Học lý thuyết

Học cài đặt: C++

DCA: Thuật toán giải các bài toán tối ưu không lồi

Min f(x) = g(x) – h(x) Diffrence of Convex functions Algorithm

Với x thuộc D

X: một véc tơ (x1, x2, x3, … xn, y1, y2,…ym, z1, z2,…zk) n+ m + k

Bước 1: X0 = một điểm khởi tạo, k=0

Repeat:

1. yk = GradientH(xk)
2. xk+1 = GradientG(yk)
3. xk = xk+1

until ||xk+1 – xk|| < epsilon

* Hiểu
* Dữ liệu như thế nào: thông thường dữ liệu đặt trong file, đọc file lưu vào biến (mảng)
* Đâu là biến, đâu là dữ liệu, kích thước của chúng, xác định các tham số
* Cài đặt:

Bước 1: Đọc được dữ liệu lên (chuẩn bị); Thiết lập các tham số: n = ? d = ? c = ?

Bước 2: Khởi tạo X0(U0, V0): Uk(cxn), Vk(cxd) (Học hàm để khởi tạo ngẫu nhiên), k=0

BƯỚC 3: VÀO DCA:

REPEAT:

Y = Xem công thức tính đạo hàm của H trong bài báo

Z = Xem công thức tính đạo hàm của H trong bài báo

Uk+1 = Trong bài báo

Vk+1 = Trong bài báo

T = ||U\_new-U|| + ||V\_new-V||

K = K+1

U = U\_new; V = V\_new;

Until T < epsilon

Code: DCA3 – FUZZY

Bước 1: Chuẩn bị bộ dữ liệu iris, c = 3;

Bước 2: Tạo project, đọc toàn bộ bộ dữ liệu lên mảng x, thiết lập tất cả các tham số của mô hình.

Bước 3: Code DCA: Code hàm tính đạo hàm H, Tính đạo hàm của G, Tính hàm mục tiêu, DCA

ĐỌC HIỂU

CODE: ĐỌC X, CHUẨN BỊ THAM SỐ

CODE: KHỞI TẠO (U, V) ngẫu nhiên

CODE: KHUNG DCA