***Практическое занятие 1*** *Анализ структурных характеристик сетей*

**Цель работы**: анализ структурных характеристик реальной сети.

**Содержание работы.**

1. Загрузка файла содержащего данные о структуре сети в программу Gephi. Изучение информации об исследуемой сети.

2. Визуализация графа в программе Gephi.

3. Расчет мер важности вершин: по степени связности, по степени близости к другим вершинам, по степени посредничества между вершинами, по степени влиятельности, по эксцентриситету вершин.

4. Расчет глобальных характеристик графа: число компонентов связности, плотность графа, средняя степень связности вершин, диаметр графа, средний коэффициент кластеризации вершин и глобальный коэффициент кластеризации вершин, среднее расстояние между вершинами.

5. Экспорт данных о графе в программу Excel и оформление отчета.

**Краткая теория и методические указания**

1. Граф *G* = (*V*, *E*) состоит из двух множеств: множества элементов *V*, называемых вершинами, и множества элементов *E*, называемых ребрами/дугами. Сетью будем называть реальную исследуемую структуру, которая моделируется графом. Узлы сети моделируются в графах вершинами, а связи – ребрами или дугами.

2. Исполняемый файл программы с открытым кодом Gephi, предназначенной для визуализации и анализа графов, доступен по адресу <https://gephi.org/users/download/>.

Программа поддерживает множество форматов представления графов, в том числе форматы таких популярных программ для анализа сетей как Pajek, UCINET, GraphVis, GML и др.

3. Файл графа, в поддерживаемом программой Gephi формате, загружается через меню *File*→*Open*. Основные окна (смотри рис. 2.1), необходимые для анализа графа, можно открыть через меню *Window*:

– подменю *Statistics* используется для расчета структурных характеристик графа;

– подменю *Graph* используется для отображения графа;

– подменю *Preview* *Settings* используется для установки параметров отображения узлов и ребер;

– подменю *Layout* используется для выбора алгоритма визуализации.

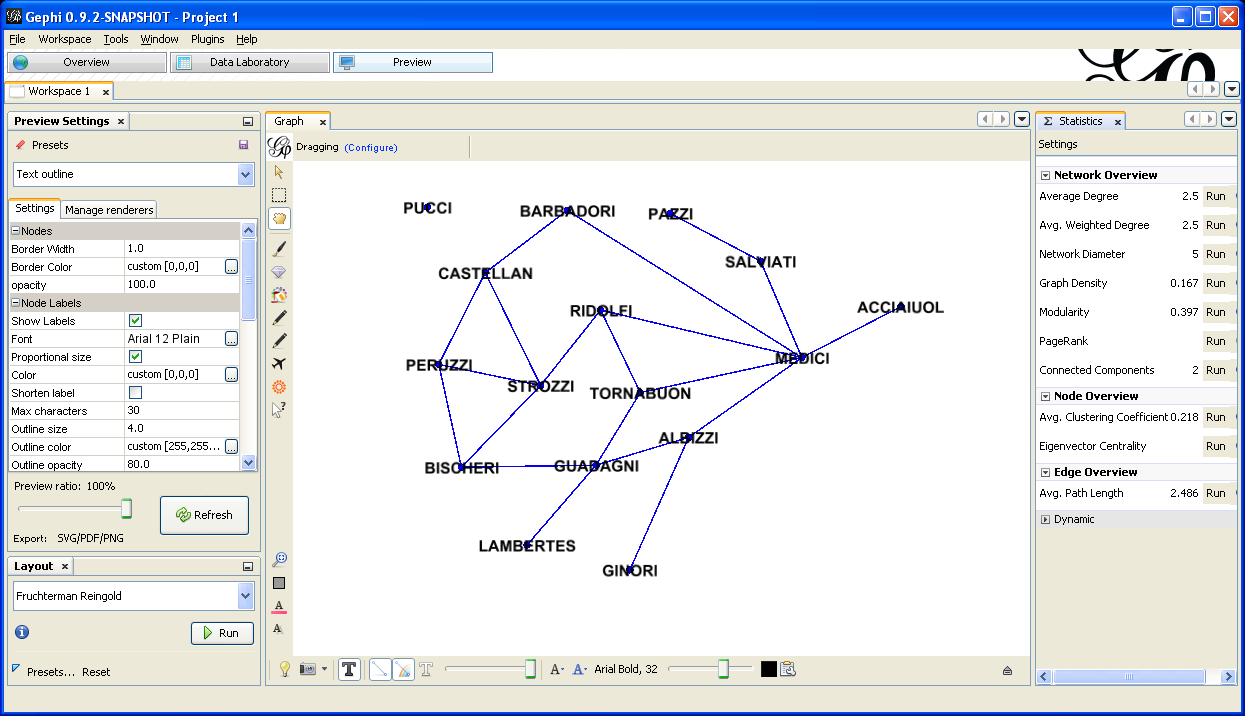


Рис. 2.1. Изображение графа, моделирующего сеть семейных кланов Флоренции в XIII-XIV вв. в программе Gephi

4. Для получения данных о структурных характеристиках вершин используется вкладка Data Laboratory, в которой в табличном виде можно получить данные о вершинах графа и экспортировать эти данные в формат .csv (рис. 2.2.).

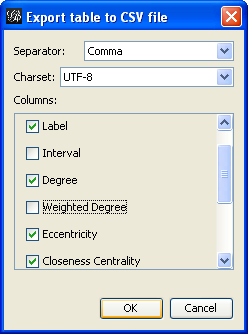
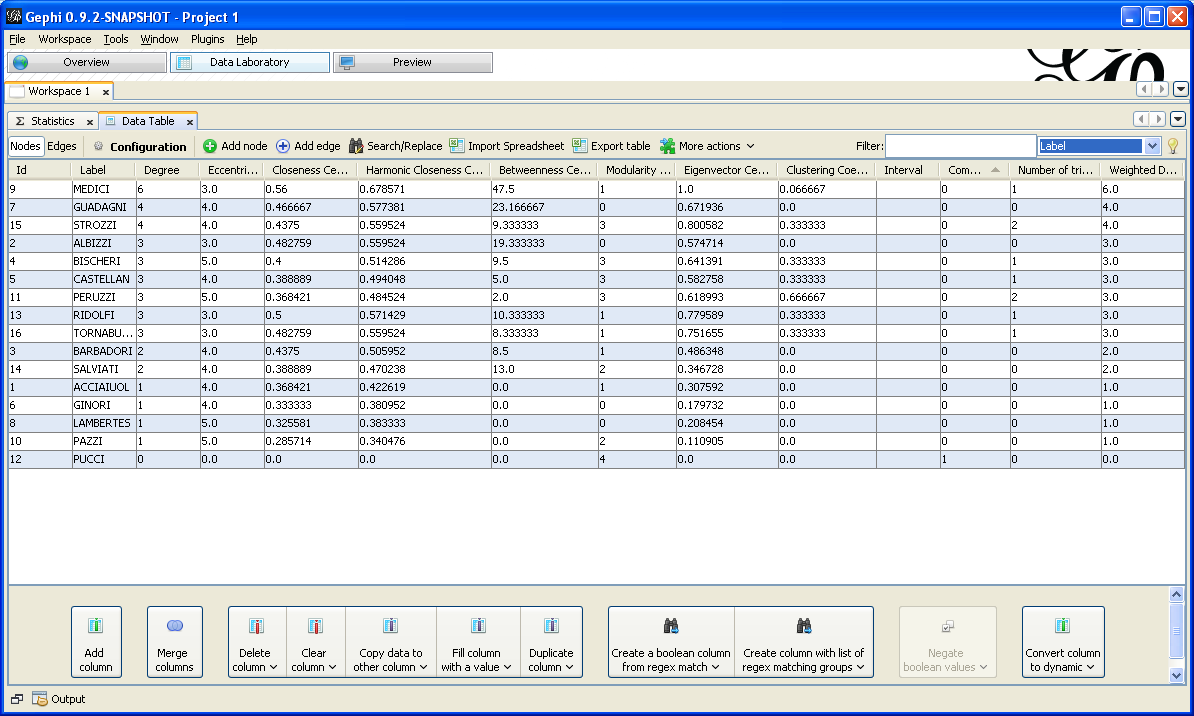


Рис. 2.2. Экспорт данных из программы Gephi в формат \*.csv, который легко импортируется в формат Excel

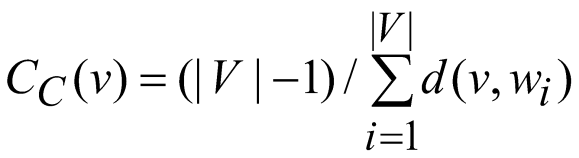
5. Будем выделять следующие характеристики.

а) Степень связности (degree centrality) *CD* – исторически первая и концептуально простая мера важности вершин. Эта мера определяется как количество deg(*v*) инцидентных данной вершине *v* ребер:

*CD*(*v*)*=* deg(*v*). (2.1)

Если говорить о моделируемой графом сети, то степень связности вершины графа определяет меру активности соответствующего узла сети в создании связей.

б) Степень близости к другим вершинам  (closeness centrality) *CC*(*v*) – обратная величина суммы кратчайших путей *d*(*v*,*wi*) от вершины *v* до других вершин *wi*:

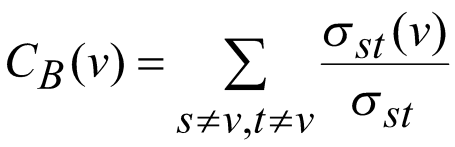
**,(2.2)

где |V| – число всех вершин графа.

Таким образом, чем более важной является вершина, тем меньше сумма кратчайших путей от нее до других вершин.

в) Степень посредничества(betweenness centrality) *СB*(*v*) – характеристика вершины, показывающая, насколько часто данная вершина лежит на кратчайших путях между другими вершинами.

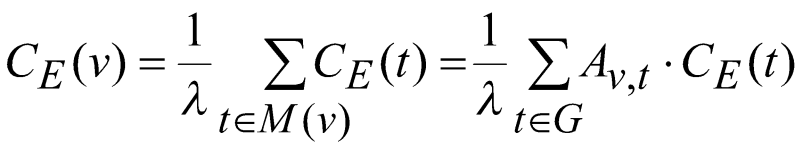
Степень посредничества *СB*(*v*) вычисляется как

, (2.3)

где σ*st* – количество кратчайших путей из вершины *s* в вершину *t*, а σ*st(v*) –количество этих путей, проходящих через вершину *v*.

Узел сети, моделируемый вершиной с высокой степенью посредничества, будет хорошо информирован при передаче информации при условии, что передача будет проходить по кратчайшим путям.

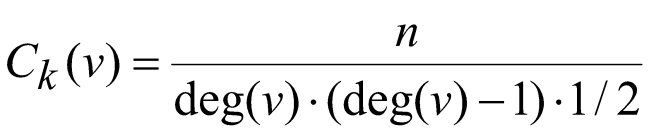
г)Влиятельность (eigenvector centrality) *СE*(*v*) – рекурсивная мера важности вершины, основанная на важности соседних вершин. Чем более влиятельны вершины, с которыми связана данная вершина, тем больше влиятельность самой вершины:

, (2.4)

где *M*(*v*) – множество соседних вершине *v* вершин, λ– константа, *Av,t* – элемент матрицы смежности (задается на основе связности вершин графа). Значения *CE*(*v*)можно получить, решив уравнение *Ax*= λ*x*, где *A* – матрица смежности, λ и *x* – соответственно, собственное значение и собственный вектор матрицы *А*. Значения элементов собственного вектора задают влиятельности соответствующих вершин.

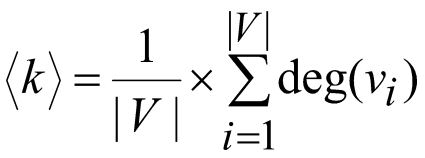
д) Эксцентриситет  *СExtr*(*v*) – минимальный кратчайший путь от даннойвершины *v* до максимально удаленной от нее вершины.

е) Коэффициент кластеризации *Сk*(*v*) вершины *v* – в простом графе характеризует вероятность того, что вершины, смежные данной вершине,являютсясмежными между собой.Если вершина *v* имеет deg(*v*)соседей с числом *n* связей между ними, то локальный коэффициент кластеризации этой вершины может быть рассчитан по формуле:

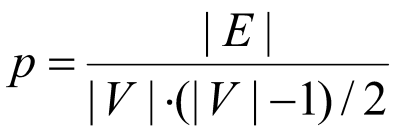
. (2.5)

6. **Глобальные характеристики графа** задают интегральные показатели, характеризующие сеть в целом. Среди них выделим следующие.

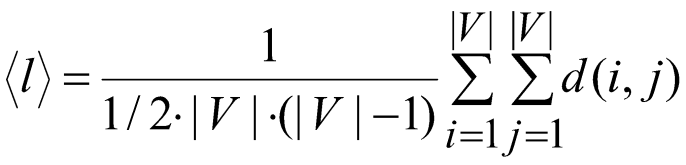
а) Средняя степень связности – характеристика, определяющая среднее арифметическое локальной степени связности deg(*vi*) вершины *vi* по всем *i* = 1,|*V*| вершинам:

. (2.6)

б) Плотность графа – это отношение числа |E| всех имеющихся ребер графа к максимально возможному количеству ребер в графе:

. (2.7)

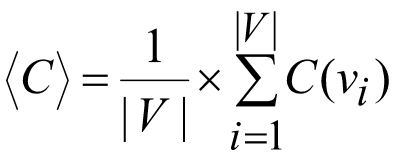
в) Среднее расстояние *l* между вершинами графа определяется как

, (2.8)

где *d*(*i*, *j*) – кратчайшее расстояние от вершины *i* до вершины *j*, измеренное в числе ребер.

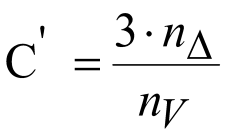
г) Диаметр графа – это максимальный из эксцентриситетов вершин графа. Для моделируемых графами сетей передачи данных этот показатель характеризует время отклика сети.

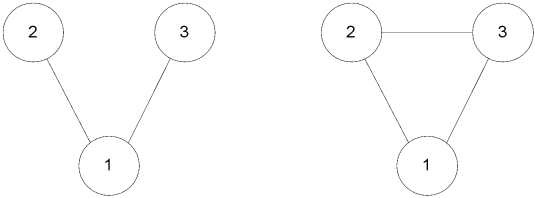
д)  Средний коэффициент кластеризации является средним арифметическим коэффициентов кластеризации всех вершин *Ci* и может быть рассчитан по формуле:

, (2.9)

где *С*(*vi*) – локальный коэффициент кластеризации вершины *vi*, задает вероятность того, что две смежные для *vi* вершины связаны между собой.

е)Глобальный коэффициент кластеризации  определяется как утроенное отношение утроенного числа  «треугольников» (см. рис 2.1) в графе к среднему числу  «вилок», т.е. путей длины 2:

 . (2.10)



a) б)

Рис 2.1 – Возможные подграфы на трех связных вершинах: а) «вилка» и б) «треугольник»

**Варианты заданий**

У всех студентов сети должны быть уникальными. При получении задания преподаватель регистрирует их в журнале. Базы данных сетей можно найти по следующим электронным адресам:

1) сети в формате Pajek: <http://vlado.fmf.uni-lj.si/pub/networks/data./>;

2) база графов Марка Ньюмана: <http://www-personal.umich.edu/~mejn/netdata/>.

**Форма отчета**

Листы в формате MS Excel, которые оформляются с пояснениями (пример оформления фрагмента листа представлен на рис. 2.3), проект Gephi, содержащий результаты анализа.

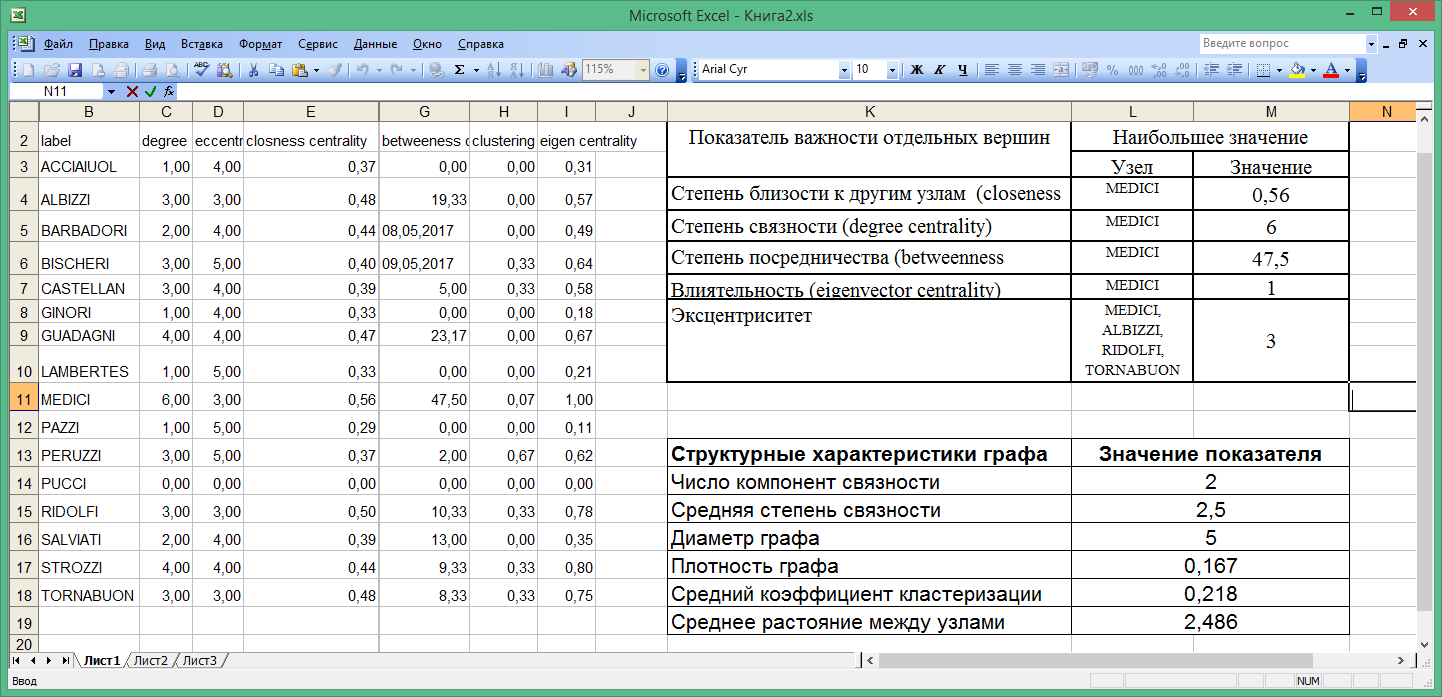


Рис. 2.3. Фрагмент листа таблицы к практическому занятию 1 (данные о сети семейных кланов Флоренции в XIII-XIV веке.)

**Контрольные вопросы**

1. Опишите исследуемую вами сеть. Какой узел является самым важным по влиятельности? Какой самый активный?

2. Создайте в Gephi простой неориентированный граф, содержащий 6-7 вершин и хотя бы один замкнутый путь. Рассчитайте в Gephi степень связности, степень близости, степень посредничества одного из узлов и проверьте расчеты вручную по формулам (2.1) – (2.3).

3. Используя созданный граф, найдите диаметр графа и плотность графа, а также найдите эти характеристики вручную.

4. Какой вероятностный смысл у глобального коэффициента кластеризации и среднего коэффициента кластеризации? Проверьте, корректно ли рассчитывает коэффициенты кластеризации Gephi.

***Практическое занятие 2*** *Анализ сообществ в сети*

**Цель работы**: анализ структуры сообществ сети в пакете igraph для среды R.

**Содержание работы.**

1. Подключение пакета igraph в среду R.

2. Загрузка файла, содержащего данные о структуре сети в среду R. Изучение информации об исследуемой сети.

3. Визуализация графа средствами пакета igraph (если размер графа не превышает 1000 вершин).

4. Расчет глобальных характеристик графа, а также структурных характеристик вершин с помощью методов пакета igraph.

5. Анализ сообществ в сети с помощью методов пакета igraph, оценка времени работы алгоритмов, модулярности решения, построение дендрограммы для оптимального по модулярности решения.

**Краткая теория и методические указания**

1. Загрузка пакета igraph в среду R осуществляется c помощью меню «Packages→LoadPackage…» (рис. 2.4). Однако, если в вашей среде R пакет igraph не установлен, то его нужно установить через меню «Packages→Install packages…».

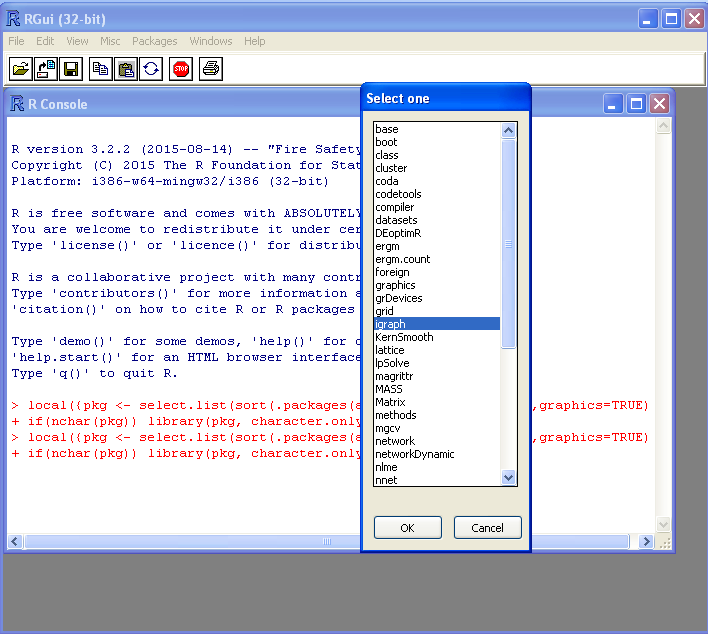


Рис. 2.4. Скриншот окна среды R в процессе подключения пакета igraph

2. Загрузка графа для расчета исследуемых характеристик вершин выполняется с помощью команд в среде R, представленных в приложении 3.

3. Для корректного расчета степени близости к другим вершинам (closeness centrality) и других характеристик необходимо удалить из графа все вершины, которые не входят в исследуемый компонент связности (исследуйте максимальный компонент связности).

4. На данный момент в библиотеке igraph разработаны и могут быть использованы следующие шесть алгоритмов выделения сообществ.

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

2) Алгоритм «label.propagation.community» основан на процедуре «распространения ярлыков» [2] 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 [3].

4) Алгоритм «spinglass.community» использует модель «спинового стекла» и «имитации отжига» (spinglass model and simulated annealing). Авторы интерпретируют такие термины как «когезия и адгезия в рамках исследования сетей в качестве меры принадлежности вершин к одному и тому же или разным сообществам» [4].

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» . Основан на вычислении собственных значений и собственных векторов того, что авторы называют «матрицей модульности», которая является «характеристикой сети и сама по себе не зависима ни от каких разделений сети на сообщества» [3]. 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» .

На рис. 2.5 приведен листинг, в котором для разбиения графа на сообщества используется алгоритм «fastgreedy.community». На рис. 2.6. приведен результат работы При этом граф отрисован с помощью алгоритма построения графов Фруктермана-Рейнголда (Fruchterman-Reingold), смотри рис.

marriage <-read.graph(file="C:\\florence.net", format="pajek")

ptm <- proc.time()

fc <- fastgreedy.community(marriage)

#fc <- fastgreedy.community(marriage)

#fc <- walktrap.community (marriage)

#fc <- label.propagation.community (marriage)

#fc <- edge.betweenness.community (marriage)

#fc <- leading.eigenvector.community (marriage)

pt<-proc.time()

pt-ptm

a=sizes(fc) # возвращает порядковый номер и размер сообществ

which.max(a) # вывод одного из них

modularity(marriage, fc$membership)

V(marriage)$color <- fc$membership+1 # изобразить граф

plot.igraph(marriage,vertex.label=V(marriage)$id,layout=layout.fruchterman.reingold, edge.arrow.size=0.5)

Рис. 2.5. Код для выделения, анализа и визуализация сообществ

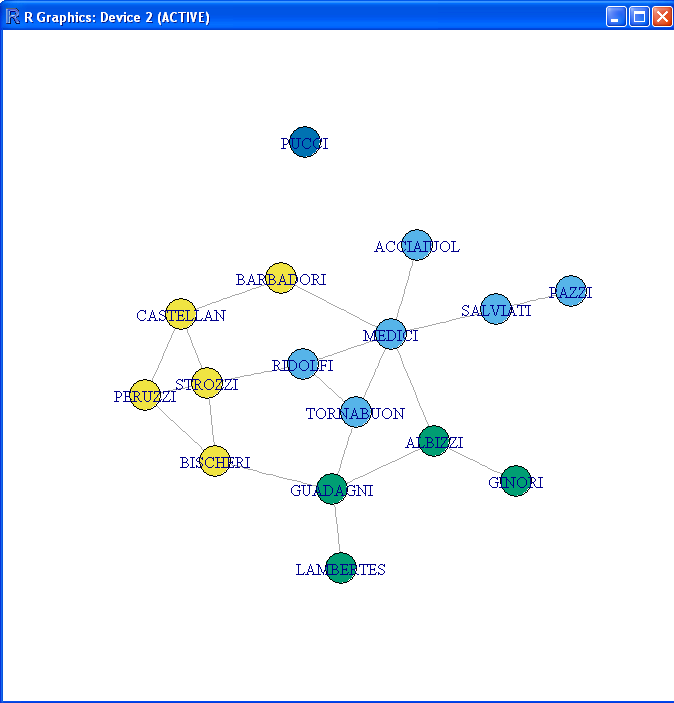


Рис. 2.6. Визуализация разбиения графа алгоритмом Фруктермана-Рейнголда

**Форма отчета**

Для защиты необходимо подготовить отчет, оформить его и передать на проверку в установленном порядке. В отчете необходимо объяснить результаты расчетов характеристик вершин графа и глобальных характеристик графа при анализе в программе Gephi и в среде R. Сделать вывод о качестве разбиений на сообщества при использовании различных алгоритмов. Данные рекомендуется хранить в формате MS Excel (рис. 2.7), которые оформляются с пояснениями.

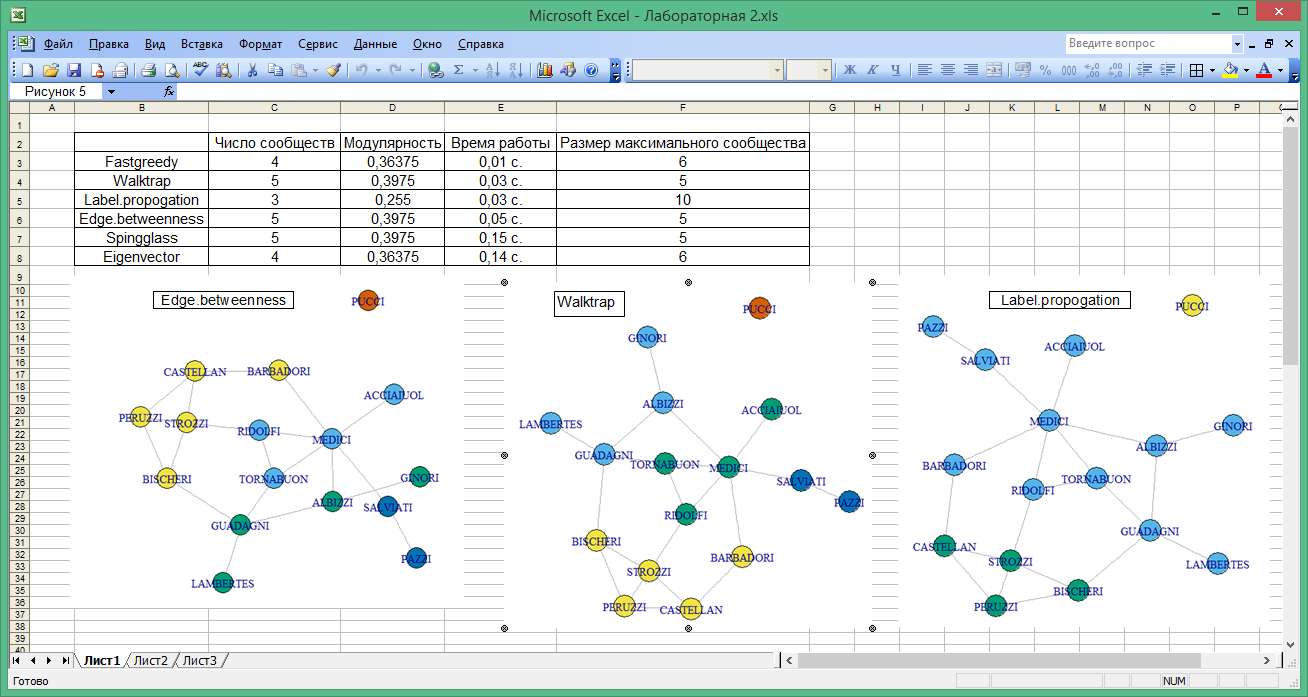


Рис. 2.7. Фрагмент листа таблицы к практическому занятию 2

**Варианты заданий**

Необходимо использовать ту же сеть, что и в практическом занятии 1.

**Контрольные вопросы**

1. Создайте с использованием igraph простой неориентированный граф, содержащий 6 вершин и хотя бы один замкнутый путь. Разбейте граф на сообщества алгоритмом **edge.betweenness.community** и отобразите дендрограмму, проверьте расчеты вручную.
2. Нарисуйте дендрограмму, если бы вы находили сообщества методом минимального сечения. Изменились ли бы результаты, если бы вы при этом использовали нормировку по числу вершин?
3. Посчитайте плотность внутренних и внешних связей одного из сообществ и сравните с плотностью графа.

***Практическое занятие 3****Моделирование процессов диффузии*

**Цель работы**: Исследование модели диффузии (модель Басса) в системе AnyLogic при использовании агентного и системно-динамического подходов.

**Содержание работы.**

1. Изучение модели Басса, зависимости числа «принявших инновацию индивидуумов» *N*(*t*) от времени моделирования.

2. Загрузка и изучение системно-динамической модели Басса в системе AnyLogic. Изменение параметров моделирования в соответствии с выданным вариантом.

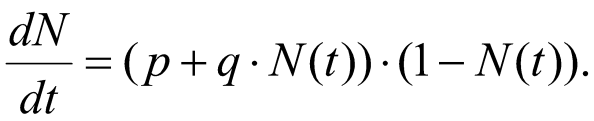
3. Загрузка и изучения анимированной агентной модели Басса в системе AnyLogic. Изменение параметров связей между агентами и согласование параметров моделирования с системно-динамическим подходом.

4. Сравнительный анализ исследуемых моделей. Оформление отчета.

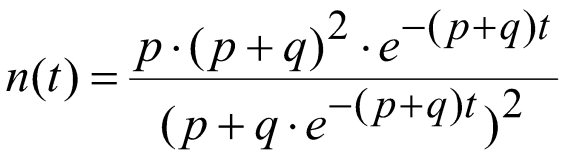
**Краткая теория и методические указания**

1. Модель Басса описывает процесс распространения инновации. Изначально инновация никем не принята, и для того, чтобы ее начали использовать, она рекламируется (или должны работать другие внешние факторы – если это новый вид антибиотиков, то о нем должны появляться публикации в медицинских журналах). В итоге определенная доля *p* индивидуумов (инноваторы) приобретает инновацию под воздействием рекламы. Также с учетом коэффициента подражания *q* часть индивидуумов (имитаторы) принимают инновацию в результате влияния тех, кто ее уже принял инновацию (внутренние факторы – авторитет других индивидуумов).

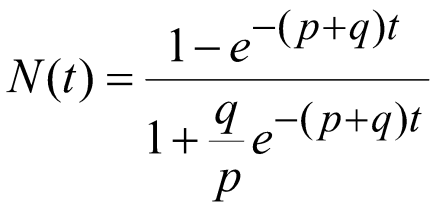
Скорость принятия инновации равна



Доля продажи в момент времени  может быть определены в виде:

. (2.11)

С использованием введенных обозначений для каждого момента времени *t* можно определить долю индивидуумов принявших инновацию с момента ее распространения, включая этот момент:

. (2.12)

2. При описании потоковой диаграммы (системно-динамический подход) используются переменные, которые накапливают значения с течением времени. В системно-динамической модели BassModel (которую можно открыть в главном окне AnyLogic через меню *Справка*→*Примеры моделей*) численность индивидуумов принявших и не принявших инновацию моделируется накопителями AdoptersA и PotentialAdoptersP, соответственно, а процесс приобретения продукта – потоком AdoptionRateAR (рис.2.8). Накопители обозначаются прямоугольниками, поток – вентилем, а динамические переменные – кружками. Стрелки обозначают причинно-следственные зависимости в модели.

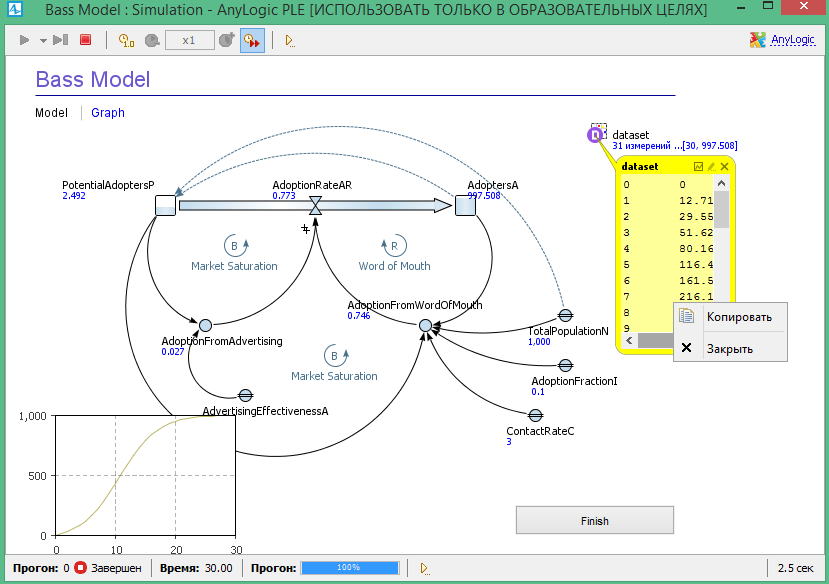


Рис. 2.8. Визуализация модели Басса при использовании системной динамики в AnyLogic

Модель зависит от параметров, представленных в таблице 2.1.

Таблица 2.1.

**Описания параметров модели Басса**

| **Обозначение параметров модели** | **Краткая характеристика** | **Используемые значения** |
| --- | --- | --- |
| TotalPopulationN | Общая численность населения | 1000 |
| AdvertisingEffectivenessA | Эффективность рекламы | 0,011 |
| ContactRateС | Число контактов в единицу времени | 3 |
| AdoptionFraction | Вероятность убеждения | 0,1 |

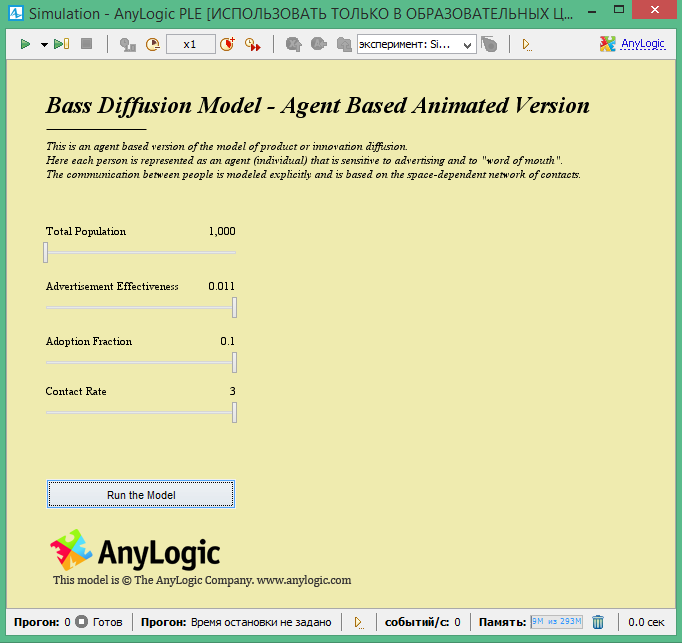
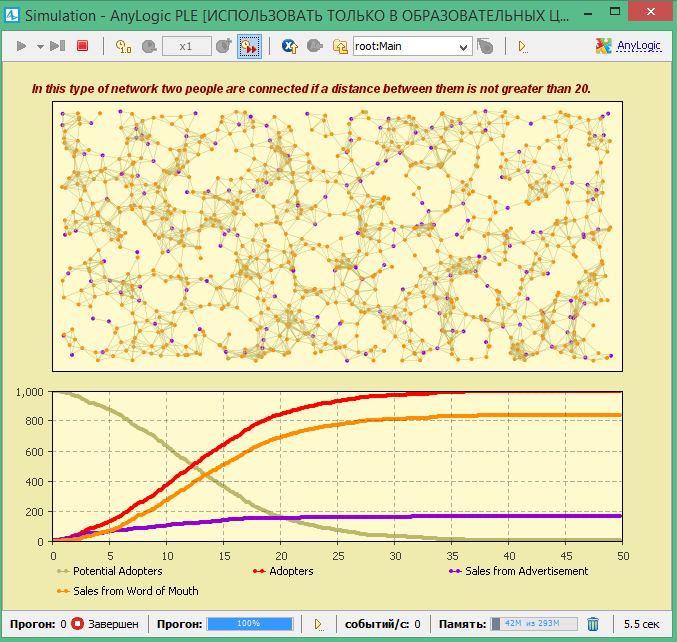
Интенсивность спонтанной инновации *p* очевидно равна параметру AdEffectiveness. Коэффициент подражания *q* равен произведению ContactRateС ×AdoptionFraction.

Вспомогательная переменная AdoptionFromWordOfMouth задает интенсивности принятия инновации имитаторами и равна:  
ContactRateC×AdoptionFractionI×PotentialAdoptersP×AdoptersA/TotalPopulationN.  
Вспомогательная переменная AdoptionFromAdvertising задает интенсивность принятия инновации индивидами-инноваторами и равна AdvertisingEffectivenessA× PotentialAdoptersP.

3. В модели BassDiffusionAgentBaseAnimated (которую можно открыть в главном окне AnyLogic через меню *Справка*→*Примеры моделей*) реализован агентный подход к построению модели Басса.

Для установки соответствующих параметров необходимо изменить визуальные элементы (рис. 2.9 а) для согласования с данными для модели Басса в системно-динамическом подходе (в зависимости от варианта меняется AdvertisingEffectivenessA, AdoptionFraction). Параметры ContactRateС и TotalPopulationN установите значения 3 и 1000, соответственно.

Шаг обновления «принявших инновацию» можно задать во вкладке Свойства (графика), к которому можно получить доступ, нажав левой кнопкой мыши на график на презентации модели (рис. 2.9 б). В режиме моделирования данные о результатах можно получить, нажав правой кнопкой мыши на график и из выпадающего меню выбрать пункт «Копировать все».

а) б)

Рис. 2.9. Визуализация модели Басса при анимированной агентной модели в AnyLogic. а) установка параметров моделирования б) визуализация процесса моделирования и построения графика изменения числа принявших инновацию агентов.

4. В агентной модели появляется возможность учитывать структуру взаимодействия агентов (в системно-динамическом подходе каждый взаимодействует с каждым). Поведение и свойства агентов задается на основе диаграммы состояний (рис. 2.10). На диаграмме задаются два условия перехода из состояния PotentialAdopter (еще не принявшего инновацию) в Adopter (принявшего инновацию) с заданной интенсивностью Ad и в результате получения сообщения WOM.

Параметры сети можно изменить в окне свойств модели Main (Вкладка «Пространство и сеть»).

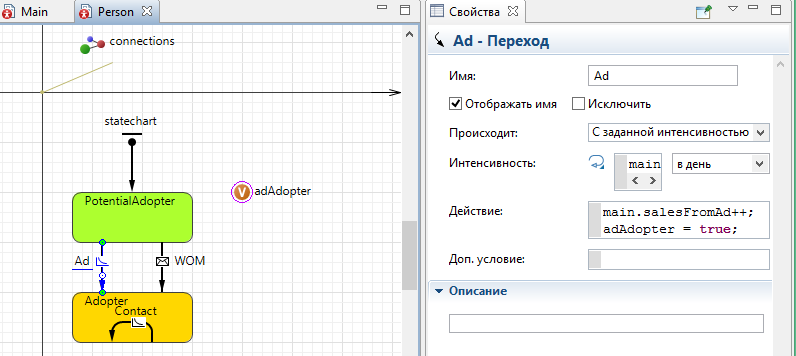


Рис. 2.10. Визуализация модели Басса при использовании системной динамики в AnyLogic

5. Для задания графа взаимодействий в свойствах модели выберите *тип расположения*: случайное; *тип сети*: согласно расстоянию; *радиус*: 20. Ширина и высота пространства 550 на 250, соответственно. Тем самым, задается случайное и равномерное распределение агентов в пределах пространства 250×550, причем если расстояние между агентами будет меньше 20, то агенты соединяются связями (см. рис. 2.9 б). Необходимо рассмотреть также случай, когда радиус равен 10 и 35.

5. Постройте графики изменения числа агентов *N*(*t*) принявших инновацию, а также изменения скорости *n*(*t*) принятия инновации (рис. 2.11).

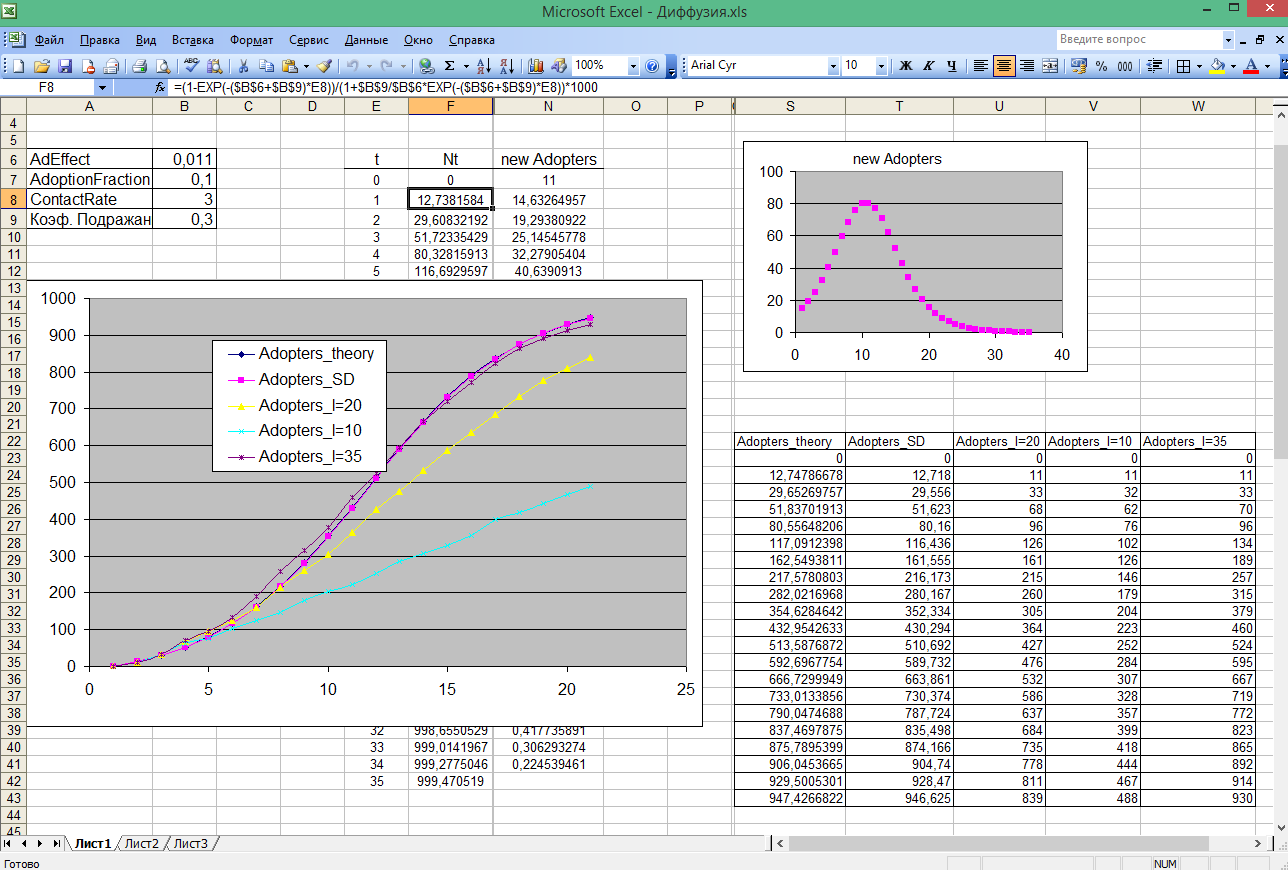


Рис. 2.11. Фрагмент листа таблицы к практическому заданию 3

**Варианты заданий**

| **№** | AdvertisingEffectivenessA | AdoptionFraction |  | **№** | AdvertisingEffectivenessA | AdoptionFraction |
| --- | --- | --- | --- | --- | --- | --- |
| **1** | **0,01** | **0,1** |  | **11** | **0,01** | **0,01** |
| **2** | **0,02** | **0,2** |  | **12** | **0,02** | **0,02** |
| **3** | **0,03** | **0,3** |  | **13** | **0,03** | **0,03** |
| **4** | **0,04** | **0,4** |  | **14** | **0,04** | **0,04** |
| **5** | **0,05** | **0,5** |  | **15** | **0,05** | **0,05** |
| **6** | **0,05** | **0,1** |  | **16** | **0,05** | **0,01** |
| **7** | **0,1** | **0,1** |  | **17** | **0,1** | **0,01** |
| **8** | **0,15** | **0,1** |  | **18** | **0,15** | **0,01** |
| **9** | **0,005** | **0,1** |  | **19** | **0,005** | **0,01** |
| **10** | **0,001** | **0,1** |  | **20** | **0,001** | **0,01** |

**Форма отчета**

Для защиты необходимо подготовить отчет, оформить его и передать на проверку в установленном порядке. В отчете привести результаты анализа числа принявших инновацию индивидуумов для аналитической модели, системно-динамической и агентной, а также скорости принятия инновации. Данные рекомендуется хранить в формате MS Excel, которые оформляются с пояснениями.

**Контрольные вопросы**

1. Дайте определение системно-динамического и агентного подходов. Какие преимущества дает использование каждого из этих подходов и какие недостатки эти подходы имеют.

2. Доработайте системно-динамическую и агентную модели, чтобы в них учитывалось, что со временем инновация может быть отвергнута, что вызовет необходимость ее повторного принятия с интенсивностью 0,01. Оцените точку равновесия, около которой колеблется число индивидуумов, принявших инновацию, в измененной модели.

3. Выведите формулу (2.12).