

Stochastic GD

درست به سراسر از دست می‌دهد local min چیست؟

نقشه شلوغ اولیه را به صورت Random می‌بینیم

با خود  $\alpha$  را نیز باید خف کنیم  $\leftarrow$  Annealing

for example:

$$\alpha_{new} = 0.999 \alpha$$

تا جایی که می‌توانیم ادامه می‌دهیم خود را با آن بررسی کنیم  $\Rightarrow$  min شبکه داده

Rosenbrock Function

ابتدا در 2D نوشته شده بود

$$f(x, y) = (a - x)^2 + b(y - x^2)^2$$

$$(x^*, y^*) = (a, a^2)$$

if putting  $a = 1$   $b = 100$  global min  $\Rightarrow (1, 1)$

این تابع یک دهه طولانی است که در  $\frac{1}{2}$  تقریباً همان min دارد

$$ND: f(\vec{x}) = \sum_{i=1}^{N-1} [-100(x_{i+1} - x_i^2)^2 + (1 - x_i)^2]$$

if  $N = 3$  1 min  $\rightarrow (1, 1, 1) \rightarrow$  Global

if  $4 \leq N \leq 7$

7 min

$\rightarrow$

Global  $\rightarrow (1, 1, \dots, 1)$

local  $\rightarrow (-1, 1, \dots, 1)$

این تابع به عنوان Gravel truth و عمل می‌کند.

به سبقت درجه در استیم باعث می‌شود نتایج min را پیدا کنیم.

GD steps:

-  $n = \text{Iteration} = \text{epochs} \sim 100$

-  $\alpha \sim 0.001$

-  $u_{\text{new}} = \text{Momentum } u \sim 0.9$

- Unsumble start =  $(A, \dots, B) \sim \text{Random} \rightarrow \text{minimum est}$

- Noise Amplitude =  $\eta \sim N(0, 1)$

$$-\vec{\nabla} L(\vec{x}) \rightarrow -\vec{\nabla} L(\vec{x}) + \underbrace{\sigma^2 \eta}_{\text{stochastic GD}} \leftarrow$$

stochastic GD

- Noise decay  $\sim 0.99 \rightarrow \text{Noise کردن}$

- Code 1 Lecture 3

- در اول جلسه 3:

پیدا کردن min در تابع RB با استفاده از GD.

دومین روش بهینه کردن که به وسیله تر از GD است.

Particle Swarm opt (PSO).

از complex system از حرکت پرندگان.

1- پرندگان شعاع امتداد خود را می بینند  
2- به سمت میانگین حرکت قبلی می روند

Visceral Model  
مدل ویسکال

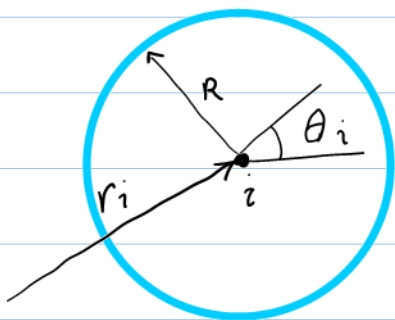
نقطه = \* پرندگان هوایی دارد که در 20 قدم گذشته با او بسته بوده

\* هر پرندگانی که به هم نزدیک می شدند بر دین = بهترین انتخاب

به یک انتخاب هم می رسد = به آن خودم رسیدیم یا دیگری

برای 2 به در مدل ویسکال داریم

مکان  $r_i$  برای در  $z$  به زاویه  $\theta_i$  به شعاع  $R$



$$\theta_i(t + \Delta t) = \langle \theta_i \rangle_R + \sigma \eta_i(t) \rightarrow N(0, 1)$$

$$r_i(t + \Delta t) = r_i(t) + v \Delta t \begin{pmatrix} \cos(\theta(t)) \\ \sin(\theta(t)) \end{pmatrix}$$

به حرکت

مرحله اول پرندگان را به عنوان نقطه اولیه می نزنیم و بعد step می زنیم

step 1.

N تعداد پرندگان

- tolerance or iteration for stopping time

$$T\text{-steps} \quad \text{or} \quad |\vec{x} - x^*| \leq 10^{-6}$$

- Inertia weight :  $\omega$

-  $c_1, c_2 \rightarrow$  Cognitive & social factors

له خویشی

صفتی دوستی

$r_1, r_2 \rightarrow$  Random

step 2:

- Initial Positions:

$$\vec{x}_i = (x_{ix}, x_{iy})$$

- Velocity:

$$\vec{v}_i = (v_{ix}, v_{iy})$$

- Personal Best

$$P_i = x_i$$

- Global Best

$$G_i = x_i$$

$\Rightarrow$  in each step  $\Rightarrow f(x, y) \rightarrow \min$

fitness ~ cost function ~ Biology

جدید update می‌شود؟

$$v_x^{t+1} = \omega v_{ix}^t + c_1 r_1 (P_{ix} - x_{ix})$$

$$+ c_2 r_2 (G_{ix} - x_{ix})$$

همین را برای y هم داریم. حال مکان‌ها را update می‌شود

$$\left\{ \begin{array}{l} x_{ix}^{t+1} = x_{ix}^t + v_{ix}^{t+1} \\ x_{iy}^{t+1} = x_{iy}^t + v_{iy}^{t+1} \end{array} \right.$$

بدون جلسه سوم: Code 2 lecture 3.

در PSO ← خروجی بهترین تر است best پرنده‌ها

برای مقایسه چاپ کردن GD قابل قبول تر است آن PSO سریع‌تر است.

روش دیگر Opt از روش‌هاست.