Finding biclusters by random projections

Stefano Lonardi *Qiaofeng Yang*
University of California, Riverside
Wojciech Szpankowski
Purdue University, West Lafayette

What is a bicluster

- Given a matrix over a finite alphabet
- Remove some of the columns and some of the rows
- Each row of what is left read the same string (row-wise)

Can you find the largest bicluster?

ABAACDADBBAABCDBBBCCABCBBAABBBDCDDCBCCAADAAB
CCACBDABDCADBBDDBAABBBBACAAACABDDCCDADDBDDBD
BBBBCCCDCDCCCDACDADABCABCACDADCBBDCDBACDDBBC
CCBCBDCCBCCAABDCBABCDBBAAAAACCDDDCCDBDBADDBD
CCCCBDCDACBCACDABCDCCBBADAABCCDDDCCDBDCDBDBD
CDCDABCACABDABACCDABCCCBACBACBBAADAAACACCBCC
CCDCBDCBACDBDCDDBCCAABBCDABCCCDDDCCDCDCACDBD
CCDCBDCCACBBBADDBCADABBABDCDCADDDCCDBDDDCDBD
CCAACACACABDDCDBDACDDCDAADCCAAACDBDBBBBABDDA
CCBCBDCCACACCDDABADBCBBABBDCCADDDCCDCDBCADBD

Stefano Lonardi
Department of CS and E
Bourns College of Engineering
University of California, Riverside

This is one … is it the largest?

ABAAC**D**ADBBAABC**D**B**B**BCCA**B**CBB**AA**BBBDC**D**D**C**BCCAADAAB
CCACB**D**ABDCADBB**D**D**B**AABB**B**BAC**AA**ACABD**D**C**C**DADDBDDBD
BBBBCCCDCDCCCDACDADABCABCACDADCBBDCDBACDDBBC
CCBCB**D**CCBCCAAB**D**C**B**ABCD**B**BAA**AA**ACCDD**D**C**C**DBDBADDBD
CCCCB**D**CDACBCAC**D**A**B**CDCC**B**BAD**AA**BCCDD**D**C**C**DBDCDBDBD
CDCDABCACABDABACCDABCCCBACBACBBAADAAACACCBCC
CCDCBDCBACDBDCDDBCCAABBCDABCCCDDDCCDCDCACDBD
CCDCBDCCACBBBADDBCADABBABDCDCADDDCCDBDDDCDBD
CCAACACACABDDCDBDACDDCDAADCCAAACDBDBBBBABDDA
CCBCBDCCACACCDDABADBCBBABBDCCADDDCCDCDBCADBD

Area=4×8=32

Stefano Lonardi
Department of CS and E
Bourns College of Engineering
University of California, Riverside

One more time …

ABAACDADBBAABCDBBBCCABCBBAABBBDCDDCBCCAADAAB
CCACBDABDCADBBDDBAABBBBACAAACABDDCCDADDBDDBD
BBBBCCCDCDCCCDACDADABCABCACDADCBBDCDBACDDBBC
CCBCBDCCBCCAABDCBABCDBBAAAAACCDDDCCDBDBADDBD
CCCCBDCDACBCACDABCDCCBBADAABCCDDDCCDBDCDBDBD
CDCDABCACABDABACCDABCCCBACBACBBAADAAACACCBCC
CCDCBDCBACDBDCDDBCCAABBCDABCCCDDDCCDCDCACDBD
CCDCBDCCACBBBADDBCADABBABDCDCADDDCCDBDDDCDBD
CCAACACACABDDCDBDACDDCDAADCCAAACDBDBBBBABDDA
CCBCBDCCACACCDDABADBCBBABBDCCADDDCCDCDBCADBD

Stefano Lonardi
Department of CS and E
Bourns College of Engineering
University of California, Riverside

Was this the one you found?

ABAACDADBBAABCDBBBCCABCBBAABBBDCDDCBCCAADAAB
CCA**CBD**ABD**C**ADBB**D**D**B**AABB**BB**ACAAA**C**AB**DDCCD**A**D**DBD**DBD**
BBBBCCCDCDCCCDACDADABCABCACDADCBBDCDBACDDBBC
CCB**CBD**CCB**C**CAAB**D**C**B**ABCD**BB**AAAAA**C**CD**DDCCD**B**D**BAD**DBD**
CCC**CBD**CDA**C**BCAC**D**A**B**CDCC**BB**ADAAB**C**CD**DDCCD**B**D**CDB**DBD**
CDCDABCACABDABACCDABCCCBACBACBBAADAAACACCBCC
CCD**CBD**CBA**C**DBDC**D**D**B**CCAA**BB**CDABC**C**CD**DDCCD**C**D**CAC**DBD**
CCD**CBD**CCA**C**BBBA**D**D**B**CADA**BB**ABDCD**C**AD**DDCCD**B**D**DDC**DBD**
CCAACACACABDDCDBDACDDCDAADCCAAACDBDBBBBABDDA
CCB**CBD**CCA**C**ACCD**D**A**B**ADBC**BB**ABBDC**C**AD**DDCCD**C**D**BCA**DBD**

Area=6×20=120

Stefano Lonardi
Department of CS and E
Bourns College of Engineering
University of California, Riverside

The general problem

- Biclustering is the problem of finding a partition of the vectors and a subset of the dimensions such that the projections along those directions of the vectors in each cluster are close to one another

- The problem requires to cluster the vectors and the dimensions simultaneously, thus the name “biclustering”

Questions

- How difficult is the problem of finding large biclusters?

- How to find them efficiently?

Stefano Lonardi
Department of CS and E
Bourns College of Engineering
University of California, Riverside

Applications

- Collaborative filtering and recommender systems
- Finding web communities
- Discovery association rules in databases
- Gene expression analysis
- …

Stefano Lonardi
Department of CS and E
Bourns College of Engineering
University of California, Riverside

Related works

- Hartigan, ‘72
- Aggarwal *et al.*, SIGMOD’99
- Cheng & Church, ISMB’00
- Wang *et al.*, SIGMOD’02
- Ben-Dor *et al.*, RECOMB’02
- Tanay *et al.*, ISMB’02
- Procopiuc *et al.*, SIGMOD’02
- Murali & Kasif, PSB’03
- Sheng *et al.*, ECCB’03
- Mishra *et al.*, COLT’03

Stefano Lonardi
Department of CS and E
Bourns College of Engineering
University of California, Riverside

Problem definition

LARGEST_BICLUSTER(***f***) problem

- **Instance:** A matrix $X \in \Sigma^{n \times m}$
- **Question:** Find a row selection *R* and a column selection *C* such that the rows of $X_{(R,\,C)}$ read the same string and $f(X_{(R,\,C)})$ is maximized

Stefano Lonardi
Department of CS and E
Bourns College of Engineering
University of California, Riverside

Examples of objective functions

$$f_1\left(X_{(R,C)}\right) = |R| + |C|$$

$$f_2\left(X_{(R,C)}\right) = |R| \text{ provided that } |C| = |R|$$

$$f_3\left(X_{(R,C)}\right) = |R||C|$$

f_1: Maximum Vertex Biclique – polytime
f_2: Balanced Biclique – hard
f_3: Maximum Edge Biclique - hard

Stefano Lonardi
Department of CS and E
Bourns College of Engineering
University of California, Riverside

Randomized search

Assume $X \in \Sigma^{n \times m}$ contains a maximal bicluster (R^*, C^*). Assume we know $|R^*| = r^*$ and $|C^*| = c^*$.

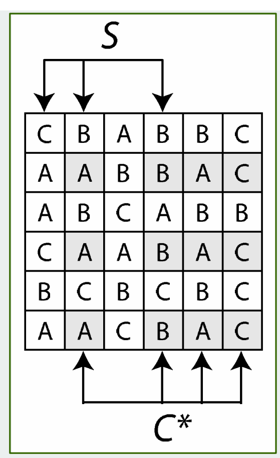
Observation:

- If we knew R^*, then C^* could be obtained
- If we knew C^*, then R^* could be obtained

Stefano Lonardi
Department of CS and E
Bourns College of Engineering
University of California, Riverside

Randomized search (step 1)

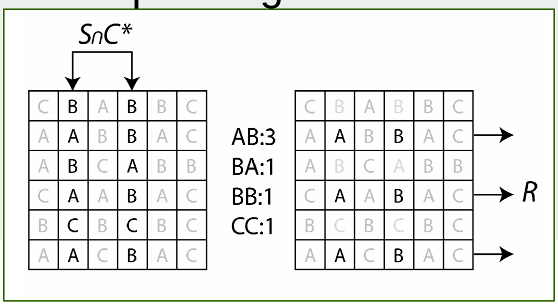
Select a random subset *S* of size *k* uniformly from the set of columns *{1, 2, …, m}*



Stefano Lonardi
Department of CS and E
Bourns College of Engineering
University of California, Riverside

Randomized search (step 2)

For all the subset of *S*, find the string *w* that appears at least $\hat{r}$ times in each subset of *S* and record the corresponding rows *R* in which *w* occurs

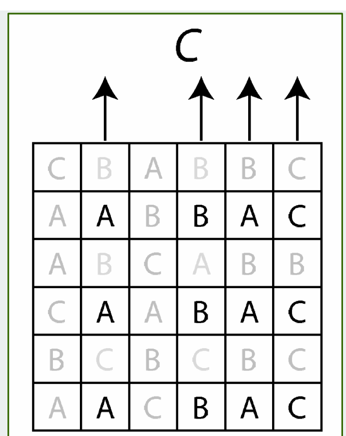


Stefano Lonardi
Department of CS and E
Bourns College of Engineering
University of California, Riverside

Randomized search (step 3)

Select the set of clean columns *C* with size at least $\hat{c}$ corresponding to each *R*

- A column *j* is *clean* with respect to *R* if the symbols in the *j*-th column of *X* restricted to the rows *R*, are identical



Stefano Lonardi
Department of CS and E
Bourns College of Engineering
University of California, Riverside

Randomized search (step 4)

Save the solutions and repeat step 1 to 4 for *t* iterations

Stefano Lonardi
Department of CS and E
Bourns College of Engineering
University of California, Riverside

Parameters

- Projection size *k* (k_{min})
- Column threshold $\hat{c}$
- Row threshold $\hat{r}$
- Number of iterations *t*

Stefano Lonardi
Department of CS and E
Bourns College of Engineering
University of California, Riverside

Selecting the projection size *k*

- Occurrences of substrings are
  - Gaussian distributed for strings shorter than $\log_a m$
  - Poisson distributed for strings longer than $\log_a m$

  where $a = |\Sigma|$
- Choose $k = \log_a m$
- Choose $k_{min} = 1$ or $k_{min} = k$

Stefano Lonardi
Department of CS and E
Bourns College of Engineering
University of California, Riverside

Selecting the number of iterations *t*

- We can miss a solution in two cases
  - *S* completely misses *C**
  - when *S* overlaps *C**, and the string *w* selected by the algorithm also appears in a row outside *R**

Stefano Lonardi
Department of CS and E
Bourns College of Engineering
University of California, Riverside

Selecting the number of iterations *t*

- The probability of missing the solution in one iteration is

$$\alpha(n,m,k,r^*,c^*,a) = \Pr\{S \cap C^* = \emptyset\} + \sum_{i=1}^{k} \Pr\{|S \cap C^*| = i \text{ and } |R| > r^*\}$$
$$= \Pr\{S \cap C^* = \emptyset\} + \sum_{i=1}^{k} \Pr\{|R| > r^* \text{ given } |S \cap C^*| = i\} \Pr\{|S \cap C^*| = i\}$$

which is

$$\alpha(n,m,k,r^*,c^*,a) = \left(\binom{m-c^*}{k} + \sum_{i=1}^{k}\left(1-\left(1-\frac{1}{a^i}\right)^{n-r^*}\right)\binom{c^*}{i}\binom{m-c^*}{k-i}\right) \Big/ \binom{m}{k}$$

Stefano Lonardi
Department of CS and E
Bourns College of Engineering
University of California, Riverside

Selecting the number of iterations t

- Given the probability of missing the solution in *t* iterations to be smaller than *e*

$$t \geq \frac{\log \epsilon}{\log \alpha(n,m,k,r^*,c^*,a)}$$

Stefano Lonardi
Department of CS and E
Bourns College of Engineering
University of California, Riverside

Selecting the number of iterations *t*

ϵ	$a=2,k=8$	$a=4,k=4$	$a=8,k=3$	$a=16,k=2$	$a=32,k=2$
0.005	18794	1342	306	179	99
0.05	10626	759	173	101	56
0.1	8168	583	133	78	43
0.2	5709	408	93	54	30
0.3	4271	305	70	41	23
0.4	3250	232	53	31	17
0.5	2459	176	40	23	13
0.6	1812	129	29	17	10
0.7	1265	90	21	12	7
0.8	792	57	13	8	4
0.9	374	27	6	4	2

Table 1: The estimated number of iterations for a matrix 256×256 with a submatrix 64×64, for different choices of ϵ, alphabet size a, and projection size k (sampling columns)

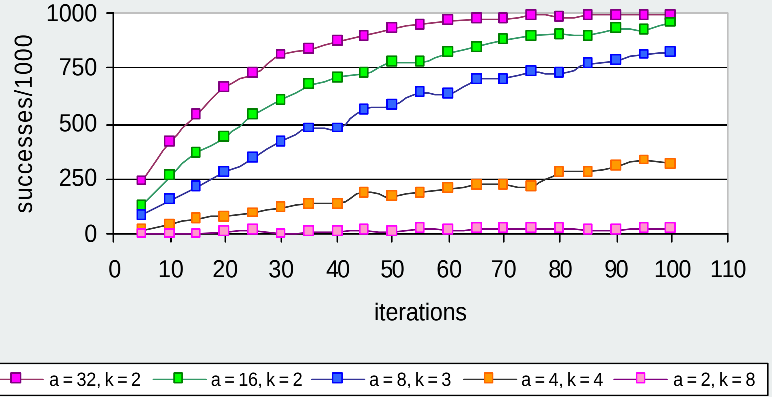
Stefano Lonardi
Department of CS and E
Bourns College of Engineering
University of California, Riverside

Simulations

- Generate 1,000 random matrices of size 256x256 over an alphabet of size *a*
- In each, embed a bicluster of size 64x64 (random content, random positions)
- Run the algorithm for *t* iterations (*t*=5,10,…,100) and compute how many successes out of 1,000

Stefano Lonardi
Department of CS and E
Bourns College of Engineering
University of California, Riverside

Simulation result (column sampling)



Performance of the randomized algorithm for different choices of the alphabet size a ($k_{min} = 1$, $k = log_a m$)

Findings

- Simple and fast randomized algorithm to find large biclusters in text matrices
- Probabilistic analysis of performance
- Simulations

- Next: approximate biclusters?

Stefano Lonardi
Department of CS and E
Bourns College of Engineering
University of California, Riverside