
Ficha - Métodos de Otimização Sem Restrições

Método do gradiente, Método do gradiente estocástico, Método do gradiente estocástico mini-batch

1. Resolva o problema

$$\underset{w \in \mathbb{R}^2}{\text{minimizar}} F(w) = w_1^2 + 2w_2^2$$

com $w^{(0)} = (1, 1)^T$.

Use o método do gradiente e para critério de paragem considere a condição $\|\nabla F(w)\| \leq \varepsilon$ com $\varepsilon = 10^{-6}$. Em cada iteração, use o algoritmo de procura de Armijo com backtracking para calcular o η_k , com $\delta = 0.1$.

Em cada iteração, grave para um ficheiro a informação:

k	$w^{(k)T}$	$\nabla F(w^{(k)})$	$\ \nabla F(w^{(k)})\ _2$	η_k	$F(w^{(k)})$
...

2. Resolva o problema

$$\underset{w \in \mathbb{R}^2}{\text{minimizar}} F(w) = (w_1 - 1)^2 + (w_1 - 1 - w_2)^2$$

com $w^{(0)} = (0, 0)^T$.

Use o método do gradiente e para critério de paragem considere a condição $\|\nabla F(w)\| \leq \varepsilon$ com $\varepsilon = 10^{-6}$. Em cada iteração, use o algoritmo de procura de Armijo com backtracking para calcular o η_k , com $\delta = 0.1$.

Em cada iteração, grave para um ficheiro a informação:

k	$w^{(k)T}$	$\nabla F(w^{(k)})$	$\ \nabla F(w^{(k)})\ _2$	η_k	$F(w^{(k)})$
...

3. Considere-se seguinte problema de *machine learning*. Dado o data set $D = (x^n, y^n)_{n=1}^N$ pretende-se determinar os coeficientes de um polinómio de grau I

$$\begin{aligned}\phi(w; x) &= w_0 + w_1x + w_2x^2 + \dots + w_Ix^I \\ &= w^T p(x)\end{aligned}$$

onde $p(x) = (1, x, x^2, \dots, x^I)^T$ que melhor ajustam o polinómio aos dados D no sentido da minimização da função MSE (*Mean Squared error*):

$$MSE(w; D) = \frac{1}{N} \sum_{n=1}^N (\phi(w; x^n) - y^n)^2.$$

Nota: O gradiente da função $MSE(w; D)$ é dado por $\nabla MSE(w; D) = \frac{2}{N} \sum_{n=1}^N (\phi(w; x^n) - y^n)p(x^n)$.

- (a) Resolver o Problema apresentado com o data set data1.csv (N=100).

Dividir o data set em duas partes: 80% para treino D_t e 20% para validação D_v . Esta seleção deverá ser aleatória. Resolva o problema com o método do gradiente batch (**Algoritmo 2**) e como aproximações iniciais aos parâmetros considere $w^{(0)} = (0, \dots, 0)$, e para critério de paragem considere

$$\|\nabla MSE(w)\| \leq 10^{-4} \text{ e } k \leq 10N_t.$$

Calcular a *learning rate* (comprimento do passo) η pelo algoritmo de procura de Armijo com backtracking, com $\delta = 0.1$. Resolva o problema considerando polinómios de grau $I = 2, \dots, 7$. Para cada um dos polinómios calcule: w^* , o erro de treino (*in-sample error*) $MSE(w^*; D_t)$, e o erro de validação (*out-sample error*) $MSE(w^*; D_v)$. Fazer o gráfico dos erros e indicar qual o grau I que fornece a melhor aproximação.

- (b) Faça o exercício 3(a) mas considere a implementação do método do gradiente estocástico (**Algoritmo 1**). Notar que terá que adaptar a condição de Armijo a este caso.
- (c) Faça o exercício 3(a) mas considere agora o método do gradiente estocástico mini-batch (**Algoritmo 3**). Considere para mini-batch 5% dos dados do D_t . Esta seleção deverá ser aleatória em cada iteração. Notar que terá que adaptar a condição de Armijo a este caso.
- (d)
 - i. Avaliar o desempenho dos algoritmos implementados com: η constante ao longo processo iterativo (exemplo: $\eta = 1, \eta = 0.1, \dots$).
 - ii. Avaliar o desempenho dos algoritmos 1 e 3 quando: $\eta = 0.1$ e reduz para 0.01, 0.001, ... de \times em \times épocas.