capire se queste due varietà sono equivalenti ed interscambiabili, non solo nella rappresentazione dello spazio iperbolico ma anche come varietà su cui effettuare ottimizzazione, a patto che la mappa conforme che lega tali varietà mantenga l'esistenza del minimo per la funzione da minimizzare, per quanto riguarda la funzione costo in questione, tale proprietà è già stata verificata nella sezione 3.3.

Gli esperimenti effettuati si suddividono in due parti che nel seguito descriveremo. Per effettuare gli esperimenti ci siamo muniti di un dataset di circa duecento istanze del problema definito su disco di Poincaré, prendendo perciò un numero fissato, comune a tutte le istanze, di punti casuali sul disco (sampling effettuato sfruttando la funzione *rand* del file *PoincareFactory* della libreria *manopt* [6]), per ogni istanza abbiamo calcolato il punto x^0 di partenza come la media aritmetica dei punti dell'istanza, abbiamo calcolato inoltre il punto limite tramite l'algoritmo a passo fisso con un α molto piccolo e come guardia di arresto abbiamo scelto la condizione che la norma euclidea del gradiente g^k scendesse sotto un valore soglia (e.g. 10^{-10}).

Una volta ottenuto il dataset, prima di poter effettivamente confrontare gli algoritmi implementati nelle due varietà abbiamo selezionato i parametri ottimi, uno per disco ed uno per iperboloide, per gli algoritmi che hanno necessità di dover modificare alcuni parametri, come α per il passo fisso e λ per l'algoritmo di Armijo. Per effettuare tale selezione abbiamo preso un sotto insieme del dataset iniziale e, sia per l'algoritmo a passo fisso che per Armijo, abbiamo proceduto come segue. Abbiamo scelto un valore che rappresenta il numero di elementi in cui si suddivide lo spazio dei possibili valori del parametro di cui si deve trovare l'ottimo (e.g. 100, partiremo da 0.01 ed arriveremo ad 0.99 con passo pari a 0.01), per ogni record del sotto dataset si è applicato l'algoritmo con un parametro incrementale creando per ogni istanza una sequenza i cui valori rappresentano il numero di passi che l'algoritmo con il parametro in questione impiega per convergere al limite, o divergere (numero di passi prima di arrivare ad avere la norma della distanza dal limite di 10^{-4} , se diverge il numero sarà il massimo delle iterazioni possibili), una volta calcolate queste sequenze ne abbiamo fatta una media

aritmetica e come parametro ottimo abbiamo scelto il valore che minimizza la sequenza media.

Per quanto riguarda la scelta di α ottimo per l'algoritmo a passo fisso abbiamo notato sperimentalmente che le sequenze di convergenza, definite per l'implementazione su disco e su iperboloide, hanno un andamento quasi identico ed anche il medesimo punto di minimo che indica il parametro ottimo.

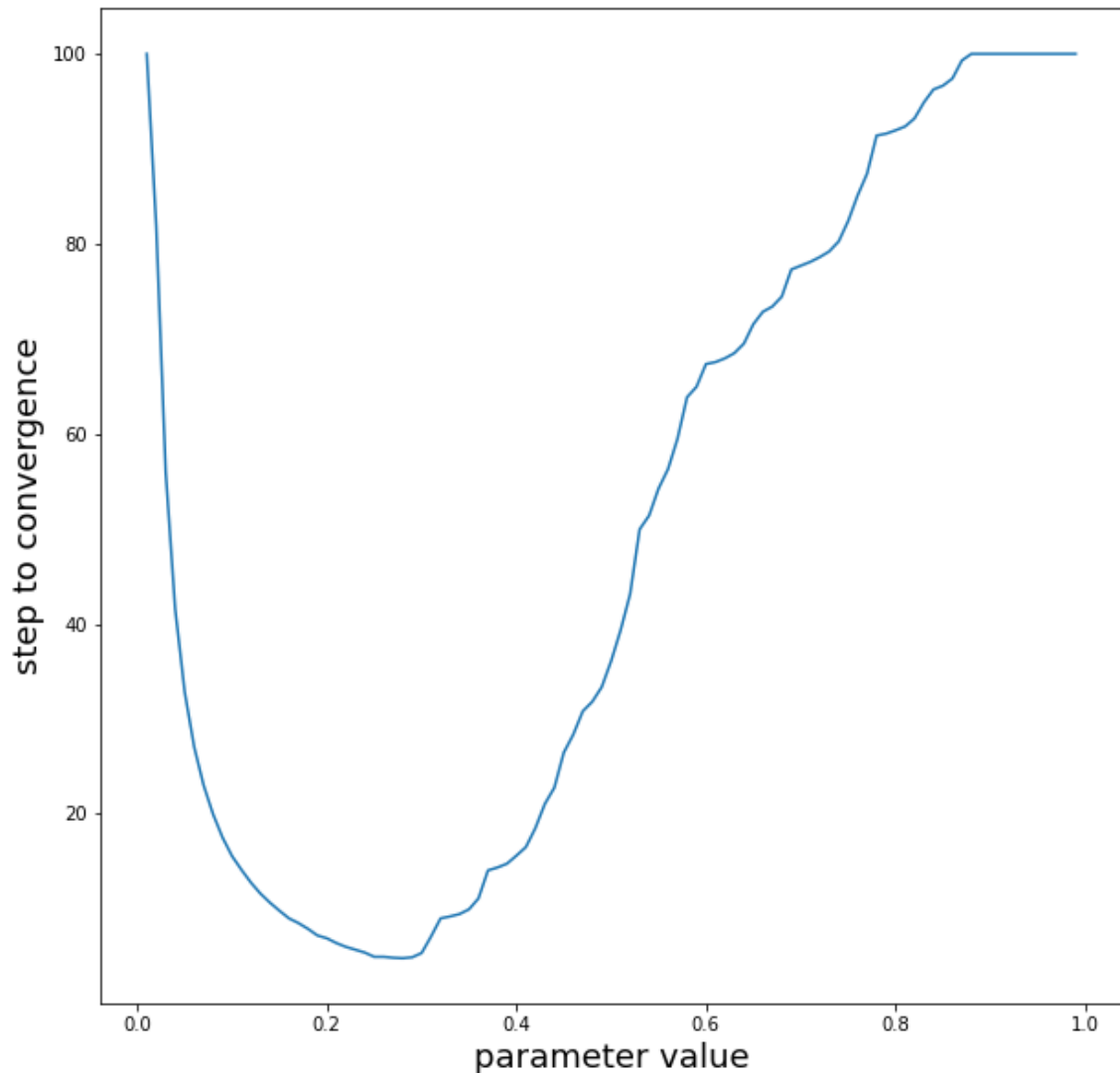


Figure 2: Sequenza relativa alla convergenza dell'algoritmo a passo fisso implementato su disco. Punto di minimo 0.28, valore di convergenza minimo 4.8.

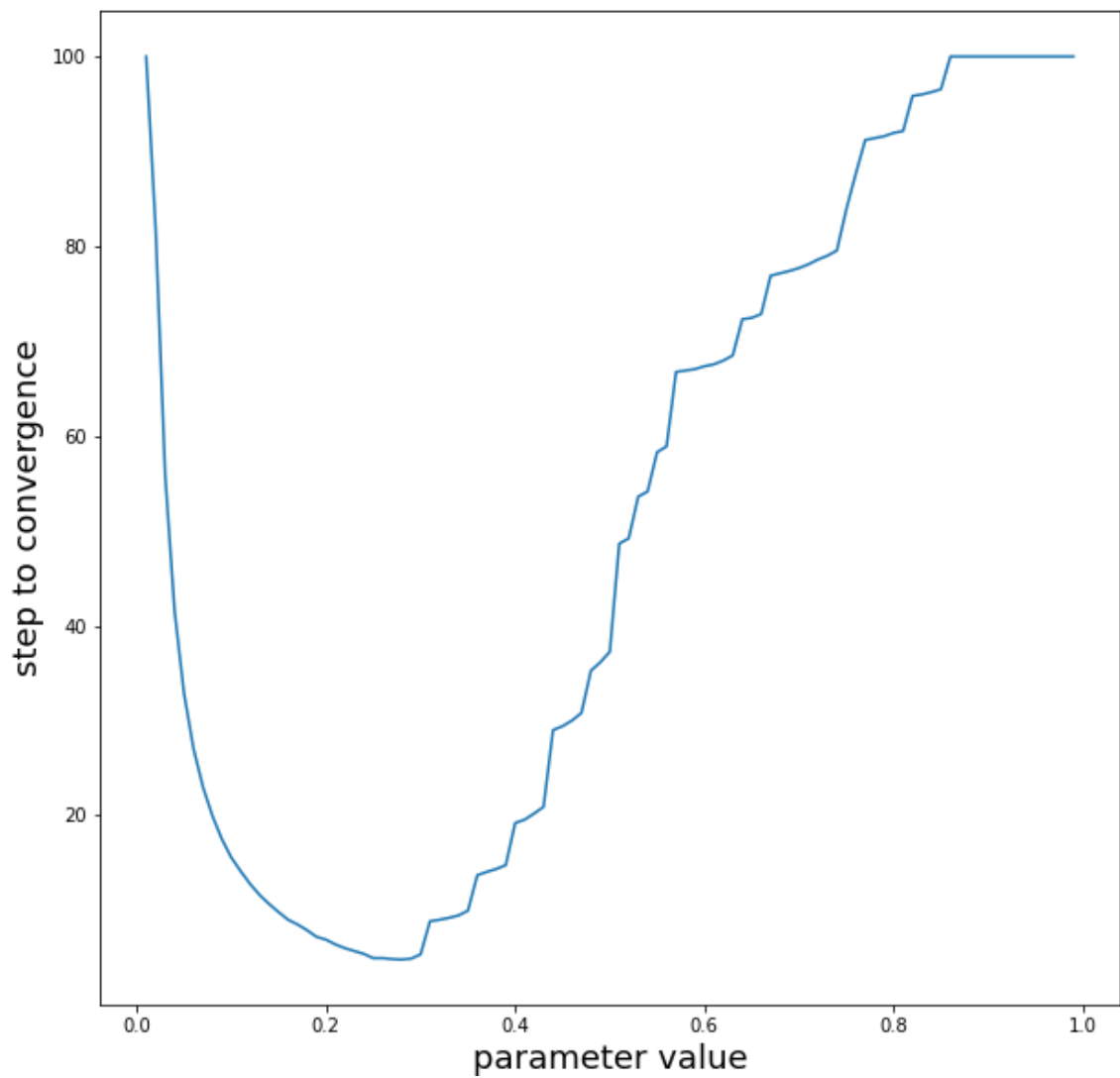


Figure 3: Sequenza relativa alla convergenza dell'algoritmo a passo fisso implementato su iperboloide. Punto di minimo 0.28, valore di convergenza minimo 4.8.

Analogamente, per quanto riguarda le sequenze definite dalla scelta di λ per l'algoritmo di Armijo, si è notato un'andamento molto simile nelle due sequenze, anche in questo caso il punto di minimo coincide.

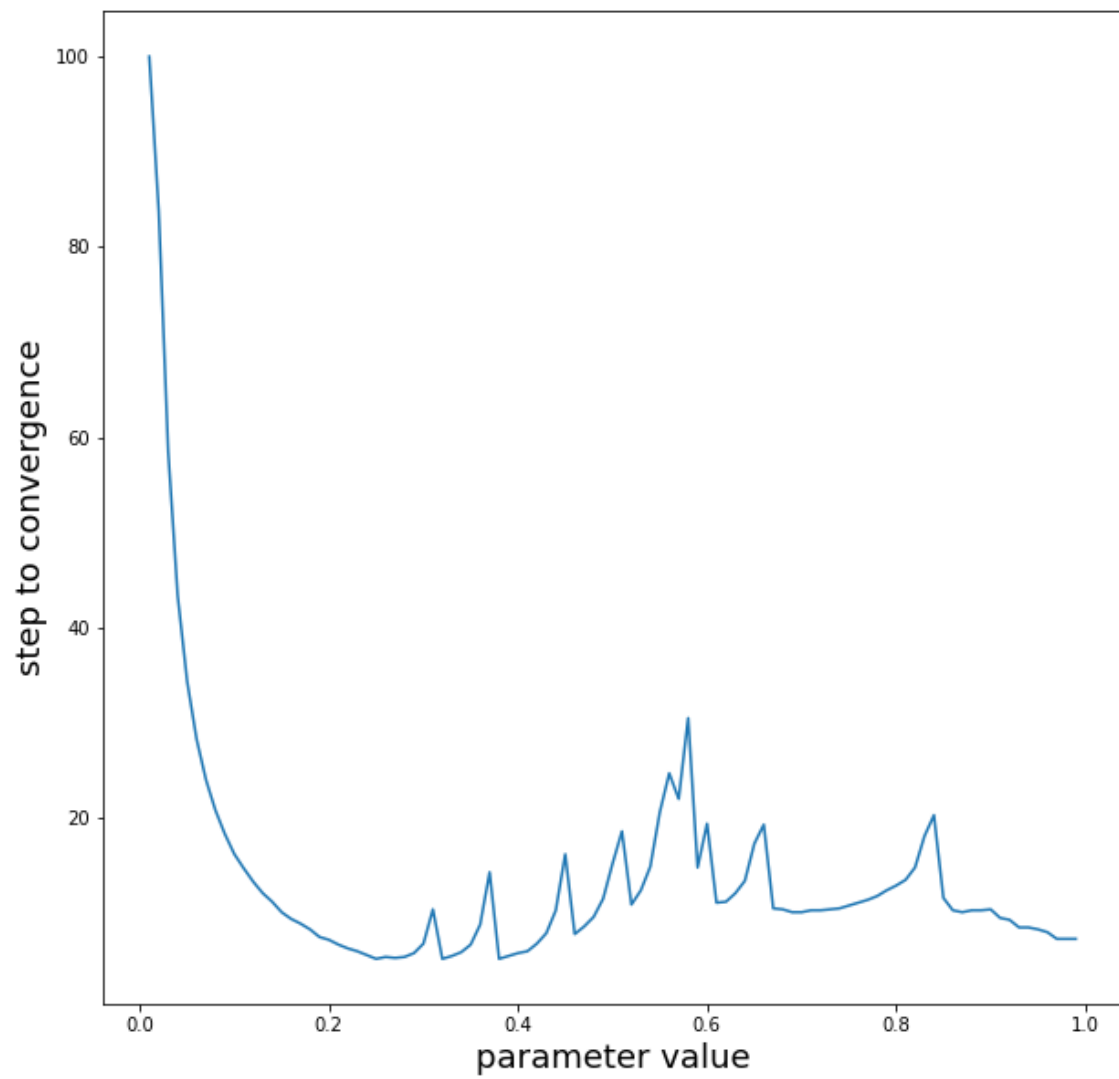


Figure 4: Sequenza relativa alla convergenza dell'algoritmo di Armijo implementato su disco. Punto di minimo 0.25 , valore di convergenza minimo 5.2.

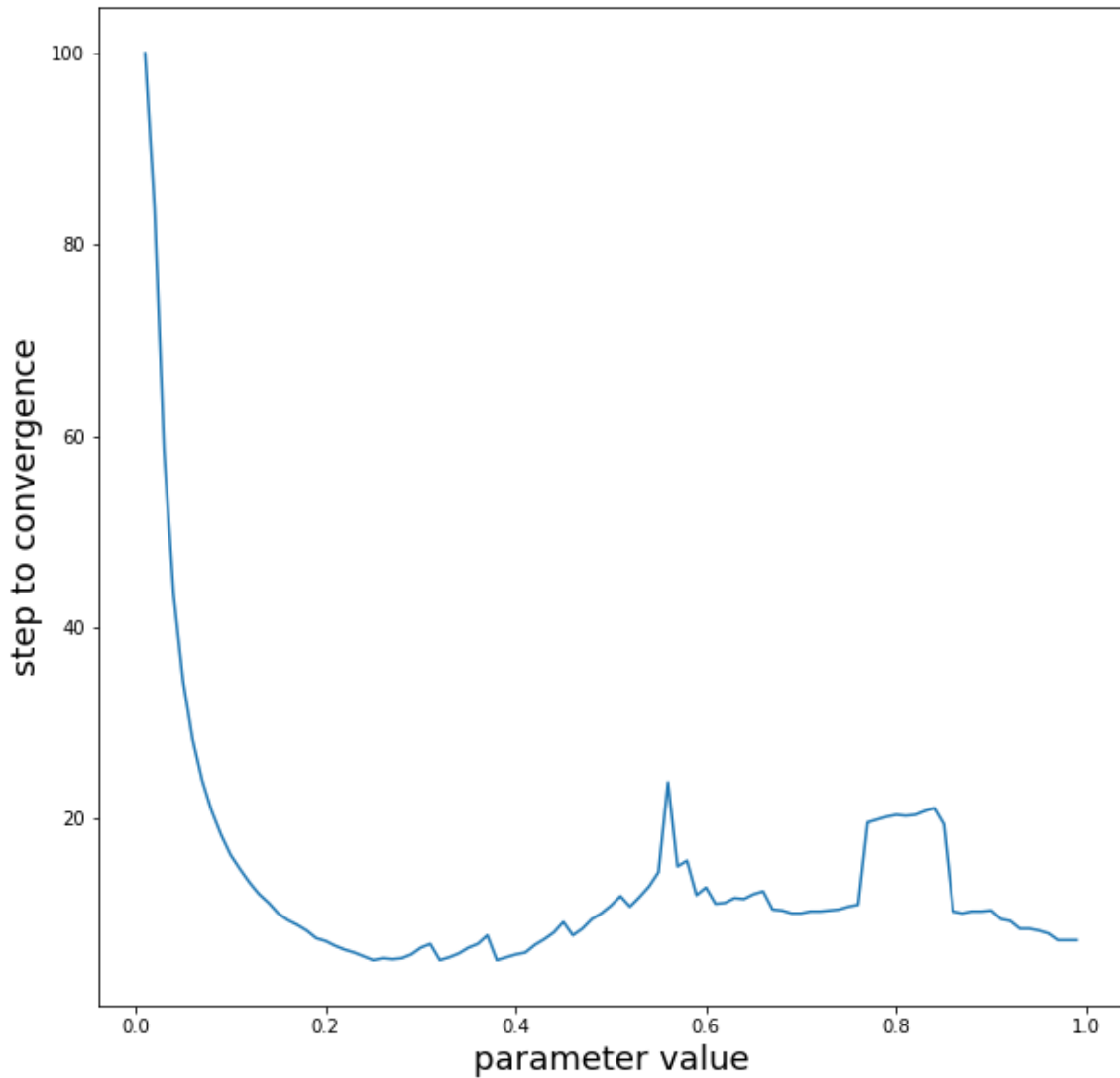


Figure 5: Sequenza relativa alla convergenza dell'algoritmo di Armijo implementato su iperboloide. Punto di minimo 0.25 , valore di convergenza minimo 5.2.

A questo punto, avendo scelto i parametri che permettono ai nostri algoritmi di performare al meglio, possiamo procedere con il confronto.

Riprendendo il dataset totale, per ogni algoritmo lo abbiamo eseguito su ogni istanza del dataset, sia per l'implementazione su disco che per quella su iperboloide, creando un punto (a, b) in R^2 le cui coordinate sono a , il numero di passi per convergere al punto limite per l'implementazione su disco, b invece, il numero di passi per convergere al punto limite per l'implementazione su iperboloide, creando così una nuvola di punti. Tramite tale nuvola possiamo verificare se le metodologie sono effettivamente equivalenti, in ogni grafico vengono inoltre costruite due rette di regressione su una versione ridotta della nuvola di punti, toglieremo quei punti che hanno una molteplicità bassa (e.g. sotto $\frac{1}{40}$ della cardinalità totale del dataset), la prima retta di regressione viene calcolata tramite il metodo dei minimi quadrati, la seconda invece viene calcolata tramite un algoritmo meno sensibile

ai valori anomali, la regressione di *Huber*. Noteremo quindi che, se le rette di regressione tendono ad avere un coefficiente angolare "vicino" ad 1 le due varietà possono essere ritenute equivalenti. Riportiamo di seguito i grafici dei nostri esperimenti, verrà riportato, per ogni grafico, un valore medio di convergenza per il disco e per l'iperboloide, tale valore rappresenterà un ulteriore indice di confronto.

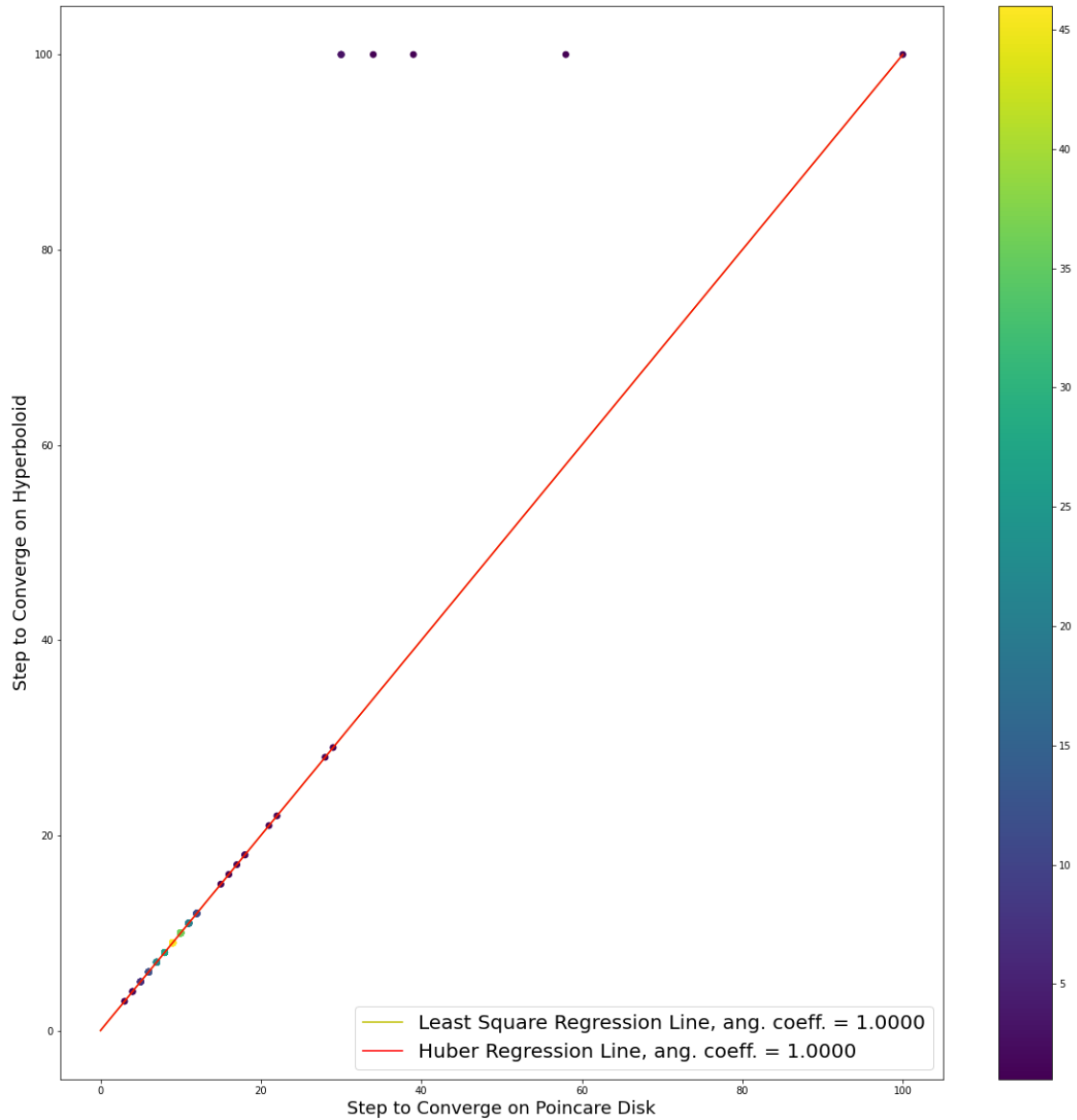


Figure 6: Nuvola di punti relativa all'algoritmo a passo fisso, notiamo che i punti generati tendono a giacere sulla bisettrice, ciò indica un'equivalenza tra le due implementazioni. Valore medio di convergenza sul disco 10.59, valore medio di convergenza su iperboloide 12.49.

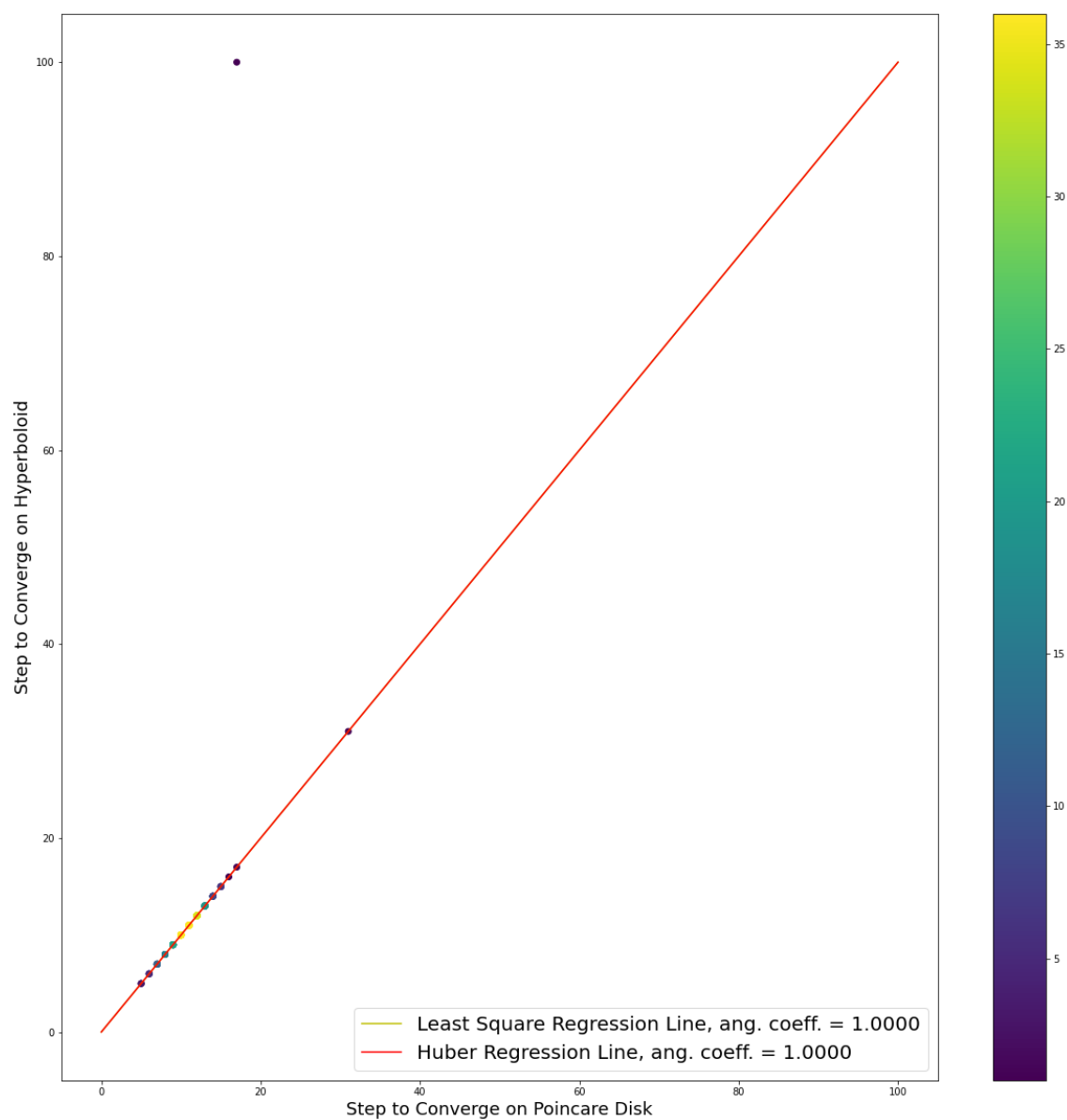


Figure 7: Nuvola di punti relativa all'algoritmo di Armijo, notiamo che i punti generati tendono a giacere sulla bisettrice, ciò indica un equivalenza tra le due implementazioni. Valore medio di convergenza sul disco 10.7, valore medio di convergenza su iperboloide 11.11.

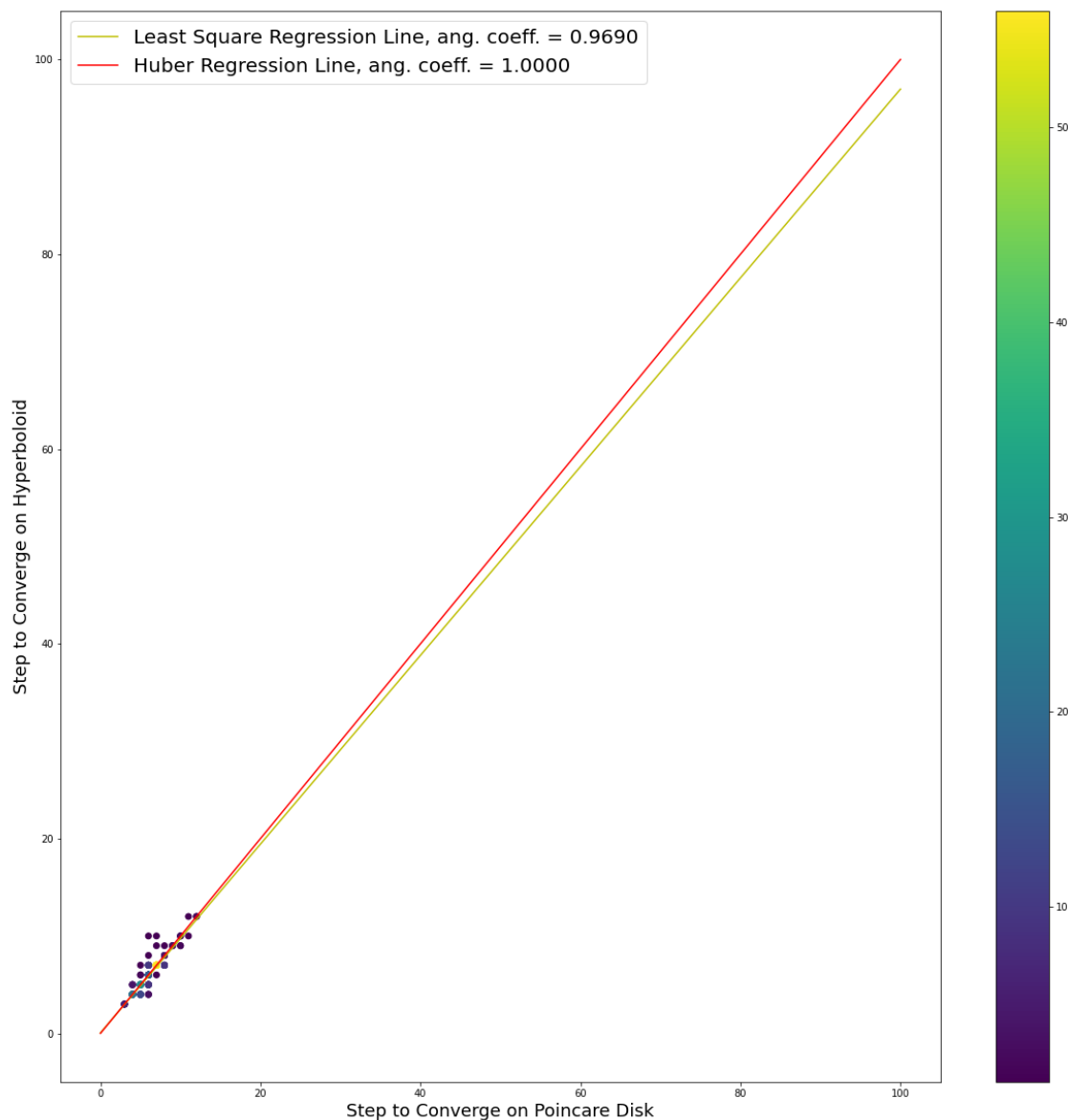


Figure 8: Nuvola di punti relativa all'algoritmo di Barzilai-Borwein, notiamo che i punti generati tendono a giacere sulla bisettrice, ciò indica un'equivalenza tra le due implementazioni. Valore medio di convergenza sul disco 6.13, valore medio di convergenza su iperboloide 6.8.

Durante le nostre sperimentazioni ci siamo resi conto che il grafico relativo all'algoritmo di Barzilai-Borwein tende ad avere i coefficienti delle rette di regressione "molto vicini" ad 1, ma non tutti i punti della nuvola tendono a giacere sulla bisettrice. Mentre i grafici relativi all'algoritmo di Armijo e l'algoritmo a passo fisso presentano una retta di regressione quasi equivalente alla bisettrice, e graficamente è facile vedere come tutti i punti, a meno di casi anomali, giacciono sulla bisettrice. Relativamente ai valori medi di convergenza, abbiamo valori abbastanza simili per tutti e tre gli algoritmi, possiamo notare una differenza di al

più un passo e mezzo di differenza nella convergenza media.

Concludendo, possiamo asserire che le implementazioni su disco e su iperboloide, degli algoritmi ottimizzazione per il calcolo della media di Fréchet, sono pressoché equivalenti, i risultati di tali sperimentazioni ci portano a congetturare che, dato un problema di ottimizzazione su una varietà è possibile trasferire il problema da tale varietà ad una varietà conforme, tenendo presente che deve essere preservata l'esistenza di una soluzione ottima per la funzione da ottimizzare.

Riportiamo il link alla repository Github contenente i codici degli esperimenti: https://github.com/Andrew-Wyn/Tesi_Ottimizzazione.

5 Appendice

5.1 Varietà

Sia M un insieme. Una bigiezione ψ di un sottoinsieme U di M in un sottoinsieme aperto di R^d è chiamata mappa d -dimensionale dell'insieme U . Data una carta (U, ψ) e $x \in U$ gli elementi di $\psi(x) \in R^d$ sono le coordinate di x nella carta (U, ψ) . Un *Atlas* (A) di M in R^d è un insieme di carte (U_a, ψ_a) dell'insieme M tali che:

- $\bigcup_a U_a = M$.
- Per ogni coppia α, β con $U_\alpha \cap U_\beta \neq \emptyset$ gli insiemi $\psi_\alpha(U_\alpha \cap U_\beta)$ e $\psi_\beta(U_\alpha \cap U_\beta)$ sono sottoinsiemi aperti di R^d ed il cambio di coordinate $\psi_\alpha \circ \psi_\beta^{-1} : R^d \rightarrow R^d$ è *smooth*.

Dato un'atlas A , sia A^+ l'insieme delle carte (U, ψ) tali che $A \cup \{(U, \psi)\}$ è ancora un'atlas. A^+ è un'atlas massimale generato da A . Un'atlas massimale dell'insieme M è chiamato strutta differenziabile su M .

Definizione 9 Una varietà d -dimensionale è una coppia (M, A^+) , dove M è un insieme e A^+ è un atlas massimale su M in R^d , tale che la topologia indotta da A^+ è di hausdorff.

Data una carta ψ su M , l'inverso ψ^{-1} è chiamata parametrizzazione locale di M .

5.1.1 Vettori Tangenti

Riprendendo il concetto di derivata direzionale possiamo generalizzarla per un funzione f definita su una varietà, sostituendo $t \mapsto (x + t\eta)$ con una curva su M attraverso $x \in M$ (punto di derivazione) ($\gamma(0) = x$) questo porta ad una derivata direzionale ben definita $\frac{\partial f(\gamma(t))}{\partial t} \big|_{t=0}$. Formalmente, sia M una varietà e γ una mappa differenziabile $\gamma : R \rightarrow M | t \mapsto \gamma(t)$ detta curva in M , data, una funzione f a valori reali definita su M , la funzione $f \circ \gamma : t \mapsto f(\gamma(t))$ è una funzione differenziabile da R in R con una ben definita derivata.

Sia $x \in M$; $\gamma(0) = x$ allora $F_x(M)$ è l'insieme delle funzioni reali differenziabili definite in un intorno di x . Il mapping $\dot{\gamma}(0)$ da $F_x(M)$ in R è definito come: $\dot{\gamma}(0)f = \frac{\partial f(\gamma(t))}{\partial t}|_{t=0}$ con $f \in F_x(M)$; è chiamato vettore tangente alla curva γ in $t = 0$.

Definizione 10 (Vettore Tangente) *Un vettore tangente ϵ_x ad una varietà M in un punto $x \in M$ è una mappa da $F_x(M)$ in R tale che esiste una curva γ a valori in M con $\gamma(0) = x$ e che soddisfa,*

$$\epsilon_x f = \dot{\gamma}(0)f = \frac{\partial f(\gamma(t))}{\partial t}|_{t=0} \quad \forall f \in F_x(M).$$

Tale curva γ è detta realizzare il vettore tangente ϵ_x

Definizione 11 (Spazio Tangente) *Uno spazio tangente ad M in x denominato come $T_x M$ è l'insieme di tutti i vettori tangenti ad M in x . Questo insieme ammette una struttura di spazio vettoriale come segue: date $\dot{\gamma}_1(0)$ e $\dot{\gamma}_2(0)$ in $T_x M$ e a, b in R allora $(a\dot{\gamma}_1(0) + b\dot{\gamma}_2(0))f = a(\dot{\gamma}_1(0)f) + b(\dot{\gamma}_2(0)f)$*

Lo spazio tangente $T_x M$ mette a disposizione un'approssimazione locale di spazio vettoriale della varietà. Perciò tramite le Retrazioni possiamo effettuare ottimizzazione sullo spazio vettoriale $T_x M$.

Definizione 12 (Tangent Bundle) *Data una varietà M , sia TM l'insieme di tutti i vettori tangenti ad M .*

$$TM = \bigcup_{x \in M} T_x M.$$

Sia $P : M \rightarrow N$ una mappa liscia tra due varietà M ed N . sia ϵ_x un vettore tangente in un punto x di M , si può far vedere che la mappa $DP(x)[\epsilon_x]$ da $F_{P(x)}(N)$ in R è definita da:

$$(DP(x)[\epsilon_x])f = \epsilon_x(f \circ P),$$

è un vettore tangente ad N in $P(x)$.

La mappa $DP(X) : T_x M \rightarrow T_{P(x)} N |_{\epsilon_x} \mapsto DP(x)[\epsilon_x]$ è una mappa lineare chiamata *differenziale* di P in x .

5.1.2 Metriche distanze e gradienti Riemanniani

I vettori tangenti ad una varietà generalizzano la nozione di derivata direzionale. Questo può essere fatto munendo lo spazio tangente $T_x M$ con un prodotto interno $\langle \cdot, \cdot \rangle_x$ bilineare, definito positivo e simmetrico. Il prodotto interno induce una norma $\| \epsilon_x \|_x = \sqrt{\langle \epsilon_x, \epsilon_x \rangle_x}$ su $T_x M$. Una varietà il cui spazio tangente è

munito di un prodotto interno prende il nome di varietà riemanniana (M, g) con g metrica riemanniana.

La lunghezza di una curva $\gamma : [a, b] \rightarrow M$ in una varietà riemanniana (M, g) è definita da,

$$L(\gamma) = \int_a^b \sqrt{g(\dot{\gamma}(t), \dot{\gamma}(t))}.$$

La distanza riemanniana in una varietà riemanniana connessa (M, g) è $dist : M \cdot M \rightarrow R | dist(x, y) \mapsto \inf_{\gamma \in \Gamma} (L(\gamma))$, dove Γ è l'insieme di tutte le curve in M che connettono x ed y .

Sia f la solita funzione liscia definita su (M, g) , il gradiente riemanniano di f in x è definito come l'elemento di $T_x M$ che soddisfa,

$$\langle \nabla^{(R)} f(x), \epsilon \rangle_x = Df(x)[\epsilon].$$

Sia M una sottovarietà immersa di una varietà riemanniana \hat{M} siccome ogni spazio tangente $T_x M$ può essere trattato come sottospazio di $T_x \hat{M}$ la metrica \hat{g} di \hat{M} induce una metrica g su M , $g_x(\epsilon, c) = \hat{g}_x(\epsilon, c)$ con $\epsilon, c \in T_x M$, perciò M è una sotto varietà riemanniana.

5.2 Topologia

Una topologia su un insieme X è una collezione T di sottoinsiemi di X chiamati insiemi aperti, tali che:

- X e \emptyset appartengono a T .
- L'unione di elementi di qualsiasi sotto collezione di T è in T .
- L'intersezione di elementi di qualsiasi sotto collezione di T è in T .

Uno spazio topologico è una coppia (X, T) dove X è un insieme e T è una topologia su X .

Definizione 13 (Hausdorff) *Un insieme X è T_2 o di Hausdorff se ogni coppia di punti distinti di X hanno un intorno distinto.*

Se X è di Hausdorff allora ogni sequenza di punti di X converge ad al più un punto di X .

5.3 Definizioni Utili

Definizione 14 (Positività) *Sia $A \in R^{n \times n}$ è definita positiva se è simmetrica e se $v'Av > 0$ se v è un vettore $\in R^n - 0$, con $v = 0$ è diretto che $v'Av = 0$.*

Definizione 15 (Convessità) Una funzione $f : \Omega \rightarrow R$ con $\Omega \subseteq V$, V (spazio vettoriale) allora è definita convessa se: $(f((\lambda)v + (1 - \lambda)w) \leq \lambda f(v) + (1 - \lambda)f(w)$, per $\lambda \in [0; 1]$ e $v, w \in \Omega$.

Definizione 16 (Derivate Parziali) Data una funzione $f : V \rightarrow R$, V (spazio vettoriale), allora una derivata parziale su $x \in V$ è definita come:

$$\frac{\partial f(x)}{\partial x_i} = \lim_{h \rightarrow 0} \frac{f(x + he_i) - f(x)}{h};$$

con x_i componente i -esima di x ed e_i i -esimo vettore della base canonica.

Se una funzione è differenziabile in x allora tutte le derivate parziali esistono in x .

Definizione 17 (Punto di accumulazione) Diciamo che $x \in R$ è un punto di accumulazione per un insieme $E \subset R$ se comunque scelto un intorno $B(x, \epsilon)$, risulta che $B(x, \epsilon)$ contiene almeno un punto di E diverso da x .

Definizione 18 (Gradiente) Consideriamo una funzione f definita su un insieme aperto $A \subset R^n$ e sia $x \in A$ se esistono in x le derivate parziali rispetto ad $\{x_i\}_{i=1 \dots n}$ allora è possibile costruire un vettore che ha per componenti le derivate parziali, perciò il gradiente è definito come.

$$\nabla f = [\frac{\partial f(x)}{\partial x_1}, \dots, \frac{\partial f(x)}{\partial x_n}].$$

Inoltre per il teorema della formula del gradiente sappiamo che, dato un versore v esiste la derivata direzionale f_v in x e che $f_v(x) = \nabla f(x) \cdot v$.

5.3.1 Teorema di Taylor

Il teorema di Taylor è un teorema che fornisce una sequenza di approssimazioni di una funzione differenziabile attorno ad un dato punto mediante i polinomi di Taylor, ovvero polinomi i cui coefficienti dipendono solo dalle derivate della funzione nel punto. Diamo un esempio pratico di approssimazione di Taylor del secondo ordine di una funzione a due variabili.

$$\begin{aligned} f(x_0 + h, y_0 + k) = & f(x_0, y_0) + f_x(x_0, y_0)h + f_y(x_0, y_0)k + \\ & \frac{1}{2}[f_{xx}(x_0, y_0)h^2 + f_{xy}(x_0, y_0)hk + f_{yy}(x_0, y_0)k^2] + \\ & R(h, k). \end{aligned}$$

con $R(h, k) \in o(\| (h, k) \|^2)$.

References

- [1] Benjamin Wilson and Matthias Leimeister. *Gradient Descent in Hyperbolic Space*.
- [2] Octavian-Eugen Ganea, Gary Becigneul and Thomas Hofmann. *Hyperbolic Entailment Cones for Learning Hierarchical Embeddings*.
- [3] Maximilian Nickel and Douwe Kiela. *Poincaré for Learning Hierarchical Representations*.
- [4] Bruno Iannazzo and Margherita Porcelli. *THE RIEMANNIAN BARZILAI-BORWEIN METHOD WITH NONMONOTONE LINE SEARCH AND THE MATRIX GEOMETRIC MEAN COMPUTATION*.
- [5] GeoOpt: Python Library for Manifold Optimization,
<https://geoopt.readthedocs.io/en/latest/>
- [6] Manopt: Python Library for Manifold Optimization,
<https://www.manopt.org/>