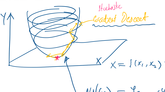
Discrete Optimization / Python

Discrete Optimization 17 Mar 2021 10:00

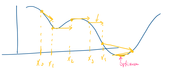


$X = \{(x_1, x_2) : x_1 \in \mathbb{R}, x_2 \in \mathbb{R}\} = \mathbb{R}^2$

$NN(x) = y$

$$\boxed{\min_{\text{weights}} \|y - \hat{y}\|_2^2}$$

$f(x)$ $\quad$ $f(x) = \|y - \hat{y}\|_2^2$

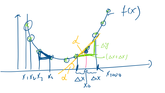


Local minima / flat

1. Scipy - minimize : 2/10
2. self.write gradient descent : 6/10
   but will give gradient
3. —— // ——
   but automatically computed gradient : 8, 9, 10 /10
   - 3.1 Finite difference
   - 3.2 Interpolation
   - 3.3 Projection

Need: Numpy : fast, broadcasting, matrix operations
Google Colab

Task 1: compute discrete gradient



$$\lim_{\Delta x \to 0} \frac{f(x+\Delta x) - f(x-\Delta x)}{2\Delta x}$$

$$\text{tangent} \approx \frac{f(x+\Delta x) - f(x-\Delta x)}{2\Delta x}$$

$(x+\Delta x) - (x-\Delta x) = 2\Delta x$

$\varepsilon = 1$

for : in range (100):

$$\frac{f(x+\varepsilon) - f(x-1)}{2\varepsilon}$$

$\varepsilon = \varepsilon/2$

```
def func(x):
    return x**2
```

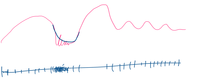
$X = [x_1, x_2, \dots, x_N]$

$f(x_i) = [f(x_1), f(x_2), \dots, f(x_N)]$ ← Pointwise approximation

$-\Delta\frac{d}{dx} = \left(\frac{d}{dx}\right)^2$

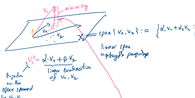
```
def grad_func(x, func):
    return func(x)[1:] - func(x)[:-1]
           / x[1:] - x[:-1]
```

$\boxed{\text{Finite-Difference Methods}}$



Gradient Computation via Projection:

Linear Algebra
Matrix - calculus
Linear - combination, span
Basis
Best - fit - Approximation (Projection)



$V^* = \alpha V_1 + \beta V_2$ — Linear combination of V_1, V_2

Regular — no the space spanned by V_1, V_2

linear span orthogonal property

$V_1^{\parallel} = \text{proj}(V_1)$

$$\boxed{\min_{\alpha, \beta \in \mathbb{R}} \|V - V_1\|_2^2}$$

V_1 V_2 $f(x)$

$$\min_{\alpha, \beta \in \mathbb{R}} \|\alpha V_1 + \beta V_2 - V_3\|_2^2$$

$f(\alpha, \beta)$

$g_{\alpha\beta}$ $\quad f(\alpha, \beta) \triangleq 0$

$(\alpha V_1 + \beta V_2 - V_3)^T (\alpha V_1 + \beta V_2 - V_3)$

Projection property

$\boxed{\langle V^{\perp}, V_3 \rangle = 0}$ ← Task: Proof

$\boxed{\langle \alpha^{\parallel} V_1 + \beta^{\parallel} V_2 - V_3, V_3 \rangle = 0}$ $\quad \alpha, \beta$

$$\nabla_\alpha \begin{pmatrix} \alpha^2 V_1 V_1^T \\ + \beta \alpha V_1^T V_2 \\ - \alpha V_1^T V_3 \\ + \beta \alpha V_2^T V_1 \\ + \beta^2 V_2^T V_2 \\ - \beta V_2^T V_3 \end{pmatrix} = 0$$

$$\nabla_\beta \begin{pmatrix} \quad \end{pmatrix} = 0$$

$\boxed{\nabla_\alpha = \|x\|^2 x}$

$\sum x \cdot x_i = x^T x$

$$\boxed{\begin{aligned} V_0 &= 1 \\ V_1 &= x \\ V_2 &= x^2 \\ &\vdots \\ V_n &= x^n \end{aligned}}$$

$x^{-1} = 1$ Vector form??

$f(x) = \sum_{i=0}^{N} \alpha_i \phi_i$, $\quad \phi^N := \text{span}\{x^i\}_{i=0}^{N} = \{ \sum_{i=0}^{N} \alpha_i x^i : \forall \alpha \in \mathbb{R} \}$

$x + 0 + 0 + x$

$$\boxed{\min_{f \in \phi^N} \|f - f\|_2^2}$$

$\|f\|_2 = \left[\int_a^b |f(x)|^2 dx\right]^{\frac{1}{2}}$ ← L_p - norm

using is this a norm?

$= \langle f - f, f - f \rangle$

$$\min_{\alpha \in \mathbb{R}} \int_{-1}^{1} \Big(f(x) - \underbrace{\sum_{i=0}^{N} \alpha_i \phi_i}_{\phi^N}\Big)^2 dx$$

eq : w.r.t. $\quad \langle f - f^N, f^N \rangle = 0$

$\Rightarrow \langle f - f^N, x^j \rangle = 0 \quad \forall j = 0, \dots, N$

$\Leftrightarrow \int (f(x) - f^N) \cdot x^j dx = 0$

$\int f \cdot x^j dx = \int f^N \cdot x^j dx$

$\int \sum_{i=0}^{N} \alpha_i \phi_i \cdot x^j dx$

$\sum \alpha_i \int \phi_i \cdot x^j dx \quad (\dots)$

$\sum \alpha_i \left[\frac{1}{i+j+1} x^{i+j}\right]_{-1}^{1} = (-)$

$\sum \frac{\alpha_i}{i+j+1} \underbrace{\left(1 - (-1)^{i+j+1}\right)}_{=0} = (-)$

$\sum_{i=0}^{N} \frac{\alpha_i}{i+j+1} = (\dots) = \underline{Ad}$

$A = [A_{ij}] = \left[\frac{1}{i+j+1}\right]_{ij}$

$\alpha = [\alpha_i]_{i=0}^{N}$

$$A = \begin{pmatrix} 1 & \frac{1}{2} & \frac{1}{3} & \cdots & \frac{1}{N} \\ \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \cdots & \frac{1}{N+1} \\ \frac{1}{3} & & \vdots & & \\ \frac{1}{N} & & & & \frac{1}{2N} \end{pmatrix}$$

← for large scaled

$\boxed{Ad = b}$

$\boxed{\alpha = A^{-1} b}$

$$Ad = \begin{cases} \int f(x) \cdot 1 dx \\ \int f(x) \cdot x dx \\ \vdots \\ \int f(x) \cdot x^N dx \end{cases}$$

$\phi_i(x) = \begin{cases} 1, & x = x_i \\ 0, & x = x_{i-1}, x_{i+1} \end{cases} = \begin{cases} \frac{x - x_{i-1}}{x_i - x_{i-1}}, & x \in [x_{i-1}, x_i] \\ \frac{x_{i+1} - x}{x_{i+1} - x_i}, & x \in [x_i, x_{i+1}] \\ 0, & x \text{ else} \end{cases}$

base fn. $\{\phi_{i-1}, \phi_i, \phi_{i+1}\}$

← we found α → we find Approximate Function

$$\boxed{f_p(x) = \sum_{i=0}^{N} (A^{-1} b)_i \cdot x^i}$$

vector

$f \approx \sum f(x_i) \cdot \phi_i$

$\boxed{f(\xi) = \sum f(x_i) \cdot \phi_i(\xi)}$

$$\partial_x f_p(x) = \partial_x \sum_{i=0}^{N} (A^{-1} b)_i \cdot x^i$$

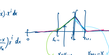
$$= \sum_{i=0}^{N} (A^{-1} b)_i (\partial_x x^i)$$

$$= \sum_{i=0}^{N} (A^{-1} b)_i \cdot i \cdot x^{i-1}$$

using this to calculate gradient-descent

$\int \sum_{i=0}^{N} f(x_i) \phi_i(x) \cdot x^j dx = \sum_{i=0}^{N} f(x_i) \int \phi_i(x) x^j dx$

$\int \phi_i(x) \cdot x^j dx = \int_{x_{i-1}}^{x_i} \frac{(x - x_{i-1})}{(x_i - x_{i-1})} x^j dx + \int_{x_i}^{x_{i+1}} \frac{(x_{i+1} - x)}{(x_{i+1} - x_i)} x^j dx$



$= \left[\frac{x}{x_i - x_{i-1}}\right] \int \left[(x - x_{i-1}) x^j dx\right] + \left[\frac{x}{x_{i+1} - x_i}\right] \int (x_{i+1} - x) x^j dx$

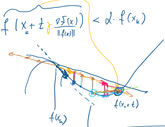
Task: Indent Indent

$\left[\frac{x^{j+1}}{j+1} - \frac{x_{i-1}}{j+1} x^{j+1}\right]_{x_{i-1}}^{x_i}$

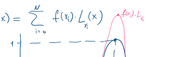
$\left[\frac{x_{i+1}^{j+2}}{j+2} - \frac{x_{i+1}}{j+1} x^{j+1} - \dots\right]$

Algorithm of gradient descent:

$$f\left(x_k + \frac{\partial f(x)}{\|f(x)\|}\right) < \alpha \cdot f(x_k)$$



$f(x) = \sum_{i=0}^{N} f(x_i) \cdot L_i(x)$



$\int f(x) \cdot x^j dx = \int \sum f(x_i) L_i(x) = \sum f$

$f(x) = \sum f(x_i) c_i$

$$\boxed{f(x_k - \alpha \cdot \partial f(x_k)) < \alpha \cdot f(x_k)}$$

$$\left| \partial \cdot \partial f(x_k) \cdot (x - x_k) - f(x_k) \right| < \delta$$

$$f\left(x_k + \xi \frac{\partial f(x)}{\|f(x)\|}\right)$$

$[t_1, t_2, t_3, \dots, t_{\text{end}}]$

Algorithm 3 Gradient Descent

Input: $f : \mathbb{R}^N \to \mathbb{R}$ a differentiable function
Output: x^* an local minimum

```
assume x^0, a local minimum of the cost function f
k ← 0
x ← x_0
while true do
    x^{k+1} ← x^k − α_k ∇f(x^k)
    if |f(x^{k+1}) − f(x^k)| < ε then
        break
    k ← k + 1
return x^k
```

end

$f(x_k - \alpha_k \partial f(x_k))$

$\alpha \cdot \partial f(x)$

$f(x_k) \cdot (x - x_k) - f(x)$

$f(x_k - \alpha \cdot \partial f(x_k)) < \alpha \cdot f(x_k)$

1. $\frac{f(x_{i+1}) - f(x_i)}{\Delta x}$ — finite-Indent
2. $\partial_x f^N(x_i)$: $\sum f(x_i) \cdot \phi_i$, $\sum f(x_i) L_i$, $\sum f(x_i) \exp(-\|x\|^2)$, $\sum f(x_i) \frac{\exp}{\sum \exp}$

3. $\left[\nabla f^N(x_i)\right]$

→ compute with fully vectorized