Конспект лекций

Юдин Е.Б., Юдина М.Н.

Лекция 2. Анализ сообществ в сети

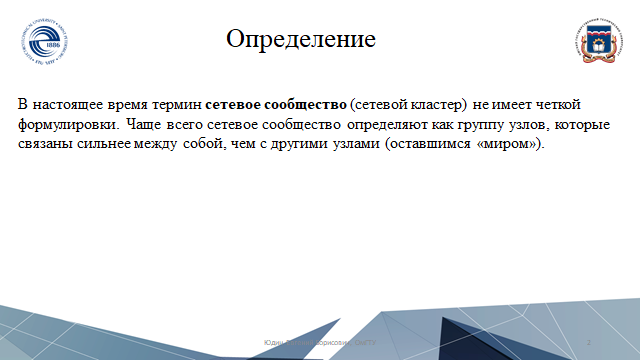
# Лекция 2. Анализ сообществ в сети”

Здравствуйте, я представляю лекцию 2 Анализ сообществ в сети по дисциплине “Модели больших сетевых структур и сетевые процессы”. Разработчиками курса являются сотрудники Омского государственного технического университета Юдин Е.Б., Юдина М.Н., Бадрызлов В.А., а также сотрудник института математики им. С.Л. Соболева Логинов К.К. Данную лекцию разработал Юдин Е.Б. в соавторстве с Юдиной М.Н.



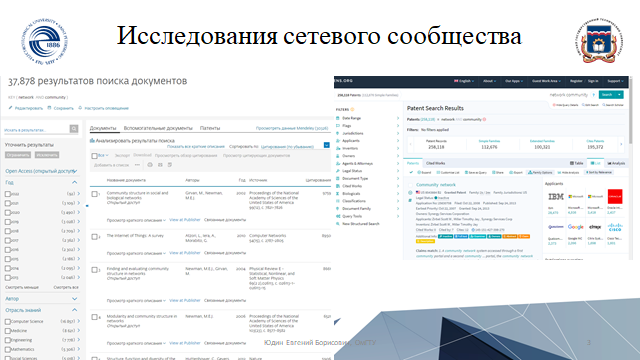
# Определение

В настоящее время термин сетевое сообщество (сетевой кластер) не имеет четкой формулировки. Чаще всего сетевое сообщество определяют как группу узлов, которые связаны сильнее между собой, чем с другими узлами.



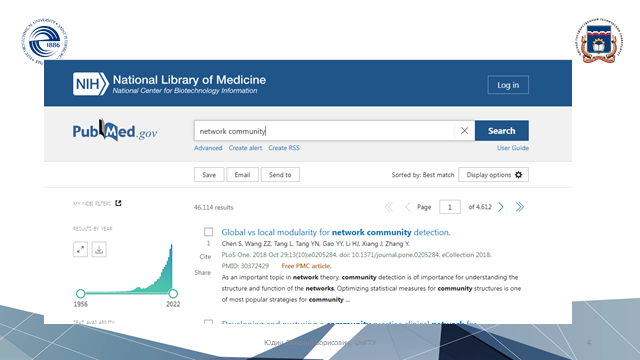
# Исследования сетевого сообщества

Как можно видеть, число публикаций год от года растет в системе научного цитирования Scopus. Как видите ключевое слово “network community”, представлены данные на ноябрь 2021 года, видно что наиболее цитируемая публикация Гирвена и Ньюмана Community structure in social and biological networks имеет почти десять тысяч цитирований. Также можно видеть в левом нижнем углу, что большая часть публикации по компьютерным наукам, на втором месте медицинские науки, на третьем инженерные науки и только потом математика. Справа показывается, что в системе LENS по ключевому слову  “network community” находится почти 260 тысяч патентов.



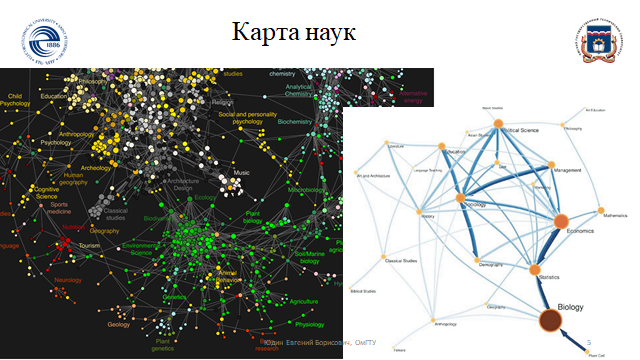
# Исследования сетевого сообщества. PubMed

Если же мы обратимся к системе Pubmed. Если кто не знает, то PubMed -  это англоязычная текстовая база данных медицинских и биологических публикаций, созданная Национальным центром биотехнологической информации (NCBI) США на основе раздела «биотехнология» Национальной медицинской библиотеки США (NLM). Так вот мы видим что по ключевому слову “network community” здесь находится более 46 тыс. записей и с начала 200-х их число растет экспоненциально.



# Карта наук

Но вернемся к самой задаче нахождения сообществ. На слайде изображена карта науки, которая получена путем анализа поведения пользователей при работе с электронными хранилищами научных статей.  Вершины представляют научные статьи из конкретной области знаний. Ребра – отношения между статьями, отношение формируется, когда некоторая часть пользователей при загрузке одной статьи в том же сеансе загружает и другую. На карте науки можно видеть, например, что ботаника (plant biology) тесно связана с экологией (ecology) и наукой о биоразнообразии окружающей среды (environmental science). Данная карта науки может быть представлена и в другом виде, когда узлы объединены в метаузлы на основе атрибута журнала, в котором публикуются статьи, этот атрибут определяет основное научное направление журнала, так ботаника и экология объединены в метаузле Биология. И это кластеры и это сообщества. Здесь всё тривиально. Принадлежность к конкретному сообществе определена тематикой журнала



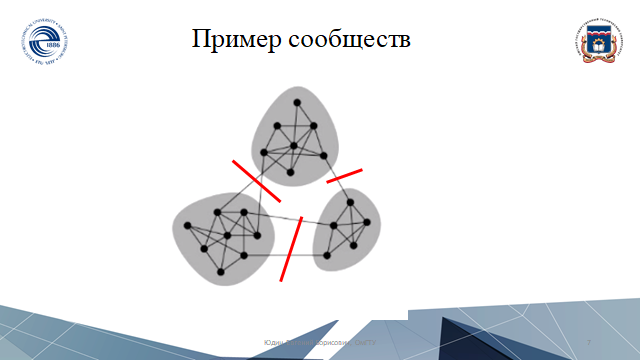
# Пример сообществ

А что если у нас нет атрибута у вершины, относящегося эту вершину к тому или иному сообществу. Можно ли в этом случае выделить сообщества. Давайте посмотрим на слайд. Где здесь могут быть сообщества?



# Пример сообществ

Обратите внимание, что мы можем сделать разрезы (удалить всего 4 ребра) и наша сеть “развалится” на три довольно компактные сети.



# Где здесь сообщества ?

А теперь посмотрим на другой граф (моделирующий некоторую сеть, естественно). Где здесь сообщества? 

# Где здесь сообщества ?

Может, следует выделить эти сильно связные компоненты?



# Где здесь сообщества ?

А, быть может, вот здесь хорошие претенденты на звание сообществ? Это изолированная полносвязная четверка вершин и достаточно большая компонента, которая связана с “большим” графов только одним ребром. Какой критерий выбора сообществ можно использовать?



# Критерий сообщества: плотность ребер

До сих пор эта задача не имеет однозначного решения, в Науке о сетях, на самом деле, используется несколько подходов. Так таким критерием может быть плотность ребер. Плотность в графе определяется как отношение числа ребер к максимально возможному их числу. Это число сочетаний из N по 2. Формула N(N-1)/2. Например у нас четыре вершины. Сколько ребер может быть между ними (нарисуйте квадрат и диагонали - 6, т.е. 4\*3/2=6!)/

Но нам придется ввести плотность внутренних и внешних связей.

Плотность внутренних связей сообщества равно отношению числа ребер внутри сообщества |Ec| к максимально возможному числу ребер Nc(Nc-1)/2 между узлами сообщества. Это по аналогии с общей плотность.

Плотность внешних связей равно отношению числа ребер между вершинами графа внутри сообщества и всеми другими вершинами графа к максимально возможному число таких ребер.

Обнаружение сообщества можно было бы свести к нахождению максимума разности внутренних и внешних связей. Однако такая максимизация часто невозможна на практике. Даже деление графа на две части является NP сложной задачей. Поэтому для решения задачи разбиения на сообщества используются специальные алгоритмы (эвристические и вероятностные алгоритмы). При этом плотность внутренних и внешних связей используется в качестве критерия качества разбиения.



# Критерий сообщества: разрезы

Но существуют и другие критерии определения сообществ. Например, число разрезов. Используемая терминология следующая.

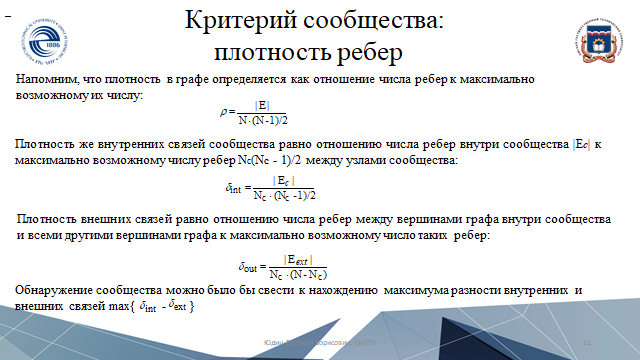
Вот у нас есть разрез, назовем его cut. Он разбивает вершины на два  множества. Одно множество V1 - одно сообщество, в другое множество V2 входят вершины, относящиеся к другому сообществу.

Рассмотрим первый критерий. Здесь нужно найти место в графе, где можно удалить наименьшее число ребер, чтобы разделить граф на части. Это первая формула. Что будет происходить. Когда мы ищем минимальный разрез? В графе мы, естественно, “отрежим” одну вершину, ну может быть больше, несколько вершин, связанных одним ребром с большой частью графа.

Но мы не хотим чтобы из нашего графа отсекались вершины поодиночке. Поэтому помимо первой формулы есть откорректированные критерии (использующие условие балансировки).  Это вторая и третья формулы.

Рассмотрим вторую формулу, это т.н. quotient cut. Идея такая. Мы берем разрезы и нормируем их на их размеры, т.е на число узлов, которые находятся в найденных сообществах.

Рассмотрим третью формулу, более сильную, normalized cut. В ней нормирование происходит не по числу узлов в сообществах, а на число ребер выходящих из узлов каждого сообщества до любых других узлов графа. С физической точки зрения это логичнее. Мы число ребер в числителе делим на число ребер в знаменателе. Т.е. получаем безразмерную величину (в quotion cut мы число ребер делим на число вершин). Метрика normalized cut хороша ещё и тем, что её можно сформулировать в терминах линейной алгебры и ее решение основывается на хорошо проработанных в математических пакетах решении задачи на нахождения собственных значений и собственного вектора. Так сформулировать можно минимальный  разрез, так сформулировать можно нормированный разрез. Так сформулировать нельзя quotient cut.

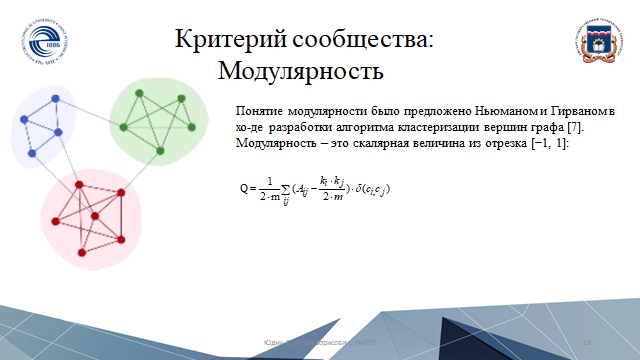


# Критерий сообщества: модулярность

Другой набор методов, который стал популярен в последние годы (степень популярности метода представлена на втором слайде презентации) это т.н. Модулярность.

Понятие модулярности было предложено Ньюманом и Гирваном в хо-де разработки алгоритма кластеризации вершин графа [7]. Модулярность – это скалярная величина из отрезка [−1, 1]. Формула её вычисления представлена на слайде. Давайте посмотрим из каких элементов она состоит.

A – матрица смежности графа, Aij – (i, j) элемент матрицы, di – степень связности вершины i графа, сi – метка вершины (номер сообщества, к которому относится вершина), m – общее количество ребер в графе, δ(ci , cj ) – дельта-функция, равная единице, если ci = cj, иначе нулю. Задача поиска выделения сообществ в графе Ньюман и Гирван предложили свести к поиску таких ci , cj , которые максимизируют значение модулярности. Плотность случайного разбиения здесь используется как относительная метрика: у сообщества плотность должна быть строго больше, чем в случайном графе. Если имеется вершина со степенью k(i), то вероятность того, что произвольное ребро присоединено к этой вершине, равна kj / 2m, а вероятность связи между ребрами равна ki kj / 2m. Иначе говоря, мы вычисляем разницу между реальным количеством ребер и ожидаемым, если бы ребра были распределены случайно. При хорошем разбиении на сообщества модулярность высокая, если сообщество одно, то модулярность равна 0.



# Понятие дендрограммы разбиения

На слайде изображен граф, и соответствующая дендрограмма, описывающая динамику разбиения на сообщества вершин графа, а также значения модулярности, для каждого разбиения.  Дендрограмма представляет собой дерево, в листьях дендрограммы находятся сообщества с одной вершиной, а в корне – сообщество, содержащее все вершины. Результатом алгоритма разбиения является ярус дендрограммы (т.е. разбиение).

На слайде представлено разбиение графа построенного на основе взаимоотношений внутри сообщества карате-клуба (Zachary's karate club). Это рисунки из статьи Ньюмана и Гирвана. Также представлена дендрограмма, так что можно видеть уровень модулярности для всех ярусов разбиения.

Т.е. сначала мы разбиваем множество вершин на два сообщества. Потом каждое из сообществ ещё на два и т.д. пока на останутся сообщества из одной вершины. Но когда нам остановится? Ответ - Когда требуемый критерий разбиения будет оптимальным. В данном случае. когда модулярность будет максимальной (потому что использовался именно этот критерий). Мы конечно могли использовать и другой - например критерий, основанный на плотности внутренних и внешних связей.



# Определение сообществ с использованием меры посредничества ребер

Рассмотрим алгоритмы нахождения сообществ, относящиеся к различным классам алгоритмов.. реализованные в популярных пакетах.

Рассмотрим пример графа. Предлагаю посмотреть на граф и попробовать придумать какую-то эвристику. Если помните, мы уже проходили понятие важности узла на основе промежуточности. Мы говорили, что эта важность определяется числом кратчайших путей проходящих через узел.В данном случае можно считать по аналогии для ребер. Т.е. считать число кратчайших путей, проходящих через ребро. Идея такая: те ребра, через которые проходят кратчайшие пути, эти ребра, вероятно, могут быть теми “мостами”, которые связывают различные сообщества. Логика такая: если у нас есть сообщества, как на слайде, которые связаны небольшим множеством ребер между собой, то кратчайшие пути между вершинами разных сообществ будут проходить именно через эти ребра. И метрика edge betweenness должно быть высокая для тех ребер, которые являются мостами для тех ребер, которые являются “мостами” между сообществами.  Суть алгоритма найти эти ребра, порезать их, и тогда граф развалится на сообщества.

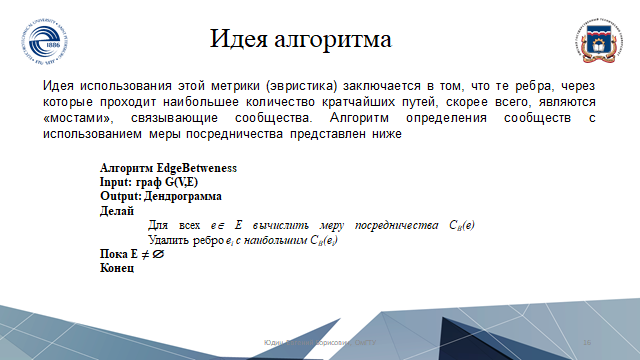


# Идея алгоритма

На данном слайде представлен алгоритм реализации поиска сообществ на основе использования меры посредничества ребер. Сложность использования этой метрики заключается в следующем. Если вычислить заранее меры посредничества ребер и отсортировать по этой мере все ребра и удалять ребра последовательно, то мы ошибемся, ведь когда мы удаляем ребра меры других ребер меняются. Поэтому правильный алгоритм следующий.

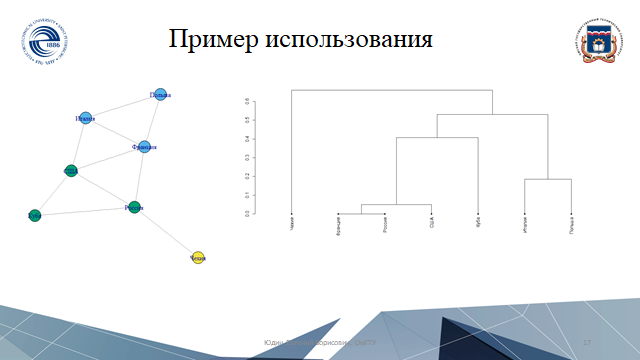
Берем граф. Находим ребра с максимальной мерой посредничества, удаляем это ребро, снова считаем  меры посредничества представлен ниже меры посредничества ребер, находим ребро с максимальным значением, удаляем его, делаем так пока граф не разделится на части (т.е. на связные компоненты)

На данном слайде представлен алгоритм разделения алгоритмом Edge betweenness. На входе граф, на выходе дендрограмма. Весь алгоритм состоит из одного цикла, в котором находится и удаляется ребро с максимальной  мерой посредничества. Если нужно алгоритм разбить на две части, то следует остановится, когда появится первые две компоненты связности.



# Пример использования

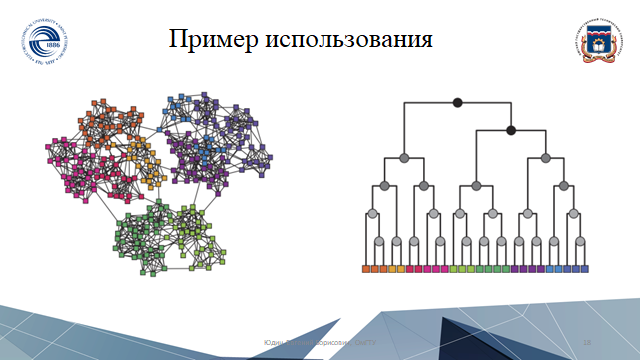
 Итак, как мы видим на слайде, если алгоритм разбивается на две части, то будет две компоненты связности. Это будет соответствовать первому ярусу дендрограммы. Но мы можем разбивать, пока у нас не останутся все вершины изолированными. Когда следует остановиться мы уже говорили. Это зависит от выбранного способа оценки качества разбиения. На данном слайде алгоритм в библиотеке igraph остановился, когда были получены три сообщества. В данном случае в igraph реализована возможность использование только модулярности, в качестве оценки качества разбиения.



# Пример использования

Вот пример решения, т.е. дендрограмма для графа, представленного ранее. Вершины одного сообщества окрашены одним цветом. Здесь видно, что будет если мы остановимся на различных уровнях разбиения. Например, если мы остановимся после первого разбиения, то остаются розовые и оранжевые в одном сообществе, а все остальные - в другом.

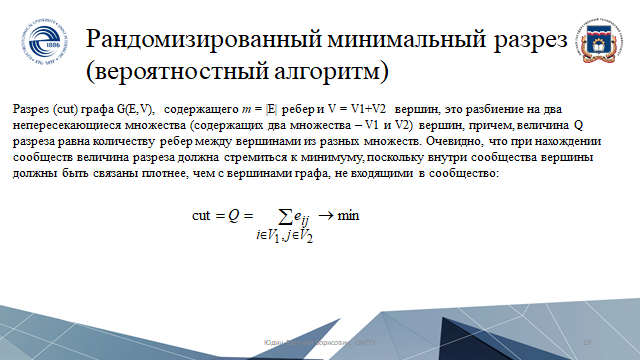
 Почему метод edge betweenness эвристический?! Просто потому, что здесь нет никакой метрики, которую мы оптимизируем, процедура оптимизация жадная (мы просто выбираем на каждом шаге оптимальное решение), кроме того сам метод основан на достаточно интуитивных рассуждениях о наличии некоторых “мостов” ребер.



# Рандомизированный минимальный разрез (вероятностный алгоритм)

Рассмотрим ещё один алгоритм, основанный на математической статистики. Он этим и сложен и интересен.

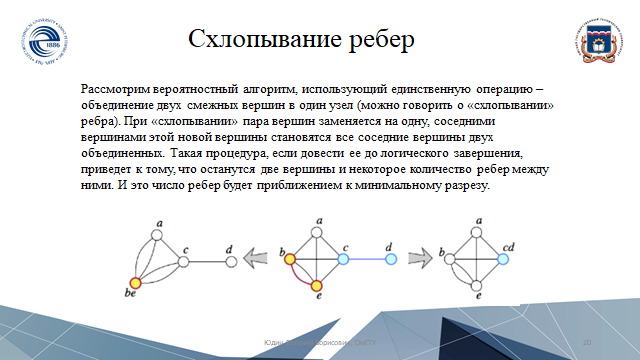
Метод основан на понятии разрезов, о которых мы уже говорили. Разрез (cut) графа G(E,V), содержащего m = |E| ребер и V = V1+V2 вершин, это разбиение на два непересекающиеся множества (содержащих два множества – V1 и V2) вершин, причем, величина Q разреза равна количеству ребер между вершинами из разных множеств. Очевидно, что при нахождении сообществ величина разреза должна стремиться к минимуму, поскольку внутри сообщества вершины должны быть связаны плотнее, чем с вершинами графа, не входящими в сообщество.



# “Схлопывание” ребер

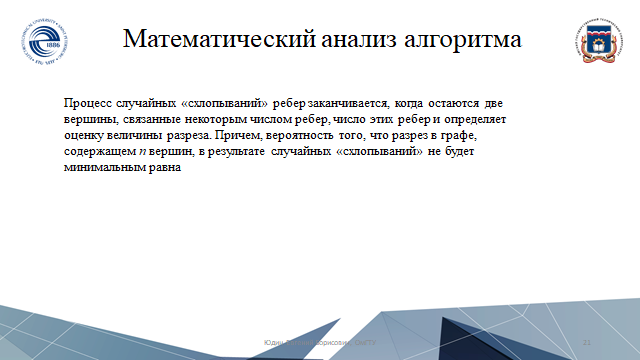
В нашем алгоритме будет использоваться единственнуая операция – объединение двух смежных вершин в один узел (можно говорить о «схлопывании» ребра). При «схлопывании» пара вершин заменяется на одну, соседними вершинами этой новой вершины становятся все соседние вершины двух объединенных. Такая процедура, если довести ее до логического завершения, приведет к тому, что останутся две вершины и некоторое количество ребер между ними. И это число ребер будет приближением к минимальному разрезу.

Действительно, рассмотрим граф на слайде слева. Мы видим, что минимальный разрез приходится между вершинами c и d. Пусть случайным образом выбирается ребро be, и в результате выбора ребро между вершинами b и e схлопывается. Остается один узел be и получается граф, изображенный на центральном рисунке, при этом величина минимального разреза в новом графе не изменилась. Пусть теперь случайным образом выбраны узлы c и d, по той же схеме получаем граф, изображенный на рисунке справа. Как мы видим, в исходном графе минимальный разрез шел по ребру cd, и его величина была равна 1. В новом графе этот разрез исчез и минимальный разрез равен трем. В данном случае минимальный разрез после схлопывания увеличился, напомним, что в графе на центральном рисунке минимальный разрез не изменился. Этот пример показывает, что схлопывание ребер не может уменьшить величину минимального разреза, минимальный разрез модифицированного графа будет больше или равен минимальному разрезу исходного графа.



# Математический анализ алгоритма

Давайте оценим точность нахождения минимального разреза. Заметим, для начала, что процесс случайных «схлопываний» ребер заканчивается, когда остаются две вершины, связанные некоторым числом ребер, число этих ребер и определяет оценку величины разреза. Причем, вероятность того, что разрез в графе, содержащем n вершин, в результате случайных «схлопываний» не будет минимальным равна 2/(n(n+1))



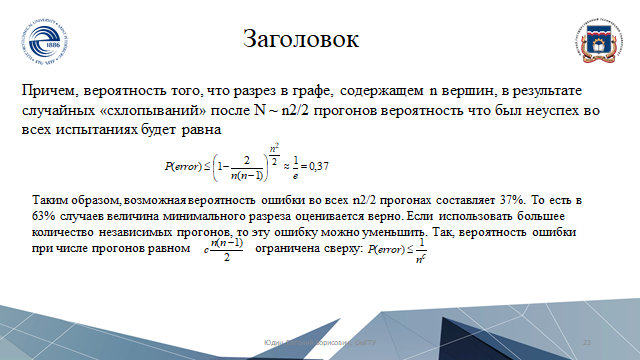
# Математический анализ алгоритма

Почему именно этому значению объясняется на данном слайде. Мы не будем на этом сегодня останавливаться. Кому интересно, может посмотреть по ссылке внизу слайда. Математические операции здесь не сложные.



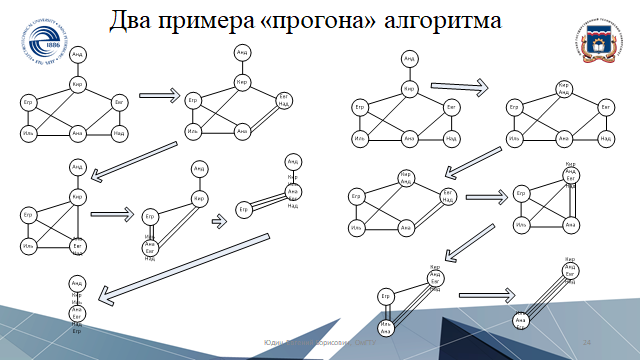
# Математический анализ алгоритма

На данном слайде оценивается вероятность того, что разрез в графе, содержащем n вершин, в результате случайных «схлопываний» после N ~ n2/2 прогонов не будет найден. Так вот вероятность что был неуспех во всех испытаниях будет может быть найдена зная вероятность того, что не будет найдена в одном испытании и она равна 37%. Таким образом, возможная вероятность ошибки во всех n2/2 прогонах составляет 37%. То есть в 63% случаев величина минимального разреза оценивается верно. Кроме того увеличивая число испытаний, мы увеличиваем вероятность нахождения минимального разреза.



# . Два примера «прогона» алгоритма

На данном слайде приведены примеры алгоритмов случайного “схлопывания” ребер. Динамика «схлопывания» методом рандомизированного минимального разреза, заканчивается слева нахождением минимального разреза, а справа  - разреза, который не является минимальным. Здесь для удобства отображения узлы обозначены тремя первыми буквами имени. На графе слева последовательно выбираются связи Евгений-Надежда, (Евгений, Надежда) -Анна, (Евгений, Надежда, Анна) - Илья, (Евгений, Надежда, Анна, Илья) - Кирилл,  (Евгений, Надежда, Анна, Илья,Кирилл) - Егор.



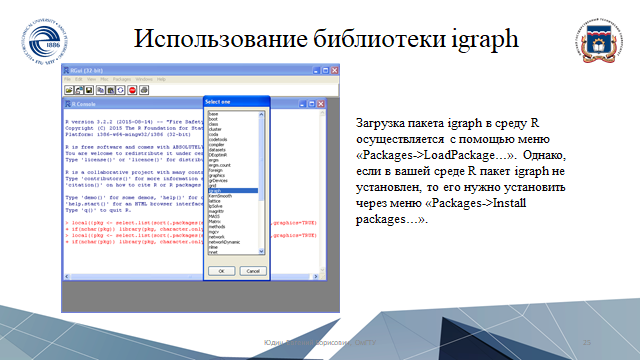
# Использование библиотеки igraph

Загрузка пакета igraph в среду R осуществляется c помощью меню «Packages->LoadPackage…». для установки в среде R пакета igraph нужно установить через меню «Packages->Install packages…».

Пример кода в igraph

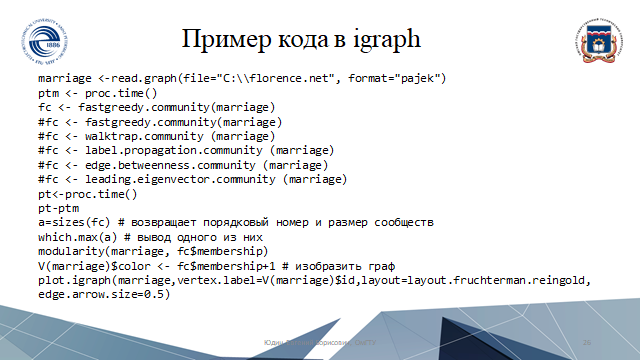
После можно писать код, вызывая функции библиотеки. На данном слайде загружается в первой строке файл семейных кланов Флоренции, который мы обсуждали ранее. Путь к файлу C:\\florence.net, формат представление  Pajek (один из наиболее популярных форматов хранения графа)

Далее запускается алгоритм поиска сообществ (этому алгоритму передается загруженный файл). В данном примере приведены 5 различных функций для вызова алгоритма. Но 4 из них закомментированы знаком #. А использоваться будет только алгоритм, описанный нами ранее, алгоритм на основе удаления ребер с максимальным значением меры важности на основе посредничества. Далее выводится номер и размер максимального сообщества, модулярность разбиения и вывод графа на экран.



# Визуализация алгоритмом Фруктермана-Рейнголда

Вывод графа на экран это отдельная научная тема. Ведь сами графы не имеют координат (за исключением специального вида графов - Евклидовы графы). Поэтому существует множество алгоритмов, одни из них позволяют минимизировать пересечения ребер, другие учитывают близость между вершинами, стараются “притянуть” или “растолкать” вершины, треть пытаются представить граф в виде, похожим на окружность, и т.д. На данном слайде представлена визуализация с использованием алгоритма Фруктермана-Рейнголда.



# Характеристики алгоритмов разбиения на сообщества в igraph

Поскольку вам могут быть интересны различные алгоритмы разбиения на сообщества дадим их краткие характеристики

1) Алгоритм «walktrap.community», основан на идее, что случайные блуждания оказываются «пойманными» в той части графа, внутри которого ребра плотнее, чем окружающие, то есть там, где находится сообщество.

2) Алгоритм «label.propagation.community» основан на процедуре «распространения ярлыков» и не требует никаких входных данных, кроме списка ребер графа. Процедура «распространения ярлыков» напоминает процесс распространения инфекции и состоит в том, что первоначально каждая вершина графа имеет свой собственный «ярлык», затем начинается итеративный процесс и на каждой итерации часть вершин принимает «ярлыки» своих «соседей», а именно тот «ярлык», который более представлен среди соседей данной вершины.

3) Алгоритм «edge.betweenness.community». В этом алгоритме итеративно исключаются ребра с наибольшей степенью промежуточности, и после каждого исключенного ребра значения степени промежуточности пересчитываются. Мы его проходили.



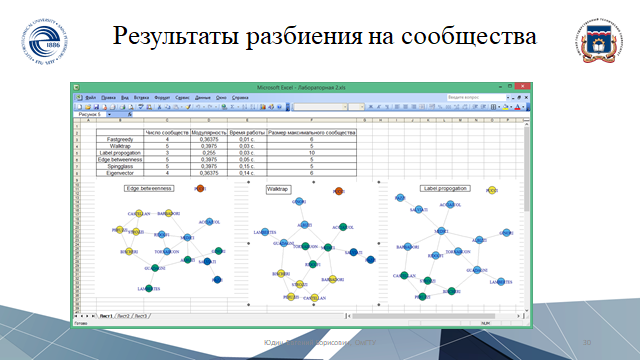
# Характеристики алгоритмов разбиения на сообщества в igraph

4) Алгоритм «spinglass.community» использует модель «спинового стекла» и «имитации отжига» (spinglass model and simulated annealing).

5) Алгоритм «fastgreedy.community» (используется только для неориентированных графов), в основе которого лежит иерархическая агломеративная кластеризация редких иерархически организованных сетей.

6) Алгоритм «leading.eigenvector.community» Для реализации данного метода «рассчитывается собственный вектор матрицы модульности для самого большого положительного собственного значения и потом разделения вершин на два сообщества, основанных на знаках соответствующих элементов в собственном векторе. Если все элементы в собственном векторе одного знака, то это означает, что сеть не имеет скрытой структуры сообществ.

# Результаты разбиения на сообщества

На данном слайде представлены результаты разбиения на сообщества при использовании различных алгоритмов. Как видите использование различных алгоритмов приводит к различным результатам как по числу найденных сообществ, размеру максимального сообщества, так и по времени работы алгоритма и полученной модулярности (качества разбиения). 

# Примеры использования сообществ в медицинских исследованиях

Но предлагаю посмотреть на примеры из биоинформатики. На данном слайде представлена визуализация сети и результат определения сообществ. (а) Сеть коэкспрессии генов, происходящая в нормальной ткани . Сети были визуализированы с помощью Cytoscape  алгоритм компоновки, без знания идентифицированной структуры сообщества. Каждому сообществу был присвоен цвет в соответствии с разделами, идентифицированными алгоритмом Лувена. (б) Сеть Коэкспрессии, происходящая в раковой ткани. (c) Нормализованные значения модулярности для нормальных сетей и сетей, полученных из опухоли тканей.



# Примеры использования сообществ в медицинских исследованиях

На данном слайде представлены результаты выявления иерархических сообществ и последующего анализа в Citoscape. (A) Иерархия, полученная из сети взаимодействия BioPlex 2.0. Красный прямоугольник : подобласть в иерархии, относящаяся к медиаторному комплексу. (B) Меню для настройки функциональной аннотации; (C) Увеличенный вид под иерархии, выделенной на панели A. Красный прямоугольник : сообщество для дальнейшего изучения. (D) Подсеть взаимодействия сообщества выделена на панели C. Белки, не известные как члены комплекса посредников, выделены ( желтым ).  Цвет узла: Перекрытие (индекс Жаккара) между сообществом и набором генов, соответствующим его аннотации (панели A и C).



# Выводы

Тема выявления сообществ - одна из центральных в направлении Network Medicine (что подтверждается ростом числа публикаций).

Однако существуют различные алгоритмы определения сообществ и оценки качества их разбиения. Эти алгоритмы доступны в различных средах для анализа сетей, таких как Citoscape или igraph.

Прежде всего анализ сообществ в медицине используется в области анализа молекулярных сетей клетки, в том числе генных сетей и сетей белковых взаимодействий.