CS246: Mining Massive Data Sets Problem Set 1

Name: Dat Nguyen
Date: 05/09/2019

By turning in this assignment, I agree by the Stanford honor code and declare that all of this is my own work.

1 Spark (25 pts)

2. My pipeline:

   - For each person 'b' in the friend list of person 'a', get a list of friends of that person 'b'. Therefore if a person 'c' in that list then 'c' will have mutual friend 'b' with 'a'.
   - Count the number of people having mutual friend with 'a' by grouping and reducing with key 'a'.
   - Process the result (sort, output at most 10 people, output empty list if a has no person having mutual friend) and output to file.

3. Recommendation for:

   - 924: 439,2409,6995,11860,15416,43748,45881
   - 8941: 8943,8944,8940
   - 8942: 8939,8940,8943,8944
   - 9019: 9022,317,9023
   - 9020: 9021,9016,9017,9022,317,9023
   - 9021: 9020,9016,9017,9022,317,9023
   - 9022: 9019,9020,9021,317,9016,9017,9023
   - 9990: 13134,13478,13877,34299,34485,34642,37941
   - 9992: 9987,9989,35667,9991
   - 9993: 9991,13134,13478,13877,34299,34485,34642,37941

2 Association Rules (30 pts)

(a) This is a drawback because if support of B is high (B appears in a lot of baskets) then there are many item A having the number of times they appear together with B and the number of times they appear by themself roughly equal. So for many items the confidence will be high. Since lift and conviction take S(B) into account so we can see the difference between Pr(B) alone and when A is given.

(b) - Confidence is not symmetric because from

$$\text{conf}(A \to B) = \frac{S(A,B)}{S(A)}$$
$$\text{conf}(B \to A) = \frac{S(A,B)}{S(B)}$$

If we choose $S(A) = 0.3, S(B) = 0.2, S(A,B) = 0.1$ then $\text{conf}(A \to B) = \frac{1}{3}$ and $\text{conf}(B \to A) = 0.5$

- Lift is symmetric because

$$\begin{aligned}\text{lift}(A \to B) &= \frac{\text{conf}(A \to B)}{S(B)} \\ &= \frac{S(A,B)}{S(A)S(B)} \\ &= \frac{\text{conf}(B \to A)}{S(A)} \\ &= \text{lift}(B \to A)\end{aligned}$$

- Conviction is not symmetric because from

$$\begin{aligned}\text{conv}(A \to B) &= \frac{1 - S(B)}{1 - \text{conf}(A \to B)} \\ &= \frac{S(A) - S(A)S(B)}{S(A) - S(A,B)}\end{aligned}$$

$$\begin{aligned}\text{conv}(B \to A) &= \frac{1 - S(A)}{1 - \text{conf}(B \to A)} \\ &= \frac{S(B) - S(B)S(A)}{S(B) - S(A,B)}\end{aligned}$$

If we choose $S(A) = 0.4, S(B) = 0.3, S(A,B) = 0.1$ then $\text{conv}(A \to B) = \frac{14}{15}$ and $\text{conv}(B \to A) = 0.9$

(c) Confidence $\text{conf}(A \to B)$ is desirable because it reaches maximum value of 1 when $S(A,B) = S(A)$ (occurence of A implies occurence of B).
Lift is not desirable because when the rule is perfect (which implies $\text{conf}(A \to B)) = 1$, the value of lift can vary with the value of $S(B)$.
Conviction is also not desiable because when $\text{conf}(A \to B) = 1$ the denominator is 0 so the value of conviction is not defined.

(d) The rules and confidence scores are

- 'DAI93865' → 'FRO40251': 1.0
- 'GRO85051' → 'FRO40251': 0.999
- 'GRO38636' → 'FRO40251': 0.991
- 'ELE12951' → 'FRO40251': 0.991
- 'DAI88079' → 'FRO40251': 0.987

(e) The rules and confidence scores are

- ('DAI23334', 'ELE92920') → 'DAI62779': 1.0
- ('DAI31081', 'GRO85051') → 'FRO40251': 1.0
- ('DAI55911', 'GRO85051') → 'FRO40251': 1.0
- ('DAI62779', 'DAI88079') → 'FRO40251': 1.0
- ('DAI75645', 'GRO85051') → 'FRO40251': 1.0

3 Locality-Sensitive Hashing (15 pts)

(a) Suppose we have randomly chosen k rows, then the probability that none of the rows having 1 is equal to the probability that all of the 1's rows are in the remaining rows. Considering the first 1's row, the probability that it is the remaining rows is

$$\frac{n-k}{n}$$

The probability that the second 1's row is in the remaining rows is

$$\frac{n-k-1}{n-1} \leq \frac{n-k}{n}$$

Therefore the probability that all of the 1's rows are in the remaining rows is at most

$$\left(\frac{n-k}{n}\right)^m \quad \text{(q.e.d)}$$

(b) We want to find smallest k such that

$$\begin{aligned}
\left(\frac{n-k}{n}\right)^m &\leq e^{-10} \\
\left(1-\frac{k}{n}\right)^{\frac{n}{k}\frac{km}{n}} &\leq e^{-10} \\
e^{-\frac{km}{n}} &\leq e^{-10} \quad \text{(Because } n \gg k\text{)} \\
k &\geq \frac{10n}{m}
\end{aligned}$$

Therefore we choose k to be $\frac{10n}{m}$

(c) We choose S1 $= \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$ and S2 $= \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$

The Jaccard similarity of S1 and S2 is 0.5

The probability that a random cyclic permutation yields the same minhash value for both S1 and S2 is $\frac{4}{5} = 0.8$

4 LSH for Approximate Near Neighbor Search (30 pts)

(a) We have

$$\begin{aligned}
\Pr\Big[\sum_{j=1}^{L} |T \cap W_j| \geq 3L\Big] &\leq \frac{\mathbb{E}\Big[\sum_{j=1}^{L} |T \cap W_j|\Big]}{3L} \qquad \text{(By Markov's inequality)} \\
&= \frac{\sum_{j=1}^{L} \mathbb{E}\Big[|T \cap W_j|\Big]}{3L} \\
&= \frac{\sum_{j=1}^{L} \mathbb{E}\Big[\sum_{t \in T} \mathbb{1}[t \in W_j]\Big]}{3L} \\
&= \frac{\sum_{j=1}^{L} \sum_{t \in T} \mathbb{E}\Big[\mathbb{1}[t \in W_j]\Big]}{3L} \\
&= \frac{\sum_{j=1}^{L} \sum_{t \in T} \Pr\Big[t \in W_j\Big]}{3L} \\
&\leq \frac{\sum_{j=1}^{L} n p_2^{\log_{1/p_2}(n)}}{3L} \\
&= \frac{\sum_{j=1}^{L} 1}{3L} \\
&= \frac{1}{3} \qquad (1) \quad \text{(q.e.d)}
\end{aligned}$$

(b) We have

$$\begin{aligned}
\Pr\Big[\forall\ 1 \le j \le L, g_j(x^*) \ne g_j(z)\Big] &= \Big(\Pr[g_1(x^*) \ne g_j(z)]\Big)^L \\
&= \Big(1 - \Pr[g_1(x^*) = g_j(z)]\Big)^L \\
&\le \Big(1 - p_1^{-\log_{p_2}(n)}\Big)^{n^{\frac{\log(p_1)}{\log(p_2)}}} \\
&< \Big(\frac{1}{e}\Big)^{p_1^{-\log_{p_2}(n)} n^{\frac{\log(p_1)}{\log(p_2)}}} \qquad (2)\ (\text{Using } (1-\frac{1}{x})^x \approx \frac{1}{e} \text{ for large x})
\end{aligned}$$

We calculate the power of (2)

$$\begin{aligned}
p_1^{-\log_{p_2}(n)} n^{\frac{\log(p_1)}{\log(p_2)}} &= n^{-\log_n(p_1)\log_{p_2}(n)} n^{\frac{\log(p_1)}{\log(p_2)}} \\
&= n^{-\log_{p_2}(p_1)} n^{\frac{\log(p_1)}{\log(p_2)}} \\
&= n^{-\frac{\log(p_1)}{\log(p_2)} + \frac{\log(p_1)}{\log(p_2)}} \\
&= 1
\end{aligned}$$

Therefore plugging in (2) we arrive at

$$\Pr\Big[\forall\ 1 \le j \le L, g_j(x^*) \ne g_j(z)\Big] < \frac{1}{e} \quad (\text{q.e.d})$$

(c) Let A be the event that all of the points in 3L points we choose belonging to T. Because event A implies event $\sum_{j=1}^{L} |T \cap W_j| \ge 3L$, we have

$$\Pr(\text{A}) \le \Pr(\sum_{j=1}^{L} |T \cap W_j| \ge 3L) \le \frac{1}{3}$$

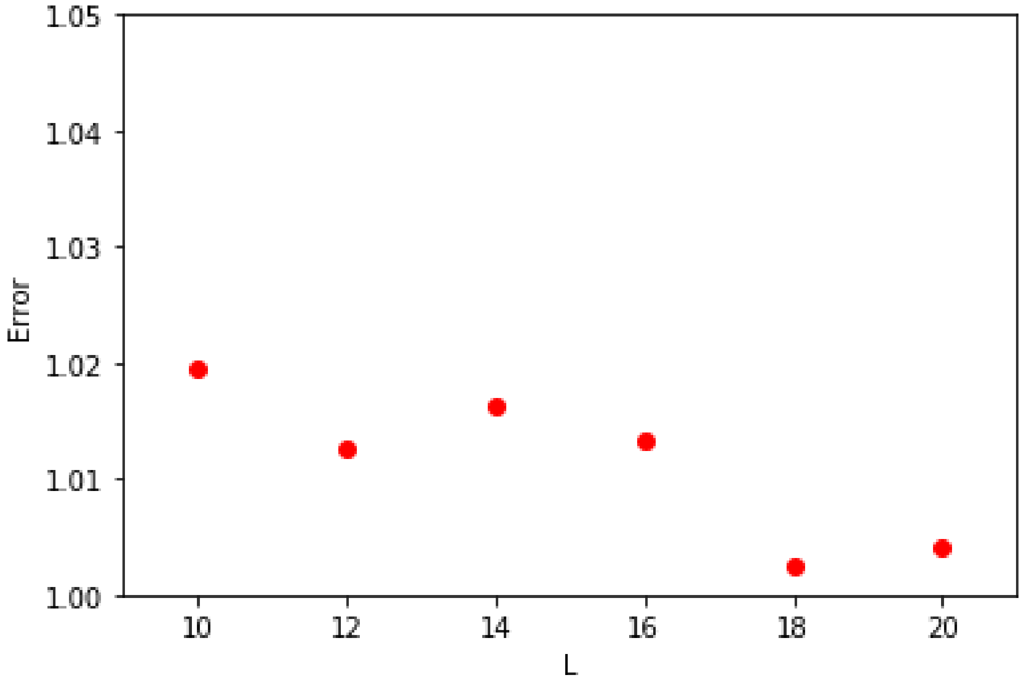
Therefore

$$\begin{aligned}
1 - \Pr(\text{A}) &\ge 1 - \frac{1}{3} \\
1 - \Pr(\text{A}) &\ge \frac{2}{3}
\end{aligned}$$

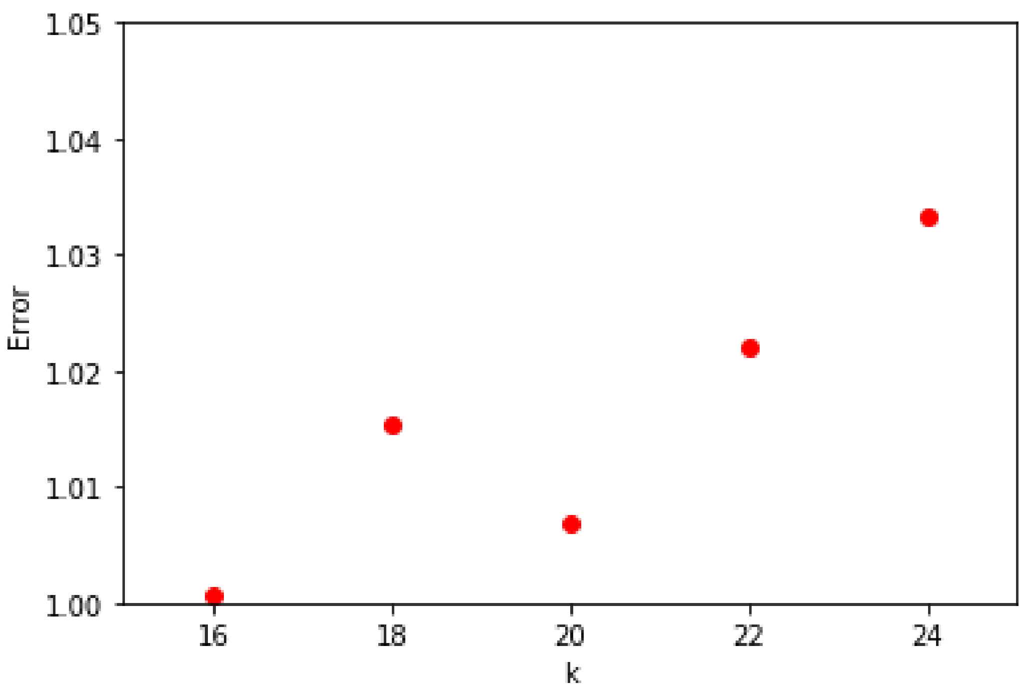
So the probability of the event that the reported point is an actual (c, λ)-ANN is greater than some fixed constant (let the constant be $\frac{1.99}{3}$).

(d)
- Average search time for LSH is 0.204s and for linear search is 0.517s

- Error value as function of L

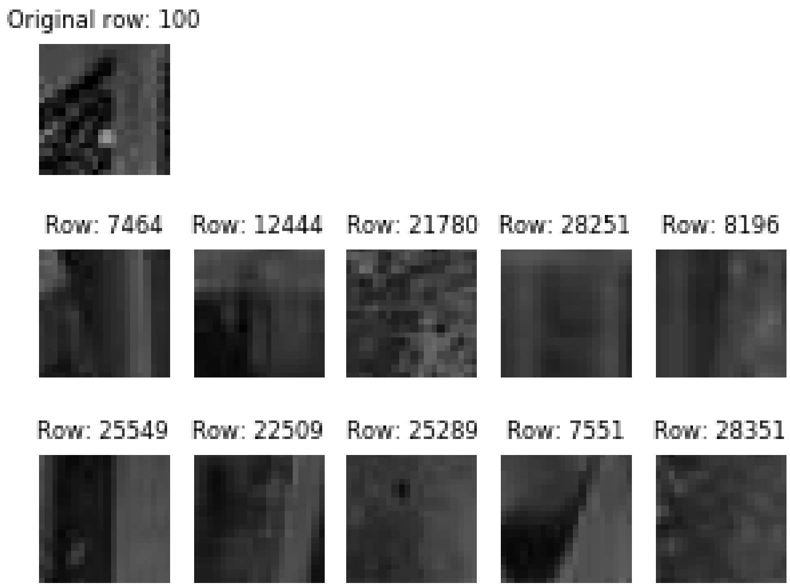


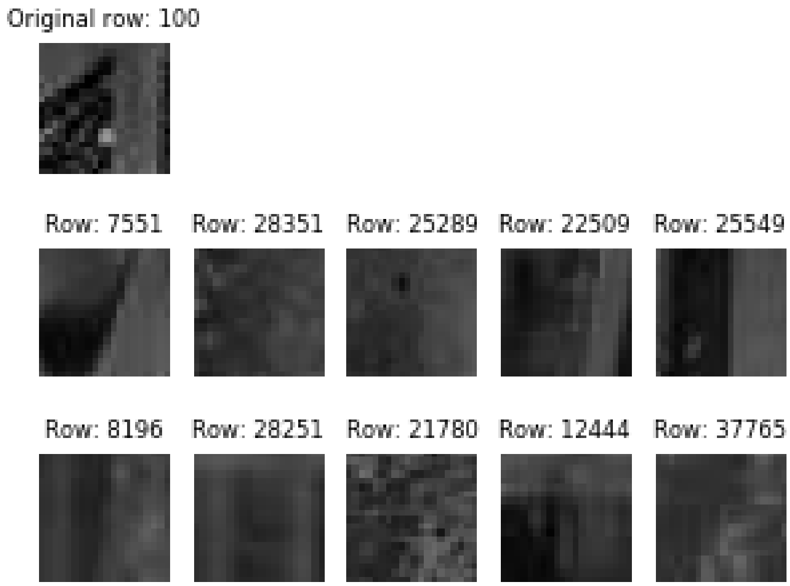
We can see the trend for larger L the error become smaller because we have more candidates for the best neighbors.
Error value as function of k



The general trend is that as k increases so does the error, because for larger k the candidates set shrinks.

- The top plot and bottom plot show 10 nearest neighbors found by linear search and lsh search respectively.





From 2 plots we can see that 9/10 neighbors found by lsh search match the ones found by linear search, and the remaining one (row 7551) reasonably resembles the original row.