COMPSCI 753

Algorithms for Massive Data

Semester 2, 2020

Tutorial 1: Locality-sensitive Hashing

Ninh Pham

1 Computing MinHash signatures and estimating Jaccard similarities

Given the 4 sets $S_1 = \{c, f\}$, S_2 tps.//spQ, S_3 pQ, S_4 oder. com

1. Present these sets as a binary management of the minhast values of	ent F	roje	ct F	Exam	Help
Solution: Assignment Add Elements	Proj	ect I	Exa	$m = \{0, \dots, M\}$	elp
Add Elements					ler
h b	1	0 1	0 0		
Add dw	eĞha	to po	0 1 WC(oder	
e f	5	1 0	0 0		

Table 1: Binary matrix presents the sets.

Integer	π	Elements	S_1	S_2	S_3	S_4
1	b	a	0	1	0	1
4	е	b	0	1	0	0
0	a	c	1	0	0	1
5	f	d	0	0	1	0
2	c	e	0	0	1	1
3	d	f	1	0	0	0

Following the procedure from the lecture note, we have the answer:

Hash function	S_1	S_2	S_3	S_4
π	a	b	c	a

2 Fast computing MinHash signatures

Since it is not feasible to permute a very large matrix explicitly, we will simulate random permutations by using random universal hash functions below:

$$h_1(x) = 2x + 1 \mod 6$$
, $h_2(x) = 3x + 2 \mod 6$, and $h_3(x) = 5x + 2 \mod 6$.

- 1. Compute the minhash values using these universal hash functions. Note that you have to map a string to an integer, e.g. $a \mapsto 0, b \mapsto 1, \ldots$
- 2. Which of these hash functions are true permutations?
- 3. How close are the estimated Jaccard similarities of the six pairs of columns to the true Jaccard similarities?

Solution:

https://powcoder.com

Integers	S_1	S_2	S_3	S_4	$h_1(x) = 2x + 1 \mod 6$	$h_2(x) = 3x + 2 \mod 6$	$h_3(x) = 5x + 2 \mod 6$
0	0	1	opc	ib	nment Proj	ect F ² vam l	Heln ²
1	_		_		· · · · · · · · · · · · · · · · · · ·	•	
2	A^1 S	\$01	\mathbf{p}_0	111	ent Project dd WeCha	Exam Hel	\mathbf{n}^{0}
3	0~	0		0 Λ	dd WeCha	t nowcode	r 5
							4
5	1	0	ht	tbs	s://powcod	er.com	3
				-			

_	Hash functions		S_1	S_2	S_3	S_4		
Add	1		ıąt	þ	O_2^1V	v <u>c</u>	oder	•
		$h_3(x)$	0	1	4	0		

Table 4: The minhash values with universal hash functions.

	S_1	S_2	S_3	S_4
S_1	1	0	0	1/4
S_2	0	1	0	1/3
S_3	0	0	1	1/4
S_4	1/4	1/3	1/4	1

Table 5: The actual Jaccard similarity values

	S_1	S_2	S_3	S_4
$\overline{S_1}$	1	1/3	1/3	2/3
S_2	1/3	1	2/3	2/3
S_3	1/3	2/3	1	2/3
S_4	2/3	2/3	2/3	1

Table 6: The estimated Jaccard similarity using these 3 hash functions.

3 Tuning the parameters for LSH

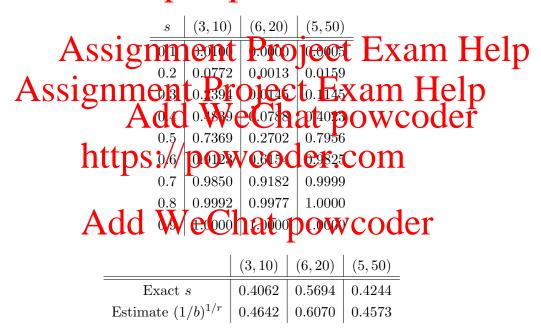
Evaluate the S-curve $1 - (1 - s^r)^b$, i.e. the probability of being a candidate pair, for $s = \{0.1, 0.2, \dots, 0.9\}$ using the following values of r and b.

- 1. r = 3 and b = 10.
- 2. r = 6 and b = 20.
- 3. r = 5 and b = 50.

For each value (r, b) above, compute the threshold, that is the value of s which the value of $1 - (1 - s^r)^b$ is exactly 1/2. How is it different from our approximation $(1/b)^{1/r}$? Which setting we should use in order to achieve the false negatives of 70%-similar pairs at most 5% and false positives of 30%-similar pairs at most 15%.

Solution:

https://powcoder.com



It is clearly that we need to use r=5 and b=50 since the probability of collision of 70%-similar pairs is 0.9999 and 30%-similar pairs is 0.1145.