8/4/24

## Ant Colony:

### Algorithm:

step 1: Initialize the no. of ants(n) and its pheromone rate(p) over aim is to find the optimized (shortest path) from source to destination based on pheromone trail and the distance(b)

step 2: for each ant i in N:

```
di = dist[source, dest]   // we cale. dist of the path of ant 'i'
pi = pheromone_trail[source, dest, i]
// we calculate the pheromone level of ant i which is released while travelling from source to destination.
a[dist]
b[pheromon_trail]
a.add[di]
b.add[pi]
// we add the current dist. travelled by 'i' in list 'a' and pheromone_trail of 'i' in list 'b'.
```

step 3:

```
res = ∞, l = ∞, k = 0
for every i in a:
    for every j in b:
        l = min(l, ai)   // find minimum dist path
        k = max(k, bi)   // find max pheromone trail.
resi = l + k
if resi < res:
    res = resi
```

step 4: return res and also return the optimized path of the ant which gives the resi.

Pseudocode:

```
def ACO():
    best_route = None
    best_length = float('inf')
    for iteration in range(ITERATIONS):
        all_routes = []
        all_lengths = []
        for ant in range(NUM_ANTS):
            start_city = np.random.randint(NUM_CITIES)
            route = construct_route(start_city)
            route_length = calculate_route_length(route)
            all_routes.append(route)
            all_lengths.append(route_length)
            if route_length < best_length:
                best_length = route_length
                best_route = route
        update_pheromones(pheromones, all_routes, all_lengths)
    return best_route, best_length.
```

8/11/24

Output:

Best Route : [1, 4, 0, 3, 2]
Best Route Length : 227.345