EE412 Foundation of Big Data Analytics, Fall 2022
HW3

Name:고건호

Student ID:20160025

Discussion Group (People with whom you discussed ideas used in your answers):

On-line or hardcopy documents used as part of your answers:

Answer to Problem 1

(a) [20 pts] Solve the following problems, which are based on the exercises in the Mining of Massive Datasets 3rd edition (MMDS) textbook.

First of all, I designed an iterator function that calculates pagerank until the error compared to the value of the previous step is less than threshold.

```
STOP_THRESHOLD = 1e-5

def incremental_analysis(M, v, e, beta):
   new_v = np.zeros(v.shape) / v.shape[0]
   iterations = 0

   while np.linalg.norm(v - new_v) > STOP_THRESHOLD:
       v = new_v
       new_v = beta * M @ new_v + (1 - beta) * e
       iterations += 1

   elements = ['a','b','c','d']
   print(f"total iterations : {iterations}")
   print(f"Page Rank :", end=' ')
   for i, ele in enumerate(v.T[0]):
       print(f"{elements[i]}= {ele:.4f}", end=', ')
   print()
```

Exercise 5.1.2

```
M = np.array([
   [1/3, 1/2, 0  ],
   [1/3, 0  , 1/2],
   [1/3, 1/2, 1/2],
])
beta = 0.8
v = np.ones((3,1)) / 3
e = np.ones((3,1)) / 3
incremental_analysis(M, v, e, beta)
```

```
total iterations : 44
Page Rank : a= 0.2592, b= 0.3086, c= 0.4321,
```

Exercise 5.3.1

```
M = np.array([
   [0  , 1/2, 1  ,0  ],
   [1/3, 0  , 0  ,1/2],
   [1/3, 0  , 0  ,1/2],
   [1/3, 1/2, 0  ,0  ],
])
beta = 0.8

# (a)
print("\n(a)")
v = np.ones((4,1)) / 4
e = np.array([1,0,0,0]).reshape((4,1))
incremental_analysis(M, v, e, beta)

# (b)
print("\n(b)")
v = np.ones((4,1)) / 4
e = np.array([1,0,1,0]).reshape((4,1)) / 2
incremental_analysis(M, v, e, beta)
```

```
(a)
total iterations : 43
Page Rank : a= 0.4285, b= 0.1905, c= 0.1905, d= 0.1905,
```

```
(b)
total iterations : 43
Page Rank : a= 0.3857, b= 0.1714, c= 0.2714, d= 0.1714,
```

(b) [20 pts] Implement the PageRank algorithm using Spark.

- **Results**
  263 0.00216
  537 0.00212
  965 0.00202
  243 0.00197
  187 0.00194
  255 0.00191
  502 0.00191
  126 0.00191
  16 0.00190
  747 0.00190
  elapsed time: 19.06s

Answer to Problem 2

(a) [20 pts] Solve the following problems, which are based on the exercises in the MMDS textbook.

Exercise 10.3.2

First define the group of nodes on the left as A, and the group of nodes on the right as B.

The number of possible subsets of B with length t:

$$nCt$$

The number of subsets of B with length t that are connected with nodes from A:

$$\sum_{i}^{A} d_i Ct \ \geq \ n \times dCt \ (d_i:\ dimension\ of\ i^{th} node\ in\ A)$$

So we can find at least s duplicated subsets from subsets of B with length t that are connected with some node from A, when the value of s is:

$$s \ = \lceil \frac{n \times dCt}{nCt} \rceil$$

(a) n=20 and d=5.

(t, s) = (1, 5), (2, 2)

(b) n=200 and d=150.

(t, s) = (1, 150), (2, 113), (3, 84), (4, 63), (5, 47), (6, 35), (7, 26), (8, 20), (9, 15), (10, 11)

Exercise 10.5.2

(a) C = {w, x}; D = {y, z}

$$L(Likelihood) \ = \ p_{wx} p_{wy} p_{xy} p_{yz} (1 - p_{wz})(1 - p_{xz})$$

$$= \ P_C P_D \epsilon^2 (1 - \epsilon)^2$$

$$\leq \ \epsilon^2 \ (Equal\ when\ P_C \ = \ P_D \ = \ 1)$$

(b) C = {w, x, y, z}; D = {x, y, z}

$$L = p_{wx}p_{wy}p_{xy}p_{yz}(1 - p_{wz})(1 - p_{xz})$$

$$= P_C^{\,2}(1 - (1 - P_C)(1 - P_D))^2(1 - P_C)^2(1 - P_D)$$

$$= P_C^{\,2}(1 - P_C)P_{CD}^{\,2}(1 - P_{CD}) \quad where,\ P_{CD} = 1 - (1 - P_C)(1 - P_D)$$

Likelihood L is maximized when:

$$P_C = \frac{2}{3}\ and\ P_{CD} = \frac{2}{3}$$

This will result in:

$$P_D = 0$$

Finally, the Maximum Likelihood is:

$$L \le \frac{2^4}{3^6} = \frac{16}{729}\ (Equal\ when\ P_c = \frac{2}{3}\ and\ P_D = 0)$$

(b) [15 pts] Implement the algorithm for finding triangles in MMDS Chapter 10.7.2. You will analyze part of the Facebook (now Meta) social network to identify communities.

- **Results**

  3501542

  elapsed time: 32.34627413749695s

Answer to Problem 3

(a) [10 pts] Solve the following problems, which are based on the exercises in the MMDS textbook.

Exercise 12.5.3

(a) GINI impurity

$$f(x) = 1 - \sum_{i=1}^{2} p_i^2 = 1 - (x^2 + (1-x)^2) = 2x(1-x)$$

$$\frac{y-z}{y-x}f(x) + \frac{z-x}{y-x}f(y) = \frac{2}{y-x}(xy - x^2y - xz + x^2z + yz - y^2z - xy + xy^2)$$

$$= \frac{2}{y-x}(xy(y-x) + z(y-x) - z(y^2 - x^2))$$

$$= 2xy + 2z - 2z(x+y)$$

$$= 2z - 2(z-x)(z-y) - 2z^2$$

$$< 2z - 2z^2 = f(z) \text{ , since } (z-x)(z-y) < 0$$

(b) Entropy

$$f(x) = \sum_{i=1}^{2} p_i log_2(1/p_i) = -xlogx - (1-x)log(1-x)$$

$$\frac{y-z}{y-x}f(x) + \frac{z-x}{y-x}f(y)$$

$$= \frac{1}{y-x}(-xylogx - (y-xy)log(1-x) + xzlogz + (z-xz)log(1-x) -$$

$$\frac{1}{y-x}(-yzlogy - (z-yz)log(1-y) + xylogy - (x-xy)log(1-y))$$

$$= \frac{1}{y-x}(xylog(1-x) - xzlog(1-x) + yzlog(1-y) - xylog(1-y)) -$$

$$\frac{1}{y-x}(ylog(1-x) - zlog(1-x) + zlog(1-y) - xlog(1-y)) -$$

$$\frac{1}{y-x}(xylogx - xzlogx + yzlogy - xylogy) - 1$$

$* \; xylog(1-x) - xzlog(1-x) + yzlog(1-y) - xylog(1-y))$

$< xy(log(1-x) - log(1-y)) + yzlog(1-z) - xzlog(1-z)$

$< (y-x)zlog(1-z)$, since $log(1-x) > log(1-z) > log(1-y)$

$* \; ylog(1-x) - zlog(1-x) + zlog(1-y) - xlog(1-y))$

$> z(log(1-y) - log(1-x)) + ylog(1-z) - xlog(1-z)$

$> (y-x)log(1-z)$, since $log(1-x) > log(1-z) > log(1-y)$

$* \; xylogx - xzlogx + yzlogy - xylogy$

$> xy(logx - logy) + yzlogz - xzlogz$

$> (y-x)zlogz$, since $logx < logz < logy$

$\Rightarrow \frac{y-z}{y-x}f(x) + \frac{z-x}{y-x}f(y) < -(1-z)log(1-z) - zlogz = f(z)$

(b) [15 pts] Implement the gradient descent SVM algorithm described in MMDS Chapter 12.3.4 using Python.

0.8331666666666667

0.01

0.01

elapsed time: 56.59s