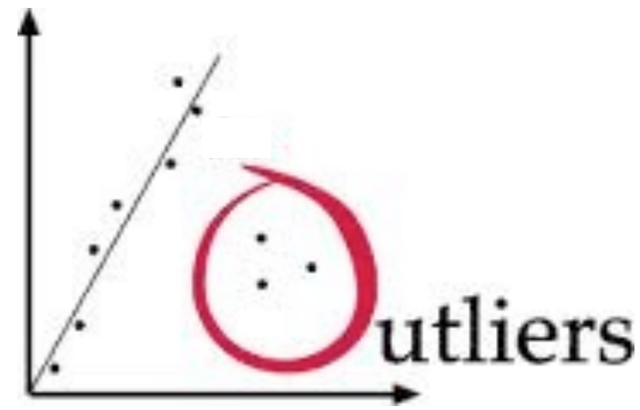
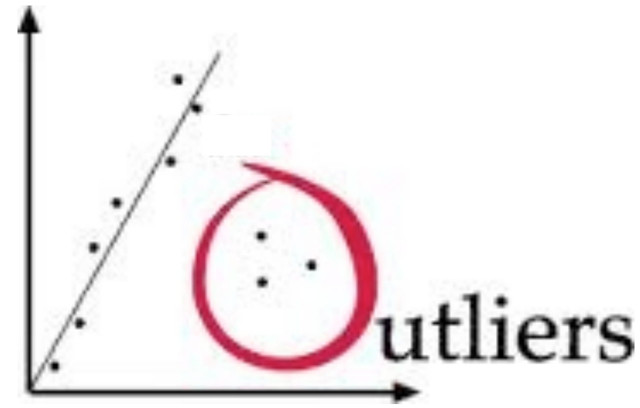
On Approximating String Selection

Problems with 

Christina Boucher, Gad M. Landau, Avivit Levy,
David Pritchard, Oren Weimann

On Approximating String Selection

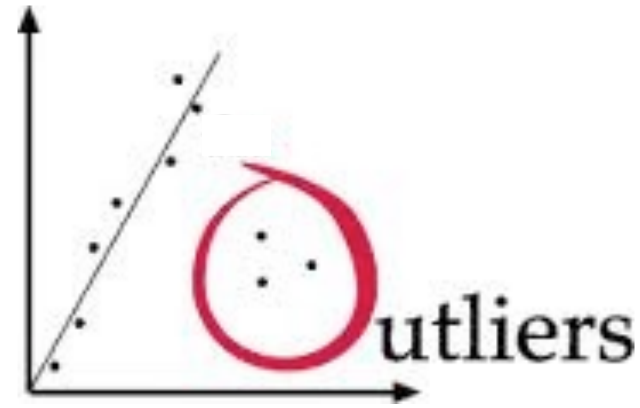
Problems with



s_1 =	b	a	n	a	n	a
s_2 =	g	a	n	a	n	a
s_3 =	a	p	p	l	e	s
s_4 =	b	a	m	a	n	a
s_n =	b	a	m	a	m	a

On Approximating String Selection

Problems with



s_1 =	b	a	n	a	n	a
s_2 =	g	a	n	a	n	a
s_3 =	a	p	p	l	e	s
s_4 =	b	a	m	a	n	a
s_n =	b	a	m	a	m	a

All the others are of small hamming distance

Problem I: CloseToMostStrings

Given d, find a string s maximizing the number of strings whose distance from s is $\leq$ d

s_1 =	b	a	n	a	n	a
s_2 =	g	a	n	a	n	a
s_3 =	a	p	p	l	e	s
s_4 =	b	a	m	a	n	a
s_n =	b	a	m	a	m	a

Problem 1: CloseToMostStrings

Given d, find a string s maximizing the number of strings whose distance from s is ≤ d

s_1 =	b	a	n	a	n	a
s_2 =	g	a	n	a	n	a
s_3 =	a	p	p	l	e	s
s_4 =	b	a	m	a	n	a
s_n =	b	a	m	a	m	a

s =	b	a	n	a	n	a

Problem I: CloseToMostStrings

Given d, find a string s maximizing the number of strings whose distance from s is $\leq d = 1$

s_1 =	b	a	n	a	n	a
s_2 =	g	a	n	a	n	a
s_3 =	a	p	p	l	e	s
s_4 =	b	a	m	a	n	a
s_n =	b	a	m	a	m	a

s =	b	a	n	a	n	a

All the others are of distance > 1 from s

Problem I: CloseToMostStrings

Given d, find a string s maximizing the number of strings whose distance from s is ≤ d

s_1 =	b	a	n	a	n	a
s_2 =	g	a	n	a	n	a
s_3 =	a	p	p	l	e	s
s_4 =	b	a	m	a	n	a
s_n =	b	a	m	a	m	a

Theorem 1: The problem has no PTAS unless ZPP = NP

Problem I: CloseToMostStrings

Given d, find a string s maximizing the number of strings whose distance from s is ≤ d

s_1 =	b	a	n	a	n	a
s_2 =	g	a	n	a	n	a
s_3 =	a	p	p	l	e	s
s_4 =	b	a	n	a	n	a
s_n =	b	a	m	a	m	a

Theorem 1: The problem has no PTAS unless ZPP = NP

~~Theorem [CPM'00]: The problem has no PTAS unless P = NP~~

Problem 1: CloseToMostStrings
≡ FarFromMostStrings

in binary alphabet

s_1 =	0	1	0	0	1	1
s_2 =	1	1	0	0	1	1
s_3 =	0	0	1	1	1	0
s_4 =	1	0	1	1	0	0
s_n =	0	0	0	1	1	0

Theorem 1: The problem has no PTAS unless ZPP = NP

~~Theorem [CPM'00]: The problem has no PTAS unless P = NP~~

Problem 1: CloseToMostStrings

$\equiv$ FarFromMostStrings

no PTAS unless P = NP

[Lanctot et al. SODA'99]

in binary alphabet

s_1 =	0	1	0	0	1	1
s_2 =	1	1	0	0	1	1
s_3 =	0	0	1	1	1	0
s_4 =	1	0	1	1	0	0
s_n =	0	0	0	1	1	0

Theorem 1: The problem has no PTAS unless ZPP = NP

~~Theorem [CPM'00]: The problem has no PTAS unless P = NP~~

Problem 1: CloseToMostStrings

≡ FarFromMostStrings

no PTAS unless P = NP

[Lanctot et al. SODA'99]

Not true in binary alphabet

s_1 =	0	1	0	0	1	1
s_2 =	1	1	0	0	1	1
s_3 =	0	0	1	1	1	0
s_4 =	1	0	1	1	0	0
s_n =	0	0	0	1	1	0

Theorem 1: The problem has no PTAS unless ZPP = NP

~~Theorem [CPM'00]: The problem has no PTAS unless P = NP~~

Problem I: CloseToMostStrings

≡ FarFromMostStrings

in binary alphabet

s_1 =	0	1	0	0	1	1
s_2 =	1	1	0	0	1	1
s_3 =	0	0	1	1	1	0
s_4 =	1	0	1	1	0	0
s_n =	0	0	0	1	1	0

no PTAS unless P = NP
[Lanctot et al. SODA'99]

no PTAS unless ZPP=NP
[here]

Theorem 1: The problem has no PTAS unless ZPP = NP

~~Theorem [CPM'00]: The problem has no PTAS unless P = NP~~

CloseToMostStrings

$s_1 =$	0	1	0	0	1	1
$s_2 =$	1	1	0	0	1	1
$s_3 =$	0	0	1	1	1	0
$s_4 =$	1	0	1	1	0	0
$s_n =$	0	0	0	1	1	0

Theorem 1: The problem has no PTAS unless ZPP = NP

Theorem 1: The problem has no PTAS unless ZPP = NP

CloseToMostStrings

s_1 =	0	1	0	0	1	1
s_2 =	1	1	0	0	1	1
s_3 =	0	0	1	1	1	0
s_4 =	1	0	1	1	0	0
s_n =	0	0	0	1	1	0

Theorem 1: The problem has no PTAS unless ZPP = NP

CloseToMostStrings

s_1 =	0	1	0	0	1	1
s_2 =	1	1	0	0	1	1
s_3 =	0	0	1	1	1	0
s_4 =	1	0	1	1	0	0
s_n =	0	0	0	1	1	0

Theorem 1: The problem has no PTAS unless ZPP = NP

Proof: Randomized reduction from Max-2-SAT

CloseToMostStrings

$s_1 =$	0	1	0	0	1	1
$s_2 =$	1	1	0	0	1	1
$s_3 =$	0	0	1	1	1	0
$s_4 =$	1	0	1	1	0	0
$s_n =$	0	0	0	1	1	0

Theorem 1: The problem has no PTAS unless ZPP = NP

Proof: Randomized reduction from Max-2-SAT

Max-2-SAT

$x_1 \vee x_2$

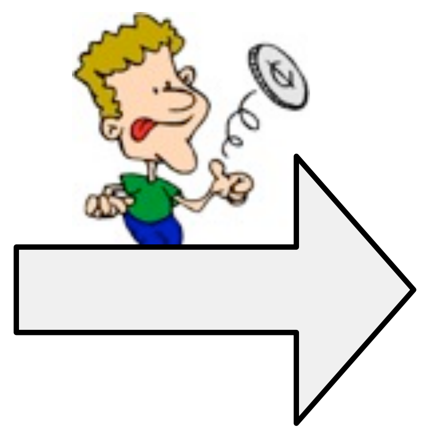
$\overline{x_1} \vee x_2$

$x_1 \vee \overline{x_3}$

$\overline{x_1} \vee \overline{x_3}$

$\overline{x_2} \vee x_3$

$x_3 \vee x_4$



CloseToMostStrings

s_1 =	0	1	0	0	1	1
s_2 =	1	1	0	0	1	1
s_3 =	0	0	1	1	1	0
s_4 =	1	0	1	1	0	0
s_n =	0	0	0	1	1	0

Theorem 1: The problem has no PTAS unless ZPP = NP

Proof: Randomized reduction from Max-2-SAT

Max-2-SAT

$x_1 \vee x_2$
$\overline{x_1} \vee x_2$
$x_1 \vee \overline{x_3}$
$\overline{x_1} \vee \overline{x_3}$
$\overline{x_2} \vee x_3$
$x_3 \vee x_4$

CloseToMostStrings

	x_1		x_2		x_3		x_4	
s_1 =	1	1	1	1	0	1	0	1
s_2 =	0	0	1	1	0	1	0	1
s_3 =	1	1	0	1	0	0	0	1
s_4 =	0	0	0	1	1	1	0	1
s_5 =	0	1	0	0	1	1	0	1
s_6 =	0	1	0	1	1	1	1	1

$$s_j(2i-1)s_j(2i) = \begin{cases} 00 & \text{if } \omega_j \text{ contains the literal } \overline{x_i}, \\ 11 & \text{if } \omega_j \text{ contains the literal } x_i, \\ 01 & \text{otherwise.} \end{cases}$$

Theorem 1: The problem has no PTAS unless ZPP = NP

Proof: Randomized reduction from Max-2-SAT

Max-2-SAT

$x_1 \vee x_2$
$\overline{x_1} \vee x_2$
$x_1 \vee \overline{x_3}$
$\overline{x_1} \vee \overline{x_3}$
$\overline{x_2} \vee x_3$
$x_3 \vee x_4$

CloseToMostStrings

s_1 =	1	1	1	1	0	1	0	1
s_2 =	0	0	1	1	0	1	0	1
s_3 =	1	1	0	1	0	0	0	1
s_4 =	0	0	0	1	1	1	0	1
s_5 =	0	1	0	0	1	1	0	1
s_6 =	0	1	0	1	1	1	1	1

uniformly random from $\{01,10\}^n$

0	1	1	0	0	1	0	1
0	1	1	0	1	0	0	1
1	0	0	1	1	0	0	1
1	0	1	0	0	0	0	1
0	1	1	0	1	0	0	1

Theorem 1: The problem has no PTAS unless ZPP = NP

Proof: Randomized reduction from Max-2-SAT

Max-2-SAT

$x_1 \vee x_2$
$\overline{x_1} \vee x_2$
$x_1 \vee \overline{x_3}$
$\overline{x_1} \vee \overline{x_3}$
$\overline{x_2} \vee x_3$
$x_3 \vee x_4$

CloseToMostStrings

s_1 =	1	1	1	1	0	1	0	1
s_2 =	0	0	1	1	0	1	0	1
s_3 =	1	1	0	1	0	0	0	1
s_4 =	0	0	0	1	1	1	0	1
s_5 =	0	1	0	0	1	1	0	1
s_6 =	0	1	0	1	1	1	1	1

0	1	1	0	0	1	0	1
0	1	1	0	1	0	0	1
1	0	0	1	1	0	0	1
1	0	1	0	0	0	0	1
0	1	1	0	1	0	0	1

define a string $\widehat{x}$ via

$$\widehat{x}(2i-1)\widehat{x}(2i) = \begin{cases} 11 & \text{if } x_i \text{ is true,} \\ 00 & \text{if } x_i \text{ is false.} \end{cases}$$

Theorem 1: The problem has no PTAS unless ZPP = NP

Proof: Randomized reduction from Max-2-SAT

Max-2-SAT

$x_1 \vee x_2$
$\overline{x_1} \vee x_2$
$x_1 \vee \overline{x_3}$
$\overline{x_1} \vee \overline{x_3}$
$\overline{x_2} \vee x_3$
$x_3 \vee x_4$

define a string $\widehat{x}$ via

$$\widehat{x}(2i-1)\widehat{x}(2i) = \begin{cases} 11 & \text{if } x_i \text{ is true,} \\ 00 & \text{if } x_i \text{ is false.} \end{cases}$$

CloseToMostStrings

s_1 =	1	1	1	1	0	1	0	1
s_2 =	0	0	1	1	0	1	0	1
s_3 =	1	1	0	1	0	0	0	1
s_4 =	0	0	0	1	1	1	0	1
s_5 =	0	1	0	0	1	1	0	1
s_6 =	0	1	0	1	1	1	1	1

at distance n from all random strings

0	1	1	0	0	1	0	1
0	1	1	0	1	0	0	1
1	0	0	1	1	0	0	1
1	0	1	0	0	0	0	1
0	1	1	0	1	0	0	1

Theorem 1: The problem has no PTAS unless ZPP = NP

Proof: Randomized reduction from Max-2-SAT

Max-2-SAT

$x_1 \vee x_2$
$\overline{x_1} \vee x_2$
$x_1 \vee \overline{x_3}$
$\overline{x_1} \vee \overline{x_3}$
$\overline{x_2} \vee x_3$
$x_3 \vee x_4$

at distance ≤ n iff satisfies the clause

CloseToMostStrings

s_1 =	1	1	1	1	0	1	0	1
s_2 =	0	0	1	1	0	1	0	1
s_3 =	1	1	0	1	0	0	0	1
s_4 =	0	0	0	1	1	1	0	1
s_5 =	0	1	0	0	1	1	0	1
s_6 =	0	1	0	1	1	1	1	1

define a string $\widehat{x}$ via

$$\widehat{x}(2i-1)\widehat{x}(2i) = \begin{cases} 11 & \text{if } x_i \text{ is true,} \\ 00 & \text{if } x_i \text{ is false.} \end{cases}$$

at distance n from all random strings

0	1	1	0	0	1	0	1
0	1	1	0	1	0	0	1
1	0	0	1	1	0	0	1
1	0	1	0	0	0	0	1
0	1	1	0	1	0	0	1

Theorem 1: The problem has no PTAS unless ZPP = NP

Proof: Randomized reduction from Max-2-SAT

Lemma 1: W.h.p any string s with distance ≤ n from cm strings is of the form $\{00,11\}^n$

Max-2-SAT ⟵ CloseToMostStrings

$x_1 \vee x_2$
$\overline{x_1} \vee x_2$
$x_1 \vee \overline{x_3}$
$\overline{x_1} \vee \overline{x_3}$
$\overline{x_2} \vee x_3$
$x_3 \vee x_4$

at distance ≤ n iff satisfies the clause

s_1 =	1	1	1	1	0	1	0	1
s_2 =	0	0	1	1	0	1	0	1
s_3 =	1	1	0	1	0	0	0	1
s_4 =	0	0	0	1	1	1	0	1
s_5 =	0	1	0	0	1	1	0	1
s_6 =	0	1	0	1	1	1	1	1

define a string $\widehat{x}$ via

$$\widehat{x}(2i-1)\widehat{x}(2i) = \begin{cases} 11 & \text{if } x_i \text{ is true,} \\ 00 & \text{if } x_i \text{ is false.} \end{cases}$$

at distance n from all random strings

0	1	1	0	0	1	0	1
0	1	1	0	1	0	0	1
1	0	0	1	1	0	0	1
1	0	1	0	0	0	0	1
0	1	1	0	1	0	0	1

Theorem 1: The problem has no PTAS unless ZPP = NP

Proof: Randomized reduction from Max-2-SAT

Lemma 1: W.h.p any string s with distance ≤ n from cm strings is of the form $\{00,11\}^n$

Proof: Uses the probabilistic method

$x_1 \lor x_2$
$\overline{x_1} \lor x_2$
$x_1 \lor \overline{x_3}$
$\overline{x_1} \lor \overline{x_3}$
$\overline{x_2} \lor x_3$
$x_3 \lor x_4$

at distance ≤ n iff satisfies the clause

s_1 =	1	1	1	1	0	1	0	1
s_2 =	0	0	1	1	0	1	0	1
s_3 =	1	1	0	1	0	0	0	1
s_4 =	0	0	0	1	1	1	0	1
s_5 =	0	1	0	0	1	1	0	1
s_6 =	0	1	0	1	1	1	1	1

define a string $\widehat{x}$ via

$$\widehat{x}(2i-1)\widehat{x}(2i) = \begin{cases} 11 & \text{if } x_i \text{ is true,} \\ 00 & \text{if } x_i \text{ is false.} \end{cases}$$

at distance n from all random strings

0	1	1	0	0	1	0	1
0	1	1	0	1	0	0	1
1	0	0	1	1	0	0	1
1	0	1	0	0	0	0	1
0	1	1	0	1	0	0	1

Problem II: ClosestTokStrings

Given k, find a string s and a subset of k input strings S such that maximum $d(s, s_i \in S)$ is minimized

$s_1 =$	b	a	n	a	n	a
$s_2 =$	g	a	n	a	n	a
$s_3 =$	a	p	p	l	e	s
$s_4 =$	b	a	n	a	n	a
$s_n =$	b	a	m	a	m	a

Problem II: ClosestTokStrings

Given k, find a string s and a subset of k input strings S such that maximum $d(s, s_i \in S)$ is minimized = 1

k = 4

s_1 =	b	a	n	a	n	a
s_2 =	g	a	n	a	n	a
s_3 =	a	p	p	l	e	s
s_4 =	b	a	m	a	m	a
s_n =	b	a	m	a	n	a

s =	b	a	n	a	n	a

Problem II: ClosestTokStrings

Given k, find a string s and a subset of k input strings S such that maximum $d(s, s_i \in S)$ is minimized = 1

s_1 =	b	a	n	a	n	a
s_2 =	g	a	n	a	n	a
s_3 =	a	p	p	l	e	s
s_4 =	b	a	n	a	n	a
s_n =	b	a	m	a	m	a

k = n

The ClosestString problem

Extensive Hardness, Approximation, and FPT research:

[Frances, Litman TCS'97], [Lanctot, Li, Ma, Wang, Zhang SODA'99], [Ma, CPM'00], [Li, Ma, Wang J. of computer and Sys. Sci. 2002], [Gramm, Niedermeier, Rossmanith Algorithmica'03], [Ma, Sun SICOMP'09], [Wang, Zhu FAW'09], [Chen, Ma, Wang COCOON'10], [Amir, Paryenty, Roditty SPIRE'11], [Lokshtanov, Marx, Saurabh SODA'11]

Problem II: ClosestTokStrings

Given k, find a string s and a subset of k input strings S such that maximum $d(s, s_i \in S)$ is minimized

s_1 =	b	a	n	a	n	a
s_2 =	g	a	n	a	n	a
s_3 =	a	p	p	l	e	s
s_4 =	b	a	n	a	n	a
s_n =	b	a	m	a	m	a

Observation 1: The known PTAS [Ma. CPM'00] for ClosesTokStrings cannot be improved to an EPTAS, unless W[1] = FPT.

Problem II: ClosestTokStrings

Given k, find a string s and a subset of k input strings S such that maximum $d(s, s_i \in S)$ is minimized

$s_1 =$	b	a	n	a	n	a
$s_2 =$	g	a	n	a	n	a
$s_3 =$	a	p	p	l	e	s
$s_4 =$	b	a	n	a	n	a
$s_n =$	b	a	m	a	m	a

$(1+\varepsilon)$-approx in $O(n^{f(\varepsilon)})$

Observation 1: The known PTAS [Ma. CPM'00] for ClosesTokStrings cannot be improved to an EPTAS, unless W[1] = FPT.

Problem II: ClosestTokStrings

Given k, find a string s and a subset of k input strings S such that maximum $d(s, s_i \in S)$ is minimized

s_1 =	b	a	n	a	n	a
s_2 =	g	a	n	a	n	a
s_3 =	a	p	p	l	e	s
s_4 =	b	a	n	a	n	a
s_n =	b	a	m	a	m	a

$(1+\varepsilon)$-approx in $O(n^{f(\varepsilon)})$

Observation 1: The known PTAS [Ma. CPM'00] for ClosesTokStrings cannot be improved to an EPTAS, unless W[1] = FPT.

$(1+\varepsilon)$-approx in $O(f(\varepsilon)\ \mathrm{poly}(n))$

Problem II: ClosestTokStrings

Given k, find a string s and a subset of k input strings S such that maximum $d(s, s_i \in S)$ is minimized

s_1 =	b	a	n	a	n	a
s_2 =	g	a	n	a	n	a
s_3 =	a	p	p	l	e	s
s_4 =	b	a	n	a	n	a
s_n =	b	a	m	a	m	a

$(1+\varepsilon)$-approx in $O(n^{f(\varepsilon)})$

Observation 1: The known PTAS [Ma. CPM'00] for ClosesTokStrings cannot be improved to an EPTAS, unless W[1] = FPT.

$(1+\varepsilon)$-approx in $O(f(\varepsilon)\ \mathrm{poly}(n))$

standard assumption in FPT

Problem II: ClosestTokStrings

Given k, find a string s and a subset of k input strings S such that maximum $d(s, s_i \in S)$ is minimized

s_1 =	b	a	n	a	n	a
s_2 =	g	a	n	a	n	a
s_3 =	a	p	p	l	e	s
s_4 =	b	a	n	a	n	a
s_n =	b	a	m	a	m	a

Proof:
Decision version has no FPT [Boucher, Ma 2011]

An EPTAS implies FPT.

Observation 1: The known PTAS [Ma. CPM'00] for ClosesTokStrings cannot be improved to an EPTAS, unless W[1] = FPT.

Problem III: FewBadColumns

Given k, find largest subset of strings with ≤ k *bad columns*

s_1 =	b	a	n	a	n	a
s_2 =	g	a	n	a	n	a
s_3 =	a	p	p	l	e	s
s_4 =	b	a	n	a	n	a
s_n =	b	a	m	a	m	a

Problem III: FewBadColumns

Given k, find largest subset of strings with ≤ k *bad columns*

k = 2

s_1 =	b	a	n	a	n	a
s_2 =	g	a	n	a	n	a
s_3 =	a	p	p	l	e	s
s_4 =	b	a	n	a	n	a
s_n =	b	a	m	a	m	a

Problem III: FewBadColumns

Given k, find largest subset of strings with ≤ k *bad columns*

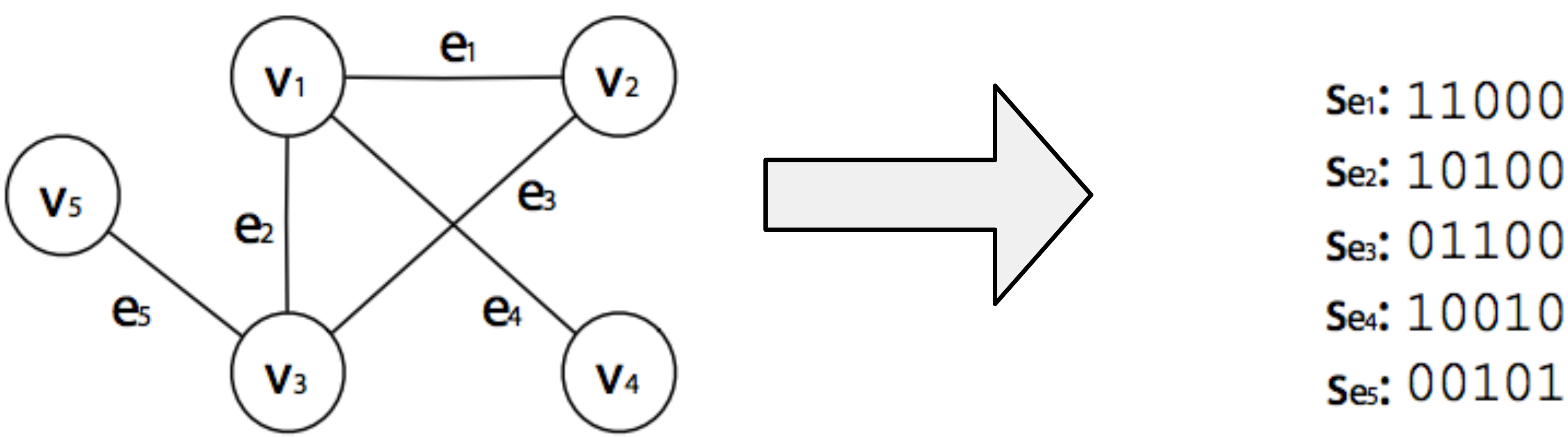
s_1 =	b	a	n	a	n	a
s_2 =	g	a	n	a	n	a
s_3 =	a	p	p	l	e	s
s_4 =	b	a	n	a	n	a
s_n =	b	a	m	a	m	a

Theorem 2: The problem has no PTAS unless P = NP

Problem III: FewBadColumns

Given k, find largest subset of strings with ≤ k *bad columns*

Densest-k-Subgraph has no PTAS [Khot SICOMP'06] → FewBadColumns



Theorem 2: The problem has no PTAS unless P = NP

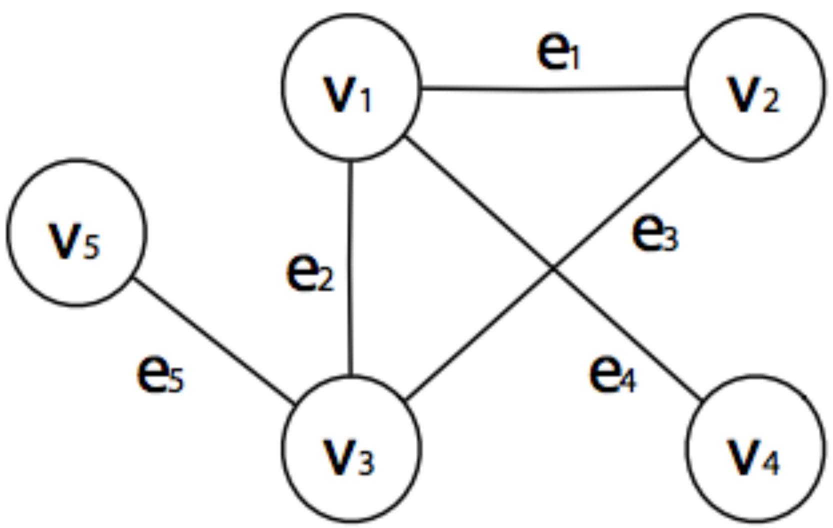
Problem III: FewBadColumns

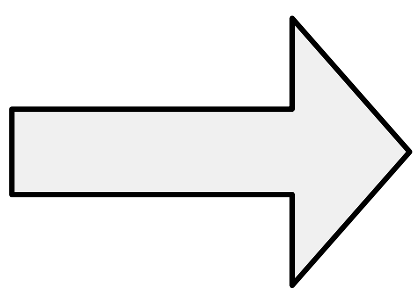
Given k, find largest subset of strings with ≤ k *bad columns*

Densest-k-Subgraph

has no PTAS [Khot SICOMP'06]

FewBadColumns





s_0: 00000
s_{e1}: 11000
s_{e2}: 10100
s_{e3}: 01100
s_{e4}: 10010
s_{e5}: 00101

Theorem 2: The problem has no PTAS unless P = NP

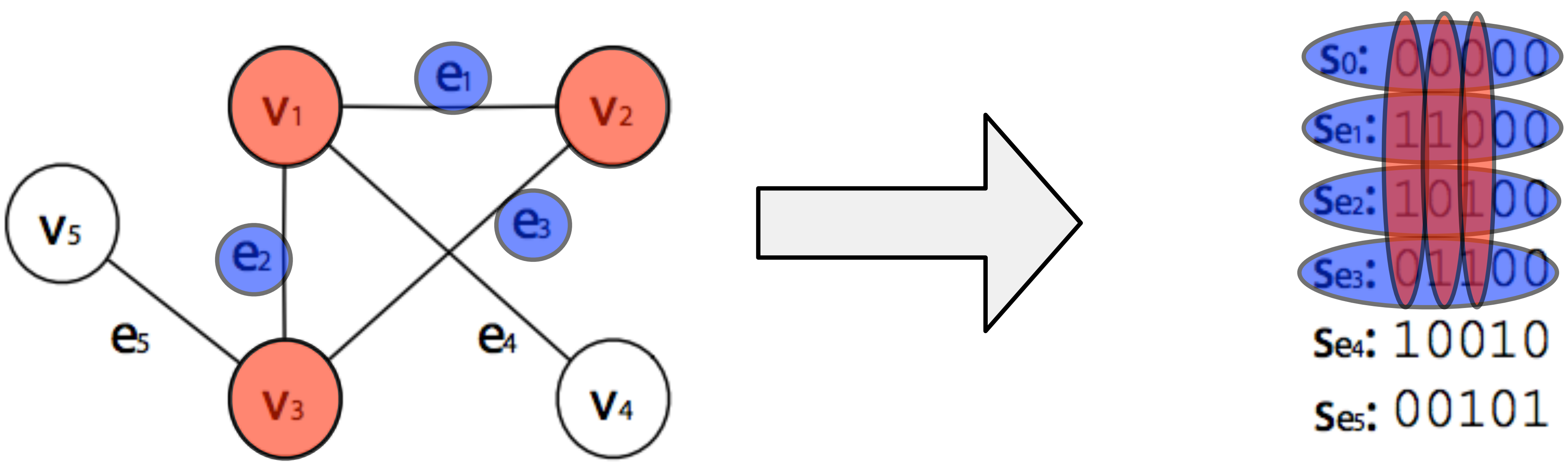
Problem III: FewBadColumns

Given k, find largest subset of strings with ≤ k *bad columns*

Densest-k-Subgraph

has no PTAS [Khot SICOMP'06]

FewBadColumns



Theorem 2: The problem has no PTAS unless P = NP

Open problems:

- Is there a deterministic reduction for CloseToMostStrings?
(to get NP=P assumption and not ZPP=NP)

- Is there a constant-factor approximation for CloseToMostStrings?
(even for binary alphabets)

- Is there a constant-factor approximation for MostStringsWithFewBadColumns?
(even for binary alphabets)

- Is there an EPTAS for CloseTokStrings for binary alphabets?

- Is there an EPTAS for ClosesestString?

Thank You!