

Pour ft linear regression

Pour ce projet, on fait un premier programme qui doit faire une descente de gradient sur un dataset de (x, y) , afin de trouver deux réels θ_0 et θ_1 , qui permettent de prédire des points \hat{y} par la fonction :

$$\hat{y} = \mathcal{F}(x) = \theta_0 + \theta_1 x$$

Pour faire cette descente de gradient, on a normalisé les x et y du dataset avec la fonction :

$$\mathcal{N}(x) = x^{(i)} = \frac{x^{(i)} - \frac{1}{m} \sum_{i=1}^m x^{(i)}}{\sqrt{\frac{1}{m-1} \sum_{i=1}^m (x^{(i)} - \frac{1}{m} \sum_{i=1}^m x^{(i)})^2}} \quad i \in [1, m]$$

Avec :

- x un vecteur de dimension m
- $x^{(i)}$ le i^{th} élément du vecteur x
- x' la version normalisé du vecteur x

Je simplifie pour faciliter l'implémentation , on a

$$x^{(i)} = \frac{x^{(i)} - \mu}{\sigma} \quad i \in [1, m]$$

Avec:

- μ est la moyenne des x
- σ est la variation standart

On sauvegarde le couple (θ_0, θ_1) et un deuxième programme se sert de ça pour prédire \hat{y} par la fonction \mathcal{F} . Mais \mathcal{F} ne fonctionne qu'avec des x' et donne des \hat{y} normalisés. J'ai dû donc sauvegarder, dans le premier programme μ et σ pour normaliser l'entrée x et « dé-normaliser » \hat{y} et afficher une valeur compréhensive à l'utilisateur.

La question est :

Peut on, dans le premier programme, calculer (θ'_0, θ'_1) qui seraient les « θ dé-normalisés » et ainsi ne pas enregistrer les moyennes et variations standards des vecteurs x et y ?