

17/12/24

Lab 6

Grey Wolf Optimization

```
initialize-wolves (search-space, num-wolves):  
    dim ← len (search-space)  
    wolves ← zeros (num-wolves, dim)
```

```
    for i ← 0 to num-wolves:  
        wolves[i] ← random-uniform (  
            search-space[:, 0],  
            search-space[:, 1])
```

```
    return wolves
```

```
fitness-function (x):
```

```
    return np.sum(x**2)
```

```
GWO-algorithm (search-space, num-wolves,  
                max-iterations):
```

```
    dim ← len (search-space)
```

```
    wolves ← initialize-wolves (search-space,  
                                num-wolves)
```

```
    alpha ← zeros (dim)
```

```
    beta ← zeros (dim)
```

```
    gamma ← zeros (dim)
```

```
    alpha-fit = beta-fit = gamma-fit ← float('inf')
```

```
    best-fit ← float('inf')
```


for iter in range(max):

$a \leftarrow 2 - (\text{iteration}/\text{max}) * 2$

for $i \leftarrow 0$ to num-wolves:

fitness \leftarrow fitness-function(wolves[i])

if fitness < alpha-fitness

copy beta ~~cutoff~~^{wolf} & fitness to gamma

copy alpha ~~cutoff~~^{wolf} & fitness to beta

alpha-wolf \leftarrow wolves[i].copy()

alpha-fitness \leftarrow fitness

elif fitness < beta-fitness:

copy beta wolf & fitness to gamma

beta-wolf \leftarrow wolves[i].copy()

beta-fitness \leftarrow fitness

elif fitness < gamma-fitness:

gamma-wolf \leftarrow wolves[i].copy()

gamma-fitness \leftarrow fitness

if alpha-fitness < best-fitness:

best-fitness \leftarrow alpha-fitness

for $i \leftarrow 0$ to num-wolves:

for $j \leftarrow 0$ to dim:

$r1$ & $r2$ random floats

$A1 \leftarrow 2 * a * r1 - a$

$C1 \leftarrow 2 * r2$

$D\text{-alpha} \leftarrow \text{abs}(C1 * \text{alpha-wolf}[j] - \text{wolves}[i, j])$

$x1 \leftarrow \text{alpha-wolf}[j] - A1 * D\text{-alpha}$

New $r1$ & $r2$ random floats

$$A2 \leftarrow 2 * a * r1 - a$$

$$C2 \leftarrow 2 * r2$$

$$D\text{-beta} \leftarrow \text{abs}(C2 * \text{beta} - \text{wolf}[j] - \text{wolves}[i, j])$$

$$x2 \leftarrow \text{beta} - \text{wolf}[j] - A2 * D\text{-beta}$$

New $r1$ & $r2$ random floats

$$A3 \leftarrow 2 * a * r1 - a$$

$$C3 \leftarrow 2 * r2$$

$$D\text{-gamma} \leftarrow \text{abs}(C3 * \text{gamma} - \text{wolf}[j] - \text{wolves}[i, j])$$

$$x3 \leftarrow \text{gamma} - \text{wolf}[j] - A3 * D\text{-gamma}$$

$$\text{wolves}[i, j] \leftarrow (x1 + x2 + x3) / 3$$

$$\text{wolves}[i, j] \leftarrow \text{disp}(\text{wolves}[i, j], \text{search-space}[j, 0], \text{search-space}[j, 1])$$

Inputs:

search-space = array([-5, 5], [-5, 5])

num-wolves = 10

max-iter = 100