САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
Кафедра Математического моделирования энергетических систем

Екимовская Мария Васильевна

Выпускная квалификационная работа бакалавра

Кооперативные игры в исследовании силы генов

Направление 010400

Прикладная математика и информатика

Научный руководитель,
кандидат физ.-мат. наук,
доцент
Лежнина Е.А.

Санкт-Петербург
2017

Содержание

Введение

Белки являются структурными компонентами клеток и тканей и могут действовать по мере необходимости как ферменты для биохимических реакций в биологических системах. Большинство структурных и функциональных единиц наследственности живых организмов - генов содержат информацию для получения конкретного белка. Эта информация кодируется генами по средствам дезоксирибонуклеиновой кислоты (ДНК).

В настоящее время появляются и совершенствуются такие технологии сбора больших массивов данных с информацией, заложенной в ДНК, как секвенирование геномов, высокопропускные методы скрининга лекарственных средств и анализ микрочипов ДНК. Технология микрочипов позволяет произвести количественную оценку экспрессии генов (уровня способности генов контролировать синтез белка) в одном биологическом состоянии (например, опухоль).

Данные полученные путем экспериментов с микрочипами могут быть использованы для изучения фундаментальных биологических явлений, таких как развитие или эволюция, для определения функций новых генов, выяснения роли отдельных генов или групп генов в появлении болезни и контроле влияния лекарств и других соединений на экспрессию генов. Обработкой полученной информации с применением математического аппарата занимается такой раздел науки как биоинформатика. Одна из задач биоинформатики заключается в изучении регуляции генов с последующим созданием профайла экспрессии белка с использованием данных из микрочипов или других технологий. Другими распространенными проблемами являются анализ мутаций при раке, эволюция вирулентности и ВИЧ-инфекции, и т.д.

В данной работе были рассмотрены некоторые приложения теории игр для анализа биологических данных. Очевидно, что такие приложения не отвечают на нормативные вопросы, как и не дают советы группам переменных (например, генам) о том, как они должны вести себя внутри биологической клетки. В данном контексте, теория игр используется для описания поведения переменных и предсказания исхода их взаимодействия.

Цель работы: изучить теорию микрочиповых игр, коалиционных игр и их применение к вычислению силы генов; применить теорию на практическом примере с реализацией алгоритма.

Постановка задачи

Определения:

Ген - структурная и функциональная единица наследственности живых организмов.

Дезоксирибонуклеиновая кислота (ДНК) — макромолекула, обеспечивающая хранение, передачу из поколения в поколение и реализацию генетической программы развития и функционирования живых организмов

Рибонуклеиновая кислота (РНК) — одна из трёх основных макромолекул, которые содержатся в клетках всех живых организмов.

Транскрипция ДНК – процесс синтеза РНК с использованием ДНК в качестве матрицы, происходящий во всех живых клетках.

Трансляция ДНК – процесс синтеза белка из аминокислот на матрице информационной (матричной) РНК (иРНК, мРНК), осуществляемый рибосомой.

Экспрессия гена – это процесс, в ходе которого наследственная информация от гена преобразуется в функциональный продукт — РНК или белок.

ДНК-микрочип – технология, используемая в молекулярной биологии и медицине для определения ДНК или РНК (обычно после обратной транскрипции), которые могут быть как белок-кодирующими, так и не кодировать белки. Измерение генной экспрессии посредством кодирующей ДНК называется профилем экспрессии, или экспрессионным анализом.

Патогенез – механизм зарождения и развития болезни и отдельных её проявлений.

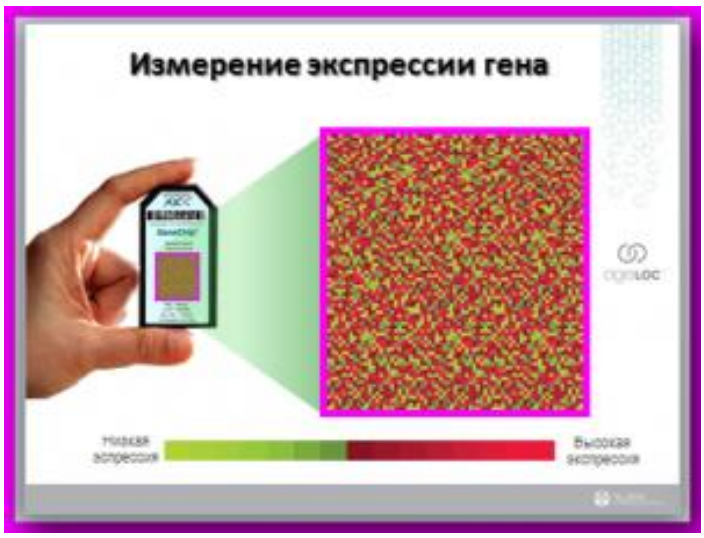

Рисунок 1

Экспрессия гена происходит, когда генетическое информация, содержащаяся в ДНК, транскрибируется в матричную РНК, а затем транслируются в белки. Рассматриваемая в данной работе технология микрочипов работает, используя способность данной мРНК (матричная рибонуклеиновая кислота) молекулы определенно связываться или гибридизироваться с ДНК-матрицей, из которой она возникла. В одном эксперименте используя массив, содержащий много образцов ДНК, можно определить, уровни экспрессии сотен или тысяч генов внутри клетки путем измерения количества мРНК, связанной с каждым сайтом в массиве. Для этого создана специальная шкала цветов (изображенная на рисунке 1 [1]), переходящая от зеленого к красному. Стоит отметить, что, если у гена экспрессия по шкале показывает красный цвет это не означает, что он анормально экспрессирующий. У каждого гена свои значения экспрессии (могут измеряться сотнями или тысячами). Сравнивать значения экспрессии различных генов без использования дискриминантного метода категорически нельзя.

Существует несколько различных экспериментальных платформ, основанных на технологии микрочипов [2]. Однако общая цель экспериментов с микрочипами состоит в том, чтобы последовательно генерировать матрицу экспрессии генов, строки которой (возможно, тысячи) индексируют гены и столбцы (обычно в порядке десятка) индекс исследуемых образцов. Числа в матрице представляют количественный уровень экспрессии генов в образцах.

Целью этой работы является решение проблемы количественного определения относительной релевантности генов в определенном сценарии - таком как патогенез генетической болезни – на основе информации, предоставленной экспериментами с использованием микрочипов, с учетом уровня взаимодействия между генами.

Сложные экспериментальные нюансы, связанные со сбором данных из микрочипов, подчеркивают необходимость предварительного анализа данных (например, оценка качества грубых данных, процедуры нормализации), с целью сокращения систематических ошибок, возникающих в результате нескольких экспериментальных процедур.

Несмотря на уменьшение экспериментальных погрешностей, после нескольких работ по анализу микрочипов за последние несколько лет [2][3][4] на практике проблема полного удаления экспериментальной изменчивости еще не решена. По этой причине в применяемом в этой работе подходе я ссылаюсь на наблюдаемый средний уровень взаимодействия группы генов, например, среднее число образцов с болезнью Паркинсона, такое что группа генов может рассматриваться как ответственная за появление болезни в соответствии с заранее определенным принципом причинности: чем выше число наблюдаемых образцов, тем ниже вероятность того, что случайность может повлиять на выводы, предоставленные моделью.

Основная идея этой модели исходит из теории коалиционных игр и используется тот же формальный язык для моделирования взаимодействия между генами (игроками), в связи с биологическим состоянием, представляющим интерес, например, патогенезом генетического заболевания.

Рассматриваемая игра происходит из сравнения двух матриц данных об экспрессии генов. Первая из образцов, представляющих интерес (с определенным заболеванием или генетической особенностью), другая со здоровыми ДНК. Сначала мы используем дискриминантный метод на каждом образце, чтобы разделить весь набор генов, то есть те гены, которые показывают в значительной степени отличающуюся от нормальных образцов экспрессию, и те, у которых уровень экспрессии соответствует здоровым образцам. На этом этапе модели для каждого отдельного гена, в качестве способа соотношения его в ту или иную группу, используются границы интервалов, содержащие большинство данных в нормальном распределении этого гена. Затем вводится отношение причинности (также называемый принцип достаточности), который непосредственно определяет характеристическую функцию игры. В качестве биологического значения индекса релевантности, используемого для измерения «силы» генов будем рассматривать такие элементы коалиционной теории игр, как вектор Шепли и индекс Банзафа.

Обзор литературы

Первым вопросом, появляющимся в начале изучения микрочипов и генетики, является «как и где берут данные об экспрессии генов?». В данной работе рассматривается метод ДНК микрочипов. В конце 90-х использование массивов ДНК высокой плотности для мониторинга экспрессии генов в масштабе генома представило собой фундаментальный прогресс в биологии. В частности, появилась возможность определить структуру экспрессии всех генов, например, дрожжей, используя микрочипы, которые гибридизируются с каждым из примерно 6000 генов в геноме. В обзоре Zhang M. Q. «Large-scale gene expression data analysis: a new challenge to computational biologists» автор рассматривает три эксперимента, связанных с регуляцией транскрипции и делает «вызов» вычислительным биологам, пытающимся извлечь функциональную информацию из таких крупномасштабных данных. В настоящее время данная технология широко применяется специалистами разных профилей.

Любой эксперимент с микрочипами включает в себя ряд различных этапов. Во-первых, это дизайн эксперимента. Исследователи должны решить, какие гены будут печататься на массивах и какие источники РНК должны быть гибридизованы с массивами. Во-вторых, после гибридизации следует несколько этапов очистки данных или «низкоуровневый анализ» данных. Полученные значения должны быть нормализованы для корректировки на смещение краски и на любые систематические изменения, отличные от тех, которые обусловлены различиями между исследуемыми образцами РНК. В-третьих, нормированные значения должны быть проанализированы с помощью различных графических и численных средств для выбора дифференциально экспрессируемых генов или для поиска групп генов, профайлы экспрессии которых могут надежно классифицировать различные РНК источники в осмысленные группы. Краткий обзор этих процессов и методов работы с ними можно найти в статье Smyth, G. K., Yang, Y.-H., Speed, T. P. «Statistical issues in cDNA microarray data analysis».

Источником технических знаний по микрочипам ДНК также является книга Dhammika Amaratunga & Javier Cabrera “Exploration and Analysis of DNA Microarray and Protein Array Data”. Эта книга является своеобразной методичкой для проведения экспериментов с помощью данной новой технологии для исследователей различных дисциплин. В качестве введения авторы рассматривают основные понятия из биологии такие как: гены, геном, ДНК, экспрессия генов и т.д., а также роль изучения генома в фармацевтической и медицинской сферах. Затем описывается процесс проведения различных видов экспериментов с микрочипами и способы оценки полученных результатов. Также в книге приведены способы обработки полученных изображений и различные методы подсчета уровня экспрессии.

Интересным примером работы над микрочипами служит статья Arfin, S. M., Long, A.D., Ito, E.T., Tolleri, L., Riehle, M. M., Paegle, E. S., and Hatfield, G. W. «Global gene expression profiling in *Escherichia coli* K12:
the effects of integration host factor». Авторы использовали микрочипы для оценки уровня экспрессии около 4000 генов одного из наиболее изученных на данный момент организмов E.coli, выращенных в минимальных средах и характеризующих глобальные изменения при кратковременном воздействии высоких температур. Были идентифицированы два набора генов, реагирующих на тепловой удар, один усиленный и другой репрессированный.

Другой труд от калифорнийских ученых Pierre Baldi & Wesley Hatfield “DNA Microarrays and Gene Expression. From experiments to Data Analysis and Modeling” предлагает уже более прикладную информацию по статистической обработке получаемых в процессе экспериментов результатов. Авторы ставят проблемы, связанные со сбором и обработкой данных и предлагают их решения. Например, большинство, если не все, гены действуют совместно с другими генами и стоит задача отделения их влияния в показаниях микрочипов ДНК. Предлагается использовать различные методы кластеризации, от метода К-средних до иерархической кластеризации, факторный анализ, деревья решений, байесовские и нейронные сети. Каждый метод имеет различные преимущества в зависимости от конкретной задачи и свойств анализируемого набора данных. Это, вместе с погрешностью оборудования(«шум»), не позволяет выбрать лучший из них.

Система кластерного анализа для генома с использованием данных экспрессии от гибридизации ДНК микрочипов, которая использует стандартные статистические алгоритмы для организации генов в соответствии с сходством в структуре генной экспрессии, описана в статье Eisen M. B., Spellman P. T., Brown P. O., and Botstein D. «Cluster analysis and display of genome-wide expression patterns.»

Другой статистический подход был использован в статье R.J. Moser, A. Reverter, Caroline A. Kerre, K.J. Beh «A mixed-model approach for the analysis of cDNA microarray gene expression data from extreme-performing pigs after infection with Actinobacillus pleuropneumoniae». Цель авторов состояла в обнаружении генов, участвующих в предоставлении наследуемых различий в восприимчивости к общим инфекциям в интенсивном производстве свиней. Анализ микрочипов был использован для идентификации двух наиболее сильно реагирующих на заражение свиней. Образцы крови и микрочипы экспрессии в течении 24 часов после инфицирования сравнивали с использованием двумерного смешанного модельного подхода, в котором взаимосвязь гена, находящегося под воздействием инфекции, и его иммунологический статус рассматривалась как случайный эффект. С помощью байесовской модели кластеризации со смесью нормальных распределений был приведен список из 307 дифференциально экспрессируемых генов, из которых 179 в последствии были изменены у наиболее восприимчивых особей. Эти результаты являются доказательством

того, что предлагаемый статистический подход полезен для повышения уровня знаний о механизмах, участвующих в иммуногенетике.

Разработка новых технологий порождает возрастающую потребность в анализе данных биологических исследований. В связи с этим появляются и совершенствуются различные технологические платформы. Книга Parmigiani, G., Garrett, E. S., Irizarry, R. A., and S. L. Zeger, S. L. «The Analysis of Gene Expression Data: Methods and Software» посвящена анализу данных об экспрессии генов, полученных с помощью технологии микрочипов. Цель труда состоит в том, чтобы дать практические рекомендации о том какие статистические подходы и пакеты могут быть использованы для биологических проектов. Книга представляет собой сборник глав, написанных авторами статистического программного обеспечения для анализа данных микрочипов. Каждая глава описывает концептуальные методологические основы инструментов анализа данных, а также их программного обеспечения. Методы затрагивают все аспекты статистики анализа микрочипов, от аннотаций и фильтрации до кластеризации и классификации.

Для получения информации по экспрессии генов в настоящее время существует множество баз данных от биологических институтов и лабораторий, в том числе находящихся в открытом доступе. Одна из таких популярных платформ - Princeton University Microarray Database. Ее плюсы: общедоступность, поддержка сложный инструментов нормализации данных. К минусам можно отнести поддержку только «spotted» микрочипов, которые позволяют делать около 10 000 проб за раз, в отличии от чипов формата affymetrix, обрабатывающих 500 000 проб за раз. Другой университетской базой данных является платформа MaxD от Мичиганского университета. Данные в ней подготовлены к обработке с помощью MySQL и языка программирования Java, но она закрыта для общего доступа. Для получения быстрой и понятной аннотации по основным функциям и характеристикам генов можно воспользоваться базой BioGPS, она интуитивно понятна и доступна всем. Платформа Bgee состоит из массивов данных по экспрессии генов разнообразных животных. В своей работе я использую публичный репозиторий данных о геномике Gene Expression Omnibus национального центра биотехнологической информации(NCBI). Этот ресурс является наиболее расширенным, с наибольшим на данный момент количеством данных, а также поддерживает стандарт MIAME (минимальная информация об эксперименте с микрочипами). Этот стандарт определяет требования к информации, необходимой для однозначной интерпретации результатов эксперимента, и для потенциального воспроизведения эксперимента. Подробнее о нем можно найти в [13].

В настоящее время существует множество областей, работающих над обработкой биологической информации, полученной с помощью микрочипов, одной из таких областей является теория игр. Обзор используемых на данный момент инструментов теории игр в применении к

биоинформатике можно найти в статье Moretti, S., Vasilakos, Athanasios V. «An overview of recent applications of Game Theory to bioinformatics». Авторы рассматривают такие области теории игр, как эволюционная теория игр, в том числе эволюционно стабильные стратегии, а также два подхода микрочиповых игр, использующих вектор Шепли.

Более подробно идею использования кооперативной теории игр рассмотрела компания итальянских ученых во главе с Stefano Moretti. Именно он защитил диссертацию на тему использования теории игр в микрочиповых играх, что послужило толчком к развитию этой темы и появлению множества статей. Например, работа «Minimum cost spanning tree games and gene expression data analysis» того же Moretti рассматривает метод основанный на задаче минимального остового дерева для представления схожести между парами генов и вводит понятие объединения коалиций в терминах данного раздела теории игр. В качестве решения вместо вектора Шепли автор использует специфическое решение для минимального остового дерева – P-value [16]. При таком подходе не требуется дискриминантного метода для вычисления матрицы анормальной экспрессии, но необходимо ввести некоторую произвольность для определения подходящего понятия сходства между парами генов и уровня подобия коалиций генов.

В статье «A game theoretical approach to the classification problem in gene expression data analysis» V. Fragnelli и S. Moretti рассматривают игру с генами в качестве игроков для анализа силы групп генов в классификации образцов в определенные классы (например, класс образцов из нормальных тканей и из тканей больных раком). Такие игры очень близки к микрочиповым играм и в некоторых численных примерах авторы используют вектор Шепли для вычисления генов с высоким влиянием в вычислении образцов.

Кооперативная теория игр впервые была использована для анализа генов в работе [18] в качестве приложения для изучения множественного возмущения вектора Шепли. Целью работы было выявление релевантности с точки зрения причинной ответственности некоторых генов при выполнении определенных функций в дрожжевых клетках. В своем подходе авторы оценивают важность каждой коалиции используя в качестве показателя эффективности системы определенную функцию или свойство (например, способность системы выдерживать УФ-облучение). Чтобы получить такие данные, они провели серию экспериментов, где гены каждого различного подмножества из n генов были нарушены одновременно. В каждом эксперименте также измерялся показатель производительности и давалась оценка соответствующему подмножеству возмущенных генов, что позволило описать кооперативную игру.

Основная идея данной работы была взята из статьи Stefano Moretti, Fiorovante Patrone и Stefano Bonassi «The class of microarray games and the relevance index for genes». В своей работе авторы ввели понятие микрочиповой игры, партнерства генов, рассмотрели аксиоматическую характеристику вектора Шепли в применении к анализу генов и произвели численный расчет индекса релевантности генов при образовании опухоли у человека. В дальнейшем индекс Банзафа так же был рассмотрен как решение микрочиповой игры в статье [22].

Глава 1. Математическая модель

1.1 Определения

Игра – математическая формализация взаимодействия нескольких участников.

Коалиция - подмножество игроков.

Кооперативная игра (N, v) определяется множеством игроков $I = \{1, 2, \ldots, n\}$ и характеристической функцией v, ставящей в соответствие каждому подмножеству игроков $S \subseteq I$ тот выигрыш, который они могут получить, объединившись в коалицию S.

Вектор Шепли – это математическое ожидание вклада каждого игрока, при предположении, что коалиции S мощности s (при $0 \leq s \leq n - 1$) возникают с одинаковыми вероятностями, и что все коалиции одной и той же мощности s также равновероятны.

Индекс Банзафа – это математическое ожидание вклада каждого игрока, при предположении, что все коалиции S возникают с одинаковыми вероятностями.

1.2 Формализация кооперативной игры в контексте микрочиповой игры

Пусть $G = \{1, 2, \dots, n\}$ набор из n генов, $S_R = \{1, 2, \dots, r\}$ множество контрольных образцов, т.е. образцы клеток из здоровых тканей, и $S_D = \{1, 2, \dots, d\}$ множество образцов из тканей представляющих интерес.

Цель экспериментов с микрочипами состоит в том, чтобы сопоставить каждому образцу $j \in S_D \cup S_R$ профайл экспрессии $(a_{ij})_{i \in G}$. Здесь $a_{ij} \in R$ относительная величина экспрессии гена i в образце j по отношении к контрольным образцам. Глобально такие величины экспрессии буду называться набором данных микрочипового эксперимента. В дальнейшем будет использоваться набор данных, предварительно обработанный методом, обычно называемым нормализацией [6], который позволяет сравнивать интенсивность экспрессии из генов из разных образцов. Целью нормализации является корректировка любого смещения, которое возникает в результате изменения микрочиповых технологий, а не биологических различий между образцами РНК или берущимися пробами.

Набор данных может быть представлен в форме двух матриц экспрессии $A^{S_R} = (A^j)_{j \in S_R}$ и $A^{S_D} = (A^j)_{j \in S_D}$, где индекс представляет столбец, являющийся профайлом экспрессии образца j.

Пусть $\mathcal{G}$ – класс всех кооперативных игр и $C \subseteq \mathcal{G}$ подкласс всех коалиционных игр. Тогда будем говорить, что рассматривая множество игроков N мы определяем класс $C^N \subseteq \mathcal{G}$ как класс кооперативных игр на C с множеством игроков N.

Микрочиповой экспериментальной ситуацией (далее МЭС) будем называть набор

$$E = < \mathcal{G},\ S_R,\ S_D,\ A^{S_R},\ A^{S_D} >.$$

В качестве первого шага анализа необходимо определить, экспрессия каких генов больных клеток выражается сильнее нормы по отношению к матрице здоровых в соответствии с определенным методом дискриминации. Ген i, экспрессия которого в образце проявляется вне нормы, можно представить как 1 в качестве значения булевой переменной b_{ij}. Анормальным профайлом экспрессии будем называть вектор $B^j = (b_{ij})_{i \in \mathcal{G}}$. Дискриминантный метод может быть выражен как отображение m, присваивающее каждому профайлу экспрессии из больных образцов соответствующий анормальный профайл. Следовательно, вся информация о различиях в экспрессии генов в образцах из S_D от образцов из S_R может быть представлена через матрицу анормальной экспрессии(далее МАЭ) с применением некоторого метода дискриминации к МЭС: $B^{E,m} \in \{0, 1\}^{\mathcal{G} \times S_D}$.

Существует множество различных дискриминантных методов. Первый самый простой и очевидный – наивный метод для двух классов: 0 и 1, где 1 обозначает анормально экспрессирующий ген, 0 нормально экспрессирующий ген:

$$(\widehat{m}(A^{S_D}, A^{S_R}))_{ij} = \begin{cases} 1, & if\ A_{ij}^{S_D} \geq max_{h\in\{1,\dots,|S_R|\}} A_{ih}^{S_R}\ or\ A_{ij}^{S_D} \leq min_{h\in\{1,\dots,|S_R|\}} A_{ih}^{S_R}, \\ 0, \text{иначе.} \end{cases}$$

Другой метод более консервативный. Для каждого $i \in N$ и каждого $j \in S_D$ выполняется:

$$(\overline{m}(A^{S_D}, A^{S_R}))_{ij} = \begin{cases} 1, & if\ A_{ij}^{S_D} \geq p_i^{25\%}\ or\ A_{ij}^{S_D} \leq p_i^{75\%}, \\ 0, \text{иначе.} \end{cases}$$

Где $p_i^{25\%}$ и $p_i^{75\%}$ соответственно 25ый и 75ый перцентили экспрессионного распределения гена i в соответствующей матрице экспрессии A^{S_R} для каждого $i \in N$.

Также необходимо ввести понятие опоры вектора. Пусть $W \in \{0,1\}^N, n \in \{1,2,\dots,n\}$. Определим опору W, обозначаемую sp(W), набором:

$$sp(W) = \{i \in \{1,\dots,n\} | W_i = 1\}.$$

Рассмотрим микроматричную экспериментальную ситуацию $E = < \mathcal{G}, S_R,\ S_D,\ A^{S_R},\ A^{S_D} >$ и дискриминантный метод m. Определим микроматричную игру как коалиционную игру (N, v), где

- Конечное множество генов N будет конечным множеством игроков,
- v – характеристическая функция, такая что v(∅) = 0. Присваивает каждой коалиции $T \in 2^N\{\emptyset\}$ среднее значение количества образцов с опухолью, определяемое по T в соответствии с принципом достаточности для групп генов.

Более точно характеристическая функция будет вычисляться по формуле:

$$v(T) = \frac{|\theta(T)|}{|S_D|}$$

Где $|S_D|$ – мощность множества больных образцов, а $|\theta(T)|$ мощность множества $\theta(T) = \{k \in S_D | Sp(B^{E,m}(k)) \in T, Sp(B^{E,m}(k)) \neq \emptyset\}$.

Условие $Sp(B^{E,m}(k)) \neq \emptyset$ обусловлено практическими соображениями относительно интерпретации принципа достаточности для групп генов на образцах, где гены не проявляют каких-либо анормальных свойств экспрессии. Предполагается, что такие образцы способствуют уменьшению уровня ассоциации больных образцов с нужной болезнью.

Класс микрочиповых игр обозначим символом $\mathcal{M}$.

Обозначим $|N|$ мощность конечного множества N. Вектором выигрыша (распределения) $(x_1, \dots, x_n)$ кооперативной игры (N, v) будет $|N|$-размерный вектор описывающий выигрыш игроков, такой что каждый игрок $i \in N$ получает x_i. Решением для класса кооперативных игр C будем называть функцию ψ определяющую вектор выигрыша $\psi(v)$: $C^N \to R^N$. В контексте микрочиповых игр, решение рассматривается как вектор ранжирования силы генов, то есть ген получивший наибольший «выигрыш» является наисильнейшим в данной выборке и так далее.

Одним из самых популярных решений на данный момент является вектор Шепли. Для его подсчета, кроме ранее введенных элементов необходимо ввести понятие личного вклада каждого гена в образование генотипа, определяемого формулой:

$$m_i(v, S) = v(S) - v(S \backslash \{i\})$$

Полученные нами данные позволяют ввести вектор Шепли на микрочиповую игру (N, v):

$$\varphi_i(v) = \sum_{S \in N : i \in S} \frac{(s-1)!\,(n-1)!}{n!} m_i(v, S)$$

где $i \in N, s = |S|$и $n = |N| -$ мощности коалиций.

Другим решением кооперативной игры является индекс Банзафа:

$$\beta_i(v) = \sum_{S \subseteq N : i \in S} \frac{1}{2^{n-1}} m_i(v, S)$$

для каждого $i \in N$.

Также, для обоснования возможности использования решений кооперативной теории игр в исследовании силы генов необходимо ввести все определения и формулы, прописанные выше в терминах простой игры.

Простой игрой будем называть игру (N, u_R) на $R \subseteq N$, где $u_R(T) = 1$, если $R \subseteq T$ и $u_R(T) = 0$, иначе. Любая кооперативная игра (N, v) может быть записана в виде линейной комбинации единогласных игр:

$$v = \sum_{S \in N, S \neq \emptyset} \lambda_S(v) u_S,$$

где $\lambda_S(\mathrm{v})$ коэффициент единогласия.

Коэффициент единогласия определяется по формуле:

$$\lambda_S = \frac{\overline{\lambda_S}}{|S_D|},$$

где $\overline{\lambda_S} = |\{k \in S_D |\ sp(B^{E,m}(k)) = S\}|$ количество возникновений коалиций S в качестве опор матрицы анормальной экспрессии $B^{E,m}$.

Альтернативная формула вектора Шепли с использованием простых игр:

$$\varphi_i(v) = \sum_{S \in N : i \in S} \frac{\lambda_S(v)}{|S|},$$

где $i \in N$

Альтернативная версия индекса Банзафа:

$$\beta_i(v) = \sum_{S \subseteq N : i \in S} \frac{\lambda_S(v)}{2^{S-1}},$$

для каждого $i \in N$.

Глава 2. Аксиоматическая характеристика решений кооперативной теории игр в применении к микрочиповым играм

2.1 Генные регуляторные сети как партнерство генов

Генные регуляторные сети (или сигнальные пути) в общем случае представляют собой сложные системы взаимодействия, состоящие из белков-лигандов (активаторов и ингибиторов), их рецепторов, посредников и факторов транскрипции, связывающихся с ДНК и активирующих транскрипцию генов-эффекторов [21]. Изучение механизмов работы генных регуляторных сетей представляет большой интерес для понимания генетического контроля развития отдельных органов или их систем. Кроме этого, понимание таких механизмов предоставляет широкие возможности для направленного изменения транскрипционной активности конкретных генов, что находит свое применение в лечении заболеваний различной этиологии. Более подробно ознакомиться с механизмами регуляции генов и различными биохимическими процессами можно в [22].

Наибольшей сложностью в понимании механизма генных регуляторных сетей является большое количество генов, вовлеченных в изучение микрочипов. Стратегия по уменьшению количества этих генов заключается в фильтрации «шумных», незначимых и излишних генов. Экспрессия «шумных» генов зависит от помех в измерении, которые происходят от вариации экспериментов. Незначимые гены это те, которые не вовлечены в заболевание, для ясности чей уровень экспрессии проявляется одинаково как в больных образцах, так и в контрольных. Излишние гены в значимой степени коррелируются с другими генами и по факту регулируются ими (такие гены рассматриваются как биологически значимые, ответственные за возникновение генетического нарушения и используются для создания моделей генных регуляторных сетей). В данной работе предполагается, что вектор Шепли, а также индекс Банзафа, могут быть использованы для определения биологически значимых генов. Обоснуем данное утверждение для вектора Шепли, используя аксиоматический подход, такой что решение микрочиповой игры характеризуется с использованием базовых свойств. Данные свойства определяются тем, как индекс значимости генов должен вести себя в различных простых ситуациях взаимодействия генов.

Для начала необходимо ввести значение генной регуляторной сети в контексте микрочиповых игр используя терминологию теории игр. Определение партнерства генов в данном случае играет ключевую роль.

Пусть $(N, v) \in \mathcal{M}^N$. Коалиция $S \in 2^N \backslash \{\emptyset\}$ такая что для каждой $T \subsetneq S$ и каждой $R \subseteq N \backslash S$

$$v(R \cup T) = v(R)$$

называется партнерством генов в микрочиповой игре (N, v).

Существует по меньшей мере две причины, поддерживающие возможность представить партнерство генов в качестве генной регуляторной сети в контексте микрочиповых игр. Первая, определение партнерства не требует никакой априорной информации о соответствующих регуляторных механизмах среди генов внутри сети. В следствии высокой сложности сигнальных путей, этот тип информации все еще не доступен для многих генов. Вторая причина в том, что определение партнерства требует того, чтобы было невозможно выявить определенную подгруппу генов, которая непосредственно взаимодействует с внешним геном или группой генов в провоцировании генетической болезни. Это необходимое условие для группы генов при составлении генной регуляторной сети, рассматриваемой как уникальную сеть генов с определенным уровнем экспрессии. С другой стороны, возможно, что совместная стоимость коалиции, созданной двумя непересекающимися партнерствами, будет больше, чем просто сумма их единичных значений, с учетом возможности взаимодействия отдельных сигнальных путей внутри клетки.

2.2 Индекс значимости генов как решение кооперативной игры

Индексом значимости будем называть решение F: $\mathcal{M}^N \to R^N$ в классе микрочиповых игр с набором генов N в качестве множества игроков. Ниже приведены некоторые свойства индекса значимости, связанные с концепцией партнерства генов.

Свойство 1. Пусть $(N, v) \in \mathcal{M}^N$. Решение F удовлетворяет свойству партнерской рациональности, если

$$\sum_{i \in S} F_i(v) \geq v(S)$$

для каждой коалиции $S \in 2^N \backslash \{\emptyset\}$, такой что она является партнерством генов в игре (N, v).

Это свойство определяет нижний предел силы партнерства, то есть общая значимость партнерства генов в определении патогенеза болезни по отдельности не может быть меньше чем среднее количество случаев болезни вызванных партнерством.

Свойство 2. Пусть $(N, v) \in \mathcal{M}^N$. Решение F удовлетворяет свойству осуществимости партнерства, если

$$\sum_{i \in S} F_i(v) \leq v(N)$$

для каждой коалиции $S \in 2^N \backslash \{\emptyset\}$, такой что она является партнерством генов в игре (N, v).

Противоположно свойству партнерской рациональности, это свойство устанавливает верхнюю границу силы партнерства. Общая значимость партнерства генов в обнаружении патогенеза болезни отдельно не может быть больше чем среднее количество случаев болезни вызванных самой большой из возможных коалиций.

Данные два свойства определяют неотрицательную меру для вычисления значимости генов, провоцирующих заболевания, назначая значение 1 партнерству генов, которое в соответствии с принципом достаточности, ответственно за появление болезни во всех больных образцах.

Критерием для сравнения значимости различных партнерств генов является их значение в микрочиповой игре, но к рассматриваемым партнерским генам могут быть также присвоены некоторые другие значения в соответствии с их ролью во всех возможных коалициях.

Свойство 3. Пусть $(N, v) \in \mathcal{M}^N$. Решение F удовлетворяет свойству монотонности партнерства, если

$$F_i(v) \geq F_j(v)$$

для каждого $i \in S$ и каждого $j \in T$, где коалиции $S, T \in 2^N \backslash \{\emptyset\}$, являются партнерствами генов в игре (N, v) и такие что: $S \cap T = \emptyset, v(S) = v(T), v(S \cup T) = v(N), |S| \leq |T|$.

Рассмотрим два непересекающихся партнерства генов, вызывающих одинаковое среднее количество случаев заболевания на заданном множестве образцов. Если гены вне объединения этих двух партнерств являются незначимыми, то гены в меньшем партнерстве должны иметь больший индекс значимости, чем гены из большей коалиции, где вероятность того что какие-то гены окажутся излишними больше.

Свойство 4. Пусть $v_1, \dots, v_k \in \mathcal{M}^N$, $k > 1$.Решение F удовлетворяет свойству равного разделения, если

$$F\left(\frac{\sum_{i=1}^{k} v_i}{k}\right) = \frac{\sum_{i=1}^{k} F(v_i)}{k}$$

Обратим внимание на то, что $\frac{\sum_{i=1}^{k} v_i}{k}$ также является просто кооперативной игрой. Доказательство этого приведено в статье []. Данное свойство предполагает, что средний индекс значимости генов в двух или более различных микрочиповых играх $v_1, \dots, v_k \in \mathcal{M}^N$ с одинаковым набором генов, возникающих из различных микрочиповых экспериментальных ситуаций проводимых в разных лабораториях, должен быть равен индексу значимости генов в усредненной игре $\frac{\sum_{i=1}^{k} v_i}{k}$.

Определение. Нулевым геном игры (N, v) будем называть ген $i \in N$ такой что $v(S \cup i) = v(S)$ для каждой коалиции $S \subseteq N \backslash \{i\}$.

Свойство 5. Пусть $(N, v) \in \mathcal{M}^N$. Решение F удовлетворяет свойству нулевого гена (НГ), если для каждого нулевого гена $i \in N$

$$F_i(v) = 0$$

Интерпретация данного свойства однозначна: если вклад гена в каждую из коалиций $S \in 2^N$ нулевой, то этот ген имеет нулевое значение.

Теорема. Пусть N конечное множество генов. Вектор Шепли на классе простых микрочиповых игр $\mathcal{M}^N$ является единственн индексом значимости, удовлетворяющим свойствам партнерской рациональности, осуществимости партнерства, монотонности, равного разделения и нулевого гена.

Доказательство данной теоремы приведено в статье [19].

Тем самым, можно утверждать, что кооперативная теория игр, включая вектор Шепли и индекс Банзафа, могут применятся в контексте микрочиповых игр.

Глава 3. Примеры

В качестве примера применения кооперативной теории игр к анализу информации полученной с помощью микрочиповых технологий были рассмотрены данные пациентов с болезнью Паркинсона. Данные о профайлах взяты с национального сайта биотехнологической информации.

Имеем две таблицы экспрессии генов, где по строкам находятся гены, по столбцам образцы: в первой таблице образцы, взятые у больных болезнью Паркинсона, во второй из здорового биологического материала (см. таблицы 1 и 2 приложения excel). Используя наивный дискриминантный метод получаем матрицу анормальной экспрессии (табл.3 приложения). Именно она будет обрабатываться в программе.

Подсчет вектора Шепли и индекса Банзафа произведен на языке C#. Код программы в приложении.

На входе программа получает матрицу анормальной экспрессии и количество рассматриваемых генов. Для начала необходимо получить массив всех возможных коалиций. Затем просчитать все необходимые элементы: личный вклад каждого игрока, следы матрицы, характеристические функции. И далее формулы вектора Шепли и индекса Банзафа. На выходе получаем два численных массива с номерами каждого гена по убыванию экспрессии и два аналогичных массива с аббревиатурами генов.

Пример 1. Поскольку данная программа имеет очень большую сложность, а язык программирования C# не поддерживает достаточное для таких вычислений количество памяти, рассмотрим пример с матрицей из 15 генов: "NDUFB2", "PHB", "RASGRF1", "AFF2", "CACNA1G", "SYN1", "APP", "MBP", "PCNT", "GNAO1", "MAPT", "UCHL1", "SNCA", "APOE", "PPIA".

Матрица анормальной экспрессии в данном случае выглядит так:

1, 1, 0, 1, 1, 1, 0, 0, 1, 1, 0, 0

0, 1, 0, 1, 1, 1, 1, 0, 0, 1, 0, 0

0, 1, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0

0, 1, 0, 0, 0, 0, 1, 0, 0, 1, 1, 0

1, 1, 0, 0, 1, 0, 1, 1, 1, 1, 1, 0

0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0

0, 1, 0, 0, 1, 1, 1, 0, 0, 1, 1, 0

0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 1

0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0

$$0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0$$

$$0, 0, 0, 0, 1, 1, 1, 0, 1, 1, 0, 0$$

$$0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0$$

$$0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 1$$

$$0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0$$

$$0, 1, 0, 1, 1, 1, 1, 0, 1, 1, 0, 0$$

На выходе имеем следующие данные:

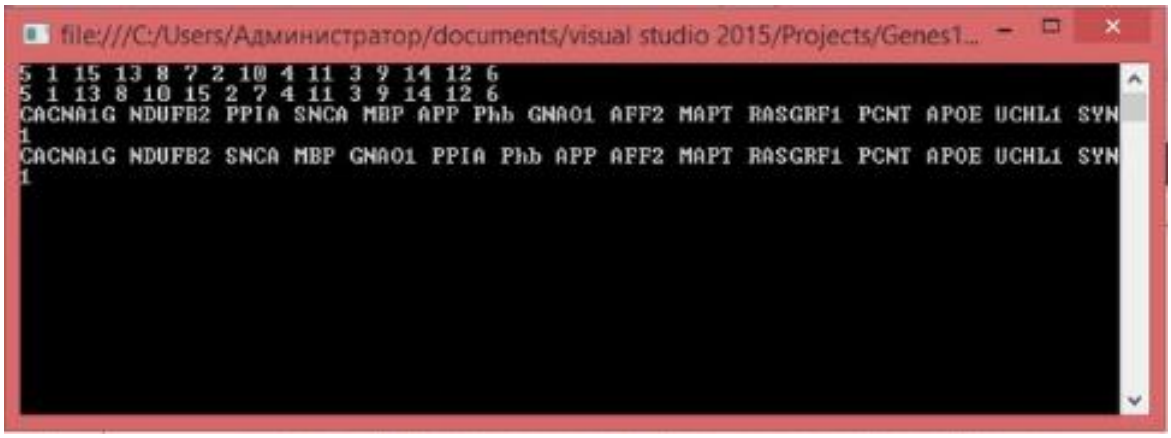

Рисунок 2

Тем самым получаем, что первые пять генов (CACNA1G, NDUFB2, PPIA, SNCA, MBP) обладают наибольшей релевантностью.

Поскольку в изначальную выборку были включены гены, нарушения в которых заведомо известно играют роль в развитии болезни Паркинсона, можно проанализировать полученный результат с помощью данной априорной информации [23].

Например, первый ген в пятерке CACNA1G является ферментом в митохондрии и участвует в процессе синтеза АТФ (аденозинтрифосфорной кислоты), отвечающей за запас энергии. Нарушение в работе этого процесса (в том числе излишняя или недостаточная экспрессия участвующих в этом генов) приводит к выпуску свободных радикалов, которые нарушают структуру ДНК и вызывают онкологические заболевания, и к запуску процесса апоптоза клетки (клеточная гибель).

Второй ген в списке: NDUFB2. Он находится в составе кальциевого канала, обеспечивающего поступление кальция в клетку и поддерживающего кальциевые процессы, такие как передача нейротрансмиттера (т.е. передача нервных импульсов через синапсы). Известно, что при мутациях этого гена у больных проявляется атаксия (нарушение согласованности движений различных мышц при условии отсутствия мышечной слабости; одно из часто

наблюдаемых расстройств моторики) и деградация клеток мозжечка (тремор). Отсюда можно предположить, что значительное изменение экспрессии этого гена также может привести к данным последствиям.

Рассмотренные в этом примере гены лишь малая часть тех, что участвуют в развитии болезни Паркинсона, и нарушения о которых сказано выше лишь малая часть всех симптомов. Но тем не менее можно утверждать, что на данной выборке вектор Шепли и индекс Банзафа достаточно точно определили наиболее значимые гены.

Пример 2.

Проанализируем все гены изначальной базы данных, приведенной в приложении, которые заведомо известно играют роль в патогенезе болезни Паркинсона: "NDUFB2", "PHB", "SLIT1", "TNFRSF21", "PEG10", "RASGRF1", "AFF2", "MT1H", "GRIA1", "CACNA1G".

На выходе получаем:

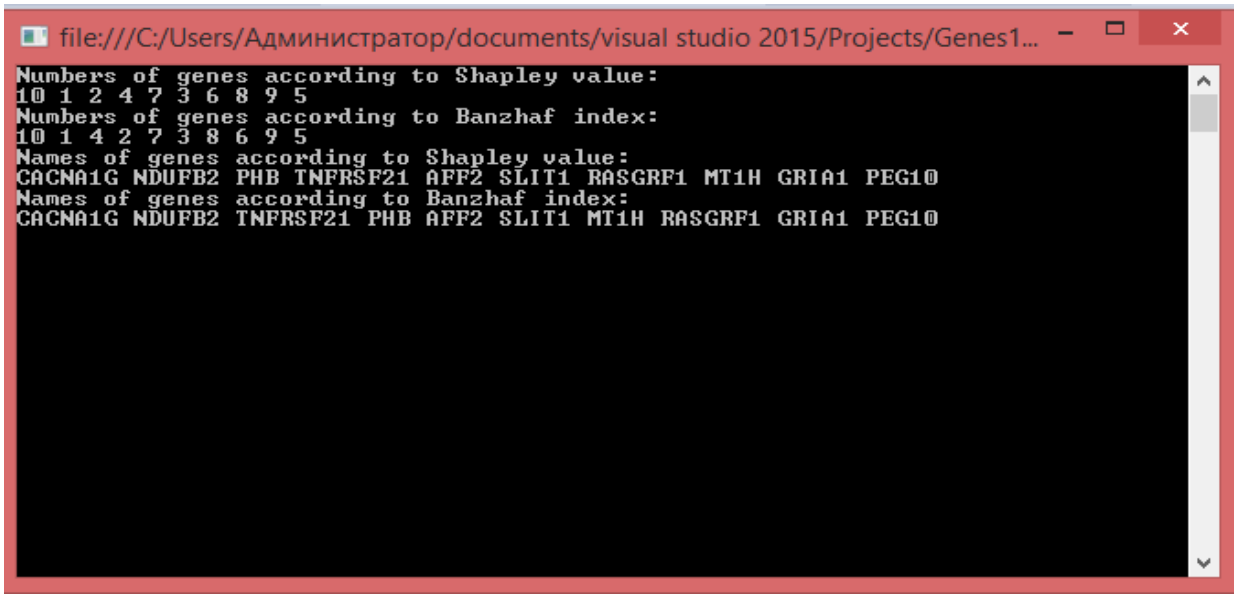

Рисунок 3

Полученная информация не отвечает на вопрос «какие гены сильнее влияют на болезнь Паркинсона», поскольку каждый из них имеет свою роль и вызывает различные симптомы. Но можно сказать, что первые гены в списке имеют наиболее анормальный уровень экспрессии.

Также данный пример иллюстрирует разницу в результатах различных решений теории игр. Как можно заметить, индекс Банзафа в отличии от вектора Шепли «меняет местами» некоторые гены. Но стоит обратить внимание, что постановка задачи определения индекса релевантности генов отличается от классического варианта теории игр: определение выигрышной коалиции. В данном случается игроки не борются за свою выгоду, а ведут

себя независимо в соответствии со своими функциями. Отсюда следует, что мы не можем применить стандартные методы определения наилучшего решения игры. Вопрос определения наилучшего решения в микрочиповой игре может быть темой дальнейших исследований.

Пример 3.

Рассмотрим набор генов и в качестве эксперимента будем изменять сравниваемые образцы. Также посчитаем ранг каждого гена (ratio). Рангом будем называть количество анормально экспрессирующих образцов для каждого гена:

$$w_i = |\{j \in S_D | B_{i,j}^{E,m} = 1\}|.$$

Будем рассматривать 10 генов ("NDUFB2", "PHB","SLIT1","TNFRSF21", "SYN1 ", "SNCA", "APOE", "PPIA", "FOLR1", "KMP3A").

В качестве основного эксперимента рассмотрим все имеющиеся образцы. Результат:

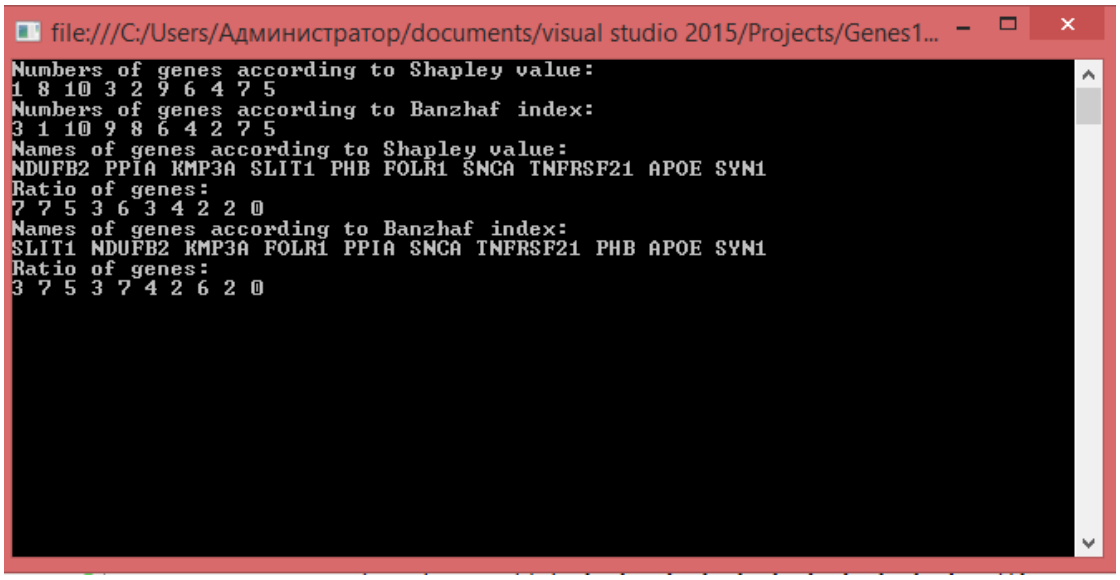

Рисунок 4

Далее уберем 1 и 2 образцы. Тем самым изменится количество столбцов в матрице, но не количество рассматриваемых генов.

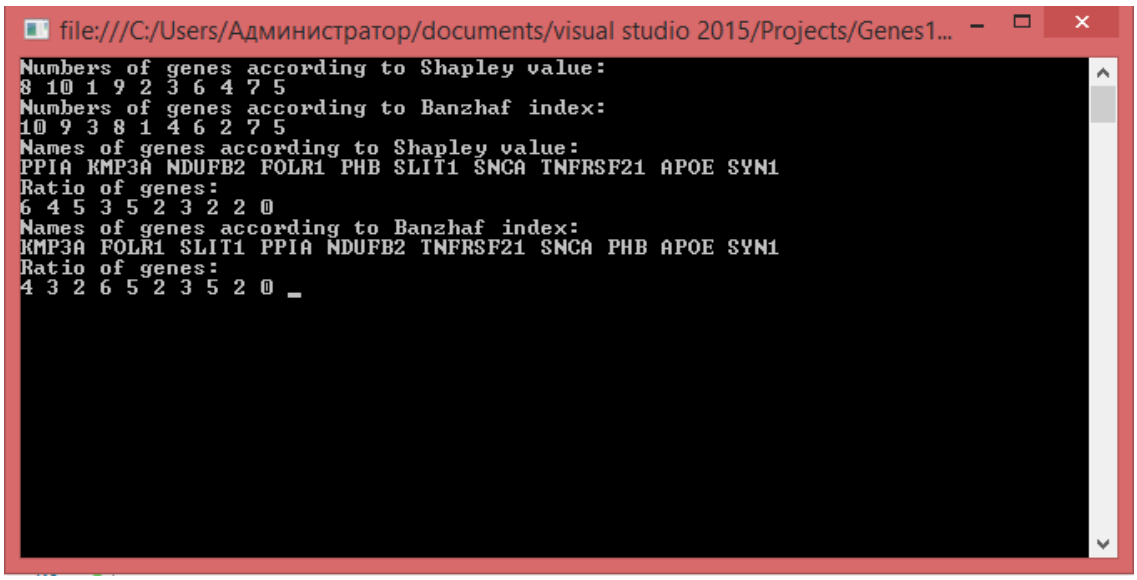

Рисунок 5

Если убрать 6 и 7 образцы, вернуть 1 и 2, получим:

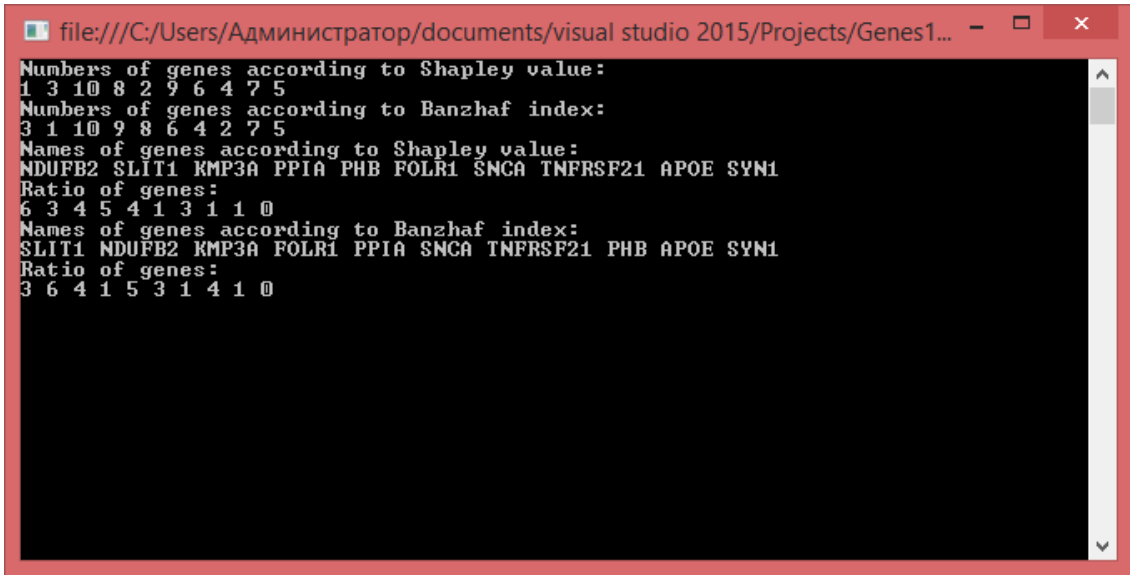

Рисунок 6

Без 5 и 10 результат такой:

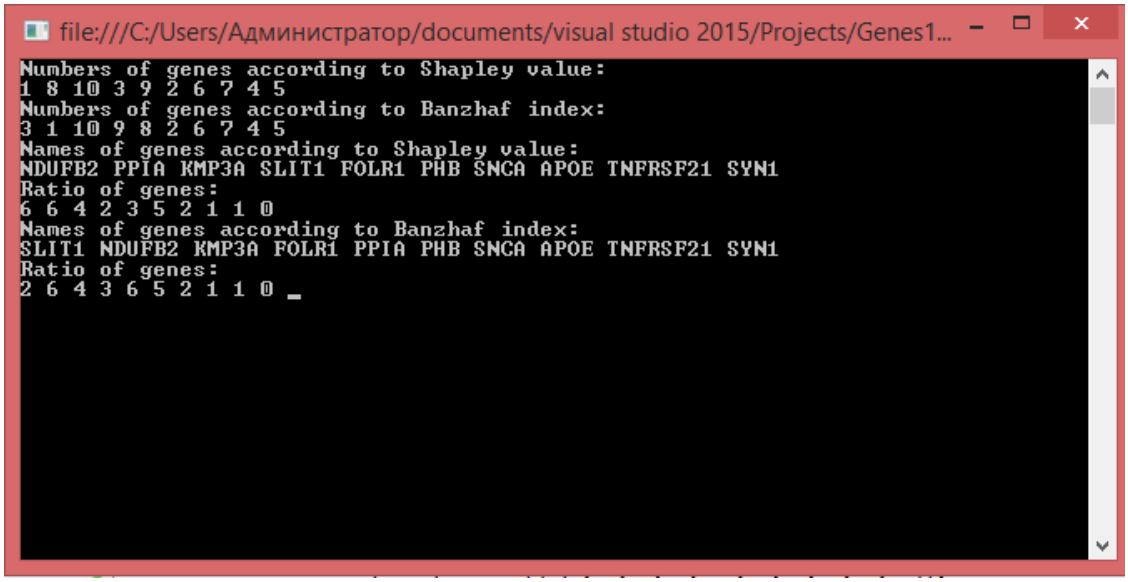

Рисунок 7

Можно видеть, что не всегда существует четкая зависимость ранга гена с его значением релевантности.

Также заметим, что порядок генов в начале и конце списков остается относительно неизменным. На различия в образцах реагируют только средние гены. Данный факт позволяет предположить, что на большей выборке генов данные изменения окажутся незначимыми, позволяя определить гены с наибольшими и наименьшими индексами релевантности.

Выводы

Новизна данного подхода по отношению к классическому заключается в двух аспектах. Первый: используемый класс кооперативных игр, называемый микрочиповыми играми, предоставляет эффективный способ описать связь между общей экспрессией каждой коалиции и генетическим заболеванием или другим состоянием, представляющим интерес, и как следствие, включить в дальнейший анализ все возможные взаимосвязи генов, связанные с биологическим состоянием. Например, возможно описать связь между анормально низкой или анормально высокой экспрессией генов в каждой коалиции и генетической болезнью, либо влиянием лекарственных средств на образцы. Учитывая все возможные подмножества генов, что означает увеличение уровня сложности анализа, невозможно сделать обоснованные выводы о вероятностном распределении экспрессии. По факту, характеристическая функция микрочиповой игры полностью опирается на рассматриваемую матрицу анормальной экспрессии. Весьма важным в данной ситуации является определение принципа достаточности, который включает критерий определения того связан ли уровень экспрессии гена в коалиции или нет с биологическим состоянием, представляющим интерес. Вся информация о связях генов хранящаяся в характеристической функции микрочиповой игры может быть использована для качественного определения роли каждого гена в каждой возможной коалиции посредством применения концепции решений кооперативных игр.

Второй аспект новизны подхода основан на идее применения решений кооперативных игр к микрочиповым играм, а также сильной связи между теорией игр и так называемым аксиоматическим подходом, используемым для изучения свойств решений. Как правило, интерпретация результатов, полученных с применением классического подхода, сильно зависят от теоретической модели, используемой для анализа, или из знаний об исследуемых образцах. Аксиоматический подход к игре предполагает возможность изменить эту точку зрения: необходима лишь малая информация об исследуемых образцах, которая является границей для правдоподобной интерпретации результатов.

В данном подходе результатом анализа является распределение вектора решения, примененного к микрочиповой игре. Этот результат интерпретируется с использованием базовых свойств, которые должны быть удовлетворены путем нахождения индекса релевантности каждого гена в взаимосвязи экспрессии коалиций и генетического заболевания.

Данный подход представляет интерес для генетического анализа, который все еще является относительно новой темой для исследований и предлагает новый математический подход для данной области.

Заключение

В данной работе было рассмотрено приложение коалиционной теории игр к анализу экспрессии генов. А именно: определены понятия микрочиповой игры, матрицы анормальной экспрессии, индекса значимости генов, партнерства генов и т.д., представлена аксиоматическая характеристика решений кооперативной теории игр в контексте микрочиповых игр, рассмотрен практический пример на основе изучения влияния генов на развитие болезни Паркинсона.

Также была реализована программа по подсчету вектора Шепли, индекса Банзафа и ранга генов, которая позволяет наглядно показать результаты применения данного отдела теории игр к исследованию силы генов.

Список литературы

1. http://www.sqlapp.ru/chto-takoe-ekspressiya-genov/
2. Parmigiani, G., Garrett, E. S., Irizarry, R. A., and S. L. Zeger, S. L. (eds.) (2003). *The Analysis of Gene Expression Data: Methods and Software.* Springer, New York. 3/2003
3. Dhammika Amaratunga, Javier Cabrera Exploration and Analysis of DNA Microarray and Protein Array Data. 2004
4. Baldi, P. and Hatfield, G. W. (2002). *DNA Microarrays and Gene Expression: From Experiments to Data Analysis and Modeling*. Cambridge University Press, Cambridge. 9/2002
5. Zhang, M. Q. Large-scale gene expression data analysis: a new challenge to computational biologists. //1999. *Genome Research* 9:681–688.
6. Smyth, G. K., Yang, Y.-H., Speed, T. P. (2003). Statistical issues in cDNA microarray data analysis. //*Methods in Molecular Biology* 224, 111-136. [PubMed ID 12710670]
7. Dhammika Amaratunga, Javier Cabrera Exploration and Analysis of DNA Microarray and Protein Array Data. 2004
8. Arfin, S. M., Long, A.D., Ito, E.T., Tolleri, L., Riehle, M. M., Paegle, E. S., and Hatfield, G. W. Global gene expression profiling in Escherichia coli K12: the effects of integration host factor. //2000. Journal of Biological Chemistry 275:29672–29684.
9. Baldi, P. and Hatfield, G. W. (2002). *DNA Microarrays and Gene Expression: From Experiments to Data Analysis and Modeling*. Cambridge University Press, Cambridge. 9/2002
10. Eisen, M. B., Spellman, P. T., Brown, P. O., and Botstein, D. Cluster analysis and display of genome-wide expression patterns. //1998. *Proceedings of the National Academy of Sciences of the USA* 95:14863–14868.
11. Moser, R. J., Reverter, A., Kerr, C. A. and Beh, K. J. (2004). A mixed-model approach for the analysis of cDNA microarray gene expression data from extreme-performing pigs after infection with Actinobacillus pleuropneumoniae. //Journal of Animal Science, 82(5), 1261-1271
12. База данных Национального центра биотехнологической информации https://www.ncbi.nlm.nih.gov/sites/GDSbrowser?acc=GDS4154
13. A Brazma, P Hingamp, J Quackenbush, G Sherlock, P Spellman, C Stoeckert, J Aach, W Ansorge, C A Ball, H C Causton, T Gaasterland, P Glenisson, F C P Holstege, I F Kim, V Markowitz, J C Matese, H Parkinson, A Robinson, U Sarkans, S Schulze-Kremer, J Stewart, R Taylor, J Vilo and M Vingron Minimum information about a microarray experiment (MIAME)—toward standards for microarray data. *//Nature Genetics* 29, 365-371. 12/2001

14. Moretti, S., Vasilakos, Athanasios V. *An overview of recent applications of Game Theory to bioinformatics.* //Information Sciences 180 (2010) 4312–4322
15. Moretti S (2006) Minimum cost spanning tree games and gene expression data analysis. //In: ACM international conference proceeding series, p 199
16. Branzei D, Seki M, Enomoto T, Rad18/Rad5/Mms2-mediated polyubiquitination of PCNA is implicated in replication completion during replication stress. //*Genes Cells* 9(11):1031-42
17. Fragnelli V, Moretti S (2008) A game theoretical approach to the classification problem in gene expression data analysis. //Comput Math Appl 55(5):950–959
18. Kaufman A, Kupiec M, Ruppin E (2004) Multi-knockout genetic network analysis: the rad6 example. //*Proc IEEE Comput Syst Bioinform Conf* :332-40
19. Moretti S, Patrone F, Bonassi S. The class of microarray games and the relevance index for genes. //Top 2007; 15:256–80.
20. Lucchetti R, Moretti S, Patrone F, Radrizzani P. The Shapley and Banzhaf value in microarray games. //Computers & Operations Research 37 (2010) 1406 – 1412
21. П.К. Головатенко-Абрамов, Е.С. Платонов, Генные регуляторные сети, контролирующие морфогенез волосяного фолликула у мыши. //Успехи современной биологии, 2009, том 129, № 2, с. 144-157
22. J.M. Bower, H. Bolouri, Computational Modeling of Genetic and Biochemical Networks. Massachusetts University of Technology, 2001
23. База данных о белках http://www.uniprot.org/

Приложение

1.Таблицы Excel (открыть: контекстное меню – объект лист – открыть):

ген\обр	119	121	123	125	127	129	120
NDUFB2	2967	648	1488	1300	339	993	1705
PHB	260	505	238	82	354	347	20
SLIT1	701	197	282	318	666	391	433
TNFRSF21	46	332	213	120	111	349	15
PEG10	851	553	963	677	245	246	462
RASGRF1	71	358	175	154	48	71	203
AFF2	189	476	155	81	133	129	420
MT1H	3275	2029	1638	2214	2050	1429	3073
GRIA1	335	131	103	88	119	238	28

2.Код программы:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Genes1
{
    class Program
    {
        public static List<int> eq(List<int> a)
        {
            List<int> temp = new List<int>();
            for (int i = 0; i < a.Count; i++)
                temp.Add(a[i]);
            return temp;
        }
        public static int fact(int a)
        {
            if (a <= 1)
                return 1;
            else
                return a * fact(a - 1);
        }
        static void Main(string[] args)
        {

            //инициализация начальной матрицы
```

```
        List<int[]> microArray = new List<int[]>();

        microArray.Add(new int[] { 1, 1, 0, 1, 1, 1, 0, 0, 1, 1, 0, 0 });//NDUFB2
        microArray.Add(new int[] { 0, 1, 0, 1, 1, 1, 1, 0, 0, 1, 0, 0 });//PHB
        microArray.Add(new int[] { 1, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0 });//SLIT1
        microArray.Add(new int[] { 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1 });//TNFRSF21
        microArray.Add(new int[] { 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 });//PEG10
        microArray.Add(new int[] { 0, 1, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0 });//RASGRF1
        microArray.Add(new int[] { 0, 1, 0, 0, 0, 0, 1, 0, 0, 1, 1, 0 });//AFF2
        microArray.Add(new int[] { 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0 });//MT1H
        microArray.Add(new int[] { 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 });//GRIA1
        microArray.Add(new int[] { 1, 1, 0, 0, 1, 0, 1, 1, 1, 1, 1, 0 });//CACNA1G

        microArray.Add(new int[] { 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 });//SYN1
        microArray.Add(new int[] { 0, 1, 0, 0, 1, 1, 1, 0, 0, 1, 1, 0 }); //APP
        microArray.Add(new int[] { 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 1 });//MBP
        microArray.Add(new int[] { 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0 });//PCNT
        microArray.Add(new int[] { 0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0 });//GNAO1
        microArray.Add(new int[] { 0, 0, 0, 0, 1, 1, 1, 0, 1, 1, 0, 0 }); //MAPT
        microArray.Add(new int[] { 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0 });//UCHL1
        microArray.Add(new int[] { 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 1 }); //SNCA
        microArray.Add(new int[] { 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0 }); //APOE
        microArray.Add(new int[] { 0, 1, 0, 1, 1, 1, 1, 0, 1, 1, 0, 0 });//PPIA
        microArray.Add(new int[] { 1, 0, 1, 0, 0, 1, 1, 0, 0, 0, 0, 0 }); //FOLR1
        microArray.Add(new int[] { 0, 1, 1, 1, 1, 1, 1, 0, 1, 1, 0, 0 }); //TUBA1B
        microArray.Add(new int[] { 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0 });//TITIN
        microArray.Add(new int[] { 0, 1, 1, 1, 1, 1, 0, 0, 0, 1, 0, 0 }); //SNORA3A
        microArray.Add(new int[] { 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0 });//GFAP
        microArray.Add(new int[] { 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0 }); //FTL
        microArray.Add(new int[] { 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0 }); //CAMK2B
        microArray.Add(new int[] { 0, 1, 0, 0, 1, 1, 0, 0, 0, 1, 1, 0 }); //KPM3A
        microArray.Add(new int[] { 0, 1, 0, 1, 1, 1, 0, 0, 1, 1, 0, 0 }); //HBB
        microArray.Add(new int[] { 0, 1, 0, 0, 1, 0, 1, 0, 0, 1, 1, 0 }); //F8

        string[] namesOfGenes = {"NDUFB2",
"PHB","SLIT1","TNFRSF21","PEG10","RASGRF1","AFF2","MT1H","GRIA1",
"CACNA1G","SYN1","APP","MBP","PCNT","GNAO1","MAPT","UCHL1","SN
CA","APOE","PPIA","FOLR1","TUBA1B","TITIN","SNORA3A","GFAP","FTL
","CAMK2B","KPM3A","HBB","F8" };//Все гены из БД
        //string[] namesOfGenes = { "NDUFB2", "Phb", "RASGRF1", "AFF2",
"CACNA1G", "SYN1", "APP", "MBP", "PCNT", "GNAO1", "MAPT", "UCHL1",
"SNCA", "APOE", "PPIA" };//Выборка из 15
        //string[] namesOfGenes = { "NDUFB2", "PHB", "SLIT1", "TNFRSF21",
"PEG10", "RASGRF1", "AFF2", "MT1H", "GRIA1", "CACNA1G" };//Все
заведомо сильные
```

```
        //Для анализа 10 генов с разными образцами
        //microArray.Add(new int[] {1, 1, 0, 1, 1, 1, 0, 0, 1, 1, 0, 0});//NDUFB2
        //microArray.Add(new int[] {0, 1, 0, 1, 1, 1, 1, 0, 0, 1, 0, 0});//PHB
        //microArray.Add(new int[] {1, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0});//SLIT1
        //microArray.Add(new int[] {0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1});//TNFRSF21
        //microArray.Add(new int[] {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0});//SYN1
        //microArray.Add(new int[] {0, 1, 0, 0, 1, 1, 0, 0, 0, 0, 0, 1}); //SNCA
        //microArray.Add(new int[] {0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0}); //APOE
        //microArray.Add(new int[] {0, 1, 0, 1, 1, 1, 1, 0, 1, 1, 0, 0});//PPIA
        //microArray.Add(new int[] {0, 0, 1, 0, 0, 1, 1, 0, 0, 0, 0, 0}); //FOLR1
        //microArray.Add(new int[] {0, 1, 0, 0, 1, 1, 0, 0, 0, 1, 1, 0}); //KPM3A

        //string[] namesOfGenes = { "NDUFB2", "PHB", "SLIT1", "TNFRSF21",
"SYN1", "SNCA", "APOE", "PPIA", "FOLR1", "KMP3A"};//10 для анализа по
различным образцам

        //число образцов
        int numOfSamples = 12;

        //число генов
        int numOfGenes = microArray.Count;

        //Создание листа всех возможных коалиций
        List<int> first = new List<int>();
        int n = numOfGenes;
        for (int i = 1; i <= n; i++)
            first.Add(i);
        List<List<int>> all = new List<List<int>>();
        bool stop = false;
        List<List<int>> ToCont = new List<List<int>>();
        for (int i = 0; i < n; i++)
        {
            List<int> tmep = new List<int>();
            tmep.Add(i + 1);
            ToCont.Add(tmep);
        }
        while (!stop)
        {
            if (ToCont.Count != 0)
            {
                List<List<int>> newToCont = new List<List<int>>();
                while (ToCont.Count != 0)
                {
                    var temp = ToCont[0];
```

```
            ToCont.RemoveAt(0);
            all.Add(temp);
            int j = temp[temp.Count - 1] + 1;
            while (j <= n)
            {
                var temp1 = eq(temp);
                temp1.Add(j);
                j++;
                newToCont.Add(temp1);
            }
        }
        ToCont = newToCont;
    }
    else
    {
        stop = true;
    }

}

//Лист следов
List<List<int>> supportOfArray = new List<List<int>>();
for (int i = 0; i < numOfSamples; i++) //здесь i столбец
{
    List<int> sp = new List<int>();
    for (int j = 0; j < microArray.Count; j++) //здесь j строка
    {
        if (microArray[j][i] == 1)
        {
            sp.Add(j + 1);
        }
    }
    supportOfArray.Add(sp);
}

//Создания листа ТетаТ
List<int> numberoflists = new List<int>();
for (int i = 0; i < all.Count; i++)
    numberoflists.Add(0);
for (int i = 0; i < all.Count; i++)
{
    for (int j = 0; j < supportOfArray.Count; j++)
    {
        bool allnumbersthere = true;
        for (int k = 0; k < supportOfArray[j].Count; k++)
```

```
            allnumbersthere &= all[i].Contains(supportOfArray[j][k]);
        if (allnumbersthere)
            numberoflists[i]++;
    }
}

//Создание листа характерестических функций
List<double> charFunc = new List<double>();

for (int i = 0; i < numberoflists.Count; i++)
{
    charFunc.Add(numberoflists[i] * 1.0 / numOfGenes);
}

//подсчет личного вклада
List<List<double>> mPower = new List<List<double>>();
for (int i = 0; i < all.Count; i++)
{
    List<double> temp = new List<double>();
    for (int j = 0; j < numOfGenes; j++)
        temp.Add(0);
    List<int> temp2 = new List<int>();
    foreach (int c in all[i])
    {
        for (int j = 0; j < all[i].Count; j++)
            if (all[i][j] != c)
                temp2.Add(all[i][j]);

        for (int j = 0; j < all.Count; j++)
        {
            if (all[j].Count == temp2.Count)
            {
                bool eq = true;
                for (int k = 0; k < all[j].Count; k++)
                {
                    bool bltmp = false;
                    for (int l = 0; l < temp2.Count; l++)
                    {
                        if (all[j][k] == temp2[l])
                        {
                            bltmp = true;
                            break;
                        }

                    }
```

```
                    eq = eq & bltmp;
                }
                if (eq)
                    temp[c - 1] = charFunc[i] - charFunc[j];
            }
        }
        temp2.Clear();
    }
    mPower.Add(temp);

}
for (int i = 0; i < n; i++)
    mPower[i][i] = 0;

//Вектор Шепли и Банзаф
List<double> Fi = new List<double>();
List<double> Bi = new List<double>();
for (int i = 0; i < numOfGenes; i++)
{
    Fi.Add(0);
    Bi.Add(0);
}
for (int i = 0; i < all.Count; i++)
    for (int j = 0; j < numOfGenes; j++)
    {
        Fi[j] += (fact(all[i].Count - 1) * fact(numOfGenes - all[i].Count)) *
mPower[i][j] / (fact(numOfGenes));
        Bi[j] += mPower[i][j] / Math.Pow(2, numOfGenes - 1);
    }

//Вывод генов(конечный результат)
List<int> nums1 = new List<int>();
List<int> nums2 = new List<int>();
for (int i = 0; i < numOfGenes; i++)
{
    nums1.Add(i);
    nums2.Add(i);
}
for (int i = 0; i < numOfGenes; i++)
{
    for (int j = i; j < numOfGenes; j++)
    {
        if (Fi[j] > Fi[i])
        {
            var temp1 = Fi[i];
```

```
            var temp2 = nums1[i];
            Fi[i] = Fi[j];
            Fi[j] = temp1;
            nums1[i] = nums1[j];
            nums1[j] = temp2;

        }
        if (Bi[j] > Bi[i])
        {
            var temp1 = Bi[i];
            var temp2 = nums2[i];
            Bi[i] = Bi[j];
            Bi[j] = temp1;
            nums2[i] = nums2[j];
            nums2[j] = temp2;

        }
    }
}

//Создание массива рангов
int r = 0;
double[] ratio = new double[numOfGenes];

for (int i = 0; i < numOfGenes; i++)
{
    for (int j = 0; j < numOfSamples; j++)
    {
        if (microArray[i][j] == 1)
            r++;
    }
    ratio[i] = r;
    r = 0;
}

Console.WriteLine("Numbers of genes according to Shapley value:");
for (int i = 0; i < numOfGenes; i++)
    Console.Write(nums1[i] + 1 + " ");
Console.WriteLine();
Console.WriteLine("Numbers of genes according to Banzhaf index:");
for (int i = 0; i < numOfGenes; i++)
    Console.Write(nums2[i] + 1 + " ");
Console.WriteLine();
Console.WriteLine("Names of genes according to Shapley value:");
for (int i = 0; i < numOfGenes; i++)
```

```
   Console.Write(namesOfGenes[nums1[i]] + " ");
Console.WriteLine();
Console.WriteLine("Ratio of genes:");
for (int i = 0; i < numOfGenes; i++)
   Console.Write(ratio[nums1[i]] + " ");
Console.WriteLine();
Console.WriteLine("Names of genes according to Banzhaf index:");
for (int i = 0; i < numOfGenes; i++)
   Console.Write(namesOfGenes[nums2[i]] + " ");
Console.WriteLine();
Console.WriteLine("Ratio of genes:");
for (int i = 0; i < numOfGenes; i++)
   Console.Write(ratio[nums2[i]] + " ");

//Вывод всех коалиций
//for (int i = 0; i < all.Count; i++)
//{
//    for (int j = 0; j < all[i].Count; j++)
//        Console.Write(all[i][j] + " ");
//    Console.WriteLine("");
//}

//Вывод изначальной матрицы
//for (int i = 0; i < microArray.Count; i++)
//{
//    for (int j = 0; j < microArray.Count; j++)
//        Console.Write(microArray[i][j] + " ");
//    Console.WriteLine("");
//}

//вывод листа следов
//for (int i = 0; i < supportOfArray.Count; i++)
//{
//    for (int j = 0; j < supportOfArray[i].Count; j++)
//        Console.Write(supportOfArray[i][j] + " ");
//    Console.WriteLine("");
//}

//Вывод ТетаТ
//for (int i = 0; i < numberoflists.Count; i++)
//{
//    Console.Write(numberoflists[i] + " ");
//    Console.WriteLine("");
//}
```

```
            //Вывод характеристических функций
            //for (int i = 0; i < charFunc.Count; i++)
            //{
            //    Console.Write(charFunc[i] + " ");
            //    Console.WriteLine("");
            //}

            Console.ReadKey();
        }
    }

}
```