# Minhashing

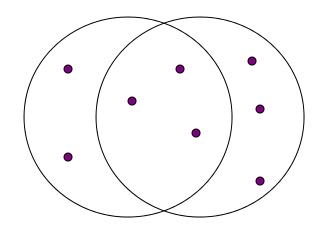
Jaccard Similarity Measure Constructing Signatures

#### **Jaccard Similarity**

- The Jaccard similarity of two sets is the size of their intersection divided by the size of their union.
- $Sim(C_1, C_2) = |C_1 \cap C_2| / |C_1 \cup C_2|$ .

雅卡尔指数(英语:Jaccard index),又称为并交比(Intersection over Union)、雅卡尔相似系数(Jaccard similarity coefficient),是用于比较样本集的相似性与多样性的统计量。雅卡尔系数能够量度有限样本集合的相似度,其定义为两个集合交集大小与并集大小之间的比例:

# **Example: Jaccard Similarity**



3 in intersection.8 in union.Jaccard similarity= 3/8

#### From Sets to Boolean Matrices

- Rows = elements of the universal set.
  - Example: the set of all k-shingles.
- Columns = sets.
- 1 in row e and column S if and only if e is a member of S.
- Column similarity is the Jaccard similarity of the sets of their rows with 1.
- Typical matrix is sparse.

#### **Example: Column Similarity**

可以想象成是顾客买了亚马逊的书,row是书的种类,col是不同的顾客。矩阵稀疏,因为每个人只会买少部分的书。 Column Similarity就可以找出顾客的相似性

Sim(
$$C_1$$
,  $C_2$ ) =  $2/5 = 0.4$ 

# Four Types of Rows

• Given columns  $C_1$  and  $C_2$ , rows may be classified as:

$$\begin{array}{cccc}
 & C_1 & C_2 \\
 a & 1 & 1 \\
 b & 1 & 0 \\
 c & 0 & 1 \\
 d & 0 & 0
\end{array}$$

- Also, a = # rows of type a, etc.
- Note  $Sim(C_1, C_2) = a/(a + b + c)$ .

#### Minhashing

https://my.oschina.net/keyven/blog/628898 在经过随机行打乱后,两个集合的最小哈希值相等的概率等于这两个集合的Jaccard相似度

- Imagine the rows permuted randomly.
- Define minhash function h(C) = the number of the first (in the permuted order) row in which column C has 1.
- Use several (e.g., 100) independent hash functions to create a signature for each column.
- The signatures can be displayed in another matrix – the signature matrix – whose columns represent the sets and the rows represent the minhash values, in order for that column.

https://blog.csdn.net/liujan511536/article/details/47729721 为了计算最小哈希,首先对特征矩阵的行进行打乱(也即随机调换行与行之间的位置),这 个打乱是随机的。然后某一列的最小哈希值就等于打乱后的这一列第一个值为1的行所在的行 号(不明白的直接看例子),行号从0开始。

# Minhashing Example

签名矩阵比特征矩阵小很多 https://blog.csdn.net/liujan511536/article/details/47729721

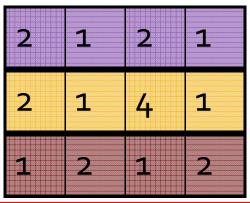
被打乱的顺序,按 照这个顺序找第一 个不为零的数

个为零	門数
	4 3
3	2 4
7	1 7
6	3 6
2	6 1
5	7 2
4	5 5

Input matrix

1	0	1	0
1	0	0	1
0	1	0	1
0	1	O	1
0	1	0	1
1	О	1	O
1	O	1	0

Signature matrix M



明显可以看出来,矩阵已经 被压缩了

#### **Surprising Property**

- The probability (over all permutations of the rows) that  $h(C_1) = h(C_2)$  is the same as  $Sim(C_1, C_2)$ .
- Both are a/(a+b+c)!
- Why?
  - Look down the permuted columns
     C<sub>1</sub> and C<sub>2</sub> until we see a 1.
  - If it's a type-a row, then  $h(C_1) = h(C_2)$ . If a type-b or type-c row, then not.

### Similarity for Signatures

- The similarity of signatures is the fraction of the minhash functions in which they agree.
  - Thinking of signatures as columns of integers, the similarity of signatures is the fraction of rows in which they agree.
- Thus, the expected similarity of two signatures equals the Jaccard similarity of the columns or sets that the signatures represent.
  - And the longer the signatures, the smaller will be the expected error.

几百次的minhash之后,这个ture jaccard和two signature 来衡量相似度的值是一样的, with 非常小的error.见17页ppt,最上

# Min Hashing – Example

#### Input matrix

1       4       3         3       2       4         7       1       7         6       3       6         2       6       1         5       7       2         4       5       5	M 100 MW 100 MW 100 MW 100 MW		
3       2       4         7       1       7         6       3       6         2       6       1         5       7       2			
3       2       4         7       1       7         6       3       6         2       6       1         5       7       2			
3       2       4         7       1       7         6       3       6         2       6       1         5       7       2			
3       2       4         7       1       7         6       3       6         2       6       1         5       7       2	8 10 L of 100 80 8		
2     6     1       5     7     2			
2     6     1       5     7     2			
2     6     1       5     7     2			
2     6     1       5     7     2			
2     6     1       5     7     2			
2     6     1       5     7     2			
2     6     1       5     7     2			
2     6     1       5     7     2		<del>                                     </del>	
2     6     1       5     7     2			
2     6     1       5     7     2			
2     6     1       5     7     2			
2     6     1       5     7     2			
2     6     1       5     7     2			
2     6     1       5     7     2			
2     6     1       5     7     2			
2     6     1       5     7     2			
<ul><li>6</li><li>7</li><li>2</li></ul>			
2     6     1       5     7     2			
2     6     1       5     7     2			
<ul><li>6</li><li>7</li><li>2</li></ul>			
<ul><li>6</li><li>7</li><li>2</li></ul>			
<ul><li>6</li><li>7</li><li>2</li></ul>		- 2	
5 7 2			
5 7 2			
5 7 2			
5 7 2			
5 7 2		ШЩ	
5 7 2			
5 7 2			
5 7 2 4 5 5			
5     7     2       4     5     5			
5     7     2       4     5     5			
5     7     2       4     5     5			
5     7     2       4     5     5			
<ul><li>5</li><li>7</li><li>2</li><li>5</li><li>5</li><li>5</li></ul>			
5 / Z 4 5 5			
<b>5 5</b>			
5 5			
5 5			
5 5			
5 5			
5 5			
5 5			
	01 (05 (06 (01 (05 (06 (01 (0		

1	O	1	0
1	0	0	1
0	1	0	1
0	1	O	1
0	1	О	1
1	О	1	0
1	0	1	0

#### Signature matrix M

iŏi	88	88	60	10	do	di	o i	8	80	100	sb		ı	********	00000	***	888	***		165	80	88	200	80	ook	00	6 10	100	bo	ö	
													п							器										8	
													н							儲										М	
													ı							日										Ħ	
		P		3	W								П	0000000	70000					18		×	=	31						81	000000000000000000000000000000000000000
				H)	я								п	000ad	8000					日				ш						И	
			12	2									ı	*****	1888					181			2	4						鹄	1000000 10000000000000000
		r	E		п								п		_					181		z	_							Ħ	1000
												88	ı							閪						88				Bl	
													ш							闘										И	
	8				18	88	81			8	88	81	п							88		88			88	81	88	188	100	揭	
ΒÜ	οÜ	οÖ	ρÖ	ηű	qü	œ	οį	Ø.		ĸ	16	a fi	ı		0000	000	000	0000	000	沒	οÜ	100	100	oli	οű	oli	40	Į.	Đ0	D)	000000000000000000000000000000000000000
				ı									٠																		
													۱							Н										Ē	
													ı							П										ä	
													t							п										й	
		ē	E	k									t							П			₹	. 1						ä	
		r	ī										ı							П			77							ã	The second secon
				7	a								ı							П		7	z	-						ä	
		r	£	r									ı							В		Z	Ξ	۰						ä	1000
													ı							П				п						ij	
													ı							Й										Ñ	
													ı							18										8	
				E	t	ı				Е	Ī	1	ı										Ħ					Е		Ħ	
							ı				ı		ı																		
			H	Hii	H	ä	늺		器		t		۱							á		蒙	욻	딂	븚				H		
ö	8	×	H	B	۰	÷	a	8	器	ä	٠	н	ı			***	*	***	***	器	8	器	쯃	8	7	88	8 20	×	to	쯃	
äi		88	Ħ	ta	t	×	Ħ	8	8		×	ä	Н							8	8		幂	81	8	B)	88	a		ä	
H				Ħ	Ħ	Ħ	a	Ħ	台		Ħ	Ħ	ı		-					8		靈	컱	젊				ti	H	ä	
äi	ä	Б	d	ő	Ħ	ä	ä	ö		ä	t	ä	ı		100					盲	篋	Е	ø	鄙		60	0 10		100	麗	
			io.	Ø		ø					Ŧ.	άĠ	۱		100					ú			ä			o li	a file			Ñ	
ãi		F			я	×	ä	8			ä	ää	ı		-					ã	8	e	2	=	8	88	86	8		ä	
Ŕ.				100	10	i i					ŧ.	a i	ı		0000					á			8	80		00	8 10			Ñ	
						Ø.					10	nii	ı							100						10.0	8 80			Ħ	
												ΒÜ	۱							10						obi	10			Ŕ	
Ħ	86	Ø.	pi	Ti.	Į.	ĸ.		86	Di.	(c)	ij.	e i				1000	2000			Si.	80	86	eei				100	100	100	Ñ	

	1-3	2-4	1-2
Col/Col	0.75	0.75	0
Sig/Sig	<b>Q</b> .67	1.00	0

就是2/3

# Implementation of Minhashing

- Suppose 1 billion rows.
- Hard to pick a random permutation of 1...billion.
- Representing a random permutation requires
   1 billion entries.
- Accessing rows in permuted order leads to thrashing.

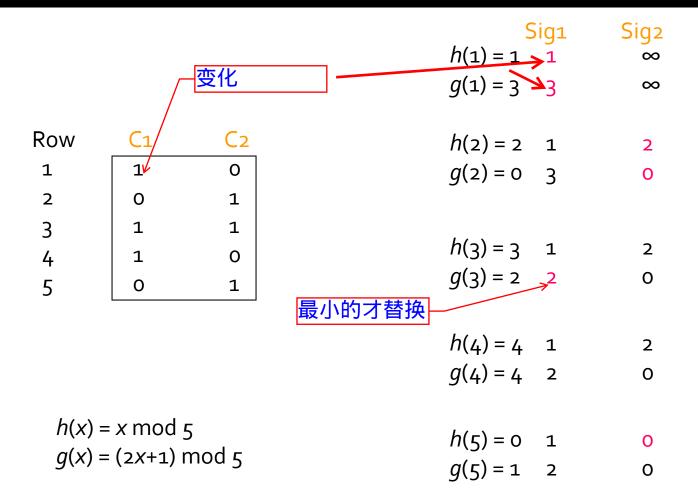
#### Implementation — (2)

- A good approximation to permuting rows: pick, say, 100 hash functions.
- For each column c and each hash function h<sub>i</sub>, keep a "slot" M(i, c).
- Intent: M(i, c) will become the smallest value of  $h_i(r)$  for which column c has 1 in row r.
  - I.e.,  $h_i(r)$  gives order of rows for  $i^{th}$  permutation.

#### Implementation – (3)

```
for each row r do begin
  for each hash function h<sub>i</sub> do
      compute h_i(r);
  for each column c
      if c has 1 in row r
        for each hash function h_i do
           if h_i(r) is smaller than M(i, c) then
              M(i, c) := h_i(r);
 end;
```

#### Example



#### Implementation – (4)

- Often, data is given by column, not row.
  - Example: columns = documents, rows = shingles.
- If so, sort matrix once so it is by row.

Start with a list of row column pairs where the ones are. Initially sort it by column, and sort these pairs by row.