Alma Mater Studiorum · Università di Bologna

SCUOLA DI SCIENZE
Corso di Laurea in Matematica

Il metodo di discesa stocastica del gradiente: convergenza e applicazioni.

Tesi di Laurea in Probabilità e Statistica Matematica

Relatore:
Chiar.mo Prof.
ANDREA PASCUCCI

Presentata da:
ALESSANDRO LAZZARA

Anno Accademico 2023/2024

Introduzione

La discesa stocastica del gradiente è un metodo iterativo che permette, sotto certe ipotesi, di ottimizzare una funzione obiettivo. I primi metodi di approssimazione stocastica risalgono al 1951 con i lavori di H. Robbins e S. Monro i quali, con il loro articolo *A Stochastic Approximation Method* svilupparono un metodo per stimare in probabilità il valore atteso di particolari variabili aleatorie. L'anno successivo J. Kiefer e J. Wolfowitz pubblicarono *Stochastic Estimation of the Maximum of a Regression Function*, un algoritmo di ottimizzazione molto simile alla discesa stocastica del gradiente. Usata in principio come modello per sviluppare algoritmi di regressione, identificazione di sistemi dinamici o controllo adattivo, negli ultimi anni la discesa stocastica del gradiente è diventata de facto l'algoritmo standard, in congiunzione all'algoritmo di *Backpropagation*, per addestrare le reti neurali artificiali nell'ambito del Machine Learning. Data la popolarità del metodo con il passare del tempo sono state sviluppate molte varianti come ad esempio *RMSProp*, *Adam*, *Adagrad* o più semplicemente le versioni con aggiunta di momento o *Mini-Batch*. In questo testo analizzeremo la convergenza dell'algoritmo in primo luogo dal punto di vista deterministico, poi da quello stocastico, evidenziando le differenze tra essi sia per forma che per ipotesi utilizzate. Tra tutti i risultati che in letteratura si possono trovare sui diversi tipi di convergenza del metodo SGD, noi tratteremo il caso della convergenza quasi certa del metodo per funzioni obiettivo non convesse, avvicinandoci alla soluzione passo per passo, di modo da poter trovare, oltre al risultato fondamentale, anche altre curiose conseguenze. È infine presente un breve capitolo su come si possa applicare la discesa stocastica del gradiente per risolvere problemi di regressione lineare semplice e dei brevi esempi di implementazione su MATLAB.

Indice

Capitolo 1

Discesa del gradiente

In questa sezione analizzeremo il metodo di discesa del gradiente dal punto di vista deterministico, metodo su cui si baserà poi la variante stocastica.

1.1 Descrizione del metodo

Vogliamo minimizzare $f : \mathbb{R}^d \longrightarrow \mathbb{R}$. Richiediamo che il problema sia ben posto, ossia che $\arg\min f \neq \emptyset$ e che la funzione in analisi sia sufficientemente regolare, definiamo quindi il nostro algoritmo:

Definizione 1.1. *Sia $x^0 \in \mathbb{R}^d$ e sia $\gamma > 0$ un parametro (detto passo di discesa). L'algoritmo di **discesa del gradiente** genera una successione $(x^t)_{t\in\mathbb{N}}$ che soddisfa:*

$$x^{t+1} = x^t - \gamma \nabla f(x^t). \tag{1.1}$$

1.2 Risultati di convergenza

Premettiamo due lemmi che ci saranno utili nella dimostrazione della convergenza del metodo.

Lemma 1.2. *Sia $f : \mathbb{R}^n \longrightarrow \mathbb{R}$ differenziabile con gradiente ∇f Lipschitz di costante $L > 0$, allora per ogni $x, y \in \mathbb{R}^d$*

$$f(y) \leq f(x) + \langle \nabla f(x), y - x \rangle + \frac{L}{2}\|y - x\|^2. \tag{1.2}$$

Dimostrazione. Fissiamo $x, y \in \mathbb{R}^d$. sia $\phi(t) \coloneqq f(x + t(y - x))$. Questa è una funzione in variabili reali sulla quale è possibile applicare il teorema fondamentale del calcolo integrale, possiamo scrivere:

$$
\begin{aligned}
f(y) &= f(x) + \int_0^1 \langle \nabla f(x + t(y-x)), y - x \rangle dt \\
&= f(x) + \langle \nabla f(x), x - y \rangle + \int_0^1 \langle \nabla f(x + t(y-x)) - \nabla f(x), y - x \rangle dt \\
&\leq f(x) + \langle \nabla f(x), y - x \rangle + \int_0^1 \|\nabla f(x + t(y-x)) - \nabla f(x)\| \|y - x\| dt \\
&\leq f(x) + \langle \nabla f(x), y - x \rangle + \int_0^1 Lt\|y - x\|^2 dt \\
&\leq f(x) + \langle \nabla f(x), y - x \rangle + \frac{L}{2}\|y - x\|^2.
\end{aligned}
$$

□

Lemma 1.3. *Sia $f : \mathbb{R}^n \longrightarrow \mathbb{R}$ differenziabile con gradiente ∇f Lipschitz di costante $L > 0$ e sia inoltre $\gamma > 0$. Allora per ogni $x, y \in \mathbb{R}^d$ vale:*

$$
f(x - \gamma \nabla f(x)) - f(x) \leq -\gamma(1 - \frac{\gamma L}{2})\|\nabla f(x)\|^2. \tag{1.3}
$$

Dimostrazione. La disuguaglianza segue da (1.2) ponendo $y = x - \gamma \nabla f(x)$:

$$
\begin{aligned}
f(x - \gamma \nabla f(x)) &\leq f(x) - \gamma \langle \nabla f(x), \nabla f(x) \rangle + \frac{L}{2}\|\gamma \nabla f(x)\|^2 \\
&= f(x) - \gamma(1 - \frac{\gamma L}{2})\|\nabla f(x)\|^2.
\end{aligned}
$$

□

Il seguente teorema, tratto da [1] (*Theorem 3.4*), ci assicura la convergenza del metodo.

Teorema 1.4. *Sia $f : \mathbb{R}^d \longrightarrow \mathbb{R}$ una funzione differenziabile, convessa e il cui gradiente ∇f sia Lipschitz di costante $L > 0$. Sia $(x^t)_{t \in \mathbb{N}}$ la successione di iterate generata da (1.1), con passo di discesa $0 < \gamma \leq \frac{1}{L}$. Allora, per ogni $x^* \in \arg\min f$ e per ogni $t \in \mathbb{N}$ abbiamo che:*

$$
f(x^t) - \inf f \leq \frac{\|x^0 - x^*\|^2}{2\gamma t}. \tag{1.4}
$$

Dimostrazione. sia $x^* \in \arg\min(f)$ un punto di minimo di f. Mostriamo innanzitutto che $f(x^t)$ è una successione decrescente. Sappiamo dal lemma 1.3 che

$$
f(x^{t+1}) - f(x^t) \leq -\gamma(1 - \frac{\gamma L}{2})\|\nabla f(x^t)\|^2 \leq 0, \tag{1.5}
$$

dove la seconda disuguaglianza segue dalla nostra assunzione $0 \leq \gamma \leq \frac{1}{L}$. Mostriamo ora che $\|x^t - x^*\|^2$ decresce. Per questo partiamo da

$$\frac{1}{2\gamma}\|x^{t+1}-x^*\|^2-\frac{1}{2\gamma}\|x^t-x^*\|^2, \tag{1.6}$$

aggiungiamo e sottraiamo all' interno della seconda norma x^{t+1} ed espandiamo il quadrato ottenendo

$$\begin{aligned}
&\frac{1}{2\gamma}\|x^{t+1}-x^*\|^2-\frac{1}{2\gamma}\|x^t-x^{t+1}+x^{t+1}-x^*\|^2\\
&=\frac{1}{2\gamma}\|x^{t+1}-x^*\|^2-\frac{1}{2\gamma}\langle(x^t-x^{t+1})+(x^{t+1}-x^*),(x^t-x^{t+1})+(x^{t+1}-x^*)\rangle\\
&=-\frac{1}{\gamma}\langle x^t-x^{t+1},x^{t+1}-x^*\rangle-\frac{1}{2\gamma}\|x^{t+1}-x^t\|^2.
\end{aligned}$$

Usando ora il fatto che $x^{t+1}=x^t-\gamma\nabla f(x^t)$ si ha che

$$\begin{aligned}
&-\langle\nabla f(x^t),x^{t+1}-x^*\rangle-\frac{1}{2\gamma}\|x^{t+1}-x^t\|^2\\
&=-\langle\nabla f(x^t),x^{t+1}-x^t\rangle+\langle\nabla f(x^t),x^*-x^t\rangle-\frac{1}{2\gamma}\|x^{t+1}-x^t\|^2.
\end{aligned}\tag{1.7}$$

Vogliamo ora limitare quanto trovato: per $\langle\nabla f(x^t),x^*-x^t\rangle$ usiamo la convessità di f in congiunzione al Lemma A.2.

$$\langle\nabla f(x^t),x^*-x^t\rangle\leq f(x^*)-f(x^t)=\inf f-f(x^t).$$

Per limitare $-\langle\nabla f(x^t),x^{t+1}-x^t\rangle$ usiamo la lipschitzianità del gradiente e (1.2)

$$-\langle\nabla f(x^t),x^{t+1}-x^t\rangle\leq\frac{L}{2}\|x^{t+1}-x^t\|^2+f(x^t)-f(x^{t+1}).$$

Usando quindi le due disuguaglianze sopra possiamo quindi limitare (1.6), ottenendo:

$$\begin{aligned}
\frac{1}{2\gamma}\|x^{t+1}-x^*\|^2-\frac{1}{2\gamma}\|x^t-x^*\|^2&\leq\frac{\gamma L-1}{2\gamma}\|x^{t+1}-x^t\|^2-(f(x^{t+1})-\inf f),\\
&\leq-(f(x^{t+1})-\inf f).
\end{aligned}\tag{1.8}$$

In particolare qui abbiamo mostrato, essendo $f(x^{t+1})\geq\inf f$ che la successione $\|x^t-x^*\|$ è decrescente. Per trovare la stima della tesi introduciamo la seguente funzione:

$$E_t:=\frac{1}{2\gamma}\|x^t-x^*\|^2+t(f(x^t)-\inf f).$$

Mostriamo ora che E_t è decrescente in $t\in\mathbb{N}$

$$\begin{aligned} E_{t+1} - E_t = \quad & (t+1)(f(x^{t+1}) - \inf f) - t(f(x^t) - \inf f) + \frac{1}{2\gamma}\|x^{t+1} - x^*\|^2 - \frac{1}{2\gamma}\|x^t - x^*\|^2 \\ = \quad & f(x^{t+1}) - \inf f + t(f(x^{t+1}) - f(x^t)) + \frac{1}{2\gamma}\|x^{t+1} - x^*\|^2 - \frac{1}{2\gamma}\|x^t - x^*\|^2. \end{aligned} \tag{1.9}$$

Usando ora (1.9), (1.5), (1.8) otteniamo

$$\begin{aligned} E_{t+1} - E_t &\leq f(x^{t+1}) - \inf f + \frac{1}{2\gamma}\|x^{t+1} - x^*\|^2 - \frac{1}{2\gamma}\|x^t - x^*\|^2 \\ &\quad \text{(usando (1.5))} \\ &\leq f(x^{t+1}) - \inf f - (f(x^{t+1}) - \inf f) \\ &\quad \text{(usando (1.8))} \\ &= 0. \end{aligned}$$

Segue che E_t è decrescente. Possiamo quindi scrivere che

$$t(f(x^t) - \inf f) \leq E_t \leq E_0 = \frac{1}{2\gamma}\|x^0 - x^*\|^2.$$

La tesi si ottiene dopo aver diviso per t.

□

Capitolo 2

Discesa stocastica del gradiente

2.1 Descrizione del metodo

Siamo interessati a minimizzare una funzione $f : \mathbb{R}^d \longrightarrow \mathbb{R}$ detta *funzione obiettivo*, che supporremo essere differenziabile. Il nostro punto di partenza è la discesa del gradiente deterministica: partiamo da una successione

$$x^{t+1} = x^t - \gamma_t \nabla f(x^t),$$

dove $\gamma_t \in \mathbb{R}$ è a priori variabile e $x^t \in \mathbb{R}^d$. In generale nelle applicazioni è molto difficile (se non impossibile) avere a disposizione il valore esatto di $\nabla f(x^t)$, introduciamo quindi una funzione $g(x, \xi)$, detta **gradiente stocastico**, la quale dipende da $x \in \mathbb{R}^d$ e da una variabile aleatoria ξ. Il tutto ha la seguente proprietà:

$$\mathbb{E}_\xi[g(x, \xi)] = \nabla f(x),$$

ciò ci dice che $g(x, \xi)$ è un *unbiased estimator* (in italiano *stimatore non distorto*) del gradiente $\nabla f(x)$. Sostituendo quindi $g(x, \xi)$ a $\nabla f(x)$ all'interno del nostro algoritmo abbiamo il metodo di **discesa stocastica del gradiente** (brevemente SGD)

$$x^{t+1} = x^t - \gamma_t g(x^t, \xi^t).$$

L'introduzione della stocasticità all'interno dell'algoritmo è un arma a doppio taglio: da un lato abbiamo un' efficienza migliore, soprattutto nel caso multidimensionale, siccome, in generale, sarà possibile computare $g(x^t, \xi^t)$ molto più velocemente rispetto $\nabla f(x^t)$, dall' altro è però vero che la nostra stima del gradiente possa essere molto rumorosa (vedremo in seguito cosa ciò vuol dire) e non dare un risultato accettabile in termini di accuratezza.

Per comprendere meglio cosa può voler significare $g(x, \xi)$ facciamo un esempio abbastanza intuitivo:

Esempio 2.1. Supponiamo che, invece di avere a disposizione $\nabla f(x)$, di avere accesso a una misurazione del gradiente viziata da un errore di tipo additivo, che chiameremo rumore, ossia:

$$g(x, \xi) = \nabla f(x) + \xi,$$

dove $\mathbb{E}[\xi] = 0$ così da avere $\mathbb{E}_\xi[g(x, \xi)] = \nabla f(x)$. Anche in questo semplice caso possiamo osservare che il rumore, sebbene abbia media nulla, possa influenzare anche in maniera importante il comportamento dell'algoritmo passo per passo.

2.2 SGD per somme di funzioni

In ambito applicativo capita spesso di voler minimizzare funzioni del tipo

$$f(x) \coloneqq \frac{1}{n} \sum_{i=1}^{n} f_i(x), \tag{2.1}$$

dove $f_i : \mathbb{R}^d \longrightarrow \mathbb{R}$. Sotto le giuste ipotesi (come ad esempio quelle del Teorema 1.4) si potrebbe applicare il metodo di discesa del gradiente classico generando una successione

$$x^{t+1} = x^t - \gamma_t \frac{1}{n} \sum_{i=1}^{n} \nabla f_i(x^t).$$

Occorre ora notare che, nel caso in cui n sia molto grande, l'esecuzione di ogni passo di computazione può essere molto onerosa sia in termini di tempo che di spazio, entra qui in gioco la discesa stocastica del gradiente: ad ogni passo scegliamo un indice $i_t \in \{1, \dots, n\}$ con probabilità uniforme e usiamo solo il gradiente ∇f_{i_t} all'interno dell'algoritmo nel seguente modo:

$$x^{t+1} = x^t - \gamma_t \nabla f_{i_t}(x^t).$$

La distribuzione uniforme con cui scegliamo i_t ci permette di avere così un *unbiased estimator* per $\nabla f(x)$, infatti $\mathbb{E}[\nabla f_i(x)] = \sum_{i=1}^{n} \frac{1}{n} \nabla f_i(x) = \nabla f(x)$. È bene osservare che in questo caso abbiamo poco controllo sulla varianza, che sarà

$$\mathbb{V}[\nabla f_i(x)] = \mathbb{E}[\nabla f_i(x)^2] - \nabla f(x)^2 = \frac{1}{n} \sum_{i=1}^{n} \nabla f_i(x)^2 - \nabla f(x)^2.$$

Sfruttiamo questo spunto per accennare a una variante del nostro algoritmo **Mini-batch**, ossia dove, al posto di usare un solo indice i_t, usiamo un sottoinsieme $I_t \subset \{1, \dots, n\}$ di cardinalità fissata $|I_t| = m$. L'algoritmo avrà forma

$$x^{t+1} = x^t - \gamma_t \frac{1}{m} \sum_{i \in I_t} \nabla f_i(x^t).$$

L'utilità di questa variante sta proprio nel fatto che si può provare che essa abbia una varianza minore, a scapito del maggior costo computazionale dato dal dover calcolare più gradienti.

Capitolo 3

Convergenza per funzioni non convesse

3.1 Ipotesi

Ora che abbiamo descritto il metodo ci chiediamo quando e come questo converga. I risultati che possiamo ottenere dipendono, come ben intuibile, dalle qualità della funzione obiettivo. Le ipotesi standard richiedono quasi sempre la lipschitzianità, la convessità o entrambe. Tra gli innumerevoli risultati che si possono trovare, noi ci concentriamo su un caso abbastanza generale, ossia la convergenza del metodo nel caso non convesso. Il motivo è anche quello di evidenziare la capacità di convergenza dell'algoritmo in un caso dove non è detto che la discesa deterministica del gradiente possa convergere. Cominciamo quindi a introdurre le nostre ipotesi:

Ipotesi 1. $f : \mathbb{R}^d \longrightarrow \mathbb{R}$ *è una funzione differenziabile, limitata dal basso, con gradiente Lipschitz di costante* L.

La nostra prossima ipotesi è richiedere che la varianza del gradiente stocastico $g(x, \xi)$ sia limitata dall'alto.

Ipotesi 2. $\mathbb{V}[g(x, \xi)] = \sigma^2 < \infty$.

Osservazione 3.1. Abbiamo come conseguenza immediata che anche la varianza condizionata è limitata dall'alto: $\mathbb{E}_\xi[\|\nabla f(x) - g(x, \xi)\|^2] \leq \sigma^2 < \infty$.

La lipschitzianità del gradiente che abbiamo appena richiesto sarà cruciale nella dimostrazione della convergenza del metodo, difatti ciò ci assicura che in corrispondenza di un punto di minimo il gradiente tenderà a zero. Viene quindi naturale porsi la seguente domanda: *Riuscirà* $\|\nabla f(x^t)\|$ *a convergere a zero con probabilità 1 quando* t *va all'infinito?*

3.2 Convergenza *average iterate*

Cominciamo ora a rispondere alla domanda che ci siamo posti alla fine della precedente sezione. Proseguiremo gradualmente come fatto in [3], ossia cominciando dalle proprietà di f e cercando di trovare dei risultati man mano più soddisfacenti. Ricordiamo che f, avendo gradiente Lipschitz, gode della proprietà 1.2, quindi:

$$\begin{aligned} f(x^{t+1}) &\leq f(x^t) + \langle \nabla f(x^t), x^{t+1} - x^t \rangle + \frac{L}{2}\|x^{t+1} - x^t\|^2 \\ &= f(x^t) - \gamma_t \langle \nabla f(x^t), g(x^t, \xi^t) \rangle + \frac{\gamma_t^2 L}{2}\|g(x^t, \xi^t)\|^2. \end{aligned}$$

Prendiamo ora il valore atteso condizionato rispetto ξ^t dato x^t, che indicheremo con $\mathbb{E}_t$:

$$\begin{aligned} \mathbb{E}_t[f(x^{t+1})] &\leq f(x^t) - \gamma_t \|\nabla f(x^t)\|^2 + \frac{\gamma_t^2 L}{2}\mathbb{E}_t[\|g(x^t, \xi^t)\|^2] \\ &= f(x^t) - \gamma_t \|\nabla f(x^t)\|^2 + \frac{\gamma_t^2 L}{2}\mathbb{E}_t[\|\nabla f(x^t) - g(x^t, \xi^t) - \nabla f(x^t)\|^2] \\ &= f(x^t) - \gamma_t \|\nabla f(x^t)\|^2 + \frac{\gamma_t^2 L}{2}(\mathbb{E}_t[\|\nabla f(x^t) - g(x^t, \xi^t)\|^2] + \|\nabla f(x^t)\|^2) \\ &\leq f(x^t) - (\gamma_t - \frac{\gamma_t^2 L}{2})\|\nabla f(x^t)\|^2 + \frac{\gamma_t^2 L}{2}\sigma^2, \end{aligned}$$

dove nell'ultima disuguaglianza abbiamo sfruttato il fatto che la varianza del gradiente stocastico sia limitata da σ^2. Prendendo ora il valore atteso non condizionato, riordinando i termini e sommando su t da 1 fino ad un T fissato otteniamo

$$\begin{aligned} \sum_{t=1}^{T}(\gamma_t - \frac{\gamma_t^2 L}{2})\mathbb{E}[\|\nabla f(x^t)\|^2] &\leq \sum_{t=1}^{T}(\mathbb{E}[f(x^t)] - \mathbb{E}[f(x^{t+1})]) + \frac{\sigma^2 L}{2}\sum_{t=1}^{T}\gamma_t^2 \\ &= f(x^1) - \mathbb{E}[f(x^{t+1})] + \frac{\sigma^2 L}{2}\sum_{t=1}^{T}\gamma_t^2 \\ &\leq f(x^1) - f^* + \frac{\sigma^2 L}{2}\sum_{t=1}^{T}\gamma_t^2, \end{aligned} \tag{3.1}$$

dove f^* indica il minimo di f. Scegliamo ora il passo di discesa costante $\gamma_t = \min(\frac{1}{L}, \frac{\alpha}{\sigma\sqrt{T}})$, qui α indica un parametro a cui noi generalmente non abbiamo accesso completamente (esso dipende da $f(x^1) - f^*$ e per ragioni ovvie non possiamo conoscere il minimo a

priori). Con questa scelta abbiamo $\gamma_t - \frac{\gamma_t^2 L}{2} \geq \gamma_t - \frac{\gamma_t}{2} = \frac{1}{2}\gamma_t$, quindi

$$\begin{aligned}\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}[\|\nabla f(x^t)\|^2] &\leq \frac{2(f(x^1)-f^*)}{\gamma_1 T} + \sigma^2 L\gamma_1 \\ &= \frac{2(f(x^1)-f^*)}{T}\max(L, \frac{\sigma\sqrt{T}}{\alpha}) + \sigma L\alpha\frac{1}{\sqrt{T}} \\ &\leq \frac{2(f(x^1)-f^*)}{T}(L + \frac{\sigma\sqrt{T}}{\alpha}) + \sigma L\alpha\frac{1}{\sqrt{T}} \\ &= \frac{2L(f(x^1)-f^*)}{T} + (\frac{2(f(x^1)-f^*)}{\alpha} + L\alpha)\frac{\sigma}{\sqrt{T}}.\end{aligned}$$

Quanto trovato mostra la convergenza non dell'ultima iterata ma della media delle norme dei gradienti: infatti questa quantità va a zero come un $O(\frac{1}{T} + \frac{1}{\sqrt{T}})$, in letteratura capita di incontrare questo tipo di convergenza con il nome di *average iterate convergence*.

Osservazione 3.2. Notiamo che la velocità di convergenza ha due termini: uno veloce di ordine $\frac{1}{T}$ e uno più lento di ordine $\frac{\sigma}{\sqrt{T}}$. Osserviamo che il termine lento è influenzato negativamente dalla varianza del gradiente stocastico: più questo rumore sarà alto, più lentamente il nostro algoritmo convergerà.

Dato che la media di un insieme di numeri è sempre maggiore o uguale al minimo, sappiamo che esiste un' iterata x^i tra $x^1, \ldots, x^T$ il cui gradiente è piccolo in valore atteso. Questo, sebbene interessante, è poco utile all' atto pratico perché non sappiamo effettivamente chi sia questa x^t. A tal proposito esiste una procedura molto particolare, chiamata **Randomly stopped SGD**, che riesce ad aggirare il problema di non sapere a priori quale sia l'iterata corretta. Vediamo come funziona: date le nostre solite iterate $x^1, \ldots, x^T$ scegliamo un' indice I da un distribuzione uniforme su $1, \ldots, T$. Prendendo il valore atteso rispetto questa distribuzione e rispetto il rumore dei gradienti stocastici abbiamo:

$$\mathbb{E}[\|\nabla f(x^I)\|^2] = \frac{1}{T}\sum_{t=1}^{T}\mathbb{E}[\|\nabla f(x^t)\|^2] \leq \frac{2L(f(x^1)-f^*)}{T} + (\frac{2(f(x^1)-f^*)}{\alpha} + L\alpha)\frac{\sigma}{\sqrt{T}}. \tag{3.2}$$

Praticamente se facciamo andare l'algoritmo per T iterazioni e alla fine, al posto di ritornare l'ultima iterata, ne torniamo una casuale, il valore atteso della norma sarà piccolo. Osserviamo che nella pratica si può ottenere questo risultato modificando leggermente l'algoritmo inserendo un tempo di fermata I preso da una distribuzione uniforme di T numeri. Vediamo che anche in questo caso non abbiamo una risposta alla nostra domanda iniziale perché abbiamo effettivamente cambiato target non considerando più l'ultima iterata.

3.3 Convergenza quasi certa

Torniamo a 3.1. Questa volta, al posto di scegliere un passo di discesa costante, scegliamo γ_t di modo che:

$$\sum_{t=1}^{\infty} \gamma_t = \infty \qquad \text{e} \qquad \sum_{t=1}^{\infty} \gamma_t^2 < \infty. \tag{3.3}$$

Queste condizioni, talvolta dette condizioni **Robbins-Monro**, sono ormai standard nello studio dell'approssimazione stocastica. La prima ci permette di allontanarci arbitrariamente lontano dall'iterata iniziale, mentre la seconda è necessaria per controllare il rumore della varianza. Un passo di discesa come $\gamma_t = \frac{\alpha}{\sqrt{t}}$ non soddisfa queste ipotesi, ma un qualcosa che decresce un po' più velocemente come $\gamma_t = \frac{\alpha}{\sqrt{t}(\ln(t+1))}$ farà al caso nostro. Con una tale scelta otteniamo

$$\sum_{t=1}^{\infty} (\gamma_t - \frac{\gamma_t^2 L}{2}) \mathbb{E}[\|\nabla f(x^t)\|^2] \leq f(x^1) - f^* + \frac{\sigma^2 L}{2} \sum_{t=1}^{\infty} \gamma_t^2 \ < \infty.$$

La seconda condizione, oltre a rendere possibile la disuguaglianza sopra, ci assicura anche che γ_t tenda a zero. Sapendo questo, possiamo dire che esiste un certo $\bar{t}$ positivo tale che per ogni $t \geq \bar{t}$:

$$\frac{\gamma_t^2 L}{2} \leq \frac{\gamma_t}{2}.$$

Vale quindi, dopo aver cambiato segno e sommato γ_t

$$\frac{\gamma_t}{2} \leq \gamma_t - \frac{\gamma_t^2 L}{2}.$$

Unendo quindi tutto otteniamo

$$\sum_{t=\bar{t}}^{\infty} \gamma_t \mathbb{E}[\|\nabla f(x^t)\|^2] < \infty,$$

e ciò implica infine

$$\sum_{t=\bar{t}}^{\infty} \gamma_t \|\nabla f(x^t)\|^2 < \infty \qquad \text{q.c.}$$

Vogliamo ora acquisire informazioni sul comportamento di $\|\nabla f(x^t)\|$. Noi sappiamo che $\sum_{t=\bar{t}}^{\infty} \gamma_t = \infty$; se per assurdo esistesse un $\epsilon > 0$ e un $t' > 0$ tali che $\|\nabla f(x^t)\| \geq \epsilon$ per ogni $t \geq t'$, allora avremmo:

$$\sum_{t=\bar{t}}^{\infty} \gamma_t \|\nabla f(x^t)\|^2 \geq \epsilon^2 \sum_{t=\bar{t}}^{\infty} \gamma_t = \infty,$$

che è assurdo. Ciò ci assicura però solo l'esistenza di una sottosuccessione di $\|\nabla f(x^t)\|$ che converge a zero, da questo possiamo dedurre che $\liminf_{t\to\infty} \|\nabla f(x^t)\| = 0$. Abbiamo

provato qualcosa di più debole di ciò che volevamo, per riuscire a ottenere un risultato più forte e che possa rispondere finalmente alla nostra domanda proseguiamo per questa strada e proviamo che anche $\limsup_{t\to\infty} \|\nabla f(x^t)\| = 0$. Per fare ciò proviamo il seguente lemma leggermente più generale.

Lemma 3.3. *Siano $(b_t)_{t\geq 1}, (\gamma_t)_{t\geq 1}$ due successioni non negative e sia $(a_t)_{t\geq 1}$ una successione di vettori in uno spazio vettoriale X. Sia $p \geq 1$ e assumiamo che $\sum_{t=1}^{\infty} \gamma_t b_t^p < \infty$ e $\sum_{t=1}^{\infty} \gamma_t = \infty$. Assumiamo anche che che esista $L \geq 0$ tale che: $|b_{t+\tau} - b_t| \leq L(\sum_{i=t}^{t+\tau-1} \gamma_i b_i + \| \sum_{i=t}^{t+\tau-1} \gamma_i a_i\|)$, dove a_t è tale che $\| \sum_{i=1}^{\infty} \gamma_i a_i\| < \infty$. Allora b_t converge a zero.*

Dimostrazione. Con ragionamenti analoghi a quanto fatto prima possiamo osservare che $\liminf_{t\to\infty} b_t = 0$ siccome $\sum_{t=1}^{\infty} \gamma_t b_t^p < \infty$ e $\sum_{t=1}^{\infty} \gamma_t = \infty$. Dobbiamo quindi provare che $\limsup_{t\to\infty} b_t = 0$. Procediamo per assurdo assumendo che $\limsup_{t\to\infty} b_t = \lambda > 0$. Supponiamo $\lambda < \infty$. Dati i valori del lim inf e lim sup, possiamo costruire due successioni di indici $(m_j)_{j\geq 1}$ e $(n_j)_{j\geq 1}$ tali che:

- $m_j < n_j < m_{j+1}$,
- $\frac{\lambda}{3} < b_k$, per $m_j \leq k < n_j$,
- $b_k \leq \frac{\lambda}{3}$ per $n_j \leq k < m_{j+1}$.

Sia $\epsilon = \frac{\lambda}{6L} \min(\frac{\lambda^{p-1}}{3^{p-1}}, 1)$. La convergenza della serie implica che la successione delle somme parziali sia di Cauchy. Quindi, esiste un $\tilde{j}$ abbastanza grande tale che per ogni $N \geq m_{\tilde{j}}$ abbiamo $\sum_{t=m_{\tilde{j}}}^{N} \gamma_t b_t^p \leq \epsilon$ e $\| \sum_{t=m_{\tilde{j}}}^{N} \gamma_t a_t\| \leq \epsilon$. Quindi, per ogni $j \geq \tilde{j}$ e per ogni $m_j \leq m \leq n_j - 1$,

$$\begin{aligned} |b_{n_j} - b_m| &\leq L(\sum_{i=m}^{n_j-1} \gamma_i b_i + \| \sum_{i=m}^{n_j-1} \gamma_i a_i\|) \\ &= L\frac{3^{p-1}}{\lambda^{p-1}}(\sum_{i=m}^{n_j-1} \gamma_i b_i \frac{\lambda^{p-1}}{3^{p-1}}) + L\| \sum_{i=m}^{n_j-1} \gamma_i a_i\| \\ &\leq L\frac{3^{p-1}}{\lambda^{p-1}} \sum_{i=m}^{n_j-1} \gamma_i b_i^p + L\| \sum_{i=m}^{n_j-1} \gamma_i a_i\| \leq \frac{\lambda}{3}. \end{aligned}$$

Usando allora la disuguaglianza triangolare vediamo che $b_m \leq b_{n_j} + \frac{\lambda}{3} \leq \frac{2\lambda}{3}$. Otteniamo infine che per ogni $m \geq \tilde{j}$ abbiamo $b_m \leq \frac{2\lambda}{3}$, ciò è assurdo perché contraddice il fatto che $\limsup_{t\to\infty} b_t = \lambda > 0$, quindi b_t va a zero. In maniera simile si dimostra il caso $\limsup_{t\to\infty} b_t = \infty$, prendendo ogni $\lambda > 0$ e ottenendo un risultato analogo che contraddice l'ipotesi.

□

Dopo aver visto questo lemma siamo quindi pronti a vedere il teorema principale. La forma del teorema è analoga al *Theorem 2* in [3]. Per una dimostrazione simile ma con ipotesi leggermente diverse segnaliamo la dimostrazione in [2].

Teorema 3.4. *Sia f funzione che soddisfi le ipotesi 1, 2, supponiamo di usare con essa l'algoritmo SGD con passi di discesa $\gamma_t > 0$ che soddisfino 3.3. Allora $\|\nabla f(x^t)\|$ tende a zero quasi certamente.*

Dimostrazione. Vogliamo usare il lemma 3.3 con $b_t = \|\nabla f(x^t)\|$. Osserviamo che per l'ipotesi 1 abbiamo:

$$\begin{aligned}
|\|\nabla f(x^{t+\tau})\| - \|\nabla f(x^t)\|| &\leq \|\nabla f(x^{t+\tau}) - \nabla f(x^t)\| \\
&\leq L\|x^{t+\tau} - x^t\| \\
&= L\| \sum_{i=t}^{t+\tau-1} \gamma_i g(x^i, \xi^i)\| \\
&= L\| \sum_{i=t}^{t+\tau-1} \gamma_i (\nabla f(x^i) + g(x^i, \xi^i) - \nabla f(x^i))\| \\
&\leq L \sum_{i=t}^{t+\tau-1} \gamma_i \|\nabla f(x^i)\| + L\| \sum_{i=t}^{t+\tau-1} \gamma_i (g(x^i, \xi^i) - \nabla f(x^i))\|.
\end{aligned}$$

Le nostre ipotesi, unite a quanto fatto, ci assicurano quasi certamente che $\sum_{t=1}^{\infty} \gamma_t \|\nabla f(x^t)\|^2 < \infty$, ciò ci suggerisce anche di porre $a_t = g(x^t, \xi^t) - \nabla f(x^t)$. Vogliamo adesso provare che $\| \sum_{t=1}^{\infty} \gamma_t a_t\| < \infty$. Osserviamo che il processo $\sum_{t=1}^{T} \gamma_t a_t$ è una martingala a media nulla, infatti:

- $\mathbb{E}[\sum_{t=1}^{T} \gamma_t a_t] = \sum_{t=1}^{T} \gamma_t \mathbb{E}[g(x^t, \xi^t) - \nabla f(x^t)] = 0$. (Media nulla)
- $\mathbb{E}[| \sum_{t=1}^{T} \gamma_t a_t|] \leq \sum_{t=1}^{T} \gamma_t \mathbb{E}[|a_t|] \leq \sigma^2 \sum_{t=1}^{\infty} \gamma_t < \infty$. (somabilità)
- $\mathbb{E}_t[\sum_{t=1}^{T+1} \gamma_t a_t] = \sum_{t=1}^{T} \gamma_t (g(x^t, \xi^t) - \nabla f(x^t)) + \gamma_t \mathbb{E}_t[g(x^{t+1}, \xi^{t+1}) - \nabla f(x^{t+1})]$
  $= \sum_{t=1}^{T} \gamma_t a_t + 0$. (Martingalità)

Si può infine provare che la varianza di $\sum_{t=1}^{T} \gamma_t a_t$ sia limitata da $\sigma^2 \sum_{t=1}^{\infty} \gamma_t$ per ogni T. Quindi per il teorema di convergenza delle martingale (Teorema A.6) il processo converge quasi certamente, allora $\| \sum_{t=1}^{\infty} \gamma_t a_t\| < \infty$. Siamo allora nelle ipotesi del lemma, possiamo finalmente dire che i gradienti delle iterate dell'algoritmo vanno a zero, ossia che per ogni $\epsilon > 0$ esiste un N_ϵ tale che $\|\nabla f(x^t)\| \leq \epsilon$ per ogni $t \geq N_\epsilon$

□

Capitolo 4

Applicazioni: regressione lineare con SGD

In ambito statistico capita spesso di chiedersi se esista qualche tipo di relazione tra le dinamiche di due o più fenomeni nel tempo. La regressione lineare è una tecnica che ci permette di trovare, se esiste, una relazione di tipo lineare tra due campioni di dati. Tratteremo il caso in cui i dati sono uno-dimensionali: prendiamo quindi un vettore $x = (x_1, \dots, x_M)$ di $\mathbb{R}^M$, che chiameremo *serie storica*. Possiamo vedere x come la realizzazione di una variabile aleatoria discreta X così definita:

$$X : I_M \longrightarrow \mathbb{R}, \qquad X(i) \coloneqq x_i, \quad i \in I_M.$$

Dove I_M, insieme degli M indici, è dotato di probabilità uniforme. Segue quindi che media e varianza sono:

$$\mathbb{E}[X] = \frac{1}{M}\sum_{i=1}^{M} x_i, \quad \mathbb{V}[X] = \frac{1}{M}\sum_{i=1}^{M}(x_i - \mathbb{E}[X])^2.$$

Prendiamo ora due serie storiche $x = (x_1, \dots, x_M)$ e $y = (y_1, \dots, y_M)$, un modo per visualizzare come esse possano essere correlate è il *grafico di dispersione*, ossia un piano cartesiano sul quale si rappresentano i punti $(x_i, y_i)_{i \in I_M}$. In questo grafico possiamo tracciare la *retta di regressione*, ossia la retta di equazione $y = ax + b$ dove a e b minimizzano l'errore quadratico tra $ax_i + b$ e y_i, ossia rendono minimo:

$$Q(a, b) = \sum_{i=1}^{M}(ax_i + b - y_i)^2.$$

Per trovare il minimo occorre annullare il gradiente

$$\nabla Q(a, b) = (2\sum_{i=1}^{M}(ax_i + b - y_i)x_i, 2\sum_{i=1}^{M}(ax_i + b - y_i))$$

e poi risolvere il sistema. Osserviamo che in questo caso abbiamo effettivamente un metodo per calcolare esattamente a e b, noi vogliamo però prendere un' altra strada. Osserviamo che $Q(a,b)$ è differenziabile, guardando la derivata seconda si può osservare che il gradiente è Lipschitz e in più $Q(a,b) \geq 0$ per cui la nostra funzione è anche limitata dal basso. Vogliamo adesso trovare un gradiente stocastico: esso dipenderà da a, b e dalla scelta da una distribuzione uniforme dell'indice i. Poniamo $g(a,b,i) \coloneqq (2M(ax_i+b-y_i)x_i, 2M(ax_i+b-y_i))$, così facendo abbiamo un *unbiased estimator* del gradiente e siamo quindi nelle condizioni per poter usare l'algoritmo di discesa stocastica del gradiente.

4.1 Implementazione e costi

Di seguito presentiamo un'implementazione MATLAB dell'algoritmo SGD:

```
%% Algoritmo SGD
function ab=sgd(x,y,ab0,m)

% il passo di discesa variabile viene definito all'interno
   dell' algoritmo
% x,y:serie storiche
% ab0:iterata iniziale
% m:massimo numero di iterazioni

n=length(x);
ab=ab0;
for i=1:m
    k=randi(n,1);
    temp=ab(1)*x(k)+ab(2)-y(k);
    ab=ab-(1/i)*[temp*x(k),temp];
    % avrei un 2n al numeratore ma il passo di discesa viene
        definito come 1/(2ni) per semplificarlo
end
```

Facciamo qualche considerazione dal punto di vista computazionale: osserviamo che all'aumentare dei dati non abbiamo necessariamente un aumento del numero di operazioni da svolgere, essendo che questo dipende dal numero di iterazioni che vogliamo eseguire. Per iterazione il nostro algoritmo esegue circa 10 operazioni, supponendo di eseguire sempre il massimo numero di iterazioni avremo quindi un costo totale di

$10m$ operazioni. Proviamo a confrontare ciò con il costo della discesa del gradiente deterministica: l'unica cosa che cambia è il calcolo del gradiente piuttosto che del gradiente stocastico; abbiamo inoltre bisogno di calcolarlo ad ogni iterazione, quindi, a meno di costanti che saranno cancellate nell' algoritmo abbiamo bisogno di computare $(\sum_{i=1}^{n}(ax_i + b - y_i)x_i, \sum_{i=1}^{n}(ax_i + b - y_i))$. In termini di operazioni $ax_i + b - y_i$ costa 3 e viene memorizzato, va poi moltiplicato per x_i e anche questo memorizzato, di questi ne dobbiamo calcolare n e poi sommarli. Alla fine calcolare il gradiente costa circa $6n$ operazioni per iterazione e il costo principale sarà dato da questo per il numero massimo di iterazioni, quindi circa $6nm$. È immediato osservare come per grandi campioni di dati la discesa stocastica del gradiente surclassi in termini di costo computazionale la discesa deterministica.

4.2 Esempi e limiti della regressione lineare

Siamo adesso pronti a mettere in pratica ciò che abbiamo mostrato. Presentiamo tre esempi di regressione lineare in ordine decrescente di efficacia. Tale scelta è motivata da due fattori: in primo luogo ciò ci permetterà di osservare realmente l'efficacia dell'algoritmo, in secondo luogo, facendo vedere anche i casi sfortunati, potremo esporre le difficoltà che la regressione lineare ha nello stimare particolari campioni di dati. Useremo lo stesso algoritmo presentato nella precedente sezione.

Per ogni esempio useremo il seguente template dove, al variare della situazione, cambieranno soltanto i valori delle serie storiche; inoltre potremmo aggiungere un rumore σ di tipo additivo per evitare eventuali risultati banali.

```
%creo i dati
n=10000; %dimensione delle serie storiche

% x e y da definire per ogni esempio

ves=reges(x,y); %calcola i coefficienti a,b tramite la
   risoluzione del sistema lineare

vsgd=sgd(x,y,[0,0],10000); %usa SGD con iterate iniziali
   (0,0) e 10000 iterazioni
```

```
%dati per creare il grafico,
plot(x,ves(1)*x+ves(2),'b:',"LineWidth",1);
hold on
plot(x,vsgd(1)*x+vsgd(2),'r');
plot(x,y,'.',"MarkerSize",0.1);
legend("regressione esatta","regressione SGD","dati")
```

Esempio 4.1. Qui simuliamo il caso in cui esista un' effettiva correlazione lineare tra le serie storiche, per farlo prendiamo x come vettore i cui elementi siano estratti da una distribuzione normale standard e y come $ax + b + \sigma$ dove a e b sono incogniti e σ è il nostro rumore. I dati da inserire nel'algoritmo saranno:

```
x=randn(1,n);
y=randi(10)*rand(1)*x+randi(10)*rand(1)+randn(1,n);
```

Sfruttiamo questo esempio per vedere il costo computazionale dell' algoritmo e il tempo di esecuzione, comparandolo a quello del metodo deterministico (un' implementazione della discesa del gradiente deterministica per la regressione lineare è presente in Appendice B).

Costo Computazionale	
Deterministico	Stocastico
$6nm$	$10m$

Tabella 4.1: Costo computazionale degli algoritmi GD e SGD; n è la dimensione dei dati, m il numero di iterazioni.

Tempo di esecuzione (S)		
Dimensione dei dati	Deterministico	Stocastico
10^4	0.608423	0.002239
10^5	6.01638	0.001814
10^6	60.577478	0.001190

Tabella 4.2: Tempo di esecuzione degli algoritmi SGD e GD al variare della dimensione dei dati; il numero massimo di iterazioni è fissato a 1000.

In accordo con quanto scritto nella sezione precedente e nella Tabella 4.1, osserviamo dalla Tabella 4.2 come il tempo di esecuzione dell'algoritmo stocastico rimanga pressoché

costante all'aumentare della dimensione dei dati, ciò perché il numero di operazioni svolte non dipende da n. Il grafico 4.1 confronta graficamente la retta di regressione e quella trovata dall'algoritmo SGD.

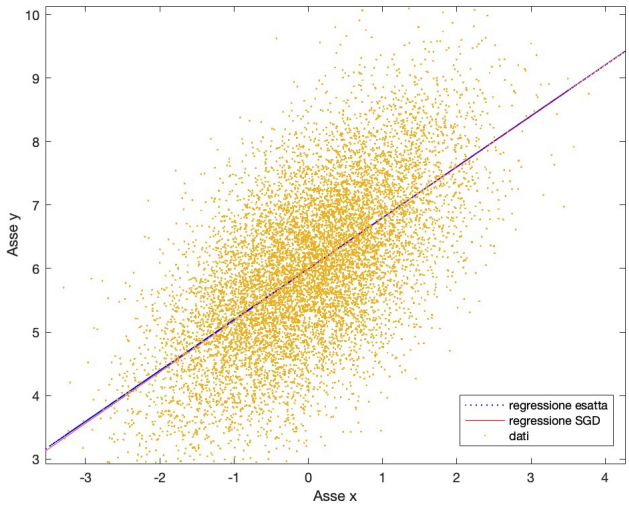

Figura 4.1: *Grafico di confronto tra regressione standard e SGD nel primo esempio.*

Esempio 4.2. In questo caso simuliamo una condizione nella quale il nostro metodo mostra le sue prime fragilità, qui x e y sono correlate, ma la loro dipendenza è più che lineare: prendiamo x come nel caso precedente, y sarà invece $e^x + \sigma$.

```
x=randn(1,n);
y=exp(x)+(1/4)*randn(1,n);
```

Come si può vedere dalla figura 4.2, la nostra retta dà un idea della correlazione presente tra i dati tanto più accurata quanto più siamo vicini all'origine. Allontanandoci da zero, però, osserviamo che la differenza di quota presente tra i punti sul grafico e la retta può diventare anche molto elevata, rendendo il nostro modello di regressione praticamente inutile in punti lontani dall'origine.

Osservazione 4.3. Riflettendo un poco possiamo spiegarci il perché il nostro metodo dia risultati in questo caso migliori in un intorno dell'origine: se prendiamo delle x molto vicine a 0 e togliamo il rumore, possiamo osservare che la retta di regressione avrà dei coefficienti molto vicini a quelli della retta $y = 1 + x$. Se adesso guardiamo lo sviluppo di Taylor di e^x centrato nell'origine, osserviamo che al primo ordine $e^x = 1 + x + o(x)$, da qui la spiegazione del fenomeno.

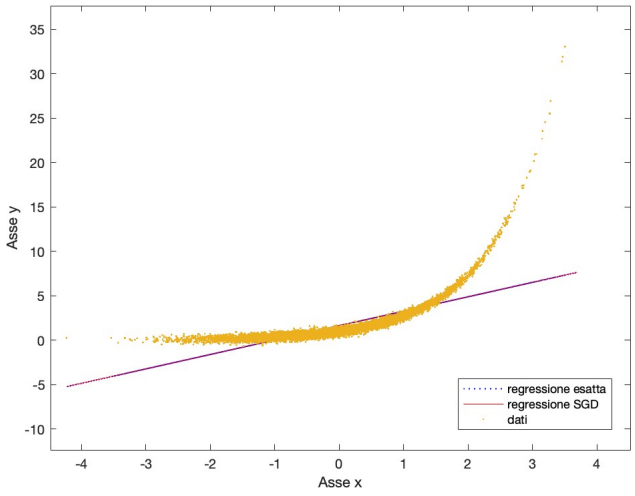

Figura 4.2: *Grafico di confronto tra regressione standard e SGD nel secondo esempio.*

Esempio 4.4. Simuliamo adesso il caso in cui x e y siano legate da una relazione quadratica, avremo: $y = x^2$.

```
x=randn(1,n);
y=x.^2;
```

La figura 4.3 aiuta a esprimere quanto la regressione lineare sia inadatta per questo tipo di dati, questo perché il metodo riesce a trovare solo l'eventuale dipendenza lineare che intercorre tra le serie storiche, ignorandone altri tipi.

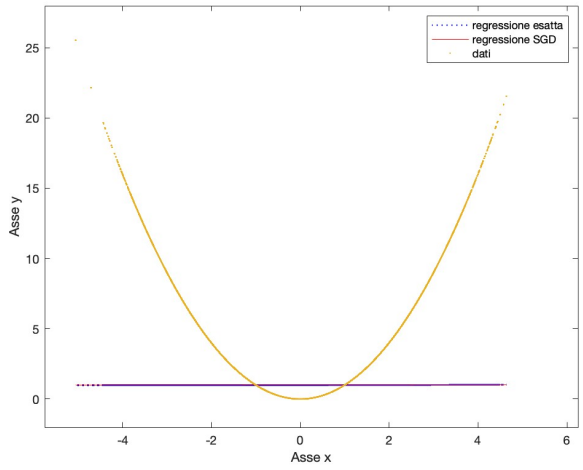

Figura 4.3: *Grafico di confronto tra regressione standard e SGD nell' ultimo esempio.*

Bibliografia

[1] G. Garrigos, R. M. Gower. *Handbook of Convergence Theorems for (Stochastic) Gradient Methods.* 2024 arXiv:2301.11235v3.

[2] D. P. Bertsekas, J. N. Tsitsiklis: *Gradient convergence in gradient methods with errors.* In: SIAM Journal on Optimization (2000), pp. 627 - 642.

[3] F. Orabona: *Almost sure convergence of SGD on smooth nonconvex functions.* Blogpost su `http://parameterfree.com`, Disponibile all'url `https://parameterfree.com/2020/10/05/almost-sure-convergence-of-sgd-on-smooth-non-convex-functions/` 2020.

[4] A. Pascucci, *Probability theory. Volume 2 - Stochastic Calculus*, Unitext, Springer, Milan, 2024.

Appendice A

Preliminari e complementi

Questa appendice serve a raccogliere delle definizioni fondamentali insieme a qualche risultato utilizzato.

A.1 Convessità

Definizione A.1. *Data $f : \mathbb{R}^d \to \mathbb{R}$, essa si dice* ***convessa*** *se per ogni $x, y \in \mathbb{R}^d$ e per ogni $t \in (0, 1)$*

$$f(tx + (1-t)y) \leq tf(x) + (1-t)f(y).$$

Lemma A.2. *Se $f : \mathbb{R}^d \to \mathbb{R}$ è convessa e differenziabile allora, per ogni $x, y \in \mathbb{R}^d$,*

$$f(x) \geq f(y) + \langle \nabla f(y), x - y \rangle.$$

Dimostrazione. La disuguaglianza si ricava dalla definizione di convessità dividendo per t e riordinando i membri. Otteniamo quindi

$$\frac{f(y + t(x-y)) - f(y)}{t} \leq f(x) - f(y).$$

Andando ora al limite per $t \longrightarrow 0$ otteniamo

$$\langle \nabla f(y), x - y \rangle \leq f(x) - f(y).$$

□

A.2 Varianza e valore atteso

Definizione A.3. *Date due variabili aleatorie $X, Y \in \mathbb{R}^d$, denotiamo:*

- *Il* ***Valore atteso*** *di X come $\mathbb{E}[X]$.*

- ***Il Valore atteso di*** X ***condizionato a*** Y *come* $\mathbb{E}[X|Y]$ *(Talvolta lo indicheremo per semplicità con* $\mathbb{E}_Y[X]$*).*
- *La* ***Varianza*** *di* X *come* $\mathbb{V}[X] \coloneqq \mathbb{E}[(X - \mathbb{E}[X])(X - \mathbb{E}[X])^*]$.

Lemma A.4. *Sia* X *una variabile aleatoria in* $\mathbb{R}^d$.

1. *Per ogni* $y \in \mathbb{R}^d$, $\mathbb{V}[X] \leq \mathbb{E}[\|X - y\|^2]$.
2. $\mathbb{V}[X] \leq \mathbb{E}[\|X\|^2]$.

Dimostrazione. Il punto 2 è una diretta conseguenza del punto 1 con $y = 0$. Per provare il punto 1 usiamo

$$\|X - \mathbb{E}[X]\|^2 = \|X - y\|^2 + \|y - \mathbb{E}[X]\|^2 + 2\langle X - y, y - \mathbb{E}[X]\rangle,$$

e prendendo poi il valore atteso concludiamo

$$\mathbb{V}[X] = \mathbb{E}\left[\|X - y\|^2\right] - 2\mathbb{E}\left[\|y - \mathbb{E}[X]\|^2\right] \leq \mathbb{E}\left[\|X - y\|^2\right].$$

□

A.3 Martingale a tempo discreto

Definizione A.5. *Sia* $X = (X_t)_{t\in I}$*, con* $I \subset \mathbb{R}$ *che supporremo discreto, un processo stocastico sullo spazio con filtrazione* $(\Omega, \mathcal{F}, P, (\mathcal{F}_t)_{t\in I})$*. Diciamo che* X *è una martingala se:*

- X *è un processo sommabile, ossia* $X_t \in L^1(\Omega, P)$ *per ogni* $t \in I$*;*
- *vale*

$$X_t = \mathbb{E}[X_T|\mathcal{F}_t] \qquad t, T \in I, t \leq T.$$

Enunciamo ora un teorema che non dimostreremo, che riguarda la convergenza quasi certa di questo tipo di processi. Una dimostrazione è presente in [4].

Teorema A.6. *Sia* $X = (X_n)_{n\in\mathbb{N}}$ *una martingala discreta tale che* $\sup_{n\in\mathbb{N}} \mathbb{E}[|X_n|] < \infty$*. Allora, quasi certamente esiste ed è finito il limite*

$$X_\infty(\omega) \coloneqq \lim_{n\to\infty} X_n(\omega).$$

Appendice B

Algoritmo GD per la regressione lineare

```
function ab=gd(x,y,ab0,maxit)

n=length(x);
ab=ab0;

for i=1:maxit
    grad=[0,0];

    for k=1:n %calcolo il gradiente
        grad2=ab(1)*x(k)+ab(2)-y(k);
        grad1=grad2*x(k);
        grad=grad+[grad1,grad2];
    end

    ab=ab-(1/(10000*i))*grad; %il passo di discesa, definito
        come (1/(10000*i)), serve a facilitare la convergenza
        per l'esempio presente nel testo

end
```