Экспериментальное исследование приближенных алгоритмов для задач кластеризации вершин графа

Моршинин Александр

Институт математики им. С.Л. Соболева СО РАН (Омский филиал)

Исследование выполнено за счет гранта Российского научного фонда No 22-71-10015

Задачи кластеризации

В *задаче кластеризации* требуется разбить заданное множество объектов на несколько подмножеств (*кластеров*) исходя из сходства объектов между собой.

Если представить объекты в виде вершин обыкновенного графа, а сходство в виде ребер графа, то мы получим формализацию задачи кластеризации – *задачу кластеризации вершин графа*.

Существует минимизационный и максимизационный вариант этой задачи. В этом варианте необходимо минимизировать количество *несогласованностей* (число ребер между кластерами + количество отсутствующих ребер внутри кластеров).

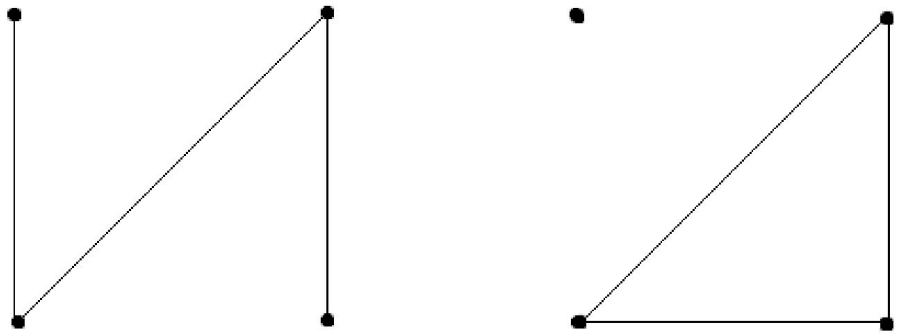
Основные определения

Кластерный граф – граф, каждая компонента связности которого есть полный граф.

Расстояние между двумя обыкновенными помеченными графами определяется как количество отсутствующих ребер в этих графах.

Окрестностью вершины v будем называть множество вершин графа $G = (V, E)$, смежных с v.

Пример кластерного графа



Для превращения графа в кластерный нужно удалить одно ребро и добавить одно ребро.

Классическая задача кластеризации

Задача $\mathbf{GC}_{\leq k}$ ([k]-GRAPH CLUSTERING). Дан произвольный граф $G = (V, E)$ и целое число k (верхнее ограничение на количество кластеров). Необходимо найти такой граф M^*, который является ближайшим к G кластерным графом с не более чем k кластерами.

3-приближенный алгоритм для задачи 2-кластеризации

Алгоритм **BBC** (Bansal-Blum-Chawla).

Шаг 1. Для каждой вершины $v \in V$ произвольного графа $G = (V, E)$ определить первый кластер как сама вершина и ее окрестность. Второй кластер – оставшиеся вершины. Полученный кластерный граф обозначим через M_v.

Шаг 2. Среди всех графов M_v выбрать ближайший к G кластерный граф M_{BBC}.

Трудоемкость алгоритма **BBC** – $O(n^2)$.

Локальный поиск для задачи 2-кластеризации

Алгоритм $\mathbf{LS}_{\leq 2}$ (Local Search for no more than 2 components).

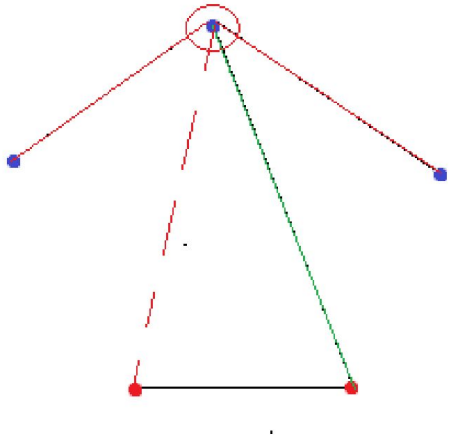
Итерация k.

Шаг 1. Для каждой вершины $v \in V$ произвольного графа $G = (V, E)$. посчитать значение целевой функции, которое получится после переноса вершины в противоположный кластер.

Шаг 2. Выбрать вершину с наибольшим уменьшением значения целевой функции. Если таковой нет, остановить алгоритм.

Трудоемкость алгоритма – $O(n^3)$.

Локальный поиск для задачи 2-кластеризации



При переносе выделенной синей вершины в красный кластер в значении целевой функции не будет участвовать зеленое ребро, но начнут участвовать красные ребра. Поэтому значение целевой функции вырастет на 2.

2-приближенный алгоритм для задачи 2-кластеризации

Алгоритм **CSW** (Coleman-Saunderson-Wirth).

Шаг 1. Пусть F – множество всех допустимых решений, построенных алгоритмом **BBC**. Применить процедуру $\mathbf{LS}_{\leq 2}$ к каждому кластерному графу из множества F.

Шаг 2. Среди всех локальных оптимумов выбрать ближайший к G кластерный граф M_{CSW}.

Трудоемкость алгоритма **CSW** – $O(n^4)$.

Экспериментальное исследование алгоритмов для задачи 2-кластеризации

Рассмотрим эвристический алгоритм $\mathbf{N1LS}_{\leq 2}$, применяющий алгоритм локального поиска лишь к лучшему допустимому решению.

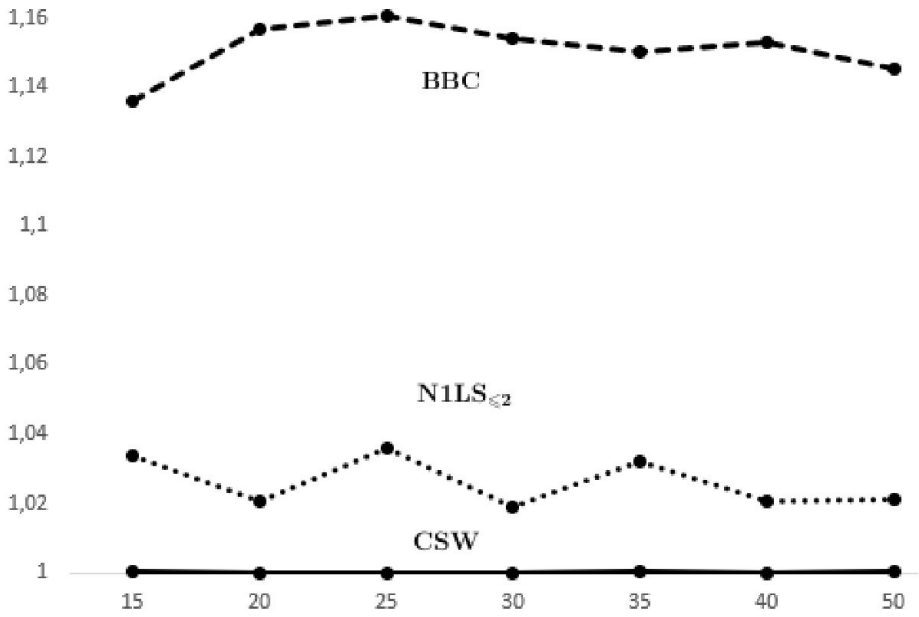
Для генерации случайных графов была применена *модель Эрдеша-Реньи* со значением параметра p из множества {0.33, 0.5, 0.67}.

Точностью алгоритма будем называть отношения значения функции на решении, полученному этим алгоритмом, к оптимальному значению целевой функции.

Эксперимент на маленьких графах

Эксперимент показал, что результаты разительно отличаются на разреженных и плотных графах. Так, при $p = 0.33$ и $p = 0.5$ средняя ошибка алгоритмов **BBC** и $\mathbf{N1LS}_{\leq 2}$ имела тенденцию к убыванию. Максимальное значение средней ошибки алгоритмов **BBC** и $\mathbf{N1LS}_{\leq 2}$ составило порядка 16% и 4 % в случае $p = 0.33$, $n = 20$. Средняя ошибка алгоритма **CSW** была около 0.

Эксперимент на маленьких графах

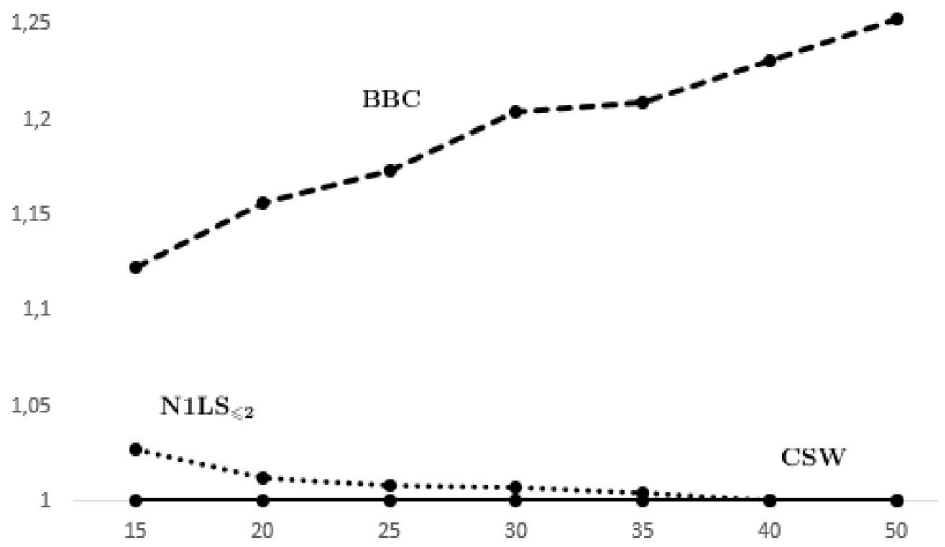


Средняя точность алгоритмов на графах малой размерности при $p = 0.33$.

Эксперимент на маленьких графах

В случае $p = 0.67$ поведение алгоритмов **BBC** и $\mathbf{N1LS}_{\leq 2}$ сильно меняется. Так, средняя ошибка алгоритма $\mathbf{N1LS}_{\leq 2}$ стремится к 0 с ростом n. Средняя ошибка алгоритма **BBC** наоборот, увеличивается с ростом n и составляет порядка 25% в случае $n = 50$. Средняя ошибка алгоритма **CSW** также остается около 0.

Эксперимент на маленьких графах



Средняя точность алгоритмов на графах малой размерности при $p = 0.67$.

Эксперимент на больших графах

В эксперименте на графах большой размерности (при n от 100 до 6000, решалось по 100 задач в серии) исследовалось поведение случайных величин $d_{\mathbf{BBC}}(n)$ и $d_{\mathbf{N1LS}}(n)$: отношение значений целевых функций, полученных алгоритмами **BBC** и $\mathbf{N1LS}_{\leq 2}$ к значению целевой функции, полученному алгоритмом $\mathbf{CSW}^{\leq 2}$ на одном и том же графе.

Гипотеза 1. При достаточно больших n математическое ожидание случайной величины $d_{\mathbf{BBC}}(n)$ не меньше 1.

Гипотеза 2. При достаточно больших n математическое ожидание случайной величины $d_{\mathbf{N1LS}}(n)$ стремится к 1.

Эксперимент на больших графах

Проверка осуществлялась по широко распространенной схеме: на основе выборочных данных были получены оценки $d_{\mathbf{BBC}}(n)$ и $d_{\mathbf{N1LS}}(n)$ математического ожидания для каждой из исследуемых случайных величин, после чего при уровне значимости $\alpha = 0.05$ рассчитаны границы доверительных интервалов.

Для расчета границ доверительного интервала использовался квантиль нормального распределения. Статистическое обоснование этому было получено путем проверки критерия согласия Колмогорова-Смирнова эмпирической функции распределения $F_{\mathbf{BBC}}(x)$ с соответствующей ей функцией нормального распределения, имеющей параметры $d_{\mathbf{BBC}}(n)$ и $\sigma_{\mathbf{BBC}}$. Тоже самое было сделано для случайной величины $d_{\mathbf{N1LS}}(n)$.

Значение статистик для $d_{\mathbf{BBC}}$

$n \setminus p$	0.33	0.5	0.67
100	0.57	0.79	0.55
200	0.56	0.55	0.52
300	0.7	0.63	0.96
400	1.01	0.73	0.78
500	0.63	0.81	0.56
600	0.6	0.77	1.01
700	0.76	0.77	0.81
800	0.54	0.61	0.53
900	1.29	0.4	0.9
1000	0.56	0.53	0.59
1200	0.69	0.46	0.83
1400	0.68	1.01	0.87
1600	1	0.63	0.79
1800	0.8	0.47	0.49
2000	0.57	0.91	0.81
2250	0.97	0.55	0.68
2500	0.53	0.71	0.95
2750	0.96	0.71	0.79
3000	0.74	0.52	1.22
3500	0.7	0.63	0.88
4000	0.92	0.55	0.72
5000	1.32	0.47	0.91
6000	0.98	0.75	0.96

Границы доверительных интервалов для $d_{\mathbf{BBC}}$

$n \setminus p$	0.33	0.5	0.67
100	[1.120354, 1.124523]	[1.107732, 1.110707]	[1.314733, 1.324574]
200	[1.099026, 1.100867]	[1.083028, 1.084592]	[1.358854, 1.364678]
300	[1.087364, 1.088806]	[1.070757, 1.071769]	[1.377802, 1.382724]
400	[1.079974, 1.081141]	[1.062563, 1.063210]	[1.387986, 1.391589]
500	[1.075121, 1.076172]	[1.056320, 1.056828]	[1.396851, 1.400028]
600	[1.071720, 1.072583]	[1.051824, 1.052259]	[1.400629, 1.403466]
700	[1.068796, 1.069815]	[1.048429, 1.048816]	[1.404510, 1.407188]
800	[1.066766, 1.067511]	[1.045628, 1.045927]	[1.408344, 1.410693]
900	[1.064499, 1.065320]	[1.043138, 1.043421]	[1.411154, 1.413458]
1000	[1.063180, 1.063974]	[1.041265, 1.041465]	[1.413391, 1.415244]
1200	[1.060619, 1.061299]	[1.037959, 1.038128]	[1.417703, 1.419182]
1400	[1.058620, 1.059116]	[1.035253, 1.035421]	[1.420417, 1.422060]
1600	[1.057041, 1.057637]	[1.033095, 1.033221]	[1.422380, 1.423910]
1800	[1.055788, 1.056339]	[1.031338, 1.031454]	[1.424256, 1.425442]
2000	[1.055066, 1.055577]	[1.029857, 1.029972]	[1.425854, 1.427060]
2250	[1.053777, 1.054279]	[1.028203, 1.028289]	[1.427102, 1.428480]
2500	[1.052807, 1.053317]	[1.026844, 1.026934]	[1.428977, 1.429927]
2750	[1.052180, 1.052603]	[1.025648, 1.025727]	[1.430258, 1.431140]
3000	[1.051492, 1.051898]	[1.024638, 1.024719]	[1.431276, 1.432182]
3500	[1.050229, 1.050654]	[1.022896, 1.022952]	[1.432746, 1.433669]
4000	[1.049369, 1.049777]	[1.021475, 1.021521]	[1.434360, 1.435101]
5000	[1.048125, 1.048469]	[1.019299, 1.019337]	[1.436198, 1.436976]
6000	[1.047211, 1.047549]	[1.017674, 1.017709]	[1.438117, 1.438726]

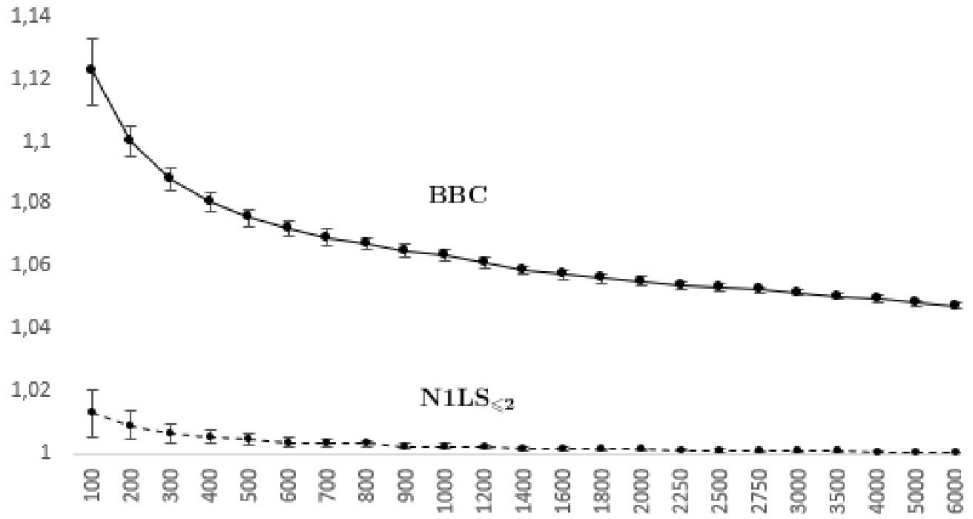
Значение статистик для $d_{\mathbf{N1LS}}$

$n \setminus p$	0.33	0.5	0.67
100	0.79	0.74	-
200	0.94	0.8	-
300	0.7	0.81	-
400	0.5	0.76	-
500	0.52	0.54	-
600	0.82	0.51	-
700	0.57	0.47	-
800	0.76	0.51	-
900	0.41	0.76	-
1000	0.77	0.69	-
1200	0.57	0.66	-
1400	0.62	0.61	-
1600	0.52	0.73	-
1800	0.87	0.63	-
2000	0.61	0.62	-
2250	0.58	0.38	-
2500	0.48	0.65	-
2750	0.84	0.42	-
3000	0.48	0.5	-
3500	0.55	0.46	-
4000	0.57	0.64	-
5000	0.6	0.81	-
6000	0.74	0.54	-

Границы доверительных интервалов для $d_{\mathbf{N1LS}}$

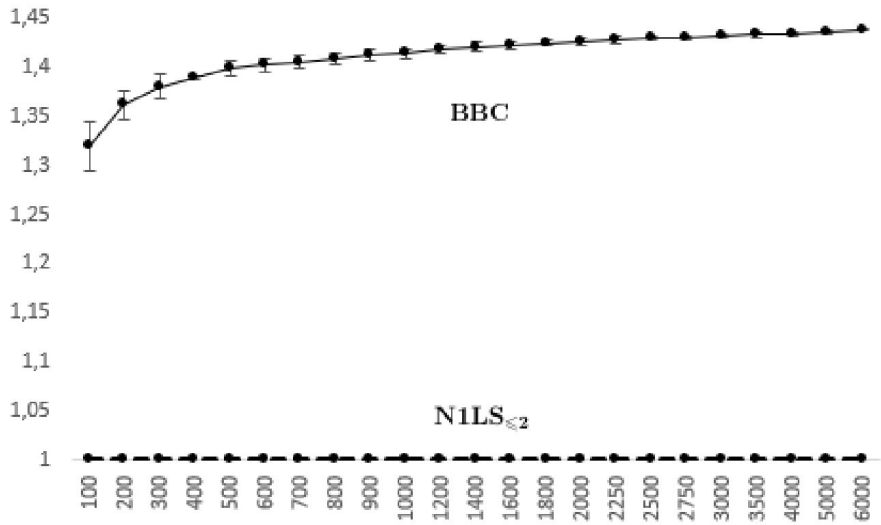
$n \setminus p$	0.33	0.5	0.67
100	[1.011620 , 1.014590]	[1.012611 , 1.015631]	[1 , 1]
200	[1.008379 , 1.010072]	[1.008866 , 1.010634]	[1 , 1]
300	[1.005938 , 1.007083]	[1.006697 , 1.007906]	[1 , 1]
400	[1.005048 , 1.005903]	[1.004967 , 1.005891]	[1 , 1]
500	[1.004286 , 1.005041]	[1.004097 , 1.004737]	[1 , 1]
600	[1.003407 , 1.003952]	[1.003582 , 1.004121]	[1 , 1]
700	[1.003153 , 1.003674]	[1.003243 , 1.003762]	[1 , 1]
800	[1.003068 , 1.003466]	[1.003000 , 1.003393]	[1 , 1]
900	[1.002367 , 1.002766]	[1.002712 , 1.003090]	[1 , 1]
1000	[1.002362 , 1.002732]	[1.002370 , 1.002757]	[1 , 1]
1200	[1.001918 , 1.002182]	[1.002029 , 1.002330]	[1 , 1]
1400	[1.001701 , 1.001911]	[1.001823 , 1.002085]	[1 , 1]
1600	[1.001567 , 1.001760]	[1.001618 , 1.001834]	[1 , 1]
1800	[1.001339 , 1.001510]	[1.001575 , 1.001765]	[1 , 1]
2000	[1.001325 , 1.001481]	[1.001431 , 1.001592]	[1 , 1]
2250	[1.001146 , 1.001294]	[1.001240 , 1.001396]	[1 , 1]
2500	[1.001034 , 1.001175]	[1.001112 , 1.001246]	[1 , 1]
2750	[1.000965 , 1.001080]	[1.000999 , 1.001147]	[1 , 1]
3000	[1.000857 , 1.000982]	[1.000955 , 1.001076]	[1 , 1]
3500	[1.000793 , 1.000898]	[1.000847 , 1.000954]	[1 , 1]
4000	[1.000700 , 1.000781]	[1.000759 , 1.000860]	[1 , 1]
5000	[1.000542 , 1.000610]	[1.000601 , 1.000677]	[1 , 1]
6000	[1.000451 , 1.000511]	[1.000502 , 1.000569]	[1 , 1]

Эксперимент на больших графах



Среднее значение случайных величин $d_{\mathbf{BBC}}$ и $d_{\mathbf{N1LS}}$ при $p = 0.33$.

Эксперимент на больших графах



Среднее значение случайных величин $d_{\mathbf{BBC}}$ и $d_{\mathbf{N1LS}}$ при $p = 0.67$.

Кластеризация с частичным привлечением учителя

Задача **SGC$_k$** (k-SEMI-SUPERVISED GRAPH CLUSTERING). Дан произвольный граф $G = (V, E)$, целое число k (количество кластеров) и множество попарно различных вершин $Z = \{z_1, z_2, \ldots, z_k\}$. Необходимо найти такой граф M^*, который является ближайшим к G кластерным графом с k кластерами и в котором все вершины из множества Z принадлежат разным кластерам.

3-приближенный алгоритм для задачи $\mathbf{SGC_2}$

Алгоритм $\mathbf{NS_2}$ (Neighborhood semi-supervised for $\mathbf{SGC_2}$).

Шаг 1. Для каждой вершины $v \in V$ произвольного графа $G = (V, E)$

(а) если $v \notin Z$, то построить 2 графа: в каждом первый будет образовывать сама вершина и ее окрестность, но в первом случае в первый кластер попадает вершина z_1, а во втором z_2.

(б) если $v \in Z$ то построить граф, в котором первый кластер образовывает сама вершина и ее окрестность без второй вершины из множества Z.

Шаг 2. Среди всех графов выбрать ближайший к G кластерный граф M_{NS}.

Трудоемкость алгоритма $\mathbf{NS_2}$ – $O(n^2)$.

2-приближенный алгоритм для задачи $\mathbf{SGC_2}$.

Алгоритм $\mathbf{NSLS_2}$ (Neighborhood semi-supervised with Local Search for $\mathbf{SGC_2}$).

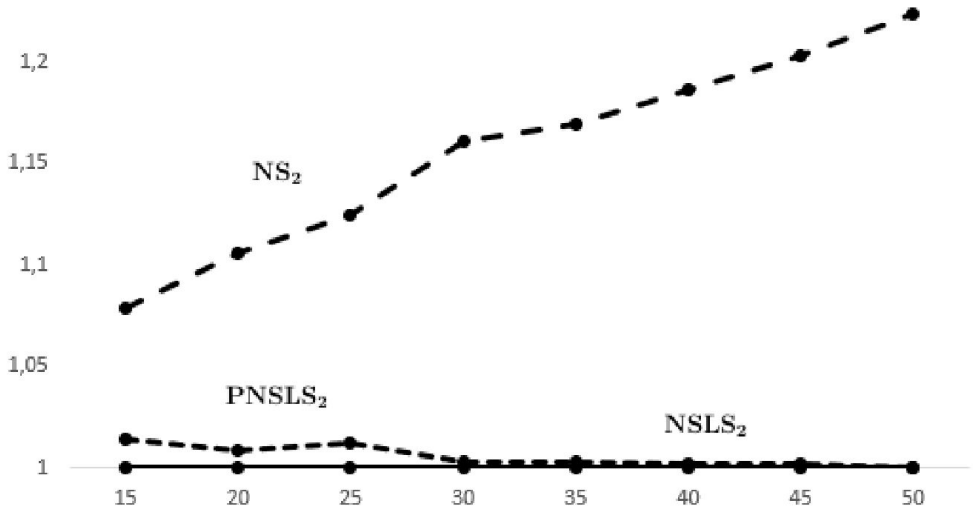
Шаг 1. Пусть F – множество всех допустимых решений, построенных алгоритмом $\mathbf{NS_2}$. Применить процедуру локального поиска $\mathbf{LS_2}$ к каждому решению из F.

Шаг 2. Среди всех локальных оптимумов выбрать ближайший к G кластерный граф M_{NSLS}.

Трудоемкость алгоритма $\mathbf{NSLS_2}$ – $O(n^4)$.

Эксперимент на маленьких графах

Рассмотрим алгоритм **PNSLS$_2$** (Pre-clustered neighborhood semi-supervised with Local Search for **SGC$_2$**). Для вершин z_1 и z_2 строятся кластерные графы по принципу алгоритма **NSLS$_2$**.



Значение статистик для $d_{\mathbf{NS}}$

$n \setminus p$	0.33	0.5	0.67
100	0.47	0.62	0.48
200	0.63	1.02	0.5
300	0.97	1.1	1.06
400	0.65	0.81	0.55
500	0.55	0.7	0.57
600	0.83	0.37	0.74
700	0.59	0.74	0.65
800	0.89	0.74	1.19
900	0.83	0.96	0.49
1000	0.89	0.47	0.49
1200	0.99	0.42	1.06
1400	0.91	0.53	1.11
1600	0.95	0.84	1.13
1800	0.91	0.75	0.77
2000	1.05	0.47	0.41
2250	0.58	0.63	0.77
2500	0.82	0.95	1.14
2750	0.84	0.84	0.62
3000	0.82	0.51	0.61

Границы доверительных интервалов для $d_{\mathbf{NS}}$

$n \setminus p$	0.33	0.5	0.67
100	[1.116443, 1.120014]	[1.104100, 1.107349]	[1.295245, 1.304322]
200	[1.097413, 1.099317]	[1.081527, 1.082789]	[1.344647, 1.351128]
300	[1.086544, 1.087923]	[1.069779, 1.070768]	[1.369551, 1.374214]
400	[1.078893, 1.080098]	[1.061764, 1.062403]	[1.381628, 1.385613]
500	[1.074676, 1.075856]	[1.056201, 1.056734]	[1.391876, 1.394756]
600	[1.070696, 1.071771]	[1.051776, 1.052128]	[1.397062, 1.400229]
700	[1.068122, 1.069153]	[1.048383, 1.048724]	[1.402694, 1.404957]
800	[1.066315, 1.067241]	[1.045385, 1.045719]	[1.405227, 1.407686]
900	[1.064078, 1.064910]	[1.043177, 1.043457]	[1.408879, 1.410733]
1000	[1.062841, 1.063677]	[1.041165, 1.041380]	[1.411142, 1.412921]
1200	[1.060195, 1.060930]	[1.037812, 1.038012]	[1.415465, 1.417280]
1400	[1.058538, 1.059111]	[1.035255, 1.035418]	[1.418731, 1.420245]
1600	[1.057010, 1.057671]	[1.033149, 1.033286]	[1.421002, 1.422349]
1800	[1.055643, 1.056264]	[1.031307, 1.031434]	[1.422915, 1.424159]
2000	[1.054950, 1.055430]	[1.029809, 1.029921]	[1.424609, 1.425802]
2250	[1.053725, 1.054194]	[1.028151, 1.028259]	[1.426209, 1.427321]
2500	[1.052839, 1.053306]	[1.026804, 1.026890]	[1.428020, 1.429106]
2750	[1.052201, 1.052666]	[1.025639, 1.025717]	[1.429130, 1.430188]
3000	[1.051367, 1.051793]	[1.024614, 1.024679]	[1.430321, 1.431192]

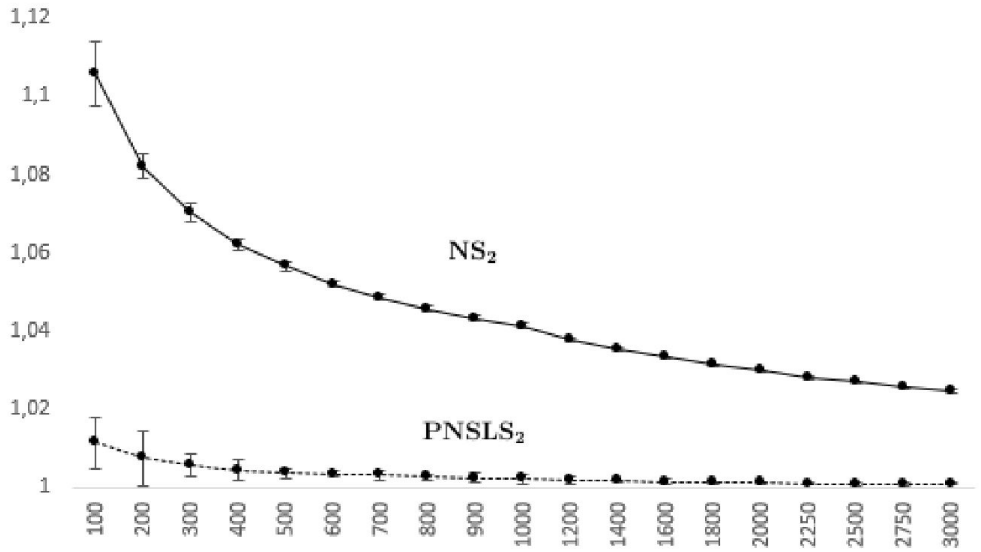
Значение статистик для $d_{\mathbf{PNSLS}}$

$n \setminus p$	0.33	0.5	0.67
100	0.68	0.81	-
200	0.52	1.05	-
300	0.6	0.79	-
400	0.51	0.64	-
500	0.68	0.74	-
600	0.5	0.65	-
700	0.63	0.51	-
800	0.46	0.62	-
900	0.89	0.55	-
1000	0.51	0.83	-
1200	0.44	0.87	-
1400	0.74	1.01	-
1600	0.49	0.72	-
1800	0.74	0.55	-
2000	0.55	0.67	-
2250	0.51	0.62	-
2500	0.65	0.67	-
2750	0.53	0.49	-
3000	0.52	0.62	-

Границы доверительных интервалов для $d_{\mathbf{PNSLS}}$

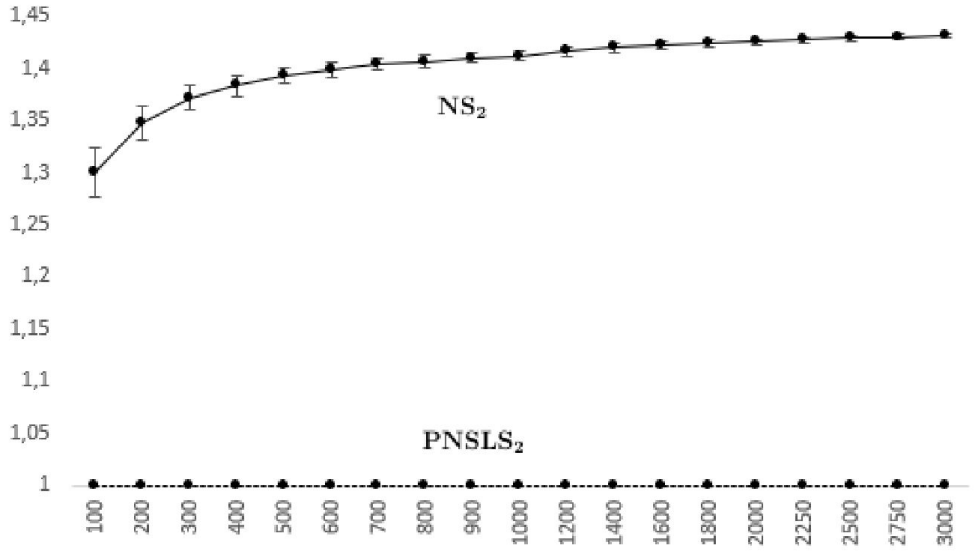
$n \setminus p$	0.33	0.5	0.67
100	[1.009466, 1.011867]	[1.009969, 1.012656]	[1, 1]
200	[1.006468, 1.007795]	[1.006929, 1.008145]	[1, 1]
300	[1.005151, 1.005966]	[1.005290, 1.006150]	[1, 1]
400	[1.003839, 1.004627]	[1.004090, 1.004844]	[1, 1]
500	[1.003458, 1.003948]	[1.003439, 1.003995]	[1, 1]
600	[1.002925, 1.003401]	[1.003160, 1.003613]	[1, 1]
700	[1.002736, 1.003169]	[1.002846, 1.003286]	[1, 1]
800	[1.002545, 1.002904]	[1.002466, 1.002829]	[1, 1]
900	[1.002232, 1.002535]	[1.002338, 1.002693]	[1, 1]
1000	[1.002040, 1.002303]	[1.002126, 1.002405]	[1, 1]
1200	[1.001696, 1.001931]	[1.001744, 1.002007]	[1, 1]
1400	[1.001449, 1.001655]	[1.001609, 1.001818]	[1, 1]
1600	[1.001327, 1.001515]	[1.001397, 1.001583]	[1, 1]
1800	[1.001319, 1.001484]	[1.001244, 1.001412]	[1, 1]
2000	[1.001168, 1.001313]	[1.001076, 1.001232]	[1, 1]
2250	[1.000979, 1.001100]	[1.001012, 1.001141]	[1, 1]
2500	[1.000926, 1.001037]	[1.000951, 1.001063]	[1, 1]
2750	[1.000865, 1.000976]	[1.000882, 1.000996]	[1, 1]
3000	[1.000777, 1.000866]	[1.000834, 1.000936]	[1, 1]

Эксперимент на больших графах



Среднее значение случайных величин $d_{\mathbf{NS}}$ и $d_{\mathbf{PNSLS}}$ при $p = 0.33$.

Эксперимент на больших графах



Среднее значение случайных величин $d_{\mathbf{NS}}$ и $d_{\mathbf{PNSLS}}$ при $p = 0.67$.