

GRADIENT DESCENT

Molti problemi in statistica si riducono a

$$\beta^* = \min_{\beta} J(\beta)$$

ossia problemi di unconstrained minimization problem

In particolare si stabilisce prima una funzione di loss e poi il rischio J non è altro che il suo valore atteso. Quindi

l'obiettivo è: *funzione di loss*

$$\beta^* = \min_{\beta} \left\{ \frac{1}{n} \sum_{D} Q_{\beta}(D) \right\}$$

Campione

(con $D = (x, y) \in \mathcal{D}$)

dataset

A' alcuni esempi in cui questo potrebbe essere utile sono:

LINEAR REGRESSION

$$Q_{\beta}(d) = Q_{\beta}(x, y) = (x^T \beta - y)^2$$

$$\Rightarrow J(\beta) = E[(x^T \beta - y)^2]$$

LOGISTIC REGRESSION (classificazione)

$$Q_{\beta}(d) = Q_{\beta}(x, y) = \log(1 + e^{-y x^T \beta})$$

$$\Rightarrow J(\beta) = E[\log(1 + e^{-y x^T \beta})]$$

$$\frac{\partial}{\partial \beta} \log(1 + e^{-y x^T \beta}) = \frac{1}{1 + e^{-y x^T \beta}} \cdot (e^{-y x^T \beta}) \cdot (-y x^T)$$

$$= \frac{1}{1 + e^{-y x^T \beta}} \cdot \frac{1}{e^{y x^T \beta}} \cdot (-y x^T) =$$

$$\frac{-y x^T}{e^{y x^T \beta} + e^{-y x^T \beta}} = \frac{-y x^T}{e^{y x^T \beta} + 1}$$

caso $\frac{y}{e^{y x^T \beta} + 1} \leftarrow$ x^T è vettore

ASSUNZIONI

Per eseguire efficacemente il gradient descent si fanno due assunzioni:

1) LIPSCHITZ - CONTINUOUS GRADIENT

$\forall \beta_1, \beta_2 :$

$$\|\nabla S(\beta_2) - \nabla S(\beta_1)\| \leq \delta \|\beta_2 - \beta_1\|$$

2) CONVESSITÀ

$\forall \alpha \in (0, 1) \quad | \quad \beta_1 \neq \beta_2$

$$J(\alpha \beta_1 + (1-\alpha) \beta_2) \leq \alpha J(\beta_1) + (1-\alpha) J(\beta_2)$$

STRETTA CONVESSITÀ

$\forall \alpha \in (0, 1) \quad | \quad \beta_1 \neq \beta_2$

$$J(\alpha \beta_1 + (1-\alpha) \beta_2) < \alpha J(\beta_1) + (1-\alpha) J(\beta_2)$$

FORTE CONVESSITÀ

$\forall \alpha \in (0, 1) \quad | \quad \beta_1 \neq \beta_2$

$$J(\alpha \beta_1 + (1-\alpha) \beta_2) \leq \alpha J(\beta_1) + (1-\alpha) J(\beta_2)$$

$$- \frac{\gamma}{2} \alpha (1-\alpha) \|\beta_1 - \beta_2\|^2$$

Ovvvero, per quanto riguarda la convessità

CONVESSA	STRETTAMENTE CONVESSA	FORTEMENTE CONVESSA
$\nabla^2 J(\rho) \geq 0$ $\forall \beta$	$\nabla^2 J(\rho) > 0$ $\forall \beta$	$\nabla^2 J(\rho) \geq \gamma I$ $\forall \beta$ "quanto" J è convessa

Da qui si arriva sia l'interpretazione
sia che

STRONG \Rightarrow STRICT CONVEXITY \Rightarrow CONVEXITY

• γ è detta costante di strong convexity

• γ è detta costante di Lipschitz

GRADIENT DESCENT

GRADIENT DESCENT (D, β_0)

- initializziamo β_0 ✓

• REPEAT

$$\beta_i = \beta_{i-1} - \mu \nabla J(\beta_{i-1})$$

UNTIL end condition

dove μ è detto STEP SIZE

In base a μ ci sono due varianti

- CONSTANT STEP-SIZE

↳ converge per un μ minore di un certo valore a zero minimo

↳ $O(p)$ con $p(\mu) \in (0, 1)$ (GEOMETRIC RATE)

- DECREASING STEP-SIZE

↳ converge al zero minimo

↳ non geometric rate (?)

con $\mu = \mu(i)$ $\sum_{i=0}^{\infty} \mu(i) = \infty$ e $\lim_{i \rightarrow \infty} \mu(i) = 0$

ad esempio

$$\mu(i) = \frac{\gamma}{i+1}$$

con $\gamma > 0$

↳ $O(1/\sqrt{i})$

Bisogna però tenere in conto che poiché non conosciamo la distribuzione di \mathcal{D} dobbiamo usare la legge dei grandi numeri, cioè:

$$J(\beta) = \frac{1}{m} \sum_{i=1}^m Q_\beta(d_i)$$

► STOCHASTIC GRADIENT DESCENT

Calcolare il costo J su tutto il dataset è davvero dispendioso, per questo utilizziamo un'heuristica:

$$\widehat{\nabla J_i}(\beta) = \nabla Q_\beta(d_i)$$

Così approssimiamo il gradiente del costo su tutto il dataset con il gradiente del costo calcolato su solo campione (cioè, effettivamente il gradiente delle loss).

Questa tecnica viene chiamata stochastic gradient descent perché l'i-esimo campione su cui viene calcolato il gradiente viene estratto casualmente ^{al} ogni step.

Quindi

STOCHASTIC GRADIENT DESCENT (D)

- inizializziamo β_0
- REPEAT for $i = 1, 2, \dots$
- prendiamo un nuovo campione CASUALE di
 - $\beta_i = \beta_{i-1} - \mu \nabla J(\beta_{i-1})$
- UNTIL end condition

Come al solito ci sono due varianti 2 seconde di come si sceglie lo step-size

• CONSTANT STEP-SIZE

- β "rimborba" intorno al minimo senza mai avvicinare per il minore di un certo valore β_i sta in un intorno di β^*
- $E[\|\beta_i - \beta^*\|^2]$ converge ad un errore a regime ad un rate $O(p)$

• DECAYING STEP-SIZE

per $\mu = \mu(i)$ $\sum_{i=0}^{\infty} \mu(i) = \infty$ e $\lim_{i \rightarrow \infty} \mu(i) = 0$

converges in the mean square sense (?)

$$\hookrightarrow \text{per } \mu(i) = \frac{\tau}{1+i} \rightarrow O\left(\frac{1}{i}\right) \text{ per } \tau > 1$$

N.B.: il decaying step-size non va

~~consigliato~~ bene per applicazioni

online perché ad un certo punto

smetterebbe di imparare

VARIANTE DEI MINIBATCH

Per migliorare l'approssimazione di $\hat{F}_{\beta}(\beta)$ si potrebbero usare più campioni invece che uno solo, cioè:

$$\beta_i = \beta_{i-1} - \frac{\mu}{|S|} \sum_{i \in S} \nabla Q_{\beta}(d_i)$$

media delle perdite
della loss sui campioni
del minibatch