Método do gradiente estocástico

<u>Problema:</u> $\min f(x)$, <u>$f:\mathbb{R}^n \to \mathbb{R}$ convexa</u>

Não temos acesso à $\nabla f(x)$ (por exemplo, porque é caro computar).

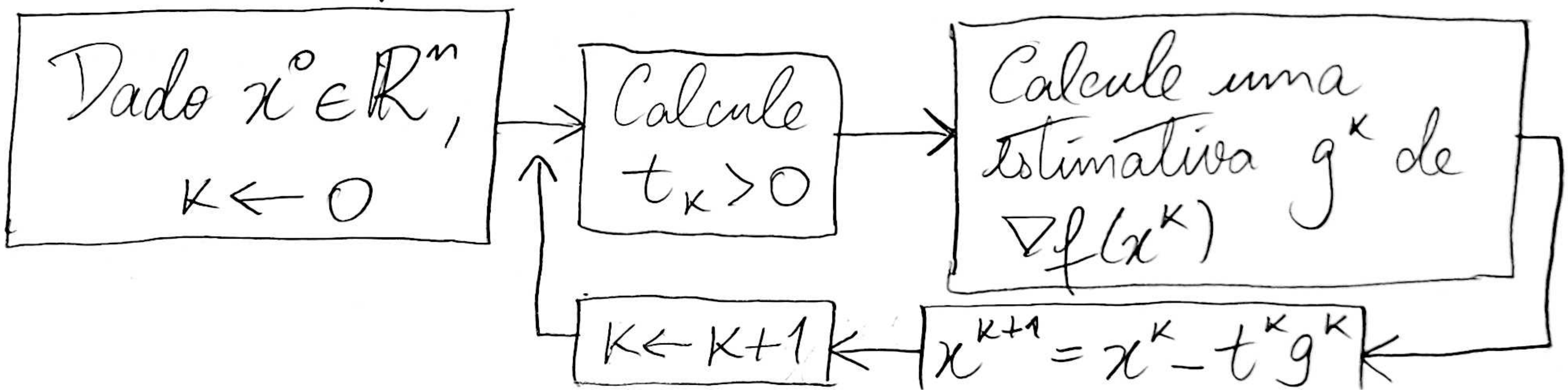
<u>Iteração:</u> $x^{k+1} = x^k - t_k g^k$, $t_k > 0$ (passo)

- g^k é uma <u>estimativa aleatória</u> de $\nabla f(x^k)$
- Assim, x^{k+1} depende da amostra de pontos.

anteriores (na verdade, está condicionado apenas à escolha de x^k e g^k)

- ponto inicial x^0 é determinístico (fornecido pelo usuário); x^k, $k \geq 1$, aleatórios.

Método do gradiente estocástico



Como deve ser a escolha de g^k?

H1) $E(g^k \mid x^k) = \nabla f(x^k)$

(a rigor, deveria escrever G^k, mas vou manter g^k para refletir a escolha no método)

Essa hipótese diz que, <u>em média</u>, g^k deve ser o gradiente $\nabla f(x^k)$, mas <u>não</u> necessariamente $g^k = \nabla f(x^k)$.

H1 é conhecida como "escolha sem viés".

Note que a esperança em H1 é condicionada à escolha $X = x^k$ (coerente com gradiente clássico, onde $g^k = \nabla f(x^k)$ só depende de x^k).

Hipótese técnica:

H2) Existe uma constante $L > 0$ tal que

$$E(\|g^k\|^2 \mid x^k) \leq L^2.$$

(compare com os métodos do gradiente incremental e do subgradiente).

Exemplo: $f(x) = \frac{1}{m} \sum_{j=1}^{m} f_j(x)$, f_j convexa $\forall j$.

Considere o seguinte estimador para g^k, dado x^k:

"escolha $i_k \in \{1, \ldots, m\}$ <u>uniformemente</u> e tome $g^k = \nabla f_{i_k}(x^k)$".

Qual a esperança de g^k dado x^k?

$$E(g^k \mid x^k) = \sum_{i=1}^{m} \nabla f_i(x^k) . \underbrace{P(i)}_{\text{prob. escolha de } i}$$

$$= \sum_{i=1}^{m} \nabla f_i(x^k) \frac{1}{m} = \frac{1}{m} \sum_{i=1}^{m} \nabla f_i(x^k)$$

$$= \nabla f(x^k).$$

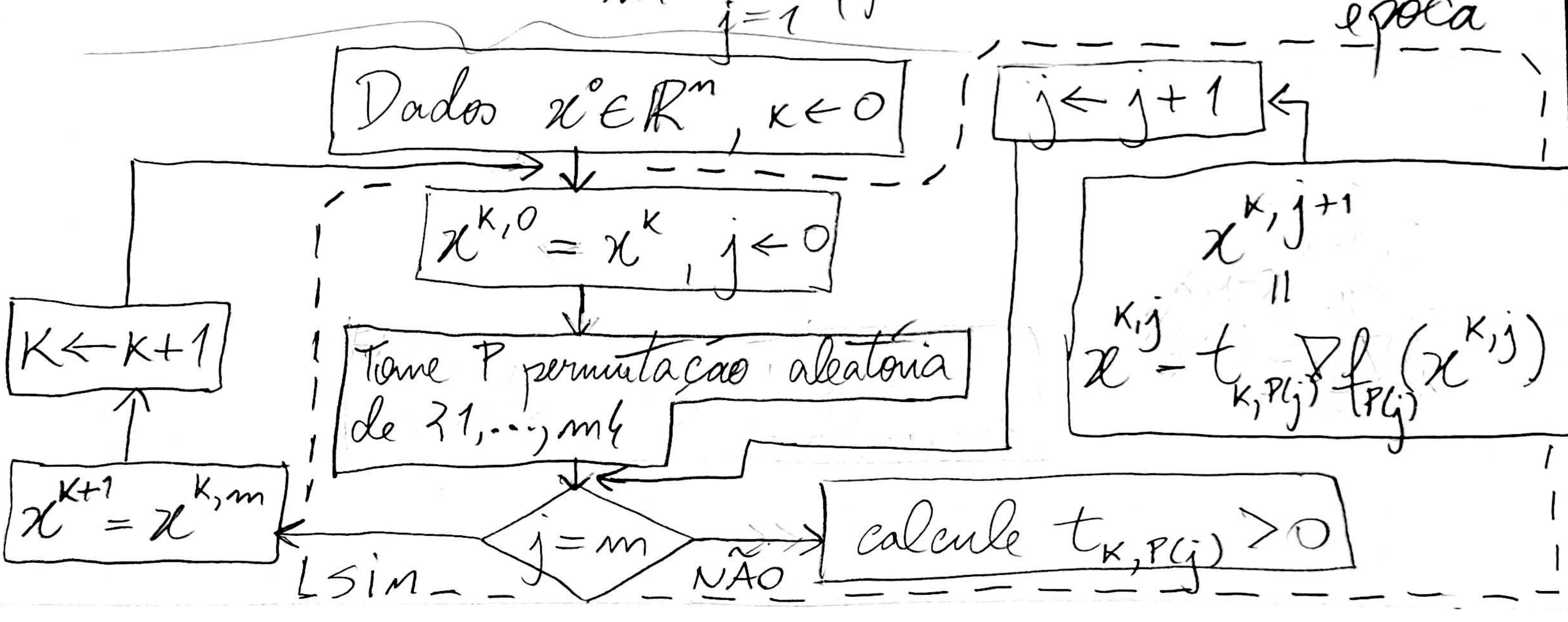
Note que é importante que a escolha de i_k seja uniforme, de modo que $P(i) = \frac{1}{m}, \forall i$.

Este exemplo indica uma escolha possível simples e sem viés para nosso problema de interesse!

Com esta escolha, o método do gradiente estocástico assemelha-se ao gradiente incremental, com a diferença que i é escolhido aleatoriamente, t_k pode ser diferente a cada subiteração, e não há garantia da escolha de <u>todos</u> os gradientes ∇f_i como no ciclo do gradiente incremental.

Quando obrigamos a escolha de todos os gradientes, o ciclo é chamado <u>época</u>.

Desta forma podemos dividir as iterações no nosso caso $f(x) = \frac{1}{m}\sum_{j=1}^{m} f_j(x)$:



Obs:

1) note que há um passo para cada subiteração

2) se $t_{k,P(j)} = t > 0$ cte $\forall k, j$, então recaímos no gradiente incremental com passo constante, com a diferença que i é aleatório.

3) Na época, todos ∇f_i são escolhidos.

4) Geralmente é assim que são as implementações

5) Porém, escolher todos ∇f_i evita viés ???

Outra forma de estimar g^k é fazer

$$g^k = \frac{1}{|S_k|} \sum_{j \in S_k} \nabla f_j(x^k) \, ,$$

onde $S_k \subset \{1, \dots, m\}$, $|S_k| \ll m$. S_k: (mini-batch) ("mini-lote")

É importante que o mini-lote seja escolhido de maneira uniforme.

Uma maneira de proceder:

1) Temos $|S| \ll m$ o tamanho dos mini-lotes (vamos supor que todos eles têm mesmo tamanho)

2) Divida os dados de treinamento em m_B mini-lotes $S_1, \ldots, S_{m_B}$ (supor $m_B = \frac{m}{|S|} \in \mathbb{N}$).

3) Escolha um mini-lote S_{i_k} onde i_k é escolhido uniformemente em $\{1, \ldots, m_B\}$

Note que $E(g^k | x^k) = \sum_{i=1}^{m_B} \left(\frac{1}{|S|} \sum_{j \in S_i} \nabla f_j(x^k) \right) P(i)$

$$= \sum_{i=1}^{m_B} \left(\frac{m_B}{m} \sum_{j \in S_i} \nabla f_j(x^k) \right) \frac{1}{m_B}$$

$P(i) = \frac{1}{m_B}$

$$= \frac{m_B}{m} \cdot \frac{1}{m_B} \sum_{j=1}^{m} \nabla f_j(x^k) = \nabla f(x^k)$$

(não há viés).

Note que aqui a época consiste em m_B escolhas de mini-lotes.

Uma outra maneira de implementar é a seguinte (usada na prática):

1) temos $|S| \ll m$ o tamanho dos mini-lotes (vamos supor que todos eles têm mesmo tamanho);

2) a cada início de época, embaralhe os dados de treinamento;

3) divida os dados embaralhados em pedaços consecutivos de tamanho $|S|$;

4) percorra sequencialmente todos os mini-lotes gerados.

Isso evita viés ???...

De qualquer forma, valem as observações:

1) $|S| = m$ (1 lote) corresponde ao método do gradiente, pois calculamos $\nabla f(x^k)$ inteiro por iteração;

2) $|S| = 1$ corresponde ao método do gradiente estocástico enunciado anteriormente

3) É imperativo $|S| \ll m$ (iteração barata), mas pode ser interessante $|S| > 1$ pois usar > 1 gradiente ∇f_j fornece uma iteração mais eficaz.

↳ o balanço de $|S|$ é empírico!

Escolha dos passos (variantes do SG).

- Método do gradiente estocástico "básico" (SGD)

$$t_{k,i_j} = \eta > 0 \quad \text{cte.}$$

≠ da ideia original de SG (1951)

- Adagrad (2011)

$$t_{k,i_j} = \frac{\eta}{\sqrt{G_{k,i_j}+\varepsilon}}, \quad \eta > 0, \ \varepsilon \in (0,1) \text{ parâmetros},$$

$$G_{k,i_j} = \sum_{\tilde{k} \le k, \forall \tilde{i_j}} \| g^{\tilde{k},\tilde{i_j}} \|^2 .$$

Ideia: diminuir t à medida que o método avança, inversamente proporcional ao acúmulo das normas dos gradientes escolhidos.
(Lembra a ideia de passo decrescente nos métodos do gradiente incremental e de subgradientes).

- Passos com "momento".

Imagine que um corpo viaje de x^{k-1} até x^k com uma velocidade v^k. No ponto x^k, o corpo não fará uma "virada brusca" pois há uma tendência a permanecer na direção $x^k - x^{k-1}$ (momento linear = massa $\times v$)

Por outro lado, a iteração $x^{k+1} = x^k - t g^k$ representa uma "virada brusca" na direção $-g^k$.

Métodos com momento buscam incorporar a ideia física na direção de busca.

A ideia é trocar g^k por um "vetor velocidade", que combina g^k com a velocidade anterior:

$$x^{k+1} = x^k - t_k v^k \quad \text{onde}$$

$$v^k = \beta v^{k-1} + (1-\beta) g^k \qquad (\beta \in [0,1) \text{ parâmetro})$$

Iniciamos a velocidade nula $v^{-1} = 0$.

- <u>SGD com momento (1999)</u>

$$t_{k,i_j} = \eta > 0 \ \text{cte},$$

$$x^{k,j+1} = x^{k,j} - \eta v^{k,j} \quad \text{onde}$$

$$v^{k,j} = \beta v^{k,j-1} + (1-\beta) g^{k,i_j}, \quad \beta \in [0,1)$$

parâmetro

- <u>RMSProp (2012?)</u>

Melhoramento do Adagrad.

- **AdaDelta (2012)**

  Outro melhoramento do Adagrad.

- **Adam (2014/15)**

  Combina passo tipo Adagrad e momento

Obs: todas variantes podem implementar mini-lotes.