

Санкт-Петербургский государственный электротехнический университет

Альтернативный экзамен.

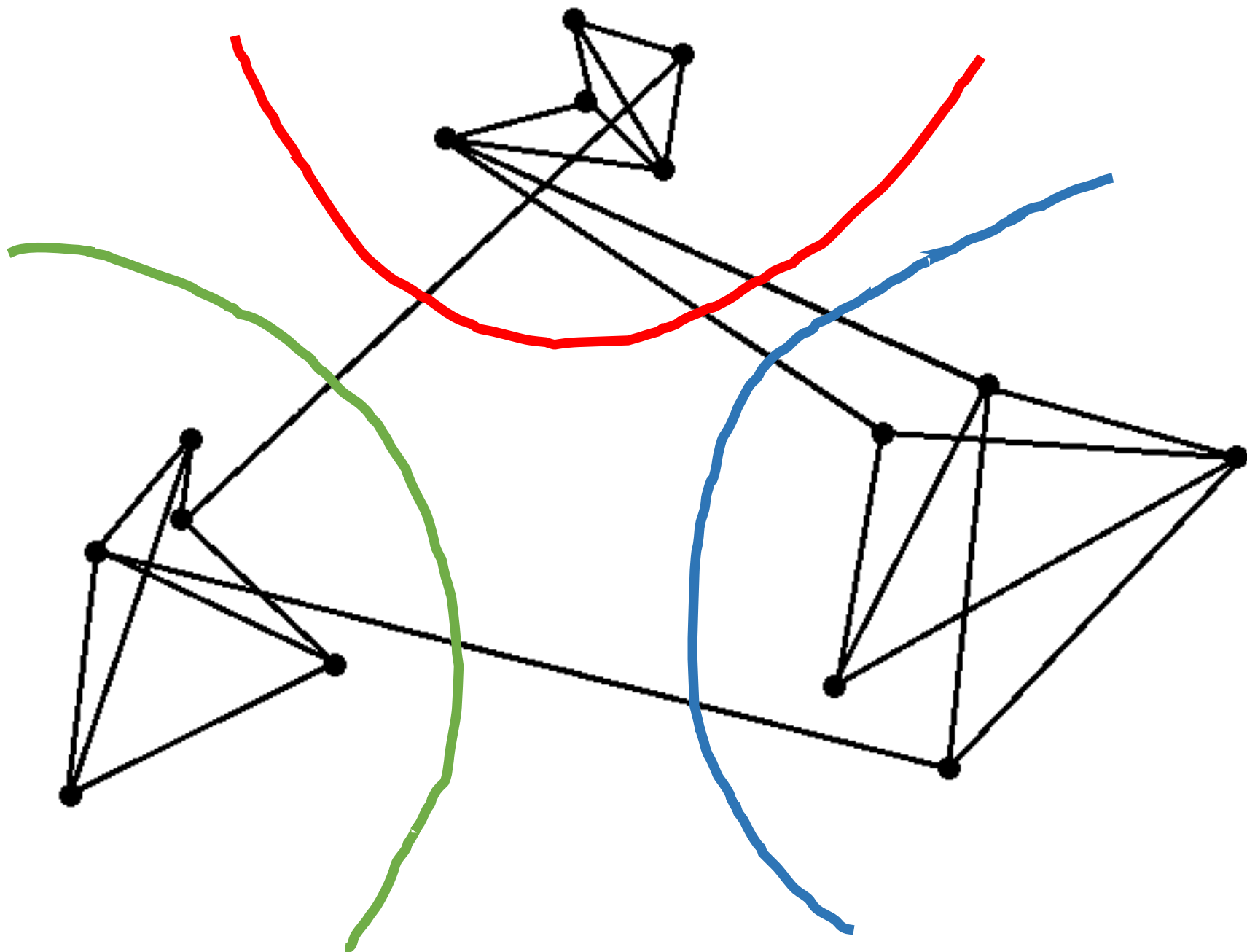
Выделение сообществ в графе.

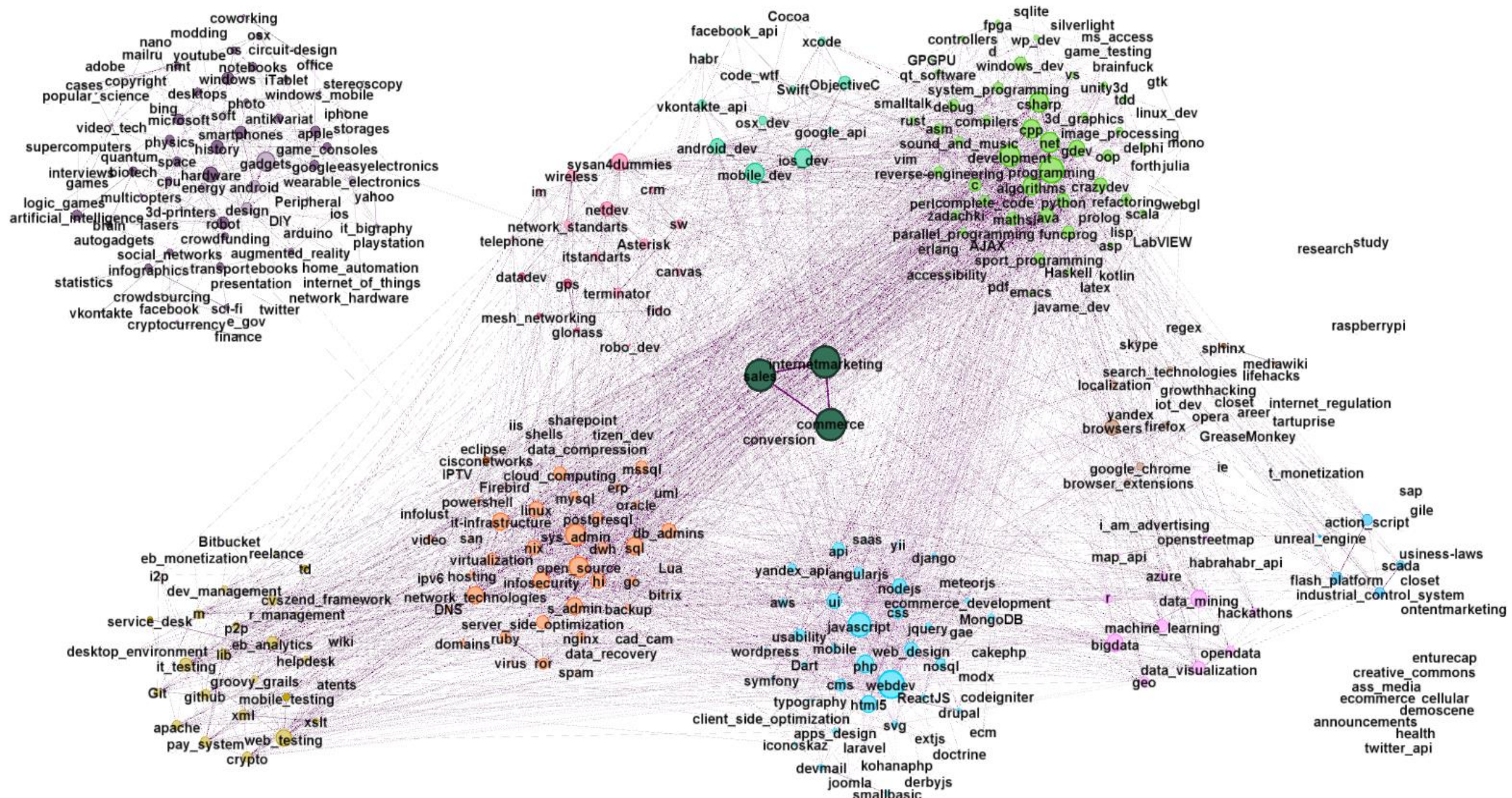
Лавров Владимир

ФКТИ

6375

2017г.





Плотность графа

$$p = \frac{m}{n(n-1) / 2}$$

Плотность сообщества

$$p_{in}(c) = \frac{m_c}{n_c(n_c - 1) / 2}$$

Плотность вне сообщества

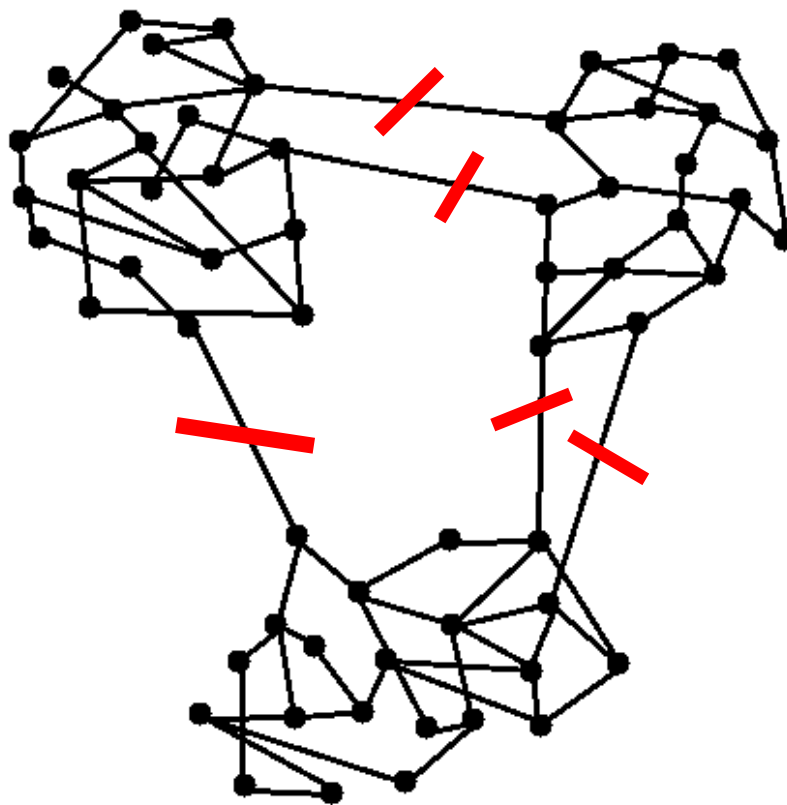
$$p_{out} = \frac{m_{out}}{n_c(n - n_c)}$$

Кластер:

$$\begin{array}{l} p_{in} > p \\ p_{out} < p \end{array} \Rightarrow \max(p_{in} - p_{out})$$

Центральность по посредничеству

$$C(e) = \sum_{s \neq t} \frac{\sigma_{st}(e)}{\sigma_{st}}$$



Модулярность

$$Q = \frac{1}{2m} \sum_{i,j} \left(A_{ij} - \frac{k_i k_j}{2m} \right) * \delta(c_i, c_j)$$

$$\delta(c_i, c_j) = \begin{cases} 1 : c_i = c_j \\ 0 : c_i \neq c_j \end{cases}$$

Maximum?

Алгоритм

1) Если исходный граф не связный, то мы изначально разбиваем его на сообщества, где каждая компонента связности – сообщество.

Для исходного графа вычисляется модулярность.

2) Для каждого ребра графа вычисляется его центральность по посредничеству.

3) То ребро, центральность которого максимальна, удаляется из графа. В случае, когда таких рёбер в графе несколько, одно из них выбирается случайным образом. Если рёбер в графе нет, то алгоритм завершает свою работу.

4) Для получившегося после удаления ребра нового графа пересчитывается функция модулярности. Если её значение увеличилось, то возвращаемся к шагу 2, иначе завершаем работу алгоритма.

Мои исследования

1. По успеваемости моих одноклассников можно было судить, какая группа учеников поедет продолжать учиться в один город.

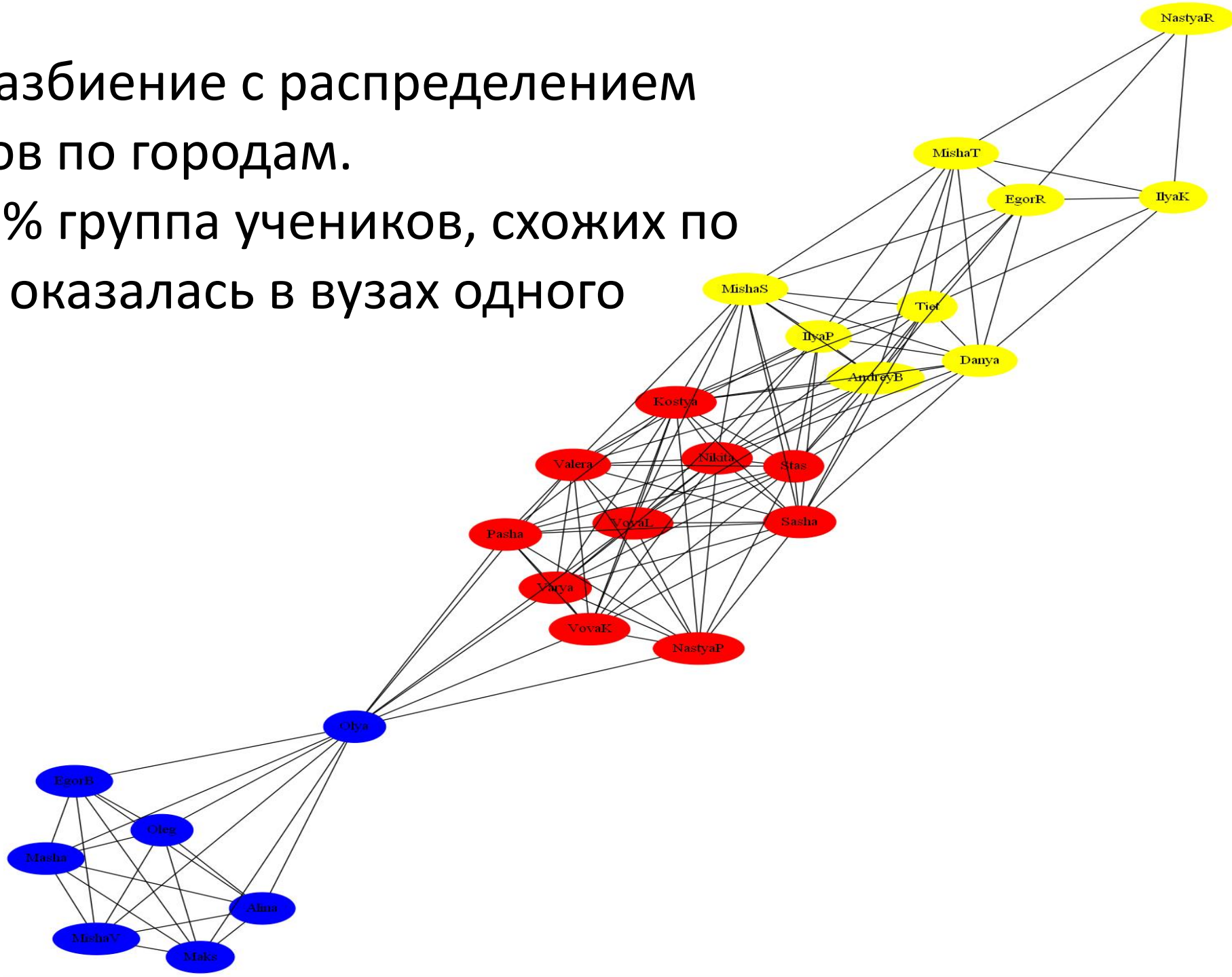
Назовем двух учеников схожими по успеваемости, если разница между суммой оценок одного и второго не превосходит 3.

Я сформировал граф, в котором вершины – мои одноклассники, а ребра соединяют схожих по успеваемости.

Применим алгоритм разбиения графа на сообщества.

Сравним это разбиение с распределением одноклассников по городам.

Результат: в 75% группа учеников, схожих по успеваемости, оказалась в вузах одного города.



	Ali	Oly	Ego	Mis	Ily	Ily	Tie	Mak	Ole	Var	Val	Mis	Sas	Mas	Nas	Dan	Vov	Nas	Mis	And	Ego	Nik	Pas	Sta	Kos	Vov
Alin	0	1	1	0	0	0	0	1	1	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0
Olya	1	0	1	0	0	0	0	1	1	1	1	0	0	1	1	0	1	0	1	0	0	0	1	0	0	1
Egor	1	1	0	0	0	0	0	1	1	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0
Mish	0	0	0	0	1	1	1	0	0	0	0	1	0	0	0	1	0	1	0	1	1	0	0	0	0	0
Ilya	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	1	0	1	0	0	1	0	0	0	0	0
Ilya	0	0	0	1	0	0	1	0	0	0	1	1	1	0	0	1	0	0	0	1	1	1	0	1	1	1
Tiet	0	0	0	1	1	1	0	0	0	0	0	1	1	0	0	1	0	0	0	1	1	1	0	1	1	0
Maks	1	1	1	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0
Oleg	1	1	1	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0
Vary	0	1	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0	0	0	0	1	1	1	1	1
Vale	0	1	0	0	0	1	0	0	0	1	0	1	1	0	1	0	1	0	0	1	0	1	1	1	1	1
Mish	0	0	0	1	0	1	1	0	0	0	1	0	1	0	0	1	0	0	0	1	1	1	0	1	1	1
Sash	0	0	0	0	0	1	1	0	0	1	1	1	0	0	1	1	1	0	0	1	0	1	1	1	1	1
Mash	1	1	1	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
Nast	0	1	0	0	0	0	0	0	0	1	1	0	1	0	0	0	1	0	0	0	0	1	1	1	1	1
Dany	0	0	0	1	1	1	1	0	0	0	0	1	1	0	0	0	0	0	0	1	1	1	0	1	1	0
Vova	0	1	0	0	0	0	0	0	0	1	1	0	1	0	1	0	0	0	0	0	0	1	1	1	1	1
Nast	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
Mish	1	1	1	0	0	0	0	1	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
Andr	0	0	0	1	0	1	1	0	0	0	1	1	1	0	0	1	0	0	0	0	1	1	0	1	1	1
Egor	0	0	0	1	1	1	1	0	0	0	0	1	0	0	0	1	0	1	0	1	0	0	0	0	0	0
Niki	0	0	0	0	0	1	1	0	0	1	1	1	1	0	1	1	1	0	0	1	0	0	1	1	1	1
Pash	0	1	0	0	0	0	0	0	0	1	1	0	1	0	1	0	1	0	0	0	0	1	0	1	1	1
Stas	0	0	0	0	0	1	1	0	0	1	1	1	1	0	1	1	1	0	0	1	0	1	1	0	1	1
Kost	0	0	0	0	0	1	1	0	0	1	1	1	1	0	1	1	1	0	0	1	0	1	1	1	0	1
Vova	0	1	0	0	0	1	0	0	0	1	1	1	1	0	1	0	1	0	0	1	0	1	1	1	1	0

max modularity: 0.53225

best partition:

1 1 1 2 2 2 2 1 1 3 3 2 3 1 3 2 3 2 1 2 2 3 3 3 3 3

(Это не конец)