

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Локтионова Оксана Геннадьевна
Должность: проректор по учебной работе
Дата подписания: 04.10.2023 10:19:37
Уникальный программный ключ:
0b817ca911e6668abb13a5d426d39e5f1c11eabbf73e943df4a4851fda56d089

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное
образовательное учреждение высшего образования
«Юго-Западный государственный университет»
(ЮЗГУ)

Кафедра биомедицинской инженерии



Системы поддержки принятия врачебных решений
Методические указания выполнения практических
работ
для студентов специальности
30.05.03 «Медицинская кибернетика»

Курск 2023

СОДЕРЖАНИЕ

Введение	4
ПРАКТИЧЕСКАЯ РАБОТА №1. СИНТЕЗ ДИАГНОСТИЧЕСКИХ РЕШАЮЩИХ ПРАВИЛ.....	5
ПРАКТИЧЕСКАЯ РАБОТА № 2. КОРРЕЛЯЦИОННЫЙ И АВТОКОРРЕЛЯЦИОННЫЙ АНАЛИЗЫ В БИМЕДИЦИНСКОЙ ПРАКТИКЕ.....	8
ПРАКТИЧЕСКАЯ РАБОТА №3. ПРОГНОЗИРОВАНИЕ РАЗВИТИЯ ЗАБОЛЕВАЕМОСТИ В РЕГИОНЕ.....	12
ПРАКТИЧЕСКАЯ РАБОТА № 4. ЛИНЕЙНЫЙ ДИСКРИМИНАНТНЫЙ АНАЛИЗ	14
ПРАКТИЧЕСКАЯ РАБОТА № 5. ПОСТРОЕНИЕ РЕШАЮЩИХ ПРАВИЛ	21
ПРАКТИЧЕСКАЯ РАБОТА №6. ИССЛЕДОВАНИЕ ПОВЕДЕНИЯ ТИПОВЫХ ФУНКЦИЙ ПРИНАДЛЕЖНОСТИ.....	28
ПРАКТИЧЕСКАЯ РАБОТА №7. СРАВНИТЕЛЬНЫЙ АНАЛИЗ СТРУКТУР ДНК.....	36
ПРАКТИЧЕСКАЯ РАБОТА №8. ПОСТРОЕНИЕ И АНАЛИЗ КОРРЕЛЯЦИОННЫХ ОТНОШЕНИЙ МЕТОДАМИ ИНТЕЛЛЕКТУАЛЬНЫХ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ.....	54
ПРАКТИЧЕСКАЯ РАБОТА №9. ИСПОЛЬЗОВАНИЕ НЕЙРОННОЙ СЕТИ В ДИАГНОСТИЧЕСКОМ ПРОЦЕССЕ.....	59
ПРАКТИЧЕСКАЯ РАБОТА № 10. РАСЧЕТ КРИТЕРИЕВ КАЧЕСТВА ДИАГНОСТИЧЕСКОГО ПРОЦЕССА.....	81
ПРАКТИЧЕСКАЯ РАБОТА № 11. ЭКСПЕРТНАЯ СИСТЕМА ДИФФЕРЕНЦИАЛЬНОЙ ДИАГНОСТИКИ ЗАБОЛЕВАНИЙ.....	85
ПРАКТИЧЕСКАЯ РАБОТА №12. СРАВНИТЕЛЬНЫЙ АНАЛИЗ ИНФОРМАЦИОННЫХ ПОРТАЛОВ, ПОСВЯЩЕННЫХ СИСТЕМАМ АВТОМАТИЧЕСКОГО УПРАВЛЕНИЯ В МЕДИЦИНЕ	115
ПРАКТИЧЕСКАЯ РАБОТА № 13. ИЗУЧЕНИЕ СТРУКТУРЫ МНОГОМЕРНЫХ МЕДИЦИНСКИХ ДАННЫХ	122

Введение

Структура методических указаний по выполнению лабораторного практикума: цель и задачи работы, информационные материалы (включая краткие теоретические сведения), методические указания по выполнению работы, библиография.

Порядок выполнения.

1. Изучить информационные материалы к занятию, включая рекомендованную литературу и лекции.
2. Выполнить работу согласно методическим указаниям.
3. Осуществить обработку полученных результатов.
4. Оформить отчет к работе.
5. Подготовиться к отчету – собеседованию с преподавателем по тематике работы (защите полученных результатов) или обсуждению результатов в студенческой группе.

Содержание отчета.

1. Цель работы. Постановка задач лабораторных исследований.
2. Этапы проведения, полученные результаты, анализ и выводы.
3. Ответы на контрольные вопросы (не менее 3), аннотации содержания информационных источников (100-200 слов по каждому).

ПРАКТИЧЕСКАЯ РАБОТА №1. СИНТЕЗ ДИАГНОСТИЧЕСКИХ РЕШАЮЩИХ ПРАВИЛ

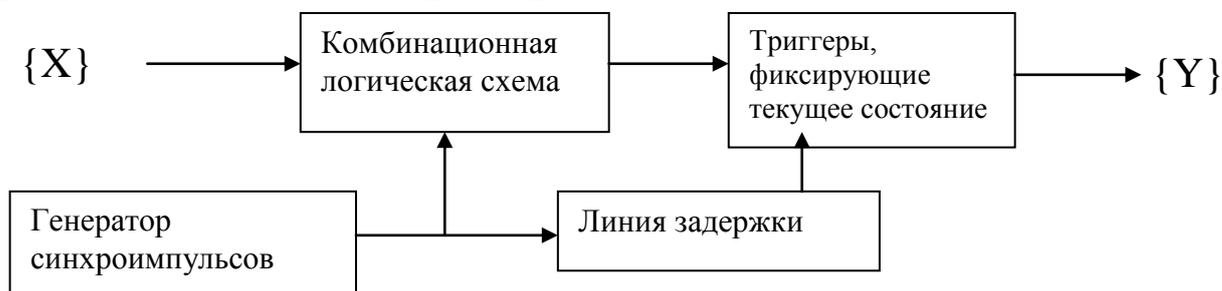
Цель работы: овладение навыками структурно-параметрической идентификации диагностических правил с применением информационных технологий средств вычислительной техники на основе биометрической информации.

Порядок выполнения работы.

1. Самостоятельно изучите теоретический материал.
2. Согласно номеру варианта (равен порядковому номеру студента в журнале группы) N сформируйте протокол мониторинга наблюдения за процессом X , следующим образом:
 Индекс переменной X | Индекс протокола Z (см. таблицу 1)

1		$\text{mod}(n,8)+1$
2		$\text{mod}(n,8)+2$
3		$\text{mod}(n,8)+3$
4		$\text{mod}(n,8)+4$
3. Выберите в качестве выходной величины, определяющей состояние процесса переменную мониторинга с индексом 4.
4. Зададитесь числом состоянием по Y - 3, X_1 -5, X_2 - 3, X_3 -2.
5. Определите диапазон изменений состояний, причем для Y_1, X_1 , - автоматически по дисперсии (здесь и далее рекомендуется использовать интегрированные среды типа EXCEL, STATISTICA), X_2 - экспертным путем анализа гистограммы, X_3 - экспериментальным путем заданием одного порога ($<, >=$).
6. Закодируйте состояние по X, Y - т.е. получите характеристические числа (вид) булевых функций X_6 и Y_6 .
7. Проанализируйте взаимозависимость между булевыми факторами X и сформируйте наиболее ортогональное факторное пространство (векторы которого в наименьшей степени зависимы между собой).
8. Идентифицируйте в минимальном виде (с помощью карт Карно) логические функции $Y_6=F(X_6)$.
9. Перейдите от булевского представления к логико-семантическому и сформулируйте диагностические правила продукционного типа.

10*. Составьте схемотехническое решение диагностических правил по следующей структуре:



Рассчитайте скважность синхроимпульсов и их характер для обеспечения устойчивой работы схем, считая время срабатывания любого логического элемента 10мс, время срабатывания триггера 50мс.

11. Оформите отчет с указанием последовательности своих действий, необходимых комментариев и выводов, кратких ответов не менее чем на 4 вопроса, аннотацию информационных источников, указанных в библиографии.

Примечание. П.10 – является заданием повышенной сложности и выполняется по желанию студента.

Контрольные вопросы:

1. Что определяет решающее правило?
2. Какие типы решающих правил применяют в диагностическом процессе при обработке результатов мониторинга?
3. В чем заключается логический способ синтеза решающего правила?
4. Каким образом осуществляется бинарное кодирование признакового пространства при синтезе логических решающих правил?
5. Как формулируется решающее правило продукционного типа?
6. Как осуществляется семантическое описание решающего правила?
7. Каким образом реализуется схемотехническая реализация решающего правила на определенной электронной базе?
8. Как проверяется качество применения решающего правила?

Таблица 1. Результаты регионального мониторинга

год	Всего родилось	Всего заболело	Врожденные пороки (ВП)	асфикс ия	Умерл о всего	Умерл о от ВП	Умерло от асфикси и
1	1657	90	22	6	17	4	5
2	2081	170	24	9	32	3	9
3	2173	201	20	25	32	7	5
4	2676	198	41	17	34	5	5
5	2557	191	51	47	21	3	10
6	2522	586	83	78	23	1	3
7	2893	252	30	19	31	3	7
8	2956	270	45	8	32	6	5
9	2650	197	38	12	25	3	6
10	3036	213	42	36	36	4	12
11	3165	230	37	32	21	2	5
12	3181	218	61	42	27	8	3
13	2930	216	65	58	18	1	6
14	2491	202	41	55	20	2	5
15	2964	185	37	39	25	2	1
16	2425	290	65	87	22	2	7
17	2432	238	50	53	19	3	4
18	2388	196	34	65	19	0	9
19	2290	197	34	58	22	4	9
20	2995	193	45	27	28	5	7
Год	Ревм. пораж. сердца	Инфаркт миокарда	Гипертан. болезнь	стенокардия	Септический эндокартит	летальность	
1	107	8	171	30	5	18	
2	147	4	151	38	9	20	
3	146	22	124	42	4	17	
4	122	27	145	56	10	20	
5	104	37	134	83	8	23	
6	77	37	110	33	11	23	
7	82	24	156	38	8	20	
8	104	37	100	40	4	13	
9	88	21	87	45	35	18	
10	75	17	111	41	6	2	
11	44	28	71	29	3	17	
12	56	4	120	38	8	12	
13	44	16	100	31	7	7	
14	44	12	104	28	5	24	
15	57	15	97	35	15	10	
16	32	17	84	33	14	5	
17	36	15	116	40	20	16	
18	44	20	176	44	2	28	
19	51	23	157	37	11	13	
20	48	18	162	43	17	17	

ПРАКТИЧЕСКАЯ РАБОТА № 2. КОРРЕЛЯЦИОННЫЙ И АВТОКОРРЕЛЯЦИОННЫЙ АНАЛИЗЫ В БИОМЕДИЦИНСКОЙ ПРАКТИКЕ

Цель работы: овладение навыками использования инструментария ПЭВМ для проведения исследований в области изучения связей между регистрируемыми характеристиками состояния биообъекта методами корреляционного и регрессионного анализов.

Порядок выполнения.

1. Изучить теоретический материал.
2. Построить корреляционные зависимости, определить коэффициенты корреляции (с оценкой значимости – ошибки первого рода), наиболее адекватные уравнения регрессии между характеристиками – показателями крови в каждом из альтернативных классов (см. приложение к лабораторной работе 1). Рассмотреть не менее 4 пар (формат пар определяет преподаватель).
3. Построить автокорреляционные функции (порядка 10-15 первых значений) для ФПГ до и после нагрузки (см. Приложение).
4. Оформите отчет, включающий в себя результаты выполнения (скрин-шоты), выводы, ответы на контрольные вопросы (не менее пяти, из них №№12,13,14 – обязательны).

Контрольные вопросы.

1. Чем отличается функциональная и корреляционная связь между признаками?
2. Что такое временной ряд биофизиологического сигнала?
3. Что такое ранжирование выборки?
4. В каком случае регрессия будет линейной?
5. В каком случае линии регрессии совпадают?
6. Что можно сказать о корреляционной связи между признаками X и Y если значение коэффициента корреляции равно 0,3?
7. Что можно сказать о корреляционной связи между признаками, если корреляционное поле имеет форму круга?
8. Какой метод применяется для нахождения коэффициентов уравнения линейной регрессии?

9. Что такое автокорреляционная функция? Как она определяется?
10. Каким образом оценивается значимость коэффициента корреляции?
11. Как строится коррелограмма?
12. Каким образом в электронной таблице осуществляется корреляционный анализ?
13. Как применяются результаты корреляционного анализа в медицине?
14. Как применяется автокорреляция в медицине?

Библиография

1. Автокорреляция. Презентация 2. /URL: <http://present5.com/prezentaciya-kafedra-avtomatizacii-obrabotki-informacii2/>
2. Автокорреляция уровней временного ряда. /URL: <http://be5.biz/ekonomika/e011u/80.htm>
3. Корреляция. Презентация. /URL: <http://informslide.ru/korrelyacionnye-zavisimosti/>
4. Корреляционные зависимости. Презентация. /URL: <http://www.myshared.ru/slide/84314/>
5. Статистическая обработка данных. Презентация. /URL: <http://informslide.ru/statisticheskaya-obrabotka-dannyx/>

Приложение
ФПГ (до и после нагрузки)

№ п.п.	До нагрузки	После нагрузки
1	0,395	0,00
2	0,37	0,03
3	0,32	0,04
4	0,30	0,17
5	0,30	0,46
6	0,34	0,84
7	0,43	1,25
8	0,53	1,62
9	0,71	1,93
10	0,96	2,23
11	1,21	2,45
12	1,47	2,65
13	1,75	2,80
14	2,02	2,92
15	2,23	2,98
16	2,49	3,02
17	2,68	3,02
18	2,83	2,99
19	2,98	2,96
20	3,07	2,89
21	3,14	2,80
22	3,19	2,72
23	3,20	2,57
24	3,23	2,45
25	3,22	2,31
26	3,19	2,15
27	3,16	2,02
28	3,10	1,92
29	3,06	1,78
30	2,99	1,69
31	2,89	1,62
32	2,82	1,58
33	2,75	1,54
34	2,67	1,55
35	2,59	1,53
36	2,51	1,55
37	2,44	1,56
38	2,39	1,58
39	2,31	1,54
40	2,25	1,52

41	2,24	1,52
42	2,19	1,50
43	2,15	1,44
44	2,13	1,42
45	2,08	1,35
46	2,04	1,28
47	2,01	1,22
48	1,98	1,15
49	1,92	1,05
50	1,87	0,97
51	1,82	0,88
52	1,74	0,77
53	1,69	0,68
54	1,63	0,58
55	1,54	0,47
56	1,46	0,39
57	1,39	0,28
58	1,30	0,17
59	1,21	0,07
60	1,16	0,02

ПРАКТИЧЕСКАЯ РАБОТА №3. ПРОГНОЗИРОВАНИЕ РАЗВИТИЯ ЗАБОЛЕВАЕМОСТИ В РЕГИОНЕ

Цель работы: овладение навыками методами идентификации и анализа прогностических моделей уровня заболеваемости в регионе на основе данных многолетних наблюдений.

Порядок выполнения работы.

1. Самостоятельно изучите теоретический материал.
2. Из приведенного в приложении временного ряда сформируйте последовательность, начиная с номера A_1 в количестве A_2 измерений. A_1 определяется как остаток от деления порядкового номера в группе на семь плюс 1. A_2 определяется как утроенное значение количества букв в Вашей фамилии.
3. С помощью инструментария Excel идентифицируйте трех аппроксимантов полученных в п.2 временных трендов: лучшая по критерию детерминированности модель в режиме «построить линию тренда»; авторегрессионную модель первого порядка; гармоническую модель по двум-трем частотам (частоты определите путем подбора соответствующих периодов циклов, полученных по анализу первой производной временного тренда согласно численному дифференцированию).
4. Постройте графики временного тренда и полученных моделей на одной плоскости.
- 4* (повышенной сложности). Постройте модель спектра Фурье.
5. Оцените средние значения относительных ошибок аппроксимации: интерполяции, экстраполяции (до и после интерполяционного интервала). Сделайте выводы.
6. Оформите отчет, включающий в себя результаты работы (возможны скриншоты), выводы, краткие ответы на контрольные вопросы, аннотацию одного из информационных источников, указанных в библиографии или иных (найденного самостоятельно).

Контрольные вопросы:

1. Что называется интерполяцией?
2. Что называется экстраполяцией?
3. Каким образом строятся гармонические модели?

4. Охарактеризуйте виды прогнозов (сиюминутный, краткосрочный, среднесрочный, долгосрочный)?
5. Как осуществляется проверка качества прогностической модели?
6. Могут ли прогностические модели быть логическими?
7. Как осуществляется прогноз во времени и пространстве?
8. Для чего необходимо прогнозировать заболеваемость в регионе?
9. Какие заболевания населения носят ритмический характер?
10. Какие природные циклы оказывают влияние на региональную заболеваемость (и почему)?

Библиография

1. Демографические прогнозирование. Презентация.
<http://900igr.net/prezentatsii/geografija/Demograficheskoe-prognozirovanie/008-Vidy-i-metody-prognozirovanija.html>
2. Методы прогнозирования. Презентация.
<http://nashaucheba.ru/v24021>

Приложение

Динамика психических заболеваний в регионе

t	0	1	2	3	4	5	6	7	8	9	10
Ур-нь	19,4	24,09	27,95	32,61	36,89	40,33	44,8	48,7	50,88	52,78	54,56
11	12	13	14	15	16	17	18	19	20		
56,42	58,15	58,94	60,03	60,39	58,68	57,43	56,83	55,37	53,70		
21	22	23	24	25	26	27	28	29	30		
51,49	48,67	45,33	40,75	36,89	34,00	29,59	24,72	20,06	14,38		

ПРАКТИЧЕСКАЯ РАБОТА № 4. ЛИНЕЙНЫЙ ДИСКРИМИНАНТНЫЙ АНАЛИЗ

Цель работы: изучение возможностей дискриминантного анализа при решении классификационных задач.

Краткие теоретические сведения

В приборостроении часто возникают задачи соотнесения неизвестного электронного устройства к определенной группе, классификации исследуемого объекта по результатам применения информационно-аналитической техники, соотнесения состояния прибора к определенной стадии функционирования (например, его дальнейшей пригодности). В простейшем случае, эти задачи могут быть решены с помощью дискриминантного анализа [2].

Дискриминантный анализ используется для принятия решения о том, какие переменные различают (дискриминируют) две или более возникающие совокупности (группы, классы). Медик может регистрировать различные переменные, относящиеся к состоянию больного, чтобы выяснить, какие переменные лучше предсказывают, что пациент, вероятно, выздоровел полностью (группа 1), частично (группа 2) или совсем не выздоровел (группа 3). Специалист в области приборостроения может, анализируя характеристики работоспособности, соотнести состояние прибора к определенному этапу его жизненного цикла.

С вычислительной точки зрения дискриминантный сходен с дисперсионным анализом. Цель дискриминантного анализа заключается в определении, отличаются ли совокупности по среднему какой-либо переменной (или линейной комбинации переменных), и затем использовать эту переменную, чтобы осуществлять прогноз для новых объектов исследования их принадлежность к определенной

Таким образом, задача о дискриминантной функции может быть перефразирована как задача однофакторного дисперсионного анализа (ANOVA). В случае одной переменной окончательный критерий значимости того, разделяет переменная две совокупности или нет, дает F -критерий.

При применении дискриминантного анализа имеются несколько переменных, и задача состоит в том, чтобы установить, какие из

переменных вносят свой вклад в дискриминацию между совокупностями. Наиболее общим применением дискриминантного анализа является включение в исследование многих переменных с целью определения тех из них, которые наилучшим образом разделяют совокупности между собой. В пошаговом анализе дискриминантных функций модель дискриминации на каждом шаге просматриваются все переменные и находится та из них, которая вносит наибольший вклад в различие между совокупностями. Эта переменная включается в модель на данном шаге, и происходит переход к следующему шагу. В процессе идентификации дискриминантной функции можно двигаться и в «обратном направлении». В этом случае, все переменные включаются в модель, а затем на каждом шаге устраняются переменные, вносящие малый вклад в предсказания.

Интерпретация функций дискриминации для двух классов осуществляется следующим образом. Для двух классов дискриминантный анализ может рассматриваться как процедура множественной регрессии (и аналогичная ей). Интерпретация результатов задачи с двумя совокупностями тесно следует логике применения множественной регрессии: переменные с наибольшими регрессионными коэффициентами вносят наибольший вклад в дискриминацию. При построении дискриминантных функций для более чем 2 классов используются подход постепенного бинарного деления групп. (Заметим, что это деление носит субъективный характер, то возможно возникновение ситуации синтеза различных функций, в зависимости от последовательности бинарного деления на классы с близкими значениями критерия качества классификации.)

Полученные значения структурных коэффициентов применяются и при интерпретации «реального смысла» дискриминирующей функции.

Значимость дискриминантной функции оценивается путем проверки числа корней, которое значимо добавляется к дискриминации между совокупностями. Для интерпретации могут быть использованы только те из них, которые будут признаны статистически значимыми. Остальные функции (корни) должны быть проигнорированы.

Таким образом, при интерпретации дискриминантной функции для нескольких совокупностей и нескольких переменных, вначале хотят проверить значимость различных функций и в дальнейшем использовать только значимые функции. Затем, для каждой значащей функции рассматриваются стандартизованные коэффициенты *бета*. Чем больше стандартизованный коэффициент *бета*, тем большим является относительный собственный вклад переменной в дискриминацию, выполняемую соответствующей дискриминантной функцией.

Дискриминантный анализ позволяет получить качественный результат при выполнении следующих требований:

1. Анализируемые переменные представляют выборку из многомерного нормального распределения. Для проверки этого требования используют специальные критерии и графики.

2. Матрицы дисперсий/ковариаций переменных однородны.

3. Применяемые критерии оценки значимости дискриминантной функции достаточно корректны. Большинство «реальных угроз» корректности применения критериев значимости возникает из-за возможной зависимости между средними по совокупностям и дисперсиями между собой. Интуитивно ясно, что если имеется большая изменчивость в совокупности с высокими средними в нескольких переменных, то эти высокие средние ненадежны. Однако критерии значимости основываются на объединенных дисперсиях, то есть, на средней дисперсии по всем совокупностям. Поэтому критерии значимости для относительно больших средних (с большими дисперсиями) основываются на относительно меньших объединенных дисперсиях и будут ошибочно указывать на статистическую значимость. Практически это происходит, если одна из изучаемых совокупностей содержит несколько экстремальных выбросов, которые сильно влияют на средние и, таким образом, увеличивают изменчивость. Для определения такого случая следует изучить описательные статистики, то есть средние и стандартные отклонения или дисперсии для таких корреляций.

Переменные, используемые для дискриминации между совокупностями, не являются полностью избыточными. При вычислении результатов дискриминантного анализа происходит

обращение матрицы дисперсий/ковариаций для переменных в модели. Если одна из переменных полностью избыточна по отношению к другим переменным, то такая матрица называется *плохо обусловленной* и не может быть обращена. Например, если переменная является суммой трех других переменных, то это отразится также и в модели, и рассматриваемая матрица будет плохо обусловленной.

Чтобы избежать плохой обусловленности матриц, необходимо постоянно проверять так называемые *значения толерантности* для каждой переменной. Значение толерантности вычисляется как $(1-R^2)$, где R^2 - квадрат коэффициент множественной корреляции для соответствующей переменной со всеми другими переменными в текущей модели (известный так же под названием – коэффициент детерминации).

Основной целью применения дискриминантного анализа является проведение классификации анализируемого объекта. Как только модель установлена и получены дискриминирующие функции, возникает вопрос о том, как хорошо они могут *предсказывать*, к какой совокупности принадлежит конкретный объект исследования? Между тем: функции классификации не следует путать с дискриминирующими функциями. Функции классификации предназначены для определения того, к какой группе наиболее вероятно может быть отнесен каждый объект. Имеется столько же функций классификации, сколько групп. Каждая функция позволяет вам для каждого образца и для каждой совокупности вычислить веса классификации по формуле:

$$S_i = c_i + w_{i,1} \cdot x_1 + w_{i,2} \cdot x_2 + \dots + w_{i,m} \cdot x_m$$

где индекс i обозначает соответствующую совокупность, а индексы 1, 2, ..., m обозначают m переменных; c_i являются константами для i -ой совокупности, $w_{i,j}$ - веса для j -ой переменной при вычислении показателя классификации для i -ой совокупности; x_j - наблюдаемое значение для соответствующего образца j -ой переменной. Величина S_i является результатом.

В общем случае наблюдение считается принадлежащим тому классу (группе) совокупности, для которой получен наивысший показатель классификации (кроме случая, когда вероятности априорной классификации становятся слишком

малыми ниже). Для определения апостериорной вероятности правильной классификации используются понятие расстояния Махаланобиса.

Расстояние Махаланобиса. Вы можете прочитать об этих расстояниях в других разделах. В общем, расстояние Махаланобиса является мерой расстояния между двумя точками в пространстве, определяемым двумя или более *коррелированными* переменными. Например, если имеются всего две некоррелированных переменные, то вы можете нанести точки (образцы) на стандартную [2М диаграмму рассеяния](#). Расстояние Махаланобиса между точками будет в этом случае равно расстоянию Евклида, т.е. расстоянию, измеренному, например, рулеткой. Если имеются три некоррелированных переменные, то для определения расстояния вы можете по-прежнему использовать рулетку (на 3М диаграмме). При наличии более трех переменных вы не можете более представить расстояние на диаграмме. Также и в случае, когда переменные коррелированы, то оси на графике могут рассматриваться как *неортогональные* (они уже не направлены под прямыми углами друг к другу). В этом случае простое определение расстояния Евклида не подходит, в то время как расстояние Махаланобиса является адекватно определенным в случае наличия корреляций.

Расстояние Махаланобиса и классификация. Для каждой совокупности в выборке вы можете определить положение точки, представляющей средние для всех переменных в многомерном пространстве, определенном переменными рассматриваемой модели. Эти точки называются *центроидами* группы. Для каждого наблюдения вы можете затем вычислить его расстояние Махаланобиса от каждого центроида группы. Снова, вы признаете наблюдение принадлежащим к той группе, к которой он ближе, т.е. когда расстояние Махаланобиса до нее минимально.

Апостериорные вероятности классификации. Используя для классификации расстояние Махаланобиса, вы можете теперь получить вероятность того, что образец принадлежит к конкретной совокупности. Это значение будет не вполне точным, так как распределение вокруг среднего для каждой совокупности будет не в точности нормальным. Так как принадлежность каждого образца вычисляется по априорному знанию модельных переменных, эти

вероятности называются *апостериорными* вероятностями. Короче, *апостериорные* вероятности - это вероятности, вычисленные с использованием знания значений других переменных для образцов из частной совокупности. Некоторые пакеты автоматически вычисляют эти вероятности для всех наблюдений (или для выбранных наблюдений при проведении [кросс-проверки](#)).

Априорные вероятности классификации. Имеется одно дополнительное обстоятельство, которое следует рассмотреть при классификации образцов. Иногда вы знаете заранее, что в одной из групп имеется больше наблюдений, чем в другой. Поэтому *априорные* вероятности того, что образец принадлежит такой группе, выше. Например, если вы знаете заранее, что 60% выпускников вашей средней школы обычно идут в колледж, (20% идут в профессиональные школы и остальные 20% идут работать), то вы можете уточнить предсказание таким образом: при всех других равных условиях более вероятно, что учащийся поступит в колледж, чем сделает два других выбора. Вы можете установить различные *априорные* вероятности, которые будут затем использоваться для уточнения результатов классификации наблюдений (и для вычисления *апостериорных* вероятностей).

На практике, исследователю необходимо задать себе вопрос, является ли неодинаковое число наблюдений в различных совокупностях в первоначальной выборке отражением истинного распределения в популяции, или это только (случайный) результат процедуры выбора. В первом случае вы должны положить *априорные* вероятности пропорциональными объемам совокупностей в выборке; во втором - положить *априорные* вероятности одинаковыми для каждой совокупности. Спецификация различных *априорных* вероятностей может сильно влиять на точность классификации.

Итог классификации. Общим результатом, на который следует обратить внимание при оценке качества текущей функции классификации, является *матрица классификации*. Матрица классификации содержит число образцов, корректно классифицированных (на диагонали матрицы) и тех, которые попали не в свои совокупности (группы).

Другие предостережения. При повторной итерации *апостериорная* классификация того, что случилось в прошлом, не очень трудна. Нетрудно получить очень хорошую классификацию тех образцов, по которым была оценена функция классификации. Для получения сведений, насколько хорошо работает процедура классификации на самом деле, следует классифицировать (*априорно*) *различные* наблюдения, то есть, наблюдения, которые не использовались при оценке функции классификации. Вы можете гибко использовать условия отбора для включения или исключения из вычисления наблюдений, поэтому матрица классификации может быть вычислена по "старым" образцам столь же успешно, как и по "новым". Только классификация новых наблюдений позволяет определить качество функции классификации (см. также [кросс-проверку](#)); классификация старых наблюдений позволяет лишь провести успешную диагностику наличия выбросов или области, где функция классификации кажется менее адекватной.

Итог. В общем, *Дискриминантный анализ* - это очень полезный инструмент (1) - для поиска переменных, позволяющих относить наблюдаемые объекты в одну или несколько реально наблюдаемых групп, (2) - для классификации наблюдений в различные группы.

Библиография:

1. Воронцов И.М., Шаповалов В.В., Шерстюк Ю.М. Здоровье. Опыт разработки и обоснование применения автоматизированных систем для мониторинга и скринирующей диагностики нарушений здоровья. – СПб.: ООО «ИПК «Коста»Б, 2006. – 432 с.
2. <http://www.statsoft.ru/home/textbook/modules/stdiscan.html>

ПРАКТИЧЕСКАЯ РАБОТА № 5. ПОСТРОЕНИЕ РЕШАЮЩИХ ПРАВИЛ

Цель работы: Овладение навыками структурно-параметрической построения решающих правил с применением средств вычислительной техники.

Краткие теоретические сведения

Под решающими правилами понимается процедура вывода заключения о соотношении состояния анализируемого объекта или процесса к определенному классу или области на основании временно-пространственной регистрации существенных характеристик.

Любой объект (процесс) с точки зрения классификации подвергается анализу со стороны исследователя, который, как правило, априори знает, какие существенные характеристики ему следует регистрировать для решения задачи. То есть, в этом случае, исследователь уже владеет набором решающих правил, которые либо опровергают, либо подтверждают выдвинутую им рабочую гипотезу о состоянии объекта. Так как о каждом состоянии объекта может выдвигаться различное количество гипотез, то, следовательно, решающие правила каждой из них не должны в случае объединения поглощать друг друга, и, вообще говоря, должны иметь минимальное количество пересечений как по регистрируемым параметрам, так и по диапазонам их изменений.

В общем случае решающее правило имеет вид, например, продукции:

Если значение $P=P_0$, то состояние $S=S_0$
 где $P=F(S, t, dS)$, (1)
 где S - состояние; t - время; dS - диагноз изменения характеристик состояния.

Если зависимость (1) достаточно хорошо идентифицирована (с заданной степенью точности или неопределенности), то нетрудно построить экспертную систему продукционного типа с указанием исследователю технологии реализации необходимой информации для достаточно достоверной классификации.

Рассмотрим логический механизм синтеза (1).

1 этап. Организация мониторинга состояния заданной глубины и полноты.

2 этап. Выделение множества ортогональных и информативных признаков с точки зрения вариативности. То есть, с одной стороны, селектируем сильно коррелированные характеристики, с другой стороны, отбираем те из них, вариативность которых (отношение дисперсии к среднему значению) выше определенного порогового уровня (например 10%).

3 этап. Кодирование состояний (лучше в двоичном коде): с учителем - то есть исследователь знает состояния, без учителя - выполняется кластер-анализ и задаются состояния или вводится пороговый принцип. Таким образом, получаем значения «логической» функции $Y=(Y_{i1}, Y_{i2}, Y_{i1})$. Если состояний не много, то рекомендуется применять унитарное кодирование с минимизацией Хеменгового расстояния соседних состояний.

4 этап. Кодирование значения признакового пространства, следующим образом (во всех случаях рекомендуется унитарный код). По каждому оставленному признаку выделяем определенный набор состояний, как попадание значения признака в определенный диапазон. Диапазон определяется либо:

1) Экспертом, исходя из его знаний и жизненного опыта.

2) Исследователем, по анализу частоты распределений значений и личного опыта. При достаточно небольшом количестве признаков анализ гистограммы рекомендуется проводить визуально, наблюдая все признаки одновременно (в концепции системный подход).

3) Автоматически (с применением ЭВМ) по следующему алгоритму.

Исследователь задает количество состояний по каждому признаку n_i (каждое из них кодируется, желательно в унитарном коде). Определяется медиана M_0 и дисперсия G_0 . Определяется удельное отклонение как $G_y = G_0/(n_{i-1})$. В качестве первого диапазона (состояния) выбирается величина внутри диапазона $M_0 \pm G_0$. Все значения X_i попавшие в данный диапазон кодируются определенным состоянием S_0 . Величина n_i декрементируется и повторяется описанный процесс над «оставшимися» данными. Так

продолжается до тех пор, пока n_i не станет равно 0 и всем оставшимся значениям будет присвоено состояние S_n . Граничные значения $M_0 \pm G_{y_0}$ либо включаются в одно из состояний, либо, что более оптимально, кодируются знаком переходной функции.

5 этап. Определяем функциональные зависимости между полученными булевыми функциями (парные и множественные) и парное Хемингово расстояние. Те признаки, у которых это расстояние равно нулю, селектируются путем оставления одного из них с наибольшей вариативностью.

Явный вид логической зависимости между булевыми переменными X_k , $k=1, m$ определяются следующим образом. На первом шаге проверяются условия независимости: поскольку каждая булева функция может иметь два значения истинности, то m булевых функций может образовывать 2^m комбинаций значений истинности. Согласно определению m -булевых функций независимы, если в совокупности при всех возможных значениях аргументов они могут принимать 2^m комбинаций значений истинности. Т.е., для проверки независимости необходимо вычислить их изображающие числа и проверить, образуют ли они полный набор чисел. Если да, то функции независимы, в противном случае - зависимы.

На втором шаге в базисе булевых функций выписывают в последовательные строки изображающие числа и определяют какие числа отсутствуют в наборе столбцов (повторяющиеся значения чисел считают один раз). Столбцы набора представляют собой комбинации значений истинности функций X_1, \dots, X_m , при которых соответствующие элементарные произведения составленные из X_1, \dots, X_m истинны.

Таким образом, если идентифицируется зависимость:

$$F(X_1, \dots, X_n) = 1 \quad (3),$$

то, следовательно, имеющиеся в наборе столбцы указывают номера тех колонок базиса в (X_1, \dots, X_n) , которые совпадают с номерами изображающего числа $\#F(X_1, \dots, X_m)$, на которых функция F истинна.

Например, пусть задан протокол мониторинга трех логических функций:

$$X_1 \quad 11001010$$

$$\begin{array}{l} X_2 \ 10101100 \\ X_3 \ 11001100 \end{array}$$

Выпишем последовательно все столбцы в этом наборе изображающих чисел как строки и укажем справа их десятичные значения:

$$111=7, 101=5, 010=2, 000=0, 111=7, 110=6, 001=1, 000=0$$

Видно, что десятичные эквиваленты 3 и 4 отсутствуют, а это означает, что по отношению и в (X_1, X_2, X_3) изображающее число связи $F(X_1, X_2, X_3) = 1$ имеет вид $\#F(X_1, X_2, X_3) = 1$.

Минимизируя полученную функцию, получаем:

$$\#F = \bar{X}_1 \bar{X}_3 + X_2 X_3 + X_1 \bar{X}_2 = 1$$

Проверяем:

X_1	X_2	\bar{X}_3	$\bar{X}_1 X_3$	$X_2 X_3$	$\bar{X}_1 X_2$	F
1	1	1	0	1	0	1
1	0	1	0	0	1	1
0	1	0	1	0	0	1
0	0	0	1	0	0	1
1	1	1	0	1	0	1
0	1	1	0	1	0	1
1	0	0	0	0	1	1
0	0	0	1	0	0	1

Таким образом, мы нашли, как логические функции связаны между собой.

6 этап. Идентифицируем логические функции $Y=F(X)$ - парная зависимость и/или $Y=F(\{X\})$ (4) - множественная зависимость. Заметим, что возможен вариант отсутствия тех или иных функциональных зависимостей.

7 этап. Переходим от полученных булевских функций либо к продукционным диагностическим правилам, либо к схмотехническому решению идентификационного диагностического устройства. Однако, второй вариант менее устойчив и мобилен в случае достаточно быстрого изменения окружающей среды, приводящего к изменению в

функционировании анализируемого объекта (системы, процесса), а, следовательно, и вида идентифицированных функций.

Как и во множественном регрессионном анализе, при синтезе зависимостей (4) для получения более строгого результата (минимизации пересечений понятий в диагностических, классификационных правилах каждого состояния) рекомендуется руководствоваться правилом максимальной организации (независимости) факторного пространства. Для этого необходимо добиться максимальной независимости X между собой, т.е. в идеале не должно существовать функциональных зависимостей между X_i . Т.е., если на пятом этапе идентифицируется $F(x)=1$ (3), то необходимо изменить множество X : либо путем исключения переменных (по критерию вариативности), что чревато в общем случае, потерей информации; либо изменить кодирование вводимых сигналов путем уменьшения количества состояний и/или изменения (экспертным путем) диагностических классов состояний. При достаточно мощной вычислительной технике и сравнительно небольшом размере факторного пространства (до 100 признаков) эти проблемы могут быть решены переборным путем. В противном случае, следует применять методы целенаправленного случайного поиска.

Как и в регрессионном анализе возможно формирование продукционных диагностических правил с учетом фактора запаздывания.

Порядок выполнения работы.

1. Самостоятельно изучите теоретический материал.
2. Согласно номеру варианта (равен порядковому номеру студента в журнале группы) N сформируйте протокол мониторинга наблюдения за процессом X , следующим образом:

Индекс переменной X	Индекс протокола Z (см. таблицу 1)
1	mod(n,8)+1
2	mod(n,8)+2
3	mod(n,8)+3
4	mod(n,8)+4

3. Выберите в качестве выходной величины, определяющей состояние процесса переменную мониторинга с индексом 4.

4. Зададитесь числом состоянием по Y - 3, X_1 -5, X_2 - 3, X_3 -2.
5. Определите диапазон изменений состояний, причем для Y_1, X_1 , -автоматически по дисперсии (здесь и далее рекомендуется использовать интегрированные среды типа EXCEL, STATISTICA), X_2 -- экспертным путем анализа гистограммы, X_3 - экспериментальным путем заданием одного порога ($<, >=$).
6. Закодируйте состояние по X, Y - т.е. получите характеристические числа (вид) булевых функций X_6 и Y_6 .
7. Проанализируйте взаимозависимость между булевыми факторами X и сформируйте наиболее ортогональное факторное пространство (векторы которого в наименьшей степени зависимы между собой).
8. Идентифицируйте в минимальном виде (с помощью карт Карно) логические функции $Y_6 = F(X_6)$.
9. Перейдите от булевого представления к логико-семантическому и сформулируйте диагностические правила продукционного типа.
10. Составьте схмотехническое решение диагностических правил по следующей структуре:

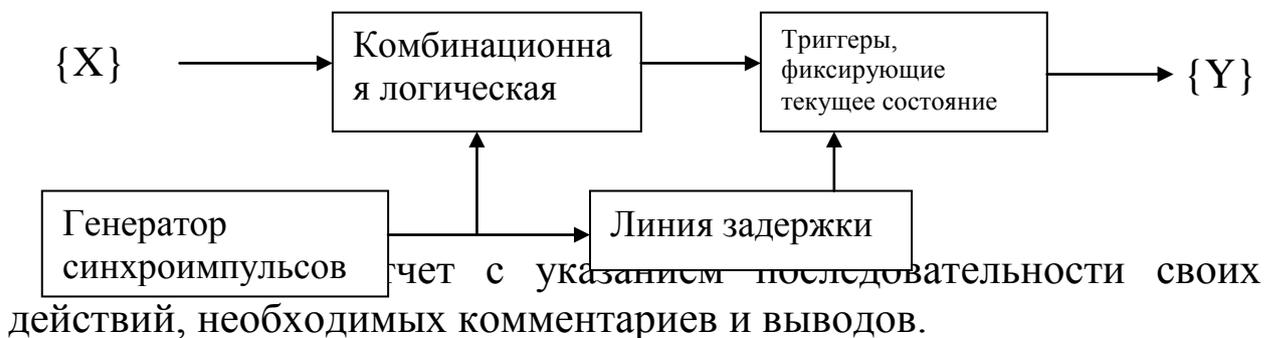


Таблица 1

T	X1	X2	X3	X4	X5	X6	X7
1965	1657	90	22	6	17	4	5
66	2081	170	24	9	32	3	9
67	2173	201	20	25	32	7	5
68	2676	198	41	17	34	5	5
69	2557	191	51	47	21	3	10
70	2522	586	83	78	23	1	3
71	2893	252	30	19	31	3	7
72	2956	270	45	8	32	6	5
73	2650	197	38	12	25	3	6
74	3036	213	42	36	36	4	12
75	3165	230	37	32	21	2	5
76	3181	218	61	42	27	8	3
77	2930	216	65	58	18	1	6

78	2491	202	41	55	20	2	5
79	2964	185	37	39	25	2	1
80	2425	290	65	87	22	2	7
81	2432	238	50	53	19	3	4
82	2388	196	34	65	19	0	9
83	2290	197	34	58	22	4	9
84	2995	193	45	27	28	5	7

T	X8	X9	X10	X11	X12	X13
1965	107	8	171	30	5	18
66	147	4	151	38	9	20
67	146	22	124	42	4	17
68	122	27	145	56	10	20
69	104	37	134	83	8	23
70	77	37	110	33	11	23
71	82	24	156	38	8	20
72	104	37	100	40	4	13
73	88	21	87	45	35	18
74	75	17	111	41	6	2
75	44	28	71	29	3	17
76	56	4	120	38	8	12
77	44	16	100	31	7	7
78	44	12	104	28	5	24
79	57	15	97	35	15	10
80	32	17	84	33	14	5
81	36	15	116	40	20	16
82	44	20	176	44	2	28
83	51	23	157	37	11	13
84	48	18	162	43	17	17

ПРАКТИЧЕСКАЯ РАБОТА №6. ИССЛЕДОВАНИЕ ПОВЕДЕНИЯ ТИПОВЫХ ФУНКЦИЙ ПРИНАДЛЕЖНОСТИ

Цель работы: Исследовать поведение интегральной функции принадлежности с точки зрения темпа ее сходимости к определенной асимптоте в зависимости от методики ее вычисления и ее составляющих.

Краткие теоретические сведения:

Нечеткие множества – это частный случай нечисловых данных.

Пусть A - некоторое множество. Подмножество B множества A характеризуется своей характеристической функцией:

$$\mu_B(x) = \begin{cases} 1, & x \in B, \\ 0, & x \notin B. \end{cases} \quad (1)$$

Что такое нечеткое множество? Обычно говорят, что нечеткое подмножество C множества A характеризуется своей функцией принадлежности $\mu_C: A \rightarrow [0;1]$. Значение функции принадлежности в точке x показывает степень принадлежности этой точки нечеткому множеству. Нечеткое множество описывает неопределенность, соответствующую точке x – она одновременно и входит, и не входит в нечеткое множество C . За вхождение - $\mu_C(x)$ шансов, за второе, т.е. за то, что точка не входит в множество, $(1 - \mu_C(x))$ шансов.

Если функция принадлежности $\mu_C(x)$ имеет вид (1) при некотором B , то C есть обычное (четкое) подмножество A . Таким образом, теория нечетких множеств является более общей или хотя бы не менее общей математической дисциплиной, чем обычная теория множеств, поскольку обычные множества – частный случай нечетких. Соответственно можно ожидать, что теория нечеткости как целое обобщает классическую математику. По степени общности обычная математика и нечеткая математика эквивалентны. Однако для практического применения, например, в теории принятия решений описание и анализ неопределенностей с помощью теории нечетких множеств весьма плодотворны.

Теория нечеткости является обобщением интервальной математики. Действительно, функция принадлежности:

$$\mu_B(x) = \begin{cases} 1, & x \in [a; b], \\ 0, & x \notin [a; b] \end{cases}$$

задает интервальную неопределенность – про рассматриваемую величину известно лишь, что она лежит в заданном интервале $[a, b]$. Тем самым описание неопределенностей с помощью нечетких множеств является более общим, чем с помощью интервалов.

Л.А. Заде рассматривал теорию нечетких множеств как аппарат анализа и моделирования гуманистических систем, т.е. систем, в которых участвует человек. Его подход опирается на предпосылку о том, что элементами мышления человека являются не числа, а элементы некоторых нечетких множеств или классов объектов, для которых переход от «принадлежности» к «непринадлежности» не скачкообразен, а непрерывен. В настоящее время методы теории нечеткости используются почти во всех прикладных областях, в том числе при управлении предприятиями, качеством продукции и технологическими процессами, при описании предпочтений и оптимизации различных процессов, в том числе диагностического характера.

Л.А. Заде использовал термин «fuzzy set» (нечеткое множество). На русский язык термин «fuzzy» переводится как нечеткий, размытый, расплывчатый, и даже как пушистый и туманный.

В качестве примера аппарата теории нечетких множеств рассмотрим определения теоретико-множественных операций над нечеткими множествами. Пусть C и D – два нечетких подмножества A с функциями принадлежности $\mu_C(x)$ и $\mu_D(x)$ соответственно. Пересечением $C \cap D$, произведением CD , объединением $C \cup D$, отрицанием \bar{C} , суммой $C + D$ называются нечеткие подмножества A с функциями принадлежности, соответственно:

$$\mu_{C \cap D}(x) = \min(\mu_C(x), \mu_D(x)), \quad \mu_{CD}(x) = \mu_C(x)\mu_D(x), \quad \mu_{\bar{C}}(x) = 1 - \mu_C(x),$$

$$\mu_{C \cup D}(x) = \max(\mu_C(x), \mu_D(x)), \quad \mu_{C+D}(x) = \mu_C(x) + \mu_D(x) - \mu_C(x)\mu_D(x), \quad x \in A.$$

Теория нечетких множеств в определенном смысле сводится к теории случайных множеств. Однако при решении прикладных

задач вероятностно-статистические методы и методы теории нечеткости

Законы де Моргана для нечетких множеств. Законами же Моргана называются следующие тождества алгебры множеств:

$$\overline{A \cup B} = \bar{A} \cap \bar{B}, \quad \overline{A \cap B} = \bar{A} \cup \bar{B}. \quad (2)$$

$$\overline{A \cup B} = \bar{A} \cap \bar{B}, \quad \overline{A \cap B} = \bar{A} \cup \bar{B}, \quad (3)$$

$$\overline{A + B} = \bar{A} \bar{B}, \quad \overline{A \bar{B}} = \bar{A} + \bar{B}. \quad (4)$$

Тождества (3) и (4) называются *законами де Моргана для нечетких множеств*.

Дистрибутивный закон для нечетких множеств. Некоторые свойства операций над множествами не выполнены для нечетких множеств. Так, $A + A \neq A$, за исключением случая, когда A – «четкое» множество (т.е. функция принадлежности принимает только значения 0 и 1).

Носителем нечеткого множества A называется совокупность всех точек $y \in Y$, для которых $\mu_A(y) > 0$.

Нечетким логическим выводом (fuzzy logic inference) называется

получение заключения в виде нечеткого множества, соответствующего текущим значениям входов, с использованием нечеткой базы знаний и нечетких операций.

В общем случае нечеткий вывод решения происходит за три (или четыре) шага:

1) *этап фаззификации.* С помощью функций принадлежности всех термов входных лингвистических переменных и на основании задаваемых четких значений из универсумов входных лингвистических переменных определяются степени уверенности в том, что выходная лингвистическая переменная принимает конкретное значение.

2) *этап непосредственного нечеткого вывода.* На основании набора правил - нечеткой базы знаний - вычисляются значения истинности для предпосылок всех правил на основании конкретных нечетких операций, соответствующих конъюнкции или дизъюнкции термов в левой части правил. В большинстве случаев это либо максимум, либо минимум из степеней уверенности термов, вычисленных на этапе фаззификации, который

применяется к заключению каждого правила. Используя один из способов построения нечеткой импликации, получается нечеткая переменная, соответствующая вычисленному значению степени уверенности в левой части правила и нечеткому множеству в правой части правила.

Основным способом построения нечеткой импликации является способ Kleene-Dienes: пусть A и B - нечеткие высказывания и μ_A, μ_B — соответствующие им функции принадлежности. Импликация $A \Rightarrow B$ определяется формулами:

$$A \Rightarrow B \equiv \bar{A} \vee B,$$

$$\mu_{A \Rightarrow B}(x, y) = \max(1 - \mu_A(x); \mu_B(y)).$$

Однако, это не единственное обобщение оператора импликации. В таблице приведены другие интерпретации этого понятия, наиболее часто применяемые на практике.

Larsen	$\mu_{A \Rightarrow B}(x, y) = \mu_A(x) \mu_B(y)$
Lukasiewicz	$\mu_{A \Rightarrow B}(x, y) = \min(1; 1 - \mu_A(x) + \mu_B(y))$
Mamdani	$\mu_{A \Rightarrow B}(x, y) = \max(\mu_A(x); \mu_B(y))$
Kleene-Dienes-Lu	$\mu_{A \Rightarrow B}(x, y) = 1 - \mu_A(x) + \mu_A(x) \mu_B(y)$

3) *этап композиции (агрегации, аккумуляции)*. Все нечеткие множества, назначенные для каждого терма каждой выходной лингвистической лингвистической переменной, объединяются вместе, и формируется единственное нечеткое множество - значение для каждой выводимой лингвистической переменной. Обычно используются функции MAX или SUM.

4) *этап дефаззификации (необязательный)*. Используется тогда, когда полезно преобразовать нечеткий набор значений выводимых лингвистических переменных к точным. В теории нечетких множеств процедура дефаззификации аналогична нахождению характеристик положения (математического ожидания, моды, медианы) случайных величин в теории вероятности. Имеется достаточно большое количество методов перехода к точным значениям.

Простейшим способом выполнения процедуры дефаззификации является выбор четкого числа, соответствующего максимуму функции принадлежности. Однако пригодность этого способа распространяется лишь на одноэкстремальные функции принадлежности. Для многоэкстремальных функций принадлежности часто используются следующие методы дефаззификации:

1) COG (Center Of Gravity) – «центр тяжести». Физическим аналогом этой формулы является нахождение центра тяжести плоской фигуры, ограниченной осями координат и графиком функции принадлежности нечеткого множества.

2) MOM (Mean Of Maximums) – «центр максимумов». При использовании метода центра максимумов требуется найти среднее арифметическое элементов универсального множества, имеющих максимальные степени принадлежности.

3) First Maximum – «первый максимум» - максимум функции принадлежности с наименьшей абсциссой.

Конкретные способы синтеза интегральных функций принадлежности определяется из характера решаемой задачи, ее целей и предметной области.

В связи с этим, на этапе разведочного анализа результатов мониторинга или в процессе диагностики целесообразно проводить исследование зависимости результирующей функции принадлежности от значений функции принадлежности аргументов в нее входящих.

Порядок выполнения работы.

1. Изучить теоретические сведения - теорию нечетких множеств, принципы выбора и построения функций принадлежности, методы вычисления интегральной функции принадлежности.

2. На абстрактном носителе x синтезировать функции принадлежности согласно Таблице 1.

3. Согласно Таблице 2 составить алгоритм вычисления интегральной функции принадлежности в случае применения N решающих правил обладающих определенными функциями принадлежностями.

4. Оценить значения скорости и ускорения сходимости интегральной функции к единице в зависимости от количества решающих правил и выбранной функции принадлежности.
5. Проанализировать полученные результаты с точки зрения: сходимости, применения определенных функций принадлежности и методов вычисления интегральной функции, количества итераций в процессе вычисления.
6. Сделать выводы.
7. Оформить отчет, включающий в себя результирующую таблицу вида:
- 8.

	Способ получения интегрального показателя
Функция принадлежности	Скорость, ускорение, график

8. В процессе самостоятельной работы в ходе выполнения лабораторной работы рекомендуется проанализировать показатели вычисления интегральной функции принадлежности в случаях изменения в итерациях как правил вычисления функций принадлежности, так и правил вычисления интегральной функции.

Таблица 1. – Типовые функции принадлежности

Название функции	Вид функции
S	$s(x; a, b, c) = \begin{cases} 0, \text{ для } x \leq a \\ 2 \cdot \left(\frac{x-a}{c-a} \right)^2, \text{ для } a \leq x \leq b \\ 1 - 2 \cdot \left(\frac{x-c}{c-a} \right)^2, \text{ для } b \leq x \leq c \\ 1, \text{ для } x \geq c \end{cases}$
π	$\pi(x; b, c) = \begin{cases} s(x; c-b, c-b/2, c), \text{ для } x \leq c \\ 1 - s(x; c, c+b/2, c+b), \text{ для } x \geq c \end{cases}$
γ	$\gamma(x; a, b) = \begin{cases} 0, \text{ для } x \leq a \\ \frac{x-a}{b-a}, \text{ для } a \leq x \leq b \\ 1, \text{ для } x \geq b \end{cases}$

t	$t(x; a, b, c) = \begin{cases} 0, & \text{для } x \leq a \\ \frac{x-a}{b-a}, & \text{для } a \leq x \leq b \\ \frac{c-x}{c-b}, & \text{для } b \leq x \leq c \\ 0, & \text{для } x \geq c \end{cases}$
L	$L(x; a, b) = \begin{cases} 1, & \text{для } x \leq a \\ \frac{b-x}{b-a}, & \text{для } a \leq x \leq b \\ 0, & \text{для } x \geq b \end{cases}$

Таблица 2 Интегральные функции принадлежности (пример)

№	T(a,b) – «пересечение»	S(a,b) - «объединение»	параметры
1	Min(a,b)	Max(a,b)	
2	a*b	a+b-a*b	
3	MAX(a+b-1,0)	Min(a+b,1)	
4	$\begin{cases} a, & b=1 \\ b, & a=1 \\ 0, & \text{если } a \neq b \end{cases}$	$\begin{cases} a, & b=0 \\ b, & a=0 \\ 0, & \text{если } a, b \neq 0 \end{cases}$	
5	$\frac{a \cdot b}{\gamma + (1-\gamma) \cdot (a+b-a \cdot b)}$	$\frac{a+b-(2-\gamma) \cdot a \cdot b}{1-(1-\gamma) \cdot (a \cdot b)}$	$\gamma > 0$
6	$\frac{a \cdot b}{\max(a, b, \alpha)}$	$\frac{a+b-a \cdot b - \min(a, b, 1-\alpha)}{\max(1-a, 1-b, \alpha)}$	$\alpha \in [0,1]$
7	$1 - \sqrt[p]{(1-a)^p + (1-b)^p - (1-a)^p \cdot (1-b)^p}$	$\sqrt[p]{a^p + b^p - a^p \cdot b^p}$	$p > 0$
8	$[1 + \lambda \sqrt[\lambda]{(\frac{1}{a}-1)^\lambda + (\frac{1}{b}-1)^\lambda}]^{-1}$	$[1 + \lambda \sqrt[\lambda]{(\frac{1}{a}-1)^{-\lambda} + (\frac{1}{b}-1)^{-\lambda}}]^{-1}$	$\lambda > 0$
9	$\max(1 - \sqrt[p]{(1-a)^p + (1-b)^p}, 0)$	$\min(\sqrt[p]{a^p + b^p}, 1)$	$p \geq 1$
10	$\log_w(1 + \frac{(w^a - 1) \cdot (w^b - 1)}{w-1})$	$1 - \log_w(1 + \frac{(w^{1-a} - 1) \cdot (w^{1-b} - 1)}{w-1})$	$w = \min(1, 1-a, 1, 1-b)$
12	$\max(\frac{a+b-1+\lambda \cdot a \cdot b}{1+\lambda}, 0)$	$\min(a+b+\lambda \cdot a \cdot b, 1)$	$\lambda \geq -1$

Контрольные вопросы:

1. Чем отличается нечеткое множество от четкого?
2. Что такое функция принадлежности?
3. Как получить интегральное значение функции принадлежности в случае анализа нескольких нечетких множеств?

4. Что такое носитель функции принадлежности?
5. Когда рекомендуется применять аппарат теории нечетких множеств при построении диагностических правил?
6. В чем заключаются алгоритмы нечеткого вывода Мамдани, Лоренса, Цукамото?
7. Что такое лингвистическая переменная?
8. Каким образом осуществляется коррекция интегральной функции принадлежности в случае неадекватно быстрого приближения ее значения к 1?

ПРАКТИЧЕСКАЯ РАБОТА №7. СРАВНИТЕЛЬНЫЙ АНАЛИЗ СТРУКТУР ДНК

Цель работы: овладение навыками разработки и анализа алгоритмов генетических последовательностей с помощью современных информационных средств и технологий.

Краткие теоретические сведения.

Нуклеотидная последовательность, генетическая последовательность - это порядок следования нуклеотидных остатков в нуклеиновых кислотах. Определяется при помощи секвенирования. Для записи нуклеотидных последовательностей ДНК по рекомендации IUPAC используются символы латинского алфавита: А =аденин; С=цитозин; G =гуанин; Т =тимин. Для записи последовательностей РНК обычно достаточно символов А, С, G, U (уридин).

Все последовательности записываются без пробелов. Сравнительный анализ нуклеотидных последовательностей позволяет судить о степени родства сравниваемых организмов. Это обстоятельство широко применяется на практике (в частности, для установления отцовства).

Изучая семьи с известной генеалогией, генетики оценивают скорость накопления различий в ДНК. В частности, большую помощь оказало исследование ДНК населения Исландии - уникальной страны, где каждый житель знает всех своих предков вплоть то первых колонистов, прибывших в Исландию из Норвегии в IX веке (причем из останков нескольких первопоселенцев тоже удалось извлечь ДНК для анализа). Теми же методами можно реконструировать историю целых народов или, к примеру, находить среди современных азиатов потомков Чингисхана.

Результаты генетического анализа при этом хорошо согласуются с сохранившимися историческими сведениями.

В ходе многочисленных исследований такого рода, где можно было непосредственно сравнить генетические данные с историческими, генетики раз за разом убеждались в достоверности оценок родства на основе сравнения ДНК, а используемые методы развивались и совершенствовались.

Генетическое родство человека и шимпанзе доказывается даже не столько сходством последовательностей, сколько характером различий между ними. Легко заметить, что характер этих различий полностью соответствует предсказаниям эволюционной теории.

Выравнивание аминокислотных или нуклеотидных последовательностей – это процесс сопоставления сравниваемых последовательностей для такого их взаиморасположения, при котором наблюдается максимальное количество совпадений аминокислотных остатков или нуклеотидов. Различают 2 вида выравнивания: парное (выравнивание двух последовательностей ДНК, РНК или белков) и множественное (выравнивание трех и более последовательностей).

Наиболее популярной серией программ для множественного выравнивания последовательностей является Clustal. Первая программа серии Clustal была создана Д.Хиггинсом в 1988 году. Затем она была усовершенствована Д. Фенгом, Р. Дулиттл и В. Тейлором путем добавления прогрессивного выравнивания, то есть созданием множественного выравнивания в результате серий попарных выравниваний, следуя ветвлению направляющего дерева, построенного методом UPGMA.

В 1992 году появилась второе поколение программ Clustal. Программа, названная Clustal V, отличалась способностью проводить сопоставления существующих выравниваний и построением направляющего дерева методом NJ. Третье поколение программ, появившееся в 1994 году и названное Clustal W, стало значительно проще в работе благодаря усовершенствованному алгоритму. Кроме этого появилась возможность выбирать матрицы сравнения аминокислот и нуклеотидов, а также устанавливать штрафы за внесение пробелов. Следует отметить, что высокая совместимость программ этого поколения с другими пакетами программ обусловлена за счет предоставления результатов выравнивания в виде формата FASTA.

Последним представителем серии является программа Clustal X, для которой характерен более удобный интерфейс и более легкая оценка результатов выравниваний. В настоящее время именно последние программы серии Clustal этого поколения (версия 1.83) позволяют создавать наиболее биологически корректные

множественные выравнивания дивергировавших последовательностей.

Программы третьего поколения серии Clustal доступны на многих серверах (<http://npsa-pbil.ibcp.fr>, <http://www.ebi.ac.uk>) в двух вариантах – интерактивном и почтовом. Интерактивный вариант предполагает ожидание пользователем получения результатов выравнивания (целесообразно применять при небольшом (<100) количестве последовательностей), а почтовый – по электронной почте (применяется при большом числе последовательностей).

Принципы работы CLUSTAL. Первоначально необходимо ввести на одном из серверов изучаемые аминокислотные или нуклеотидные последовательности в одном из 7 возможных форматов (NBRF/PIR, EMBL/SWISSPROT, Pearson (Fasta), Clustal (*.aln), GCG/MSF (Pileup), GCG9/RSF, GDE). Наиболее часто используется формат FASTA, сущность которого заключается во введении знака «>» перед названием каждой последовательности, а затем (с новой строки) однобуквенном обозначении аминокислот и нуклеотидов. Суммарная длина вводимых последовательностей не должна превышать 40000 для WWW и 60000 для e-mail серверов.

При использовании данной программы выравнивание состоит из трех этапов: парных выравниваний, построения направляющего дерева и множественного выравнивания.

В ходе парных выравниваний предварительно сравниваются все возможные пары изучаемых последовательностей. На основании проведенных сравнений вычисляются показатели сходства в соответствии с выбранными матрицами.

Существуют 2 разновидности парного выравнивания: медленное (slow) и быстрое (fast). Медленное выравнивание является более точным, но его не рекомендуется применять в случае большого количества (более 20) последовательностей значительной длины (более 1000 остатков). Медленное выравнивание характеризуется 4 параметрами:

- штрафом на внесение делеции (gap open penalty). Уменьшение этого параметра способствует внесению разрывов в выравнивание, что ухудшает качество. Увеличение – приводит к тому, что выравнивание будет представлять собой длинные участки последовательностей почти без вставок или делеций.

- штраф на продолжение делеции (gap extension penalty). Этот параметр контролирует возможность внесения длинных вставок или делеций.

- матрица сравнений нуклеотидов (DNA weight matrix, Clustal W 1.6). В наиболее широко используемой матрице DNA identity совпадение нуклеотидов оценивается в 1 балл, а несовпадение – -10000 баллов. Такой высокий штраф за несоответствие облегчает внесение пробелов.

- матрица сравнения аминокислот (protein weight matrix) – PAM, Blosum и Gonnet.

Выбор матрицы оказывает большое влияние на получаемые результаты, так как каждая матрица представляет отражение отдельных эволюционных гипотез. Известно, что все замены аминокислот не являются равновероятными и в ходе эволюции чаще происходят замены на сходные по физико-химическим свойствам аминокислоты. Так в ходе эволюции гидрофобный изолейцин достаточно часто заменяется на гидрофобный валин и редко на гидрофильный цистеин.

Исследования эволюционных изменений различных белковых семейств позволили установить частоты фиксированных мутаций аминокислот и нуклеотидов и обобщить полученную информацию в виде матриц. В настоящее время используются серии белковых матриц Blosum, PAM и Gonnet. Матрицы серии Blosum преимущественно используются при проведении локальных выравниваний (поиск сходных последовательностей по базам данных).

Матрицы серии PAM, предложенные М. Дэйхофф, широко используются с 70-х годов. Основными отличиями матриц PAM и Blosum являются:

- 1) использование матрицами PAM простой эволюционной модели (подсчет замен на ветвях филогенетического дерева);
- 2) матрицы PAM основаны на учете мутаций по принципу глобального выравнивания (в высококонсервативных и высокомутабельных участках), а матрицы Blosum – локального (только высококонсервативных участков);
- 3) для матриц PAM замены в группах последовательностей подсчитываются сходным образом.

Матрицы этих двух серий сопоставимы следующим образом РАМ 100 – Blosum 90, РАМ 120 – Blosum 80, РАМ 160 – Blosum 60, РАМ 200 – Blosum 52, РАМ 250 – Blosum 45. Наиболее часто используются матрицы Blosum 62 и РАМ 160 (при среднем сходстве последовательностей). При выравнивании близко родственных последовательностей следует использовать матрицы Blosum с большим порядковым номером и матрицы РАМ с меньшим номером.

Матрицы Gonnet (рис. 4) представляют собой усовершенствованный вариант матриц Дэйхофф, основанный на большей базе данных. Использование этой матрицы наиболее целесообразно для инициальных сравнений.

Геном - совокупность информации, передаваемой живыми существами по наследству. Геномика – это наука, изучающая геном. В её задачи входят, среди прочего, секвенирование геномов и определение механизма связи между генами и признаками. Мутация - это самопроизвольное или спровоцированное изменение генома, передающееся по наследству. Мутации могут быть точечными, а могут затрагивать большие участки ДНК. Считается, что мутации являются необходимым фактором эволюции и образования новых видов.

В настоящее время существует несколько способов электронного представления генетических данных. Среди них наиболее популярны — форматы FASTA и FASTQ. Оба эти формата текстовые и кодируют каждый нуклеотид одной буквой.

Существующие алгоритмы позволяют выравнивать биологические последовательности. Это позволяет оценить их степень сходства, но без учёта не точечных мутаций.

На молекулярном уровне гены представляют собой участки молекул ДНК или РНК. Большая часть живых существ хранят свои гены в нескольких длинных молекулах ДНК, находящихся в клеточном ядре. При экспрессии генов соответствующий участок ДНК транскрибируется, в результате чего создаётся молекула транспортной РНК (тРНК), которая в дальнейшем может использоваться для синтеза молекулы белка.

Генетический код - способ кодирования структуры белков при помощи последовательности нуклеотидов. Молекула белка

представляет собой цепочку из аминокислот. Хотя общее число различных аминокислот достаточно велико, лишь небольшое число аминокислот, так называемые стандартные аминокислоты, могут становиться частью молекул белка. Всего известно 20 стандартных аминокислот.

Поскольку число возможных триплетов больше, чем число стандартных аминокислот, некоторые аминокислоты кодируются несколькими возможными триплетами (до шести триплетов для одной аминокислоты). Это придаёт коду некоторую степень помехоустойчивости - некоторые изменения последовательности нуклеотидов не приводят к изменению последовательности аминокислот, которую она кодирует.

Представление генетической информации в электронном виде. Задача электронного представления генетической информации встала перед исследователями, когда появились устройства, способные считывать эту информацию. Поскольку различных нуклеотидов и стандартных аминокислот немного, логично, что их стали кодировать одним символом. Обычно для кодирования нуклеотида используется первая буква в его названии. В то же время, названия многих аминокислот начинаются с одинаковых букв, поэтому коды многих аминокислот не совпадают с из первыми буквами — используются те буквы, которые не заняты.

Формат FASTA. FASTA — один из наиболее популярных форматов для представления последовательностей нуклеотидов или аминокислот. Файл в формате FASTA - простой текстовый файл. Первая строка должна начинаться с символа «>» или «;». Она содержит имя последовательности и некоторую дополнительную информацию, предназначенную для идентификации. Другие строки, начинающиеся с «;», являются комментариями и игнорируются.

После первой строки начинается, собственно, описание последовательности. При кодировании последовательности нуклеотидов, буквы А, С, G, Т и U кодируют, соответственно, аденин, цитозин, гуанин, тимин и урацил. Также некоторые буквы кодируют позиции, в которых находится один нуклеотид из некоторого множества (это используется, если неизвестно, какой

именно нуклеотид там находится). Символ - (дефис) кодирует неизвестную последовательность произвольной длины.

Формат FASTQ. FASTQ - формат представления биологической последовательности совместно с данными о качестве. Этот формат используется для представления данных секвенирования, так как позволяет представить как саму последовательность, так и вероятность, что каждый из элементов последовательности указан правильно. Для этого, кроме символов, кодирующих элементы последовательности, используется символы, кодирующие уровень качества.

Уровень качества - целое число в некотором диапазоне. Известно два различных способа выразить уровень качества через вероятность ошибки. Чаще всего используется следующая формула: $Q = -10\log(p)$ Здесь Q - уровень качества, p - вероятность, что этот элемент последовательности - ошибочный.

Сам же файл содержит четыре строки для каждой последовательности. Первая строка начинается с символа «@», после которого идёт описание последовательности, как и в формате FASTA. Следующая строка содержит последовательность символов, кодирующих саму последовательность, аналогично формату FASTA. За ней идёт строка, начинающаяся с символа «+», после которого может идти описание последовательности (третья строка будет отличаться от первой только тем, что первый символ заменён на «+»), а может ничего не идти.

Последняя строка содержит уровни качества. Её длина равна длине второй строки, а каждый символ кодирует информацию о качестве элемента последовательности, закодированного соответствующим символом второй строки. Сами же уровни кодируются таким образом: ASCII-код символа равен уровню качества плюс некоторая константа. Константа обычно имеет значение 33 или 64. В любом случае, код символа не должен превышать 127.

Формат GenBank. Формат GenBank позволяет представить больше дополнительной информации о последовательности. Файл в формате GenBank состоит из нескольких записей, каждая из которых может занимать несколько строк. Все строки в записи, кроме первой, начинаются с пробела, это позволяет легко находить границы записей. Каждая запись начинается с имени, за которым

идёт значение (через один или несколько пробелов). Некоторые записи могут содержать подзаписи. Они форматируются аналогично записям.

После этого в начало каждой строки записывается несколько пробелов. Таким образом, значения подзаписей выравнены правее значений записей, а те выравнены правее имён записей. Имена записей имеют предопределённое значение. Например, запись DESCRIPTION хранит описание последовательности, запись SOURCE идентифицирует особь, с которой считана последовательность, а запись REFERENCE (их может быть несколько) используется для ссылок на публикации. Запись ORIGIN содержит саму последовательность.

Каждая строка, кроме последней, содержит 60 элементов, разбитых пробелами на группы по 10. Для представления элементов используются строчные буквы. В начало каждой строки добавляется текущая позиция, начиная с единицы, и пробел.

Существующие методы поиска гомологий в биологических последовательностях.

Гомология - структурное сходство. В генетике под гомологиями понимаются участки белков или ДНК, имеющую сходную последовательность аминокислот или нуклеотидов. Существующие методы поиска гомологий умеют находить участки (подстроки) двух последовательностей, которые отличаются не очень сильно.

Алгоритм Нидлмана-Вунша. Алгоритм Нидлмана-Вунша впервые был опубликован в 1970 году и позволяет определять степень сходства последовательностей, а также находить глобальное выравнивание, - находить, какой именно символ из одной последовательности соответствует некоторому символу из другой последовательности.

Для своей работы алгоритм использует матрицу сходства, которая указывает, насколько схожими считать разные нуклеотиды. Для различных нуклеотидов используются отрицательные элементы. Поэтому последовательности должны содержать некоторую долю совпадающих нуклеотидов, для того чтобы быть признанными гомологичными.

Использование матрицы позволяет придавать разный вес разным заменам нуклеотидов. Например, поскольку транзиции более

вероятны, чем трансверсии, логично считать последовательности, отличающиеся заменой пурина на пурин или пиримидина на пиримидин, более схожими, чем те, которые отличаются заменой пурина на пиримидин или наоборот.

Вообще, матрица позволяет приписать любой вес любым заменам. Обычно используется симметричная матрица, однако применение несимметричной матрицы позволяет различать замены в одну и в другую сторону.

Пример матрицы сходства:

	А	Г	Т	Ц
А	10	-1	-4	-3
Г	-1	7	-3	-5
Т	-4	-3	8	0
Ц	-3	-5	0	9

Здесь А, Г, Т и Ц обозначают, соответственно, аденин, гуанин, тимин и цитозин, а числа в матрице указывают степень сходства между двумя нуклеотидами.

Алгоритм Нидлмана-Вунша способен сопоставлять символы двух последовательностей так, что сумма значений сходства для соответствующих символов максимально. Кроме того, алгоритм может учитывать вставки и удаления. При этом считается, что символ в одной строке, которому не соответствует никакой символ из другой строки, имеет некоторый уровень сходства, который является параметром алгоритма (например, -5).

Алгоритм Смита-Вотермана. Алгоритм Смита-Вотермана аналогичен алгоритму Нидлмана-Вунша, но при этом решается задача локального выравнивания: находит подстроки первой и второй строк, обладающие максимальным сходством, а также выравнивает их. Алгоритм был опубликован в 1981 году. Как и алгоритм Нидлмана-Вунша, алгоритм Смита-Вотермана использует матрицу сходства. Это позволяет учитывать различные замены с различным весом. Также он позволяет учитывать разные добавления и удаления по-разному в зависимости от того, какой именно нуклеотид был добавлен или удалён.

Как и алгоритм Нидлмана-Вунша, алгоритм Смита-Вотермана всегда находит оптимальное решение. Однако время работы и занимаемая память делают эти алгоритмы неприемлемыми для работы с большим количеством генетического материала.

Критерий сходства биологических последовательностей.

Чтобы оценивать сходство двух последовательностей, необходимо придумать некоторую меру сходства. Предложенная в данной работе мера сходства - число от нуля до единицы, где единица соответствует максимальному сходству, а ноль — минимальному. Кроме того, мера удовлетворяет следующим свойствам:

- Сходство последовательности с самой собой равно единице.
- Сходство несхожих последовательностей обычно невелико.
- Сходство не изменяется значительно, если одну из последовательностей подвергнуть мутации. Чем менее вероятна мутация, тем сильнее изменяется мера сходства.
- Существует способ эффективно вычислять меру сходства для практически встречающихся последовательностей.
- Сложность разработки подобной меры заключается в том, что нелокальные мутации могут приводить к значительным перестановкам частей последовательности.

Рассмотрим типовой способ решения проблемы. Назовём одну из последовательностей исходной, а другую — целевой. Будем рассматривать все возможные разбиения целевой последовательности на фрагменты (подстроки) и для каждого из них подсчитаем степень различия — число, тем большее, чем хуже, с точки зрения данного разбиения, целевая последовательность аппроксимирует исходную. Затем найдём минимум степени различия по всем разбиениям.

В каждом разбиении каждый фрагмент можно считать свободным или связанным. От этого зависит вклад этого фрагмента в степень различия. Вклад свободного фрагмента в степень различия пропорционален его длине с константой, являющейся параметром метода, и не зависит от его содержимого. Вклад же связанного фрагмента зависит от его содержимого: каждому связанному фрагменту сопоставляется подстрока исходной последовательности или комплементарной к ней, и вклад фрагмента в степень различия равен редакционному расстоянию между содержимым фрагмента и этой подстрокой. Здесь редакционное расстояние рассматривается в обобщённом смысле: можно приписывать различные веса разным заменам, вставкам и удалениям.

Как и разбиение на фрагменты, назначение типов фрагментов и соответствующих подстрок происходит таким образом, чтобы минимизировать суммарную степень различия. Кроме того, чтобы сделать большое число коротких фрагментов менее оптимальным, к степени различия добавляется ещё одно слагаемое, пропорциональное общему числу фрагментов в разбиении.

Мера сходства вычисляется по степени различия путём применения линейного преобразования, переводящего нулевую степень различия в единицу, а максимально возможную степень различия в ноль. Поскольку одним из разбиений является разбиение на один свободный фрагмент, степень различия не превосходит такую, которая соответствует этому разбиению, поэтому удобно принимать это значение соответствующим нулевой степени сходства.

Если редакционное расстояние между связанным фрагментом и соответствующей подстрокой достаточно велико, то такое разбиение заведомо не оптимально: можно сделать этот фрагмент свободным и таким образом уменьшить суммарную степень различия.

Более того, даже если фрагмент содержит немного изменений относительно его длины, но все эти изменения расположены недалеко друг от друга, можно разбить этот фрагмент на три фрагмента, вырезав из него кусок, содержащий большую часть изменений, и заменив его свободным фрагментом. В результате, во многих случаях можно не считать (или не продолжать считать) редакционное расстояние между фрагментами, поскольку и так понятно, что оптимальное разбиение не содержит такой фрагмент.

Кроме того, использование редакционного расстояния не даёт локальным мутациям существенно изменять степень различия: если мутация попала в свободный фрагмент, то степень различия может лишь слегка измениться за счёт изменения длины фрагмента, а если мутация попала в связанный фрагмент, то, опять же, соответствующее изменение редакционного расстояния не будет большим.

В отличие от локальных мутаций, которые сказываются на редакционном расстоянии, нелокальные мутации непосредственно

сказываются на самом разбиении. Например, если подвергнуть последовательность дубликации (вставить в неё копию её подстроки), то в соответствующее разбиение можно тоже вставить копию его подстроки (при этом крайние фрагменты могут оказаться обрезанными). При этом вклад в степень различия от добавленных фрагментов не превосходит вклад от исходных фрагментов. Кроме того, некоторый вклад в степень различия происходит из-за того, что тот фрагмент, в котором находится место вставки, оказывается разделён на два.

Другие нелокальные мутации учитываются аналогично. При учёте инверсий используется то, что подстроки, соответствующие связанным фрагментам, можно брать как из исходной последовательности, так и из комплементарной к ней.

Представление последовательности нуклеотидов в памяти компьютера. Существует несколько форматов файлов, предназначенных для хранения нуклеотидных последовательностей. Однако, эти форматы создавались для удобного восприятия человеком, а также для совместимости с программами и протоколами, рассчитанными на работу с текстовыми данными.

В то же время, для работы с нуклеотидной последовательностью желательно такое представление, с которым можно быстро выполнять следующие операции:

- Индексация - определение нуклеотида по порядковому номеру.
- Взятие подстроки - получение последовательности, содержащей те из нуклеотидов исходной последовательности, порядковые номера которых лежат в заданном диапазоне.
- Сравнение - определение, совпадают ли две заданные последовательности.
- Хеширование - операция, получающая по заданной последовательности число таким образом, чтобы одинаковым последовательностям соответствовали одинаковые числа, а разным, по возможности, разные.

Форматы типа FASTA допускают наличие в файле произвольного числа символов перевода строки между символами, кодирующими нуклеотиды, что препятствует эффективной индексации и сравнению, поэтому этот формат не подходит для выполнения

операций, при которых необходима возможность произвольного доступа к последовательности.

Вместо этого, используется двоичное представление последовательности. Поскольку в одной последовательности возможны четыре разновидности нуклеотидов, каждый нуклеотид кодируется двумя битами. Таким образом, каждый байт кодирует сразу четыре нуклеотида. Поскольку современные процессоры обрабатывают информацию блоками сразу по четыре или даже по восемь байт, подобное кодирование позволяет в несколько раз быстрее выполнять операции, которые не требуют обращения к индивидуальным нуклеотидам, например, сравнение, которое можно делать поблочно.

Кроме того, для представления подстрок используется механизм ссылок: вместо копирования данных при взятии подстроки создаётся объект, ссылающийся на исходные данные, с указанием смещения и длины подстроки.

Алгоритм сравнения генетических последовательностей.
Сравнение нуклеотидных последовательностей происходит в два этапа.

На первом этапе создаётся словарь, позволяющий быстро находить участки исходной последовательности, совпадающие с заданными короткими последовательностями (в качестве которых используются подстроки целевой последовательности).

На втором этапе ищется оптимальное разбиение целевой последовательности на фрагменты. При этом словарь используется, для того чтобы быстро находить подстроки исходной последовательности, соответствующие связанным фрагментам. Поскольку в оптимальном покрытии редакционное расстояние между связанным фрагментом и соответствующей ему подстрокой невелико, в них найдётся подстрока, совпадающая полностью. Каждое такое совпадение расширяется до тех пор, пока ошибок не станет настолько много, что дальнейшее расширение не имеет смысла.

Для подсчёта степени различия используется динамический алгоритм: подсчитывается степень различия для каждого префикса целевой последовательности, а фрагменты генерируются от начала целевой последовательности к её концу, что позволяет эффективно

пересчитывать степени различия. Результатом является степень различия для всей целевой последовательности.

Создание словаря.

Для быстрого поиска схожих участков в данной работе используется словарь - структура, позволяющая по нуклеотидной последовательности определённой длины (ключу) быстро найти все её вхождения в исходной подпоследовательности, а также в последовательности, комплементарной к ней.

В качестве словаря используется хеш-таблица, в которой ключами служат последовательности, а значениями — списки вхождений. Для каждого вхождения указывается, в какой из строк находится это вхождение (исходной или комплементарной), а также смещение этого вхождения относительно начала последовательности.

Для создания словаря перебираются все подстроки исходной последовательности некоторой длины, а также подстроки последовательности, комплементарной к ней. Длина ключа является параметром алгоритма и подбирается так, чтобы обеспечить оптимальное потребление ресурсов: если ключи будут слишком короткими, то каждой подстроке будет соответствовать слишком длинный список позиций, поэтому увеличится время работы.

Если же ключи будут слишком длинными, то алгоритм будет потреблять слишком много памяти. Кроме того, из-за слишком длинных ключей алгоритм может начать пропускать фрагменты, в которых нет полностью совпадающих участков достаточной длины. На практике оптимальная длина ключа получается около $\log_4(ls)-1$, где ls - длина исходной последовательности.

Для каждой подстроки соответствующей длины проверяется, есть ли она в словаре. Если есть, то в соответствующий список добавляется новое вхождение. Если нет, то в словарь добавляется новая запись, в котором ключом служит подстрока, а значением — новый список, содержащий единственное вхождение.

Поиск оптимального покрытия.

Поиск оптимального покрытия производится методом динамического программирования: для каждого префикса в целевой строке хранится минимальная степень различия и пересчитывается на основе информации о возможных фрагментах.

Кроме минимальной степени различия хранится минимальная степень различия для случаев, когда в конце префикса находится свободный фрагмент. Это включает случаи, когда этот фрагмент имеет нулевую длину. Это требуется для того, чтобы легче было учитывать свободные фрагменты.

Пусть g_i — минимальная степень различия для префикса длины i , а h_i — минимальная степень различия для префикса длины i при условии, что последний фрагмент - свободный.

Тогда выполняется неравенство: $g_i \leq h_i \leq g_i + Df$.

Здесь Df — коэффициент к добавке к степени различия, пропорциональной числу фрагментов.

Во всех точках, кроме нуля, эти значения инициализируются бесконечностями. В нуле используются значения $g_0 = -Df$ (ноль фрагментов, поэтому отрицательное число) и $h_0 = 0$ (здесь один пустой свободный фрагмент). Заметим, что g_0 — единственный из всех g_i и h_i может быть отрицательным, поскольку остальные соответствуют покрытиям, содержащим хотя бы один фрагмент.

Первое неравенство выполняется, потому что g_i - минимум на множестве покрытий префикса длины i , а h_i - минимум на подмножестве этого множества. Второе неравенство выполнено, так как к любому покрытию можно добавить свободный фрагмент длины ноль, что увеличит степень различия на Df .

В процессе работы алгоритма поддерживается инвариант, что для всех префиксов длины меньше некоторого i числа g_i и h_i уже вычислены и в дальнейшем не будут изменяться, а для остальных префиксов значения ещё не вычислены, поэтому не будут использоваться. В процессе работы алгоритма i может только увеличиваться. Для того чтобы выполнить пересчёт для соответствующего фрагмента, необходимо выполнить операцию $g_e \leftarrow \min(g_e, g_b + d + Df)$, где b - индекс начала фрагмента, e - индекс его окончания, d - его вклад в степень различия. Существенно, что g_b уже вычислено, а g_e ещё нет.

При таком методе требуется выполнять пересчёт для каждого фрагмента в тот момент, когда позиция с индексом i лежит внутри него. Это условие выполняется автоматически, если начинать поиск фрагмента с ключа, начинающегося с позиции i . После того, как массивы g и h обновлены с учётом найденных связанных

фрагментов, необходимо увеличить i на один. Перед этим следует пересчитать g_{i+1} и h_{i+1} для учёта свободных фрагментов. При этом g_{i+1} вычисляется по формуле: $g_{i+1} \leftarrow \min(g_{i+1}, h_i + D_l)$ (можно взять значение, полученное при пересчёте или продолжить свободную последовательность). Здесь D_l — вклад единицы длины свободного фрагмента в степень различия. Число h_{i+1} вычисляется по формуле: $h_{i+1} \leftarrow \min(h_i + D_l, g_{i+1} + D_f)$ (можно продолжить свободную последовательность или начать новую). Заметим, что g_{i+1} может присваиваться до этого пересчёта, поэтому использование этого значения до присваивания осмысленно.

Алгоритм заканчивает работу, когда $i = l_t + 1$ — когда пройдена вся целевая строка. Тогда $D = g_{l_t+1}$ по индукции.

Сканирование целевой строки

Чтобы быстро учесть все связанные фрагменты, получающиеся из вхождения ключа, используется такой метод: пусть ключ входит, начиная с позиции i целевой строки и j исходной. Поскольку сам ключ совпадает полностью, редакционное расстояние между фрагментом и соответствующей подстрокой складывается из двух слагаемых: редакционное расстояние между участками до ключа и после ключа.

Соответственно, если s и e — индексы начала и конца фрагмента в целевой строке, то выполняется неравенство:

$$g_e \leq g_s + d_{s,i} + d_{i,e}.$$

Здесь $d_{s,i}$ и $d_{i,e}$ — вклад в редакционное расстояние участков от s до i и от i до e . Чтобы обеспечить это неравенство, можно было бы перебирать все возможные значения s и e в некотором диапазоне, но можно сделать это намного быстрее: вначале найти $\min_s (g_s + d_{s,i})$ перебором s , а затем перебирать только значения e .

Для определения границ перебора s и e используется изложенный выше критерий оптимальности: если для какого-то s выгодно заменить подстроку от s до i свободным фрагментом, то дальнейшее уменьшение s не имеет смысла. Аналогично для e .

Оценка сложности алгоритма. Поскольку представленный алгоритм использует эвристики, его сложность существенно зависит от входных данных. В частности, чем более схожи исходная и целевая последовательности, тем медленнее будет работать алгоритм.

Порядок выполнения работы:

1. Изучите теоретический материал.
2. Составьте блок-схему алгоритма сравнения двух генетических текстов по 4 «буквам» с подсчетом доли совпадений.
3. В информационных источниках найти две любых генетических последовательности и выполнить трассировку разработанного алгоритма. Результаты трассировки оформить в виде таблицы.
4. Средствами Excel выполните сравнение генетических текстов и сравните результаты, полученные в п.3. Сделайте выводы.
5. Оформите отчет, включающий результаты выполнения п.3 и п.4. и краткие ответы на контрольные вопросы (не менее 4).
6. Составьте аннотацию не менее 3 информационных источников, найденных в сети интернет по тематике лабораторной работы (объемом 150-350 слов).

Примечание. Сравнение генетических текстов в Excel можно осуществить следующим образом: преобразовать последовательность символов в генетических текстах, найти алгебраическую разницу между полученными кодами, подсчитайте количество подряд стоящих «0», если таковых более трех, оцените процент совпадений.

Контрольные вопросы.

1. Что характеризует нуклеотидная генетическая последовательность?
2. Как осуществляется выравнивание нуклеотидных последовательностей? Какие компьютерные программы для этого используются в настоящее время?
3. В чем заключаются принципы работы программного инструментария CLUSTAL?
4. Как наука изучает информацию заключенную в геноме? Характеристики объектов и методологии исследования.
5. Общие и отличия форматов представления генетических данных FASTA, FASTQ и GenBank.
6. Каким образом осуществляется представление генетической информации в электронном виде.

7. Охарактеризуйте основные существующие методы поиска гомологий в биологических последовательностях?
8. Опишите математический аппарат обработки биоинформации в алгоритме Нидлмана-Вунша.
9. Опишите математический аппарат обработки биоинформации в алгоритме Смита-Вотермана.
10. По какому критерию оценивается сходство биологических последовательностей?
11. Охарактеризуйте этапы алгоритма сравнения генетических последовательностей?
12. Каким образом создается словарь для сравнения генетических последовательностей?
13. Каким образом осуществляется поиск оптимального покрытия?
14. Как осуществляется сканирование целевой строки?

ПРАКТИЧЕСКАЯ РАБОТА №8. ПОСТРОЕНИЕ И АНАЛИЗ КОРРЕЛЯЦИОННЫХ ОТНОШЕНИЙ МЕТОДАМИ ИНТЕЛЛЕКТУАЛЬНЫХ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

Цель работы: овладение навыками построения и анализа корреляционных отношений с целью анализа произошедших в процессе терапии изменений корреляций между динамиками регистрируемых характерных параметров.

Краткие теоретические сведения

Состояние системы в процессе ее функционирования меняется – реакция на окружающую среду (внешние воздействия), что характеризуется изменениями синхронности взаимодействия разнообразных функциональных систем. Поскольку данный процесс отслеживается путем регистрации определенных характеристических параметров, то следует предположить, что изменение синхронизации отражается как количественных сдвигом величин данных параметров, так и изменением значений величин, прямо или косвенно отображающих парную или множественную связи между «временными рядами» параметров. В статистике, в качестве таких величин, могут использоваться коэффициент парной корреляции и коэффициент корреляционного отношения, отражающей нелинейную регрессию.

В этом случае, предлагается использовать для количественной оценки происходящих корреляционных сдвигов Максимальный Градиент Функциональных Различий, который вычисляется следующим образом. Он вычисляется как наибольшее значение межранговых отклонений упорядоченных по убыванию связей показателей, выбранных для характеристики ФС (например, значения различных корреляционных коэффициентов). Показатель, имеющий наибольшую связность называется системообразующим для данного класса. В качестве меры связности рассматривается обычно значения коэффициента парной корреляции, превышающие порог определенного уровня статистической значимости и

ранжированные определенным образом (например, в алфавите – от 0 до 9).

Например, пусть два некоторых класса имеют следующую структуру – Рис.1

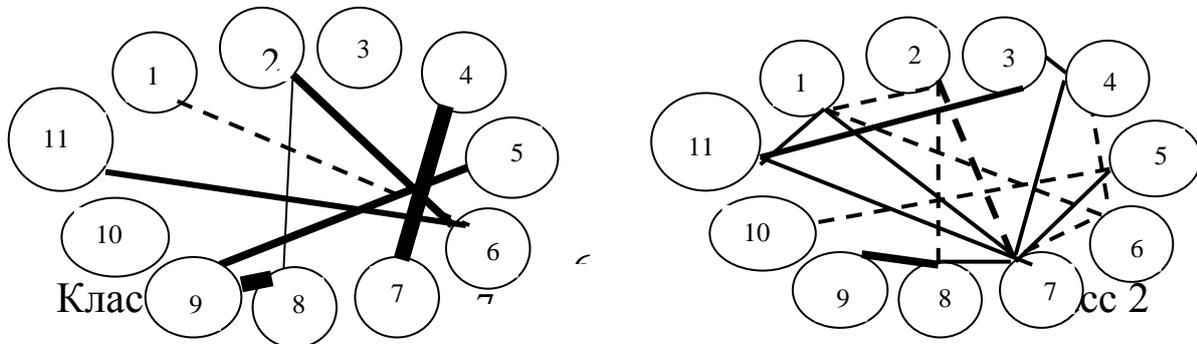


Рис. 1 Корреляционные связи между показателями 1-11 (толщина линии отражает величину коэффициента корреляции, пунктир – обратная связь).

Для рассматриваемого примера: системообразующими показателями являются: в Классе 1 – 6 показатель, в Классе 2 – 7. МГФР определяется следующим образом. Упорядочиваем показатели по мере связности – соответственно получаем: в Классе 1- 6,9,8,7,4,2,5,11,1,(3,10); в Классе 2 – 7,1,11,8,6,2,4,5,9,3,10. Таким образом, смещение показателей (или градиент – в терминологии []) составило в абсолютном значении: для 1 показателя – 7, для 2 – 0, для 3 – 0, для 4 – 5, для 5 – 1, для 6 – 4, для 7 – 3, для 8 – 1, для 9 – 7, для 10 – 0 и для 11 – 5. Т.о., МГФР Класса 2 относительно Класса 1 равен 7.

Чем больше значение МГФР, тем более произошедшие в процессе терапии изменения в синхронности работы функциональных систем организма.

Порядок выполнения.

1. В среде EXCEL сформировать таблицу данных протокола мониторинга.
2. Построить корреляционную матрицу.
3. Построить матрицу парных корреляционных отношений.
4. Отобразить графически (в виде связанного графа, вершинами которых являются регистрируемые характеристиками, дугами –

статистически значимые корреляционные связи, количество дуг между двумя вершинами – ранг связности, определяемый по формуле $rangs_{xy} = round(6 * \pi * (\cosh(R_{xy}) - 1))$, пунктир дуги означает «отрицательную» связь) построенные матрицы.

5. Рассчитать МГФР

6. Сделать выводы, оформить отчет.

Исходные данные

№	X1	X2	X3	X4	X5	Состояние
1	21	100	-15	518	204	Состояние 1
2	19	99	-11	482	208	Состояние 2
3	19	96	-5	477	209	Состояние 3
4	20	95	-18	517	204	Состояние 1
5	15	89	-3	475	203	Состояние 1
6	21	94	-7	479	193	Состояние 2
7	25	88	-7	549	203	Состояние 3
8	21	89	-11	494	194	Состояние 2
9	24	110	-18	487	195	Состояние 1
10	23	88	-17	486	190	Состояние 1
11	21	66	-31	544	213	Состояние 3
12	19	98	-16	465	190	Состояние 2
13	19	100	-22	482	113	Состояние 2
14	23	82	-14	485	211	Состояние 1
15	21	99	-14	492	204	Состояние 3
16	22	79	-8	516	195	Состояние 3
17	20	109	-9	493	197	Состояние 3
18	20	96	-10	481	196	Состояние 1
19	18	85	-21	479	205	Состояние 2
20	19	105	-14	497	189	Состояние 2
21	29	111	-29	417	159	Состояние 2
22	29	110	-31	427	160	Состояние 3
23	28	128	-27	434	171	Состояние 2
24	26	126	-38	427	163	Состояние 1
25	35	106	-37	390	173	Состояние 3
26	30	123	-34	387	145	Состояние 1
27	33	121	-30	421	165	Состояние 1
28	30	114	-39	460	169	Состояние 3
29	-18	1078	-6	2089	210	Состояние 2
30	-21	1060	2	300	220	Состояние 2
31	-21	991	14	32	224	Состояние 3
32	-20	948	-14	2065	210	Состояние 2
33	-26	780	20	-74	208	Состояние 3
34	-19	921	10	159	182	Состояние 1
35	-15	742	10	3635	208	Состояние 1
36	-19	773	3	904	184	Состояние 3
37	-16	1364	-13	540	188	Состояние 2
38	-17	755	-10	491	175	Состояние 3
39	-18	857	1	3395	232	Состояние 2
40	-21	1047	-9	-544	176	Состояние 1
41	-22	1083	-21	307	231	Состояние 3
42	-16	579	-5	434	229	Состояние 2
43	-19	1063	-4	791	211	Состояние 1
44	-17	500	8	2012	188	Состояние 1
45	-20	1354	7	861	192	Состояние 2
46	-20	983	5	265	191	Состояние 1
47	-23	663	-19	134	212	Состояние 3
48	-21	1242	-4	1031	172	Состояние 2
49	-31	2	55	5631	158	Состояние 1
50	-31	-29	50	6146	160	Состояние 1

51	-32	482	58	6506	187	Состояние 3
52	-34	424	34	6139	166	Состояние 2
53	-24	-128	38	4307	292	Состояние 2
54	-31	345	45	4127	124	Состояние 1
55	-26	282	53	5839	172	Состояние 3
56	-30	85	33	7815	184	Состояние 1
57	-35	203	74	7406	158	Состояние 2
58	-26	158	48	4095	171	Состояние 3
59	-30	82	30	6128	166	Состояние 1
60	-34	-106	33	5468	164	Состояние 1
61	-33	-205	43	4699	158	Состояние 2
62	-31	341	48	4406	151	Состояние 3
63	-22	-231	42	6714	166	Состояние 2
64	-32	225	43	5662	173	Состояние 1
65	-29	136	44	7854	176	Состояние 1
66	-30	375	52	5645	145	Состояние 3
67	-31	-167	38	5892	163	Состояние 2

ПРАКТИЧЕСКАЯ РАБОТА №9. ИСПОЛЬЗОВАНИЕ НЕЙРОННОЙ СЕТИ В ДИАГНОСТИЧЕСКОМ ПРОЦЕССЕ

Цель работы: овладение навыками применения искусственных нейронных сетей в диагностическом процессе на примере персептрона и сетей Кохонена.

Краткие теоретические сведения

Нейронные сети представляют собой новую и весьма перспективную вычислительную технологию, дающую новые подходы к исследованию динамических и классификационных задач в системах поддержки принятия решений в диагностическом процессе и-или при обработке результатов мониторинга функционирования биообъекта (включая экологические аспекты).

Способность к моделированию нелинейных процессов, работе с зашумленными данными и адаптивность дают возможности применять нейронные сети для решения широкого класса задач.

Приложения нейронных сетей охватывают самые разнообразные области интересов: распознавание образов, обработка зашумленных данных, дополнение образов, ассоциативный поиск, классификация, оптимизация, прогноз, диагностика, обработка сигналов, абстрагирование, управление процессами, сегментация данных, сжатие информации, сложные отображения, моделирование сложных процессов, машинное зрение, распознавание речи.

Первый интерес к нейросетям был обусловлен пионерской работой МакКаллока и Питса, изданной в 1943 году, где предлагалась схема компьютера, основанного на аналогии с работой человеческого мозга. Они создали упрощенную модель нервной клетки – *нейрон*. Мозг человека состоит из белого и серого веществ: белое – это тела нейронов, а серое – это соединительная ткань между нейронами, или *аксоны и дендриты*. Мозг состоит примерно из 10^{11} нейронов, связанных между собой. Каждый нейрон получает информацию через свои дендриты, а передает ее дальше только через единственный аксон, разветвляющийся на конце на тысячи *синапсов*. Простейший нейрон может иметь до 10000 дендритов, принимающих сигналы от других клеток. Таким образом,

мозг содержит не менее 10^{15} взаимосвязей между нейронами, обеспечивающих его целевое востребуемое функционирование.

Искусственным нейроном называется простой элемент, сначала вычисляющий взвешенную сумму V входных величин x_i :

$$V = \sum_{i=1}^N W_i * x_i = W * \bar{X}$$
, где N – размерность пространства входных сигналов.

Затем полученная сумма сравнивается с пороговой величиной W_0 , вслед за чем вступает в действие нелинейная функция активации f . Коэффициенты $\{W_i\}$ во взвешенной сумме обычно называют *синаптическими коэффициентами* или *весами*. Саму же взвешенную сумму V мы будем называть потенциалом нейрона i . Выходной сигнал тогда имеет вид $f(V)$.

Величину порогового барьера можно рассматривать как еще один весовой коэффициент при постоянном входном сигнале. В этом случае говорится о *расширенном входном пространстве*: нейрон с N -мерным входом имеет $N+1$ весовой коэффициент. Если ввести в уравнение пороговую величину W_0 , то оно переписывается так:

$$V = \sum_{i=1}^N W * x + W_0$$

В зависимости от способа преобразования сигнала и характера активации возникают различные виды нейронных структур. Существуют *детерминированные нейроны*, когда активизирующая функция однозначно вычисляет выход по входу, и *вероятностные нейроны*, состояние которых в момент t есть случайная функция потенциала и состояния в момент $t-1$.

В искусственных нейронах могут быть различные функции активации, к наиболее типовым из которых относятся следующие виды функций:

- ◆ Линейная: выходной сигнал нейрона равен его потенциалу,
- ◆ пороговая: нейрон выбирает решение из двух вариантов: активен / неактивен,
- ◆ Многопороговая: выходной сигнал может принимать одно из q значений, определяемых $(q-1)$ порогом внутри предельных значений.

◆ Сигмоидная: рассматриваются два вида сигмоидных функций:

$$s = f(V) = \frac{1}{1 + \exp(-bV)}$$

с выходными значениями в промежутке $[0,1]$ и

$$s = f(V) = \frac{\exp(bV) - 1}{\exp(bV) + 1}$$

с выходными значениями в промежутке $[-1,1]$.

Коэффициент b определяет *крутизну* сигмоида. Поскольку сигмоидная функция является гладким отображением $(-\infty, \infty)$ на $(-1, 1)$, то крутизну можно учесть через величины весов и порогов, и без ограничения общности можно полагать ее равной единице.

Графические изображения простейшего нейрона и виды функций с их графиками приведены на рис. 1.

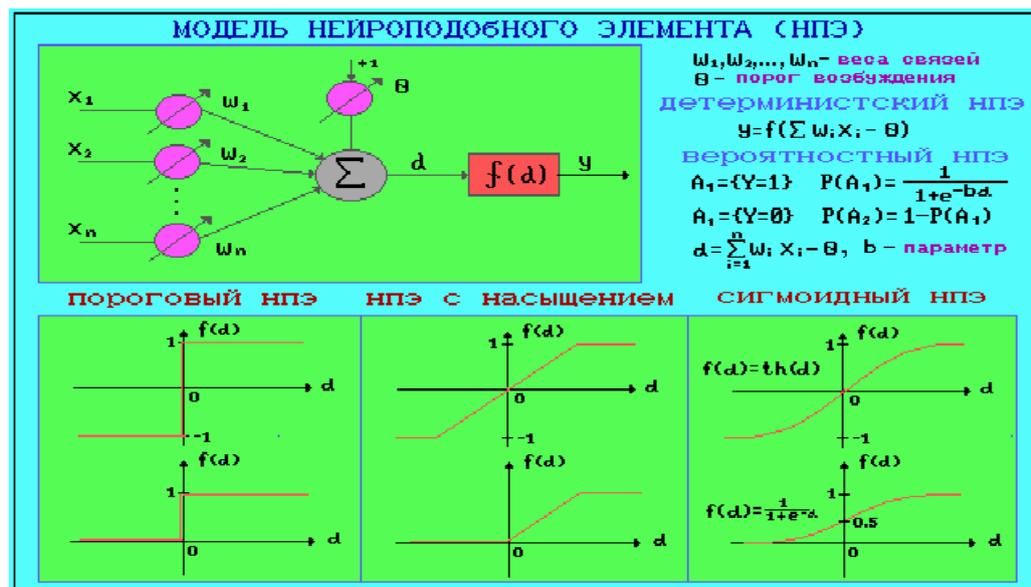


Рис.1 Изображение искусственного нейрона

Из точек на плоскости и соединений между ними можно построить множество графических фигур, называемых *графами*. Если каждую точку представить себе как один нейрон, а соединения между точками – как дендриты и синапсы, то получим нейронную сеть. Но не всякое соединение нейронов будет работоспособно или вообще целесообразно. По архитектуре связей нейросети могут быть сгруппированы в два класса: сети прямого распространения, в которых связи не имеют петель, и сети

рекуррентного типа, в которых возможны обратные связи(см. рис. 2).

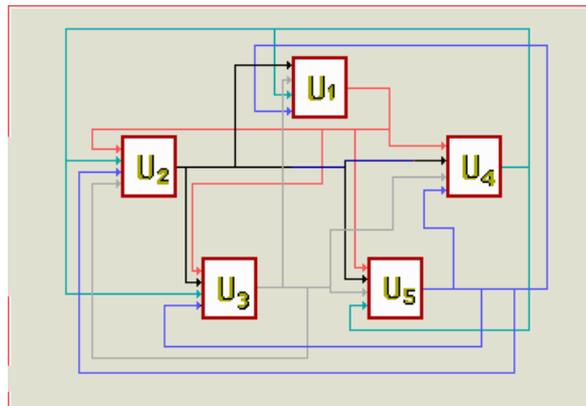


Рис. 2 Рекуррентная сеть

Сети прямого распространения подразделяются на однослойные перцептроны (сети) и многослойные перцептроны (сети). Название перцептрона для нейросетей придумал американский нейрофизиолог Ф. Розенблатт, придумавший в 1957 году первый нейропроцессорный элемент (НПЭ), то есть нейросеть. Он же доказал сходимости области решений для перцептрона при его обучении. Сразу после этого началось бурное исследование в этой области и был создан самый первый нейрокомпьютер Mark I. Многослойные сети отличаются тем, что между входными и выходными данными располагаются несколько так называемых скрытых слоев нейронов, добавляющих больше нелинейных связей в модель.

Рассмотрим устройство простейшей многослойной нейросети (схема представлена на рис.3). Любая нейронная сеть состоит из **входного слоя** и **выходного слоя**. Соответственно подаются независимые и зависимые переменные. Входные данные преобразуются нейронами сети и сравниваются с выходом. Если отклонение больше заданного, то специальным образом изменяются веса связей нейронов между собой и пороговые значения нейронов. Снова происходит процесс вычислений выходного значения и его сравнение с эталоном. Если отклонения меньше заданной погрешности, то процесс обучения прекращается. Помимо входного и выходного слоев в многослойной сети существуют так называемые **скрытые слои**. Они представляют собой нейроны, которые не имеют непосредственных входов исходных данных, а связаны только с выходами входного слоя и с

входом выходного слоя. Таким образом, скрытые слои дополнительно преобразуют информацию и добавляют нелинейности в модели.

Рассмотрим более подробно функционирование многослойного персептрона, представленного на рис. 3.

Если однослойная нейросеть очень хорошо справляется с задачами классификации, так как выходной слой нейронов сравнивает полученные от предыдущего слоя значения с порогом и выдает значение либо ноль, то есть меньше порогового значения, либо единицу - больше порогового (для случая пороговой внутренней функции нейрона), и не способен решать большинство практических задач (что было доказано Минским и Пейпертом), то многослойный персептрон с сигмоидными решающими функциями **способен аппроксимировать любую функциональную зависимость** (это было доказано в виде теоремы). Но при этом не известно ни нужное число слоев, ни нужное количество скрытых нейронов, ни необходимое для обучения сети время. Эти проблемы до сих пор стоят перед исследователями и разработчиками нейросетей. Поведение рекуррентных сетей описывается дифференциальными или разностными уравнениями, как правило, первого порядка. Это расширяет области применения нейросетей и способы их обучения. Сеть организована так, что каждый нейрон получает входную информацию от других нейронов, возможно, и от самого себя, и от окружающей среды. Этот тип сетей имеет важное значение, так как с их помощью можно моделировать нелинейные динамические системы.

Среди рекуррентных сетей можно выделить **сети Хопфилда и сети Кохонена**.

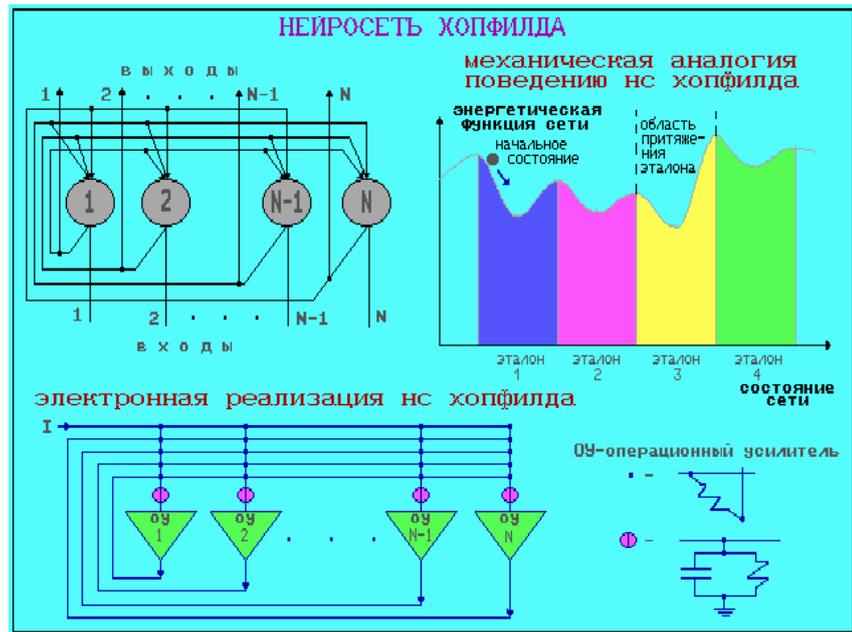


Рис.4 Структура нейросети Хопфилда

С помощью сетей Хопфилда можно обрабатывать неупорядоченные (рукописные буквы), упорядоченные во времени (временные ряды) или пространстве (графики) образцы. Рекуррентная нейросеть простейшего вида была введена Хопфилдом и построена она из N нейронов, связанных каждый с каждым кроме самого себя, причем все нейроны являются выходными. Нейросеть Хопфилда можно использовать в качестве **ассоциативной памяти**. Архитектура сети Хопфилда изображена на рис. 4. Сеть Кохонена еще называют «самоорганизующейся картой признаков». Сеть такого типа рассчитана на самостоятельное обучение во время обучения сообщать ей правильные ответы необязательно. В процессе обучения на вход сети подаются различные образцы. Сеть улавливает особенности их структуры и разделяет образцы на кластеры, а уже обученная сеть относит каждый вновь поступающий пример к одному из кластеров, руководствуясь некоторым критерием «близости». Сеть состоит из одного входного и одного выходного слоя. Количество элементов в выходном слое непосредственно определяет, сколько различных кластеров сеть сможет распознать. Каждый из выходных элементов получает на вход весь входной вектор. Как и во всякой нейронной сети, каждой связи приписан

некоторый синаптический вес. В большинстве случаев каждый выходной элемент соединен также со своими соседями. Эти внутрислойные связи играют важную роль в процессе обучения, так как корректировка весов происходит только в окрестности того элемента, который наилучшим образом откликается на очередной вход. Выходные элементы соревнуются между собой за право вступить в действие и «получить урок». Выигрывает тот из них, чей вектор весов окажется ближе всех к входному вектору.

Главное отличие и преимущество нейросетей перед классическими средствами прогнозирования и классификации заключается в их способности к обучению. На этапе обучения происходит вычисление синаптических коэффициентов в процессе решения нейронной сетью задач, в которых нужный ответ определяется не по правилам, а с помощью примеров, сгруппированных в обучающие множества. Таким образом, нейросеть на этапе обучения сама выполняет роль *эксперта* в процессе подготовки данных для построения экспертной системы. Предполагается, что *правила* находятся в структуре обучающих данных.

Для обучения нейронной сети требуются *обучающие данные*. Они должны отвечать свойствам *представительности и случайности или последовательности*. Все зависит от класса решаемой задачи. Такие данные представляют собой ряды примеров с указанием для каждого из них значением выходного параметра, которое было бы желательно получить. Действия, которые при этом происходят, можно назвать *контролируемым обучением*: «учитель» подает на вход сети вектор исходных данных, а на выходной узел сообщает желаемое значение результата вычислений. Контролируемое обучение нейросети можно рассматривать как решение оптимизационной задачи. Ее целью является минимизация функции ошибок E на данном множестве примеров путем выбора значений весов W . Достижение минимума называется *сходимостью* процесса обучения. Именно возможность этого и доказал Розенблатт. Поскольку ошибка зависит от весов нелинейно, получить решение в аналитической форме невозможно, и поиск глобального минимума осуществляется посредством итерационного процесса -

так называемого *обучающего алгоритма*. Разработано уже более сотни разных обучающих алгоритмов, отличающихся друг от друга стратегией оптимизации и критерием ошибок. Обычно в качестве меры погрешности берется средняя квадратичная ошибка (СКО):

$$E = \sqrt{\frac{\sum_{i=1}^M (d_i - y_i)^2}{M}}, \text{ где } M - \text{ число примеров в обучающем множестве.}$$

Минимизация величины E осуществляется с помощью градиентных методов. Изменение весов происходит в направлении, обратном к направлению наибольшей крутизны для функции:

$$W(t+1) = W(t) + \varepsilon * \frac{\partial E}{\partial W}, \text{ где } \varepsilon - \text{ определяемый пользователем параметр, который называется коэффициентом обучения.}$$

Одним из самых распространенных алгоритмов обучения нейросетей прямого распространения является алгоритм обратного распространения ошибки (BackPropagation, BP). Этот алгоритм был переоткрыт и популяризован в 1986 г. Румельхартом и Мак Клелландом из группы по изучению параллельных распределенных процессов в Массачусетском технологическом институте.

Основная идея состоит в том, чтобы вычислять чувствительность ошибки сети к изменениям весов. Для этого нужно вычислить частные производные от ошибки по весам. Пусть обучающее множество состоит из P образцов, и входы k -го образца обозначены через $\{x_k^i\}$. Вычисление частных производных осуществляется по *правилу цепи*: вес входа i -го нейрона, идущего от j -го нейрона, пересчитывается по формуле:

$$\Delta W_{ij} = -\varepsilon * \sum_{k=1}^P \frac{\partial E_k}{\partial W_{ij}} = -\varepsilon * \sum_{k=1}^P \frac{\partial E_k}{\partial V_k^i} * \frac{\partial V_k^i}{\partial W_{ij}} = -\varepsilon * \sum_{k=1}^P \delta_k^i * x_k^i,$$

где ε - длина шага в направлении, обратном к градиенту.

Если рассмотреть отдельно k -ый образец, то соответствующее изменение весов равно:

$$\Delta W_{ij} = -\varepsilon * \frac{\partial E_k}{\partial W_{ij}} = -\varepsilon * \delta_k^i * x_k^i.$$

Множитель δ_k^i вычисляется через аналогичные множители из последующего слоя, и ошибка, таким образом, передается в обратном направлении.

Для выходных элементов получим:

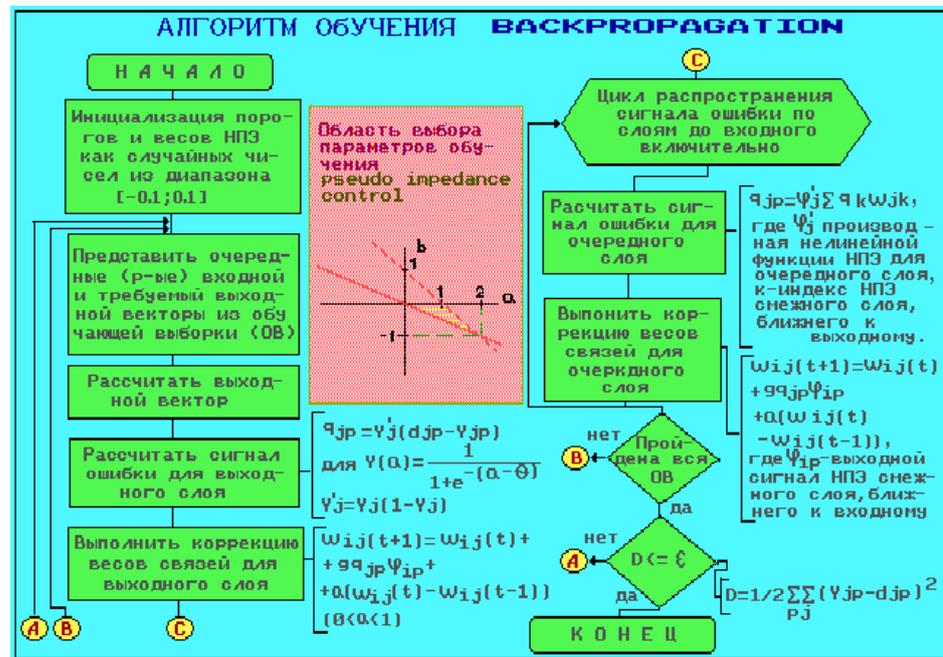
$$\delta_k^i = -\frac{\partial E_k}{\partial v_k^i} = -\frac{\partial E_k}{\partial x_k^i} * \frac{\partial x_k^i}{\partial v_k^i} = -(d_k^i - x_k^i) f'(v_k^i).$$

Для скрытых элементов множитель δ_k^i определяется так:

$$\delta_k^i = -\frac{\partial E_k}{\partial v_k^i} = -\sum_h \frac{\partial E_k}{\partial v_k^h} \frac{\partial v_k^h}{\partial v_k^i},$$

где индекс h пробегает номера всех нейронов, на которые воздействует i -ый нейрон.

Алгоритм обратного распространения ошибки представлен на рис.



5.

Рис.5 – Алгоритм обучения многослойной нейронной сети

Перед тем, как начинать процесс обучения нейронной сети, необходимо присвоить весам начальные значения. Цель состоит в том, чтобы найти как можно более хорошее начальное приближение к решению и таким образом сэкономить время обучения и улучшить сходимость. Классический подход к этой проблеме состоит в том, чтобы случайным образом выбрать малые значения для всех весов, чтобы быть уверенным, что ни один из сигмоидных элементов не перенасыщен. Однако это не дает

полной гарантии, что такое приближение приведет к глобальному минимуму или уменьшит время сходимости. Чтобы обучение не двигалось в ложном направлении при обработке задачи классификации или распознавания, но не задачи аппроксимирования временных рядов, данные нужно перемешивать случайным образом. Иначе нейросеть «выучит» последовательность случайно оказавшихся рядом значений как истинное правило, и потом будет делать ошибку.

Иногда при изменении весов связей нейронов кроме текущего изменения веса к нему прибавляют вектор смещения с предыдущего шага, взятый с некоторым коэффициентом. В этом случае говорят, что учитывается предыдущий импульс движения. Формула изменения веса связи будет выглядеть следующим образом:

$$\Delta W_{ij}(t+1) = \mu * \Delta W_{ij}(t) - (1 - \mu)\varepsilon \frac{\partial E}{\partial W_{ij}}.$$

где μ - число в интервале (0,1), которое задается пользователем.

Меру точности обучения сети определяет «шаг» изменения параметров в процессе управления обучением. Чем он больше, тем более грубым будет следующее уменьшение суммарной ошибки сети. Чем он меньше, тем больше времени сеть будет тратить на обучение и тем более возможно ее попадание в окрестность локального минимума ошибки. Поэтому управление шагом имеет важное значение для улучшения сходимости нейронной сети. В современных нейросетевых пакетах пользователь может сам определять, как будет изменяться величина шага. Очень часто по умолчанию берется линейная или экспоненциальная зависимость величины шага от количества итераций сети.

Одной из самых больших проблем при использовании нейросетей является невозможность предварительного определения оптимального количества скрытых слоев и нейронов в них. Если нейронов будет слишком мало, то это равносильно потере каких-то нелинейных связей в модели, если нейронов будет много, то это может привести к «переобучению» сети, то есть она просто «выучит» данные, а не распознает их структуру. Поэтому применяется два основных подхода:

- деструктивный подход: берется сеть заведомо большего размера, чем нужно, и в процессе обучения из нее удаляются связи и даже сами нейроны;
- конструктивный подход: первоначально берется маленькая сеть, и к ней, в соответствии со структурой и сложностью задачи, добавляются новые элементы.

Диапазон выходных значений решающих функций нейронов лежит в интервале $(0,1)$ либо $(-1,1)$. Поэтому для лучшей работы сети следует предварительно масштабировать данные обучающей выборки к интервалу от 0 до 1. Это позволит уменьшить ошибки и ускорить функционирование нейросети на этапах обучения и эксплуатации.

Из теоремы об отображении практически любой функции с помощью многослойной нейросети следует, что обучаемая нами нейронная сеть **в принципе** способна сама подстроиться под любые данные с целью минимизации суммарной квадратичной ошибки. Чтобы этого не происходило при обучении нейросетей используют следующий способ проверки сети. Для этого обучающую выборку еще перед началом обучения разбивают случайным образом на две подвыборки: обучающую и тестовую. Обучающую выборку используют собственно для процесса обучения, при этом изменяются веса нейронов. А тестовую используют в процессе обучения для проверки на ней суммарной квадратичной ошибки, но при этом не происходит изменение весов. Если нейросеть показывает улучшение аппроксимации и на обучающей, и на тестовой выборках, то обучение сети происходит в правильном направлении. Иначе может снижаться ошибка на обучающей выборке, но происходить ее увеличение на тестовой. Последнее означает, что сеть «переобучилась» и уже не может быть использована для прогнозирования или классификации. В этом случае немного изменяются веса нейронов, чтобы вывести сеть из окрестности локального минимума ошибки.

Самоорганизующиеся карты. Самоорганизующиеся карты (Self Organizing Maps - SOM) это одна из разновидностей нейросетевых алгоритмов. Основным отличием данной технологии от рассмотренных нами ранее нейросетей, обучаемых по алгоритму обратного распространения, является то, что при обучении

используется метод обучения без учителя, то есть результат обучения зависит только от структуры входных данных. Нейронные сети денного типа часто применяются для решения самых различных задач, от восстановления пропусков в данных до анализа данных и поиска закономерностей, например, в финансовой задаче.

Основы самоорганизующихся карт.

Алгоритм функционирования самообучающихся карт представляет собой один из вариантов кластеризации многомерных векторов. Примером таких алгоритмов может служить алгоритм ближайших средних (с-means). Важным отличием алгоритма SOM является то, что в нем все нейроны (узлы, центры классов) упорядочены в некоторую структуру (обычно двумерную сетку). При этом в ходе обучения модифицируется не только нейрон-победитель, но и его соседи, но в меньшей степени. За счет этого SOM можно считать одним из методов проецирования многомерного пространства в пространство с более низкой размерностью. При использовании этого алгоритма вектора, схожие в исходном пространстве, оказываются рядом и на полученной карте. SOM подразумевает использование упорядоченной структуры нейронов. Обычно используются одно- и двумерные сетки. При этом каждый нейрон представляет собой n -мерный вектор-столбец w_n , где n определяется размерностью исходного пространства (размерностью входных векторов). Применение одно- и двумерных сеток связано с тем, что возникают проблемы при отображении пространственных структур большей размерности (при этом опять возникают проблемы с понижением размерности до двумерной, представимой на мониторе).

Обычно нейроны располагаются в узлах двумерной сетки с прямоугольными или шестиугольными ячейками. При этом, как было сказано выше, нейроны также взаимодействуют друг с другом. Величина этого взаимодействия определяется расстоянием между нейронами на карте.

При реализации алгоритма SOM заранее задается конфигурация сетки (прямоугольная или шестиугольная), а также количество нейронов в сети. Некоторые источники рекомендуют использовать максимально возможное количество нейронов в карте. При этом

начальный радиус обучения (*neighborhood* в англоязычной литературе) в значительной степени влияет на способность обобщения при помощи, полученной карты. В случае, когда количество узлов карты превышает количество примеров в обучающей выборке, то успех использования алгоритма в большой степени зависит от подходящего выбора начального радиуса обучения. Однако в случае, когда размер карты составляет десятки тысяч нейронов, время, требуемое на обучение карты, обычно бывает слишком велико для решения практических задач, таким образом, необходимо достигать допустимого компромисса при выборе количества узлов.

Перед началом обучения карты необходимо проинициализировать весовые коэффициенты нейронов. Удачно выбранный способ инициализации может существенно ускорить обучение и привести к получению более качественных результатов. Существуют три способа инициирования начальных весов:

1. инициализация случайными значениями, когда всем весам даются малые случайные величины;
2. инициализация примерами, когда в качестве начальных значений задаются значения случайно выбранных примеров из обучающей выборки;
3. линейная инициализация. В этом случае веса иницируются значениями векторов, линейно упорядоченных вдоль линейного подпространства, проходящего между двумя главными собственными векторами исходного набора данных. Собственные векторы могут быть найдены, например, при помощи процедуры Грама-Шмидта.

Обучение карт состоит из последовательности коррекций векторов, представляющих собой нейроны. На каждом шаге обучения из исходного набора данных случайно выбирается один из векторов, а затем производится поиск наиболее похожего на него вектора коэффициентов нейронов. При этом выбирается нейрон-победитель, который наиболее похож на вектор входов. Под похожестью в данной задаче понимается расстояние между векторами, обычно вычисляемое в евклидовом пространстве. После того, как найден нейрон-победитель, производится корректировка весов нейро-сети. При этом вектор, описывающий нейрон-

победитель, и векторы, описывающие его соседей в сетке, перемещаются в направлении входного вектора.

Использование искусственных нейронных сетей в медицине используется в основном в двух направлениях: синтеза классификационных решающих правил для решения диагностических задач (включая прогнозирование возможных состояний больного при различных терапевтических процедурах и технологиях лечения) и прогнозирования значений регистрируемых показателей, характеризующих функционирование как определенной физиологической или функциональной системы так и организма в целом. Разумеется задача прогнозирования качественно разрешается в экстраполяционном временном интервале.

Для решения задачи классификации используются различные пакеты программного обеспечения: Statistica, Matlab Simulink. Neurosolution, специализированные средства для решения определенных задач.

Рассмотрим примеры синтеза нейронных сетей в универсальных оболочках.

1. В пакете STATISTICA:

Для решения задач классификации используется *Мастер решения задач*.

Первый шаг: открывается подготовленный файл данных; ряд переменных (столбцов) являются значениями параметров, категориальная переменная (последний столбец) обозначает диагностический класс.

Второй шаг: открывается окно «*Тип задачи*» («*Problem Type*»); указывается тип задачи и нажимается кнопка «*Next*».

Третий шаг: в окне выбирается зависимая переменная (диагностического класса) и нажимается кнопка «*Next*».

Четвертый шаг: в окне выбираются входные (независимые) переменные; после нажатия кнопки «*Next*» *Мастер решения* автоматически разобьет выборку на обучающую, контрольную и тестовую подвыборки, выделив их черным, синим и красным цветом, соответственно, и производит перемешивание наблюдений для обеспечения репрезентативности подвыборок.

Пятый шаг: на экране появляется окно «*Длительность поиска*» («*Duration of Design Process*»), в котором задается длительность поиска: быстрый, средний, полный, ограниченный по времени.

Шестой шаг: в открывшемся окне «*Сохранение сетей*» («*Saving Networks*») определяется способ сохранения сетей, например: сохранить сети с лучшим качеством решения, максимальное число сохраняемых сетей и т.д.; затем, в следующем открывшемся окне указываются опции представления результатов.

Седьмой шаг: после нажатия кнопки «*Finish*» STATISTICA производит вычисления и предоставляет итоговый результат в виде таблицы - например, если был выбран на предыдущем шаге режим «сохранить 10 сетей с лучшим качеством», то на экране появятся следующие столбцы: – *Type* – указывается тип сетей, *RBF* – указываются радиальные базисные функции, *Linear* – линейные, *MLP* – многослойный персептрон, *ошибка*, *входы*, *скрытые*, *Perfomance* – качество (указаны доли правильно классифицированных диагностических состояний) – определяется по контрольному подмножеству (доля правильно классифицированных наблюдений – диагностическая эффективность).

Лучшая сеть отмечается знаком «*». Для удаления из набора сетей неудовлетворительного качества используется правая кнопка мыши и команда «*Удалить*» («*Delete*»).

Выделенная нейронная сеть делается активной с помощью команды всплывающего меню «*Выбрать*» («*Select*»).

Если набор нейронных сетей заполнен, то программа *ST Neural Networks* должна определить, какие из имеющихся сетей заменяются вновь создаваемые. После нажатия кнопки *Options-Опции* в диалоговом окне *Редактор набора сетей – Network Set Editor*. На экране появиться окно *Параметры набора сетей – Network Set Options*, в котором задается максимальное количество сетей в наборе (по умолчанию – 30). Установив необходимые значения параметров набора сетей нажимается кнопка *Закреть – Close*.

Для исследования степени информативности входных переменных в обученной сети применяют *анализ чувствительности* – для этого используют команду *Чувствительность – Sensitivity* из

выпадающего меню. Программа строит таблицу, в которой указывается чувствительность сети по отношению к каждой переменной: ранг, ошибка и отношение. Показатели чувствительности определяются отдельно для обучающего и контрольного набора (подвыборки) наблюдений. В качестве столбцов в окне выступают переменные исходного файла данных.

2. Классификация с помощью персептрона в пакете MatLab.

Классификация состоит из этапов: ввод данных, обучение сети, классификация. Для оценки качества классификации (диагностики) первоначально результаты обследования (и-или результаты мониторинга за состоянием) биообъекта разделяются на две подвыборки: обучающая и тестовая. Если руководствоваться принципом «золотого сечения», то соотношение размеров указанных выборок -0,62:0,38.

Рассмотрим пример 1, реализованный в пакете MatLab 7 SP 2 + Simulink 6, для обучения **персептрона**:

```
>> % одному из двух классов
>> P=[-0.5 -0.5 +0.3 - 0.1;-0.5 +0.5 -0.5 +1.0];
>> T=[1 1 0 0];
>> plot (P,T);
>> % графическое представление исходных данных
>> % создание персептрона
>> % с указанием границ изменений и одним нейроном
>> My_net=new([1 1; -1 1], 1); % инициализация персептрона
>> % организация цикла адаптивной настройки персептрона
>> % с показом графика линии разделения классов
>> % while (sse(E))
    [My_net, Y,E]=adapt(My_net, P, T);
    Linehandle=plotpc(My_net.IW{1}, My_net.b{1});
    Drawnow;
    End;
```

В приведенной программе исходными данными являются входные векторы с указанием их принадлежности к одному из двух классов – с индексами 0 и 1. P – матрица, столбцы которой соответствуют 4-м входным векторам, T – вектор, элементы которого указывают на принадлежность вектора к нулевому или первому классу.

Итоговый результат отображает обучающие вектора и разделяющую их линию, формируемую обученным персептроном. Для проверки качества созданного классификатора подадим, например, на вход обученного персептрона контрольный вектор $p = \{0.4; 0.5\}$ и проанализируем реакцию сети. Моделирование иллюстрирует следующий программный фрагмент.

```
>> p=[0.4; 0.5];
>> a = sim(My-net,p);
>> a
a=
0
```

Его работа показывает, что последовала реакция сети $a=0$, т.е., предъявленный контрольный вектор относится к нулевому классу, что является правильным (и графически подтверждается рисунком).

Построение графика реализуется следующим программным фрагментом:

```
>> plot(p,a);
>> hold on;
>> plotpv(P,T);
>> Linehandle=plotpc(My_net.IW{1}, My_net.b{1});
>> hold off
```

Для оценки диагностического качества здесь и далее с помощью статистических вычислений ошибок первого и второго рода классификации необходимо на контрольной выборке для ряда объектов выполнить анализ правильности срабатывания персептрона и рассчитать значения показателей диагностических чувствительности, специфичности и эффективности.

Рассмотрим пример 2, реализованный в пакете MatLab 7 SP 2 + Simulink 6, для обучения **сети со слоем Кохонена**:

Задание: Используя встроенные функции пакета нейронных сетей математической среды Matlab, построить нейронную сеть со слоем Кохонена, которая множество входных данных разделит на кластеры и выявит их центры. На обученную сеть подать новый входной вектор и определить, к какому кластеру он относится.

Для создания нейронной сети со слоем Кохонена воспользуемся встроенной в среду Matlab функцией *newc*:

```
X=[0 1; 0 1];
clusters=5;
points=5; %Задание количества точек в кластере
std_dev=0.01;
p=nngenc(X,clusters,points,std_dev);%Моделирование          ВХОДНЫХ
данных
h=newc([0 1;0 1],5,.1); % создание слоя Кохонена
h.trainParam.epochs=50; %Задание количества циклов обучения
h=init(h);
h=train(h,p);
w=h.IW{1}; % вывод графиков исходных данных и выявленных
центров кластеров
plot(p(1,:),p(2,:),'^r'),grid;
hold on;
plot(w(:,1),w(:,2),'ob');
xlabel('p(1)');
ylabel('p(2)');
A=0.6
B=0.5
p=[A;B];
plot(A,B,'^k');
y=sim(h,p) % Опрос сети
A =0.6000
B =0.5000
y = (2,1)
```

Результат работы программы отображается в командном окне.

Предъявленный вектор отнесен ко второму кластеру. Теперь данный алгоритм применим к реальной (хоть и элементарной) задаче кластеризации. На вход нейронной сети будем подавать данные весоростовых показателей людей и попробуем выявить три класса (кластера): нормальный весоростовой показатель; избыток веса; недостаток веса. Программный код выглядит следующим образом:

```
%входные данные (первая строка матрицы - рост; вторая - вес)
```

```

p=[175 180 182 175 183 176 183 176 183 176 175 180 178 180 178 182
178 182 179 174 172 179; 70 75 100 99 42 48 76 72 40 45 92 96 70 69
95 90 79 82 80 50 96 91] %создаем НС Кохонена с 3 кластерами
(нормальный весоростовой показатель, избыток веса и недостаток
веса)
h=newc([0 200;0 100],3,.1);
h.trainParam.epochs=500; %Задание количества циклов обучения
h=train(h,p);
w=h.IW{1};
plot(p(1,:),p(2,:),'^r');
hold on;
plot(w(:,1),w(:,2),'ob');
xlabel('Rost');
ylabel('Ves');
% Задание нового входного вектора
%Опрос сети
A=181
B=65
p=[A;B];
plot(A,B,'+r')
y=sim(h,p)
A =181
B = 65
y = (2,1) 1

```

Результат работы программы представлен в командном окне: $y=(2,1)$ 1. Предъявленный вектор отнесен ко второму кластеру.

3. Теперь рассмотрим использование *самоорганизующей карты* на примере двумерных векторов. Используя самоорганизующиеся карты, двумерные векторы разбить на кластеры и выявить их центры, затем подать на вход самоорганизующей карты новый вектор и определить кластер, к которому он относится.

```

P=rand(2,100); %Задание случайных двухмерных входных
векторов
figure(1);
hold on;
plot(P(1,:),P(2:,:),'+r') %визуальное изображение входных векторов

```

```

%Создание НС с 3*4 нейронами
%По умолчанию функция TFCN = 'hextop', то есть нейроны
располагаются в узлах двумерной сетки с шестиугольными
ячейками
net=newsom([0 1;0 1],[3 4]);
net.trainParam.epoch=1 %Задание числа циклов настройки
net=train(net,P); % настройка сети
A=0.5
B=0.3
p=[A;B]; % Задание нового входного вектора
plot(A,B,'^k') %прорисовка на рисунке входного вектора (черный
треугольник)
figure(2)
plotsom(net.iw{1,1},net.layers{1}.distances)
a=sim(net,p) % опрос сети
A = 0.5000
B = 0.3000
a = (5,1) 1

```

4. Результат работы программы представлен в командном окне: $a = (5,1) 1$ / Предъявленный вектор отнесен к пятому кластеру.

Порядок выполнения работы.

1. Изучить теоретический материал и принципы построения искусственных нейронных сетей, включая язык программирования в MatLab 7 SP 2 + Simulink 6.

2. Сформировать выборки исходных данных по Таблице 1, характеризующие результаты общего анализа крови. Выборки состоят из:

- для пакета STATISTICA - матрицы, включающей три показателя крови (согласно варианта задания – определяется преподавателем) и идентификаторов состояния («здоров», «болен»), трех матриц типа «показатель», «состояние»;

- для MatLab 7 SP 2 + Simulink 6 - трех матриц типа «показатель», «состояние».

3. Построить нейронные сети в пакетах STATISTICA и MatLab 7 SP 2 + Simulink 6 (или иных версиях MatLab) – персептрон, сеть Кохонена, самоорганизующие карты для всех сформированных в п.2 матриц.

4. По результатам п.3 сравнить качество работы сетей – в качестве показателя качества использовать диагностическую эффективность.
5. По зафиксированным скрин-шотам работы с программными инструментариями (представляются в отчете по выполнению лабораторной работы) и результатами п.4 сделать выводы о перспективности применения полученных сетей для диагностического процесса.
6. Оформить отчет.

Контрольные вопросы:

1. Что понимается под кластеризацией?
2. Для чего применяются искусственные нейронные сети?
3. Каким образом классифицируются искусственные нейронные сети?
4. Для решения каких задач предназначены искусственные нейронные сети?
5. В чем заключается закон обучения Кохонена?
6. Опишите встроенные операторы Matlab для кластеризации.
7. Зачем используются самоорганизующиеся карты? Чем отличаются сети Кохонена от SOM?
8. Как устроен персептрон?
9. В чем заключается принцип обратного распространения ошибки в искусственных нейронных сетях?
10. Что такое – «скрытые слои» искусственной нейронной сети?
11. Как проверяется диагностическое качество работы искусственной нейронной сети?
12. Перечислите основные недостатки и преимущества применения искусственных нейронных сетей в диагностическом процессе?
13. В чем состоят отличия искусственных нейронных и иммунных сетей?

Приложение

Таблица 1. Данные для построения искусственной нейронной сети.

Больные					
№ п.п.	эритроциты	гемоглобин	Цветовой показатель	моноциты	СОЭ
1	4,6	142	0,93	2	9
2	4,2	115	0,82	2	16
3	3,4	107	0,94	2	8
4	5	170	1,02	4	2
5	3,6	105	0,88	2	22
6	5,3	108	0,61	2	6
7	4,16	132	0,95	4	10
8	4,3	145	1,01	3	4
9	5	160	0,96	1	4
10	3,1	108	1,05	4	12
11	4	124	0,93	3	10
12	5,1	164	0,96	6	12
13	4,25	138	0,97	4	26
14	4,3	150	1,05	9	9
15	4,8	153	0,96	2	9
16	3,6	114	0,95	1	14
Здоровые					
1	4,2	137	0,98	3	3
2	4,05	125	0,93	2	7
3	3,9	125	0,96	9	5
4	3,85	120	0,94	5	5
5	4,2	134	0,96	8	4
6	3,65	117	0,96	3	9
7	4,4	146	0,995	4	3
8	4	122	0,92	2	10
9	4,7	150	0,96	3	6
10	4,6	144	0,94	6	3
11	4,26	140	0,99	6	3
12	4,2	140	1	6	6
13	4,5	140	0,93	9	1
14	4,5	150	1	4	4
15	4,4	148	1,01	10	2
16	4,4	143	0,98	8	4

ПРАКТИЧЕСКАЯ РАБОТА № 10. РАСЧЕТ КРИТЕРИЕВ КАЧЕСТВА ДИАГНОСТИЧЕСКОГО ПРОЦЕССА

Цель работы. Овладение навыками оценки вычисления типовых критериев качества классификации биологических объектов.

Краткие теоретические сведения.

В биомедицинских исследованиях основополагающую роль является правильное соотнесение биообъекта или процесса к определенному классу. Указанное соотнесение называется классификацией или диагностикой.

В конечном итоге диагностика осуществляется путем применения определенных решающих правил. В случае возможности формализации синтезируются базы знаний соответствующих автоматизированных систем поддержки принятия решений (АСППР).

АСППР диагностики состояния биообъекта находят все большее применение в медицинской практике на различных этапах лечебно-диагностического процесса, профилактике и скрининге заболеваний.

Для определения классификационных (диагностических) способностей системы поддержки принятия решений на экзаменационной выборке с заранее определенными кластерами (так называемо, «обучение с учителем») осуществляется сбор статистического материала, по которому определяются показатели качества диагностической системы – применения заложенных в базе знаний СППР решающих правил.

Диагностика определенной нозологической группы проводится либо относительно некоторого класса условно здоровых людей (класс А) и класса, характерного для определенной нозологии (класс Б). (В случае дифференциальной диагностики осуществляется сравнение между различными нозологиями – в этом случае под «классом А» понимается одна из нозологий, под «классом Б» - другая).

По результатам тестирования составляется таблица вида:

Таблица Распределения результатов диагностики

Обследуемые	«золотой стандарт» (истина)		Всего
	Болен	здоров	
Болен (положительный результат теста)	Истинно-положительный результат a	Ложно- положительный результат b	a+b
Здоров (отрицательный результат теста)	Ложно-отрицательный результат c	Истинно- отрицательный результат d	c+d
Всего	a+c	b+d	a+b+c+d

В качестве показателей качества, характеризующих статистическую достоверность медицинской экспертной системы, при первичной оценке качества ее работы, выбираются: диагностическая чувствительность (ДЧ), диагностическая специфичность (ДС), прогностическая значимость положительных ($ПЗ^+$) и отрицательных ($ПЗ^-$) результатов испытаний и диагностическая эффективность (ДЭ).

Указанные показатели качества рассчитываются в соответствии со следующими выражениями:

$$\begin{aligned}
 ДЧ &= \frac{a}{a+c} ДС = \frac{d}{d+b} ДЭ = \frac{a+d}{a+b+c+d} \\
 ПЗ^+ &= \frac{a}{a+b} ПЗ^- = \frac{d}{c+d}
 \end{aligned}$$

где: a – истинно положительный результат равный количеству пациентов из класса заболеваний Б правильно классифицируемых;
 b – ложноположительный результат равный количеству относительно здоровых людей класса А ошибочно отнесенных экспертной системой к классу Б;

c – ложноотрицательный результат равный количеству людей из класса Б отнесенных экспертной системой к классу А;

d – истинно отрицательный результат равный количеству людей из класса А правильно классифицированных экспертной системой.

Заметим, что соотношение $\frac{a+b}{c+d}$ существенно влияет на значения показателей качества.

Порядок выполнения работы

1. Изучить теоретический материал.
 2. Согласно данным приложения №2 лабораторной работы № 2 синтезировать диагностическое решающее правила вида: «ЕСЛИ большее количество значений показателей входит в доверительный интервал, ТО пациент принадлежит к данному классу, В ПРОТИВНОМ СЛУЧАЕ он принадлежит другому классу». Рассчитайте показатели качества. Сделайте выводы.
 3. Изменяя значения a, b, c, d случайным образом в диапазоне [30, 60] порядка 20-30 раз, определите значения критериев качества и постройте и проанализируйте графики зависимости указанных показателей от величины $\frac{a+b}{c+d}$. (построение осуществите на одной плоскости). Сделайте выводы.
 4. Оформите отчет, включающий результаты выполнения работы и краткие ответы на контрольные вопросы.
 5. Составьте аннотацию не менее 2 информационных источников, найденных в сети интернет по тематике лабораторной работы (объемом 150-350 слов).
- Примечания: для расчетов и их графической интерпретации используйте любые офисные средства.

Контрольные вопросы

1. Что такое диагностический процесс?
2. Какие решающие правила применяются в автоматизированных системах поддержки принятия решений?
3. Как определяются критерии качества диагностического процесса?
4. В чем заключается семантическая нагрузка показателей качества диагностического процесса?
5. Каким образом значения показателей качества связаны с ошибками первого и второго рода?

Библиография

1. Доказательная медицина для всех /URL: <http://medspecial.ru/>
2. Доказательная медицина. Презентация /URL: <http://www.myshared.ru/slide/181227/>

3. Лекции по медицинской информатике. /URL: http://kingmed.info/lektsii/Meditsinskaya_informatika_i_biostatistika/lecture_772/Dokazatelnaya_medsina_alternativa_medsine_mneniy
4. Разработка критериев оценки качества оказания медицинской помощи в дневном стационаре в педиатрии /URL: <http://cyberleninka.ru/article/n/razrabotka-kriteriev-otsenki-kachestva-okazaniya-medsinskoy-pomoschi-v-dnevnom-statsionare-v-pediatrici>
5. Рассчитать диагностическую чувствительность и специфичность теста. /URL: <http://allrefs.net/c49/3t4ln/p1/>
6. Ступаков, И. Н. Доказательная медицина в практике руководителей всех уровней системы здравоохранения [Текст] / под ред. В. И. Стародубова. - М.:МЦФЭР, 2006. - 448 с.
7. [Триша Гринхальх](#). Основы доказательной медицины Издательство [ГЭОТАР-Медиа](#), 2015. – 336 .

ПРАКТИЧЕСКАЯ РАБОТА № 11. ЭКСПЕРТНАЯ СИСТЕМА ДИФФЕРЕНЦИАЛЬНОЙ ДИАГНОСТИКИ ЗАБОЛЕВАНИЙ

Цель работы: изучение принципов работы, проектирования и овладение навыками эксплуатации экспертной системы диагностики заболеваний.

Краткие теоретические сведения

Успехи врачевания в первую очередь определяются качеством распознавания болезней. В связи с этим правомочен афоризм: «Правильная диагностика – хорошее лечение».

Непрерывный научно-технический прогресс во всех научных областях, в том числе и в медицине, дает для расширения диапазона диагностических возможностей врача множество современных методов обследования больного. Современные методы исследования болезни позволяют более глубоко и детально изучить их, способствуют открытию новых патологических состояний, дают возможность уточнить этиопатогенетические механизмы их возникновения. Однако качество диагноза определяется не только открытием и внедрением инструментально-технических и других методов распознавания болезней в клиническую медицину, но и умением практических врачей мыслить. С увеличением информации о болезнях усложняется процесс мыслительной деятельности врача, повышаются требования к логической обоснованности, своевременности, завершенности и точности установления диагноза. Проблема удовлетворения современным требованиям, предъявляемым к диагнозу, является самой главной, определяющей задачей врача, которая реализуется с помощью методической части клинической медицины – диагностики - науки о распознавании болезней.

- **Диагностика** – раздел медицинской науки изучающей методы распознавания болезней в процессе исследования, наблюдения больного и мышления врача с целью установления диагноза

Решая задачу распознавания болезни, врач имеет несколько возможностей, использование которых обеспечивает высокое качество диагностики:

- основательный уровень владения методами и техникой врачебного исследования;

- внедрение в клиническую медицину новых современных методов исследования;
- совершенствование и углубление знаний о признаках болезни и их диагностической ценности;
- совершенствование мыслительных операций с выявленными признаками болезни научно обоснованными способами диагностики;
- совершенствование и разработка методов диагностических решений (создание диагностического алгоритма, схем эвристического поиска, диагностических тестов и др.).

Анализ клинической картины начинается с процесса обследования больного. Важно понимать, что обследование – отнюдь не механическое, а творческое занятие. Его творческий компонент неразрывно связан с глубиной характеристики симптома. Группировка и классификация симптомов являются основными моментами анализа клинической картины. Процедура систематизации подразумевает группировку симптомов по системам органов. В этом процессе проверяется полнота обследования больного и вырабатывается привычка к строгому, последовательному обследованию и изложению обнаруженных фактов.

Компоненты синдрома – симптомы, полученные при сборе анамнеза и физикальных исследований. Синдром синтезируется из симптомов, с целью локализовать зону поражения и выявить патологию. Например, установленное при обследовании системы дыхания локализованное притупление перкуторного звука, бронхиальное дыхание и усиление бронхофонии свидетельствуют об уплотнении легочной ткани. Важнейшая предпосылка успеха диагноза – глубокий патогенетический анализ клинической ситуации.

Дифференциальная диагностика – это сравнение клинической картины заболевания у данного конкретного больного со всеми заболеваниями, имеющими сходную клиническую картину, для исключения всех болезней, кроме одной.

Это сравнение должно проводиться не хаотично, не стихийно, а по строгим правилам и в определенном порядке. Иными словами, мыслительный процесс при дифференциальной

диагностике должен подчиняться такой же строгой схеме, как, например, проведение физикального исследования или анализ электрокардиограммы. Практически все авторы, занимавшиеся этой проблемой сходятся, с небольшими отличиями, в том мнении, что проведение дифференциальной диагностики состоит из следующих пяти этапов:

- группировка симптомов для выделения синдромов;
- выделение ведущего синдрома;
- генерация диагностической гипотезы;
- построение дифференциально-диагностического ряда;
- исключение синдромно-сходных заболеваний и формулировку окончательного диагноза.

Рассмотрим последовательность действий на каждом этапе проведения дифференциальной диагностики.

Первый этап – группировка симптомов для выделения синдромов, был достаточно описан выше, но хотелось бы подчеркнуть, что при выделении синдрома следует оценить его особенности. Подробная патогенетическая характеристика синдромов абсолютно необходима для плодотворной деятельности на следующих этапах.

На втором этапе ведущий синдром выделяется по тем же критериям, что и ведущий симптом. Главное здесь – плодотворность генерации диагностической гипотезы.

На третьем этапе формируется диагностическая гипотеза. Диагностическая гипотеза – это, в сущности, рабочий диагноз, который создается методом аналогии на основе синдромного анализа.

Характерными чертами диагностической гипотезы являются:

- неполнота аргументации;
- уникальность диагностической гипотезы.

Неполнота аргументации означает что рабочий диагноз формулируется у постели больного на базе данных анамнеза и физикального обследования. Следует иметь в виду, что как анамнестическое, так и физикальное обследование, несмотря на стереотипизм их реализации, не предусматривают некоторых вопросов, приемов, методов, которые имеют существенное значение для диагностики относительно редких заболеваний. Например, проведение провокационных проб для диагностики

кровооточивости петехиально-пятнистого типа, оценка ширины пространства между относительной и абсолютной тупостью сердца для верификации гидроперикарда и т.д.

Под уникальностью диагностической гипотезы в данном случае, имеется в виду то, что диагностическая гипотеза генерируется в процессе работы с каждым больным. Она может звучать одинаково, при работе с довольно большим количеством больных, например «гипертоническая болезнь», но аргументация в пользу этого предположения каждый раз создается заново.

Считается, что в процессе генерации диагностической гипотезы возможен ряд ошибок логического порядка:

- ошибка поспешного обобщения;
- ошибка в умозаключении по аналогии;
- ошибка ложной последовательности.

Ошибкой поспешного обобщения является учет диагностической гипотезой не всех, а лишь некоторых клинических данных. Ошибка в умозаключении по аналогии происходит, когда обобщаются процессы, имеющие разный патогенез. Ошибка ложной последовательности возможна, если умозаключение движется по пути «после этого, значит вследствие этого», тогда как далеко не всегда последовательное возникновение симптомов означает наличие между ними причинно – следственной связи.

На четвертом этапе строится дифференциально - диагностический ряд, что требует от врача широкой эрудиции с одной стороны и навыков системного мышления с другой. Следует обратить особое внимание, что дифференциально - диагностический ряд - не простое перечисление заболеваний, имеющих один отличительный признак (выделенный в качестве ведущего синдрома), на большие или меньшие группы. Разделение это осуществляется на патогенетической основе.

Такого рода внутрисиндромные классификации чрезвычайно полезны, т.к. позволяют в дальнейшем проводить дифференциальную диагностику, исключая заболевания не по одному, а целыми группами. Работа по составлению дифференциально-диагностического ряда сопровождается его написанием. Это служит гарантией того, что ни одно из

заболеваний не будет забыто в процессе дифференциальной диагностики.

На пятом этапе осуществляется процесс исключения синдромно - сходных заболеваний в два этапа. На первом этапе производятся межгрупповые сопоставления и исключение всех групп заболеваний, кроме одной. На второй фазе происходит сопоставление заболеваний внутри оставшейся группы и исключение всех болезней, кроме одной.

Определяются пять основных принципов дифференциальной диагностики.

Первый принцип – принцип существенного различия.

Принцип существенного различия означает, что наблюдаемый случай болезни не принадлежит к сравниваемому с ним виду заболеваний, т.к. в клинической картине у данного больного отсутствует синдром, обязательно имеющий место при сравниваемой болезни, или группе болезней. Принцип существенного различия имеет и другую трактовку: заболевание не относится к сравниваемому с ним виду, т.к. у больного имеется синдром (симптом), никогда не встречающийся при сравниваемом заболевании.

Второй принцип – исключения через противоположность. Этот принцип можно сформулировать так: у больного имеется симптом, прямо противоположный тому, который закономерно выявляется при сравниваемой болезни.

Третий принцип – принцип несовпадения признаков – важнейший и наиболее часто используемый. В данном случае речь идет о существенном сходстве симптома и в то же время – о его отличительных характеристиках при различных заболеваниях.

Четвертый принцип – индуктивного дифференциального диагноза -предусматривает постоянную работу в библиотеке. По прочтении соответствующей предполагаемому заболеванию литературы неизбежно возникает потребность повторить обследование больного с фиксацией внимания на специфических проявлениях болезни.

Пятый принцип – проверки диагноза. В зависимости от случая, проверка диагноза может быть либо очень короткой (в самых

простых случаях она не нужна), либо продолжительной, трудной для врача и больного. Иногда приходится двигаться от простых методов к все более и более сложным, если простые и необременительные способы диагностики на момент их использования не дают определенных результатов.

Применение методики дифференциальной диагностики по одному ведущему синдрому плодотворна, но иногда, особенно при сложных, полисиндромных клинических картинах моно-синдромный дифференциальный диагноз не приводит к успеху. Диагностическая гипотеза (рабочий диагноз) не оказывается ни подтвержденной, ни отвергнутой. В этой ситуации все приходится начинать сначала, выделяя другой синдром в качестве основного. При безуспешности этой попытки следует перейти к дифференциальной диагностике по третьему синдрому и т.д.

Вообще при полисиндромных процессах иногда целесообразно взглянуть на вещи более широко: все многообразие клиники может быть обусловлено не одним заболеванием, а их сочетанием.

После сбора анамнеза, проведения физикального исследования с целью проверки диагностических версий, врач использует дополнительные специальные диагностические тесты, применяя лабораторные и инструментальные методы исследования. Диагностический тест необходим для подтверждения или исключения болезни.

В зависимости от цели выделяют специальные методы исследования – инвазивные и неинвазивные. Выделяют также следующие методы исследования :

- функциональные методы, оценивающие ту или иную функцию органа или системы;
- индикаторные, указывающие на наличие какого-либо патологического процесса или состояния (воспаление, цитолиз, иммунные нарушения и т. д.);
- визуализирующие методы, позволяющие оценить макроструктуру органа;
- морфологические — прижизненно оценивающие микроструктуру органа.

При дополнительном обследовании больного необходимо учитывать чувствительность, специфичность и информативность

используемых методов. Информативность исследования – неколичественная характеристика метода, отражающая объем полезной информации, получаемой при применении данного метода в конкретной патологии и, следовательно, позволяющей судить, насколько данный метод необходим в конкретном случае.

Наибольшую ценность в диагностике имеют диагностические тесты высокой чувствительности и специфичности, позволяющие обнаруживать высокоспецифические и патогномоничные диагностические стигматы.

Оценка надежности различных диагностических методов (тестов) проводится в специальных научных исследованиях, где изучаемый метод сравнивают с эталонным. Однако, прежде чем определить критерии надежности результатов диагностического теста, необходимо оценить априорную вероятность наличия болезни.

Она особенно важна в тех случаях, когда чувствительность и специфичность применяемого метода низкая, а эталонный тест либо отсутствует, либо его трудно осуществить. Оценка априорной вероятности основывается на данных эпидемиологических характеристик болезни: степени ее распространенности, особенности ее течения в той или иной местности, факторах риска и степени соответствия клинической картины данного больного классическому описанию болезни. В практической деятельности врач, оценивая априорную вероятность болезни ограничивается приблизительным количественным ориентиром. Эта ориентировочная, приблизительная, количественная оценка априорной вероятности выводится на основании знаний литературных источников, подсказанной интуиции, клинических наблюдений и накопленного опыта.

Вероятность болезни, определяемая по результатам проведения диагностического теста, называется апостериорной вероятностью.

Результат теста может быть либо положительным (патология), либо отрицательным (норма), а заболевание может либо присутствовать, либо отсутствовать. Возможны четыре варианта толкования результатов теста – два истинных и два ложных. Правильный ответ – это положительный результат, если заболевание есть, или отрицательный, если заболевание отсутствует. Однако результаты могут быть и ложноотрицательными (болезнь есть, но тест ее

исключает), и ложноположительными (болезни нет, но тест ее подтверждает) (таблицы 1,2). Оценка точности теста основана на сравнении его результата с критерием, заслуживающим доверия истины, - эталоном. С этой целью определяют чувствительность и специфичность теста, предсказательную ценность положительного и отрицательного теста, а также диагностическую точность метода.

Таблица 1 - Варианты оценки результатов теста

Тесты	Болезнь есть	Болезни нет
Положительный	A - истинноположительный	B - ложноположительный
Отрицательный	C - ложноотрицательный	D - истинноотрицательный

Таблица 2 - Оценка точности результатов теста

Чувствительность теста	$A / (a + c)$ - доля истинно положительных тестов среди больных
Специфичность теста	$D / (b + d)$ - доля истинно отрицательных тестов среди здоровых
Предсказанная ценность положительного теста	$A / (a + b)$ = доля истинно положительных тестов среди всех положительных тестов
Предсказанная ценность отрицательного теста	$D / (c + d)$ = доля истинно отрицательных тестов среди всех отрицательных тестов
Диагностическая точность метода	$(a + d) / (a + b + c + d)$ = доля истинных результатов среди всех результатов теста

Поскольку точные способы установления диагноза почти всегда трудоемкие, дорогие и сопряжены с риском исследования, оправдано назначение более простых и доступных тестов, особенно на ранних этапах исследования больного. Вероятность положительного результата диагностического теста в присутствии болезни называется чувствительностью метода, а вероятность отрицательного результата в отсутствие болезни – его

специфичностью. При выборе диагностического теста врач должен принимать во внимание как чувствительность, так и его специфичность.

Чувствительность метода для выявления конкретного признака определяется соотношением количества больных с наличием признака по данным применяемого метода исследования и количества обследованных больных с действительным наличием этого признака, выраженным в процентном отношении. Если метод или тест обладает высокой чувствительностью, то он выявляет патологию у большинства больных с данным заболеванием (истинно положительные результаты) и лишь у незначительного количества больных дает ложноотрицательные результаты. Чувствительность признака при конкретном заболевании – показатель, отражающий частоту встречаемости признака при данной патологии. Специфичность метода для выявления конкретного признака – соотношение количества обследованных здоровых пациентов, у которых признак отсутствует, и общего количества здоровых пациентов, выраженное в процентном отношении. Таким образом, специфичность отражает частоту «ошибок» метода исследования у здоровых людей. Специфичность признака при данном заболевании – показатель, отражающий насколько частый признак бывает у людей, не имеющих этого заболевания. Чувствительный тест следует убрать, если есть риск пропустить опасную, но излечимую болезнь. Чувствительные тесты также рациональны на ранних стадиях диагностического поиска с целью сужения круга вероятных заболеваний].

Диагностический метод безупречен, если и чувствительность и специфичность равны 100 %. Однако таких совершенных методов исследования мало или они труднодоступны. Если проводится эталонный тест, при котором чувствительность и специфичность равны 100 %, определение априорной вероятности подозреваемой болезни почти не играет роли. И наоборот, чем меньше чувствительность и специфичность, тем значительнее предварительная информация о больном.

Таким образом, в одних случаях решающим фактором для интерпретации результатов диагностического теста будет как можно более точная оценка априорной вероятности, т. е.

вероятности еще до проведения диагностического теста, а в других определяющее значение в верификации болезни будет иметь эталонный тест или тест с высокой чувствительностью и его специфичностью.

При положительном результате диагностического теста апостериорная вероятность болезни идентифицирует предсказательную ценность

Из данных таблицы 3 видно, что апостериорная вероятность болезни значительно зависит от оценки априорной вероятности. Следовательно, при оценке эффективности положительных результатов для подтверждения диагноза и отрицательных – для его исключения необходимо учитывать вероятность подозреваемой болезни у обследуемого больного еще до проведения диагностического теста, особенно, если диагностические тесты несовершенны (низкой чувствительности и специфичности).

Таблица 3 - Пример диагностического теста

Априорная вероятность	2%		30%		70%	
Болезнь	Есть	Нет	Есть	Нет	Есть	Нет
Чувствительность и специфичность	ДЧ	ДС	ДЧ	ДС	ДЧ	ДС
Положительный тест	18	196	270	140	630	90
Отрицательный тест	2	784	30	560	70	210
Всего	20	980	300	700	700	300
Предсказанная ценность положительного теста	8,4% 18/(18+196)		65,85% 270/(270+140)		87,5% 630/(630+90)	

В таблице ДЧ и ДС – диагностические чувствительность и специфичность, соответственно.

Считается, что вероятность болезни тем выше, чем больше свидетельств в ее пользу. Поэтому, если результат диагностического теста сомнительный, необходимо полагаться на сумму доказательств. Но эти данные повышают диагностические возможности только в том случае, если назначаемые диагностические тесты дают независимые друг от друга информативные доказательства.

Экспертные системы дифференциальной диагностики предназначены для решения следующих основных задач: интерпретация, прогноз, диагностика, проектирование, планирование, обучение, наблюдение, управление.

Экспертные системы имеют в своем составе обширную базу данных – факты выбранной предметной области, а также базу знаний, в которой отражены профессиональные навыки и умения специалистов высокого уровня в данной области.

В качестве примера российской разработки можно привести комплекс РОФЭС (Регистратор Оценки Функционально-Эмоционального Состояния). Система состоит из комплекса датчиков и программного обеспечения, с помощью которых производится диагностика и анализ состояния организма. Она может осуществить подбор препаратов для конкретного человека, провести частотную и цветотерапию (с использованием обратной связи), определить гормональный профиль и риск опасных видов заболеваний. Таким образом, РОФЭС производит не только оценку, но и управление состоянием здоровья..

В качестве примера современных экспертных систем, использующихся в медицинских учреждениях, можно привести следующие:

1 Экспертная система ДИН. С целью повышения качества диагностики и эффективности лечения критических состояний у детей в Московском НИИ педиатрии и детской хирургии создана экспертная система ДИН, ориентированная на решение задачи распознавания неотложного состояния у детей в терминах синдрома или нескольких синдромов (точнее, характеризующих их состояний, которые отражают степень выраженности синдрома) при предъявлении признаков заболеваний, под которыми следует понимать анамнестические, клинические и лабораторные проявления. База знаний ЭС содержит описания 34 синдромов, которые включают 84 состояния. Для системы это список диагностических предположений-гипотез. База экспертных знаний врача-реаниматолога содержит более 1000 диагностических критериев и заключений о динамике развития неотложного состояния. С целью ускорения работы системы, ориентированной на угрожающие жизни состояния, все множество синдромов подразделено на 14 групп в соответствии с преобладающими нарушениями со стороны той или иной системы жизнеобеспечения, и каждая группа имеет свои дифференцирующие симптомы. Знания о синдромах представлены таким образом, что охватывают:

условие выдвижения гипотезы о возможности возникновения синдрома, которое является некоторой логической комбинацией симптомов; клиническую картину, то есть симптомы заболевания; необходимые и достаточные условия для подтверждения гипотезы; информацию о дополнительных синдромах, состоящих в некоторых отношениях с рассматриваемым:

- 1) причинно-следственные связи, предполагающие информацию о синдромах, которые могут быть причиной данного синдрома или, наоборот, являться его следствием;
- 2) временные связи, позволяющие как прогнозировать состояние ребенка, так и восстанавливать возможный анамнез болезни;
- 3) ассоциативные связи, дающие возможность учитывать, на фоне каких состояний может развиваться данный синдром, фоном для каких синдромов он может служить и с какими синдромами может быть совместим, то есть какие синдромы могут встречаться одновременно;
 - информацию о состояниях, взаимоисключающих друг друга;
 - сведения о дифференцируемых синдромах (синдромах-конкурентах).

Все знания о синдромах подразделяются на декларативные - для описания самого синдрома (клиническая картина, дополнительные синдромы) - и процедурные, указывающие на то, как использовать знания в процессе диагностики.

Функциональная схема ДИН соответствуй схеме классической ЭС: интерфейс, блок представления знаний, блок метазнаний, блок механизма логического вывода, блок рабочей области, блок пополнения и модернизации знаний.

2. Информационно-диагностическая система по наследственным болезням у детей «Диаген». Система «ДИАГЕН» - предназначена для консультативной помощи врачам при диагностике редко встречающихся и трудно распознаваемых заболеваний наследственной природы на долабораторном этапе обследования. База знаний системы включает сведения о 1200 моногенных и хромосомных болезнях, а модуль визуального представления данных - более 1000 фотографий больных с наследственными болезнями. Система включает три основных блока: диагностический, выдающий дифференциально-диагностический

ряд; справочник, который предоставляет полную информацию о признаках и синдромах, хранящихся в системе (ориентирован в первую очередь на врача-педиатра широкого профиля); архив, который обеспечивает хранение и повторное использование данных о диагностированных больных.

Система может быть использована при решении вопроса о предварительном (долабораторном) диагнозе при подозрении на моногенную и хромосомную патологию как справочная система, а также в учебном процессе.

Система "ДИАГЕН" реализована для IBM-совместимых персональных компьютеров. Программное обеспечение написано на языке Borland C++ и функционирует под MS DOS.

3 Экспертная система Мутант.

Система Мутант обеспечивает проведение первичного скрининга заболеваний с целью получения диагностической информации, необходимой для организации более углубленного обследования пациентов. Предлагаемый подход к дифференциальной диагностике основан на интервьюировании пациента по определенной схеме. Вначале пациенту предлагается ответить на первичный (основной) опросник, включающий паспортные и некоторые антропометрические данные (масса тела, пульс, артериальное давление), личный и семейный анамнез, перечень основных жалоб, обычно приводящих пациента к врачу-терапевту. На основании компьютерной обработки ответов пациента делаются заключения двух типов:- о наличии риска определенных заболеваний (например, ишемической болезни сердца, сахарного диабета, хронического бронхита, рака легких и др.). В этом случае при отсутствии жалоб указывается, какие дополнительные исследования следует провести;- при наличии тех или иных жалоб пациент адресуется к соответствующему дополнительному (специализированному) опроснику, содержащему углубленную проработку каждого симптома по различным характеристикам, например, точная локализация болей, их характер, иррадиация, продолжительность, сопутствующие явления, условия прекращения и курирования. Для этого по каждому разделу предлагается набор альтернативных ответов. При обработке полученных ответов дополнительно используются такие элементы основного опросника

как, например, сведения о поле, возрасте, анамнезе, факторах риска и других жалобах. На этом основании формируется диагностическое заключение в форме синдрома (например, сердечная или легочная недостаточность, синдром малоабсорбации и др.) или определенной нозологической единицы с указанием степени его вероятности в соответствии с количеством информационных синдромов, полученных от больного (50-75%, 76-95%, >95%). Врач должен учитывать, что диагностическое заключение носит рекомендательный характер. В дальнейшем врач может провести более углубленное обследование пациента, заведя в систему данные объективного осмотра пациента (осмотр головы, живота, ног и т.д.) и результаты дополнительных методов исследования (анализы крови, мочи, кала, мокроты и т.д.). В результате система уточняет раннее выставленные диагнозы.

Экспертная система Мутант состоит из следующих подсистем:

- подсистема консультаций (ПК);
- подсистема пополнения знаний (ППЗ).

Моделью представления знаний описываемой ЭС является продукционная система.

Фактологическими знаниями (базой фактов) ЭС Мутант являются:

- карта первичного опроса (КПО) и специализированные опросники (СО);
- список диагностируемых заболеваний;
- список анализов, рекомендуемых системой;
- список анализов, которые могут быть сделаны в данном медицинском учреждении;
- список врачей - специалистов, рекомендуемых системой;
- список врачей - специалистов, которыми располагает данное медицинское учреждение.

Процедурные знания (базы знаний) для системы содержат правила двух типов:

- правило 1-го типа - Если (условие), то (медицинское заключение);
- правило 2-го типа - Если (условие), то (список опросников). Здесь условием является список выбранных экспертом вопросов из карты первичного опроса и одного из специализированных опросников.

Медицинское заключение включает в себя:

- список предполагаемых диагнозов;

- список рекомендуемых системой анализов;
- список рекомендуемых системой врачей-специалистов.

Список опросников содержит номера специализированных опросников, которые необходимо предложить пациенту для уточнения медицинского заключения.

База знаний состоит из двух частей, в каждой из которых хранятся правила только одного типа. Каждое правило представлено одним термом. Все термы, соответствующие правилам, относящимся к одному опроснику, и группируются в одну цепь. Каждое правило при занесении в БЗ автоматически получает уникальный номер, который не может быть модифицирован, и удаляется из БЗ только при удалении этого правила.

Механизм вывода ЭС Мутант заключается в сопоставлении ответов пациента на вопросы карты первичного опроса и специализированных опросников с условной частью правил из БЗ. Поскольку задачей данной системы является выявление всех возможных предварительных заключений, то для ее решения применяется прямой вывод. При этом в зависимости от отношения количества ответов пациента, совпавших с условной частью правила 1-го типа, к общему количеству условий в данном правиле, система различает три степени достоверности диагнозов:

- диагноз маловероятен (50-75%);
- диагноз вероятен (76-95%);
- диагноз весьма вероятен (более 95%).

Для работы со знаниями врач-эксперт использует две подсистемы ведения БЗ, каждая из которых предназначена для работы со знаниями 1-го или, соответственно, 2-го типа.

Каждая из подсистем предлагает эксперту следующие виды работы:

- ввод нового правила в БЗ;
- коррекция правила в БЗ;
- удаление правила из БЗ;
- просмотр БЗ.

Применением базы знаний системы Мутант было ее совместное использование с базой данных медицинских карт. Данный подход позволяет не только хранить результаты сеанса работы системы с пациентом, но и дает возможность врачу занести в БД свои собственные наблюдения, результаты анализов и т.п.

В этих системах реализован интерактивный подход в использовании экспертной системы, когда в результате ответов пациента непосредственно формируется заключение. Однако авторы сочли возможным использование экспертной системы Мутант и в скрытом режиме в качестве контролирующей системы [11, 29].

4. Экспертная система FIRSTConsult.

Ресурс FIRSTConsult (прежнее название - PDxMD) является частью MD Consult Clinical Knowledge System. Эта система научно обоснованной электронной клинической информации о первичной медицинской помощи предназначена для улучшения процесса принятия решений специалистами посредством предоставления специализированных диагностических средств и непрерывно обновляемых данных о последних тенденциях в области диагностики, вариантах ведения пациентов и результатах лечения пациентов. Она обеспечивает эффективное принятие решений, сочетая подход на основе передовой практики с гибкостью, необходимой для учета индивидуальных различий между пациентами и предпочтений врача; ускоряет клинические исследования с целью постановки диагноза и оптимизирует результаты, предлагая планирование научно обоснованного лечения и повышая эффективность использования времени и ресурсов; снижает риск ошибок и улучшает клиническую документацию, предоставляя всеобъемлющую и постоянно обновляемую информацию. Ресурс FIRSTConsult доступен в трех форматах: он-лайн, для карманных компьютеров и в печатном. Формат FIRSTConsult Online обновляется еженедельно и состоит из инструмента «Интерактивная дифференциальная диагностика», научно обоснованных Файлов по медицинским состояниям, Файлов с информацией о пациентах, Справочных центров и Процедурных файлов. Формат FIRSTConsult Handheld включает в себя инструмент «Интерактивная дифференциальная диагностика» и Файлов по медицинским состояниям, обновляемых ежеквартально. Файлы по медицинским состояниям охватывают более 450 медицинских состояний, которые чаще всего наблюдаются врачами первичной медицинской помощи. Каждое состояние включает в себя самую необходимую краткую информацию из других

разделов; справочную информацию, включая коды ICD 9 и эпидемиологию; диагностику, включая дифференциальную диагностику, признаки и симптомы, предлагаемые вопросы для истории болезни, ссылки на литературу, полезную при принятии решений медицинскими учреждениями, и клинические тесты, которые можно провести; терапию с кратким перечнем терапевтических вариантов, подробными сведениями о лекарствах и проблемах пациентов и медицинских учреждений; результаты, включая эффективность терапии, прогноз и осложнения; профилактику с указанием факторов риска, рекомендаций в отношении образа жизни и сохранения здоровья, и скрининг; ресурсы, включая основную литературу и ссылку на информацию для пациента; а также научные данные, увязанные с библиографическими ссылками и клиническими руководствами.

Файлы по дифференциальной диагностике позволяют пользователям ознакомиться с таблицей диагностических алгоритмов, в которой перечисляются потенциальные диагнозы с разбивкой по возрасту и распространенности для более чем 330 признаков и симптомов. Потенциальные критические состояния показаны красным. Выбор диагноза выводит на экран краткую сводку, включая клиническую картину начала заболевания, соотношение процентов среди мужчин и женщин, особенности у разных этносов, характер, географические особенности, клинический картину течения заболевания и сопутствующие заболевания со ссылкой на файлы по медицинским состояниям. Файлы с информацией о пациентах содержат раздаточные материалы для пациентов, написанные в формате вопросов и ответов. Эти материалы имеют ссылку на файлы по медицинскому состоянию и могут быть адаптированы под конкретного пациента. Раздаточные материалы доступны различных языках в различных форматах и предназначаются для пациента, у которого было диагностировано данное состояние, либо для общего сведения.

Рассмотрим обобщенный алгоритм работы диагностической экспертной системы.

В алгоритме учитывается, что система медицинской диагностики должна позволять на основе неполных данных ставить диагноз с

некоторой точностью, и выдавать, насколько точно она определила болезнь.

Входные данные можно разбить на два больших блока. Первый блок - это данные, поступающие из пользовательского интерфейса. Второй блок - это содержимое базы знаний, заполненной экспертом. База знаний должна храниться на жестком диске в виде четырех файлов. Структура базы знаний будет описана более подробно в одном из последующих пунктах.

Таким образом, входными данными для разработки алгоритма программы системы медицинской диагностики являются:

ответы пользователя на вопросы системы;

база данных с описаниями симптомов;

база данных с описанием болезней;

таблица соответствий между болезнями и симптомами;

таблица «весов» (вероятностей) симптомов для болезней;

Выходными данными экспертной системы является диагноз, построенный на основе наблюдаемых симптомов и базы знаний о болезнях. Этот диагноз появляется на экран дисплея ПВМ.

В процессе работы система предлагает несколько сгенерированных рабочих версий окончательного диагноза, и по требованию пользователя «отсеивает» лишние гипотезы, которые имеют вес, меньший, чем некоторое значение, заранее заданное разработчиком программы или опытным пользователем.

Например, если в процессе работы сформировалось 4 версии с вероятностями от 71 до 93 %: диагноз №1 – 93%; диагноз № 2 – 86%; диагноз № 3 – 79%; диагноз № 4 – 64% и порог уверенности, заданный разработчиком программы – 80%, то система показывает все версии, вероятности которых больше или равны 80% (например, их 2 - Диагноз №1 и Диагноз №2 из сформированных).

При заполнении базы знаний необходимо привлекать эксперта.

База знаний представляет собой совокупность следующих баз данных:

база данных с описаниями диагнозов.

база данных с описаниями симптомов.

таблица соответствий симптомов и диагнозов.

таблица весов симптомов для диагнозов.

Вес – это относительная величина значимости каждого симптома для конкретного диагноза.

Кроме того, в экспертной системе предусмотрен порог уверенности – число, заранее определенное пользователем, означающее максимальный вес диагноза в данном случае, ниже которого гипотезы не рассматриваются.

Например, структуры данных представлены в Paradox 7.0 показаны в таблицах 1–2.

Работу с этими таблицами обеспечивают DBE и DBD, поставляемые в комплекте с Borland Delphi для работы с базами данных распространенных форматов.

Таблица 1 – Таблица диагнозов

№	Наименование поля	Тип	Длина	Краткое описание
1	Num	Numeric		Идентификационный номер диагноза
2	Name	Character	15	Наименование диагноза
3	FileName	Character	12	Имя файла с рекомендациями

Таблица 2 – Таблица симптомов

№	Наименование поля	Тип	Длина	Краткое описание
1	Num	Numeric		Идентификационный номер
2	NameS	Character	15	Наименование симптома
3	FileName	Character	12	Имя файла с рекомендациями

Все значения полей таблицы соответствий имеют тип numeric. Число в пересечении колонок является «указателем» на тот диагноз, на который указывает определенный симптом, либо на тот симптом, на который указывает определенный диагноз. Всем значениям полей таблицы весов назначается тип numeric. Таблица имеет размерность $[n,m]$, где n – число диагнозов, m – число симптомов. Число в пересечении колонок является весом данного симптома для данного диагноза.

Работа программы начинается с функции инициализации, которая ответственна за присвоение данным начального значения. После чего управление получает процедура «Главного меню». Далее, в

зависимости от действий пользователя, происходит вызов одной из ниже перечисленных процедур:

работа с базами знаний;

непосредственный запуск экспертной системы;

настройки;

контекстно-зависимая справочная система.

На каждом шаге алгоритма у системы предусмотрен «откат», т.е. пользователю предоставлена возможность вернуться в предыдущее состояние системы, нажав на программную кнопку «Назад». На вход системы изначально поступает неполная информация в виде данных, введенных пользователем с клавиатуры. Вследствие чего система не может однозначно поставить диагноз.

Для решения проблемы выбора диагноза используется следующий алгоритм:

1. Сбор предварительной информации. При входе в систему пользователю предоставлен список всех симптомов, имеющихся в базе. Напротив выборочных пунктов списка симптомов пользователь должен выставлять «галочки» отмечая симптомы, которые он наблюдает. При нажатии пользователем программной кнопки «Далее» система переходит к следующему пункту.

2. Составление первоначального списка диагнозов, к которым подходит данный набор симптомов. После ввода пользователем симптомов, система анализирует, к каким диагнозам принадлежат данные симптомы по следующему алгоритму:

а) создается список пустой первоначальный список диагнозов

б) открывается таблица соответствий;

в) для каждого i -го симптома, $i=1..m$, где m —максимальное число симптомов по таблице соответствий циклом делаем полный перебор всех элементов (диагнозов), стоящих в столбце данного симптома и определяем предварительный диагноз;

г) если диагноз уже имеется в списке диагнозов, то переходим к пункту е);

д) если элемент таблицы равен 0, то диагноз в список диагнозов не включается;

е) переходим к проверке следующего симптома.

ж) если все симптомы (i) перебраны, то первоначальный список болезней сформирован;

з) закрываем таблицу соответствий.

Каждый симптом может принадлежать сразу нескольким диагнозам, поэтому в первоначальный список диагнозов следует уточнить, используя таблицу весов. И следующим пунктом обобщенного алгоритма будет уточнение полученных данных.

3. Уточнение первоначального списка диагнозов.

Имея начальный список диагнозов, система осуществляет их дифференциацию. Самыми распространенными методами логического вывода являются прямая цепочка рассуждений (прямой вывод) и обратная цепочка рассуждений (обратный вывод). В основном, при решении задач диагностики используется обратный вывод. Можно сказать, что обратный вывод более эффективен, когда пользователь должен выбирать из набора возможных последствий в случае медицинской диагностики. В предлагаемой системе реализуется механизм смешанного вывода, который позволяет и прямой вывод от фактов к заключениям, и обратный – чтобы подтвердить или опровергнуть гипотезу. В процессе уточнения информации система, задавая пользователю вопросы, осуществляет «отсеивание» лишних гипотез, имеющих малый вес. Для просчета веса гипотез система должна открыть таблицу весов. В таблице весов указывается вес данного симптома для данного диагноза на основе априорных знаний эксперта. Порог уверенности заранее задается в настройках приложения и использованием интуитивно понятного интерфейса.

4. Принятие окончательного решения.

В процессе предыдущих шагов выявляется несколько версий окончательного результата, которые система должна распределять по порядку возрастания вероятности того или иного диагноза. Вероятности диагнозов считаются по таблице весов. Предлагается следующий алгоритм подсчитывания веса:

- а) выбирается диагноз из списка диагнозов, сформированного на предыдущих этапах;
- б) система просматривает, какие симптомы из списка симптомов имеют отношение к данному диагнозу;
- в) суммируются веса, определенные по таблице весов, всех симптомов, имеющих отношение к данному диагнозу;
- г) фиксирование конечного веса диагноза;

д) после подсчета весов всех диагнозов выбирается диагноз, имеющий максимальный вес;

е) система выбирает те диагнозы, оценки вероятности которых выше «порога».

Для конечного подсчета вероятности предлагается суммирование всех элементов таблицы весов имеющих отношение к болезни и нормировка всех конечных вероятностей с целью «укладывания» их в промежуток от 1 до 100 с целью получения окончательного ответа в процентах.

5. Представление конечного результата.

Система предоставляет врачу диагнозы, которые были выбраны в предыдущем пункте в порядке процентного убывания на экран монитора ПВМ и предоставлять пользователю возможность корректировать базу знаний на основе новых экспертных знаний и своего опыта.

На основании рассмотренного обобщенного алгоритма разработана экспертная система дифференциальной диагностики внутренних болезней.

База данных состоит из четырех таблиц .

Заполнение базы данных производится экспертом в разработанном приложении. Заполнению подлежат таблицы диагнозов, таблицы клинико-лабораторно-инструментальных показателей и расстановка соответствующих весов в таблице весов. Пример заполнения представлен на рисунке 7.

Симптом: Боль в подложечной области: тупая
Показатель: Острый холецистит
Вес: 0

Диагноз	Симптом	Вес
Острый холецистит	анализ мочи: пиурия	0
Острый холецистит	болезненная диффузная припухлость	0
Острый холецистит	болезненность при пальпации	3
Острый холецистит	боли в животе: в верхней части с иррадиацией влево	0
Острый холецистит	боль в животе: спастический характер	20
Острый холецистит	боль в животе: строго локализованная, связанная с приемом	0
Острый холецистит	боль в животе: умеренная боль в правой подвздошной обл	0
Острый холецистит	боль в подложечной области: тупая	0

Диагнозы: Инфаркт миокарда, Остеоартроз, **Острый аппендицит**, Острый панкреатит, Острый холецистит, Пиелонефрит

Клинико-лабораторно-инструментальные показатели: снижение аппетита, Боль в грудной клетке: продолжительность > 20-30, Боль в суставе: спонтанные боли, усиливающиеся п, УЗИ: внутрисосудистое исследование выявило обес, УЗИ: лоцируются конкременты в чашечно-лоханоч, УЗИ: признаки гидронефроза

Диагноз: Острый аппендицит
Клинико-лабораторно-инструментальный показатель: Edt1

Описание диагноза: Острый аппендицит проявляется умеренной болью, которая начинается либо непосредственно в правой подвздошной области, либо дебютирует в эпигастрии и перемещается в правую подвздошную область спустя 2-3 часа от начала заболевания

Рисунок 7 - Пример заполнения базы данных

Пример работы с базой данных представлен на рисунке 8.

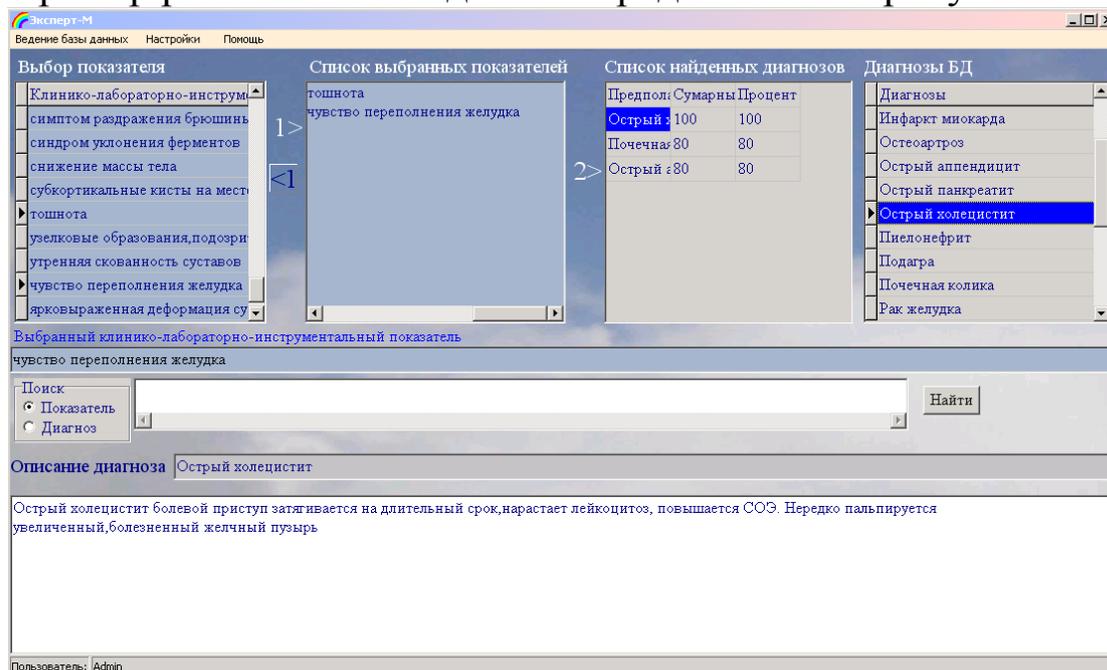


Рисунок 8 - Пример работы с базой данных.

Обобщенная структура экспертной системы представлена на рисунке 9.

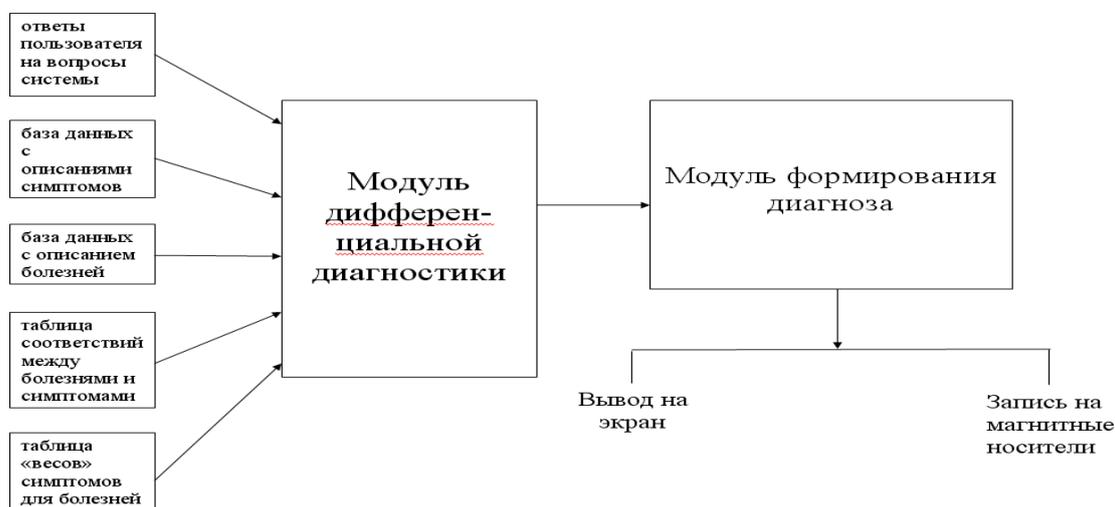


Рисунок 9 - Обобщенная структура экспертной системы

В модуль дифференциальной диагностики поступают данные из комплекса программ, реализующих диалог пользователя с ЭС как на стадии ввода информации, так и получения результатов (интерфейс пользователя) а также данные из базы данных с описаниями симптомов и болезней и данные из таблиц весов

симптомов и болезней и соответствия между болезнями и симптомами. Модуль дифференциальной диагностики - программа, моделирующая ход рассуждений эксперта на основании знаний, имеющихся в БЗ.

Итогом работы модуля дифференциальной диагностики является совокупность возможных диагнозов, соответствующих всем выбранным пользователем симптомам.

Далее в модуле формирования диагноза происходит анализ и отсеивание диагнозов суммарный вес которых, ниже порога чувствительности.

Полученные результаты выводятся на экран или записываются на магнитные носители.

Инструкция пользователя по эксплуатации экспертной подсистемы для дифференциальной диагностики внутренних болезней.

При запуске приложения открывается окно, представленное на рисунке 10.

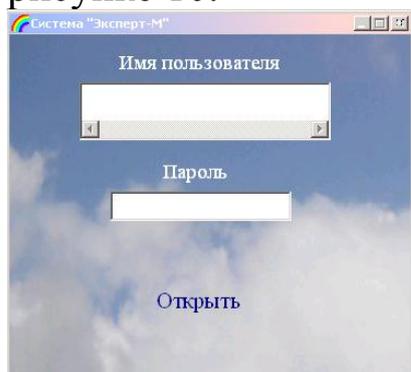


Рисунок 10 – Окно входа в систему

При вводе имени пользователя и пароля открывается основное рабочее окно программы, представленное на рисунке 11.

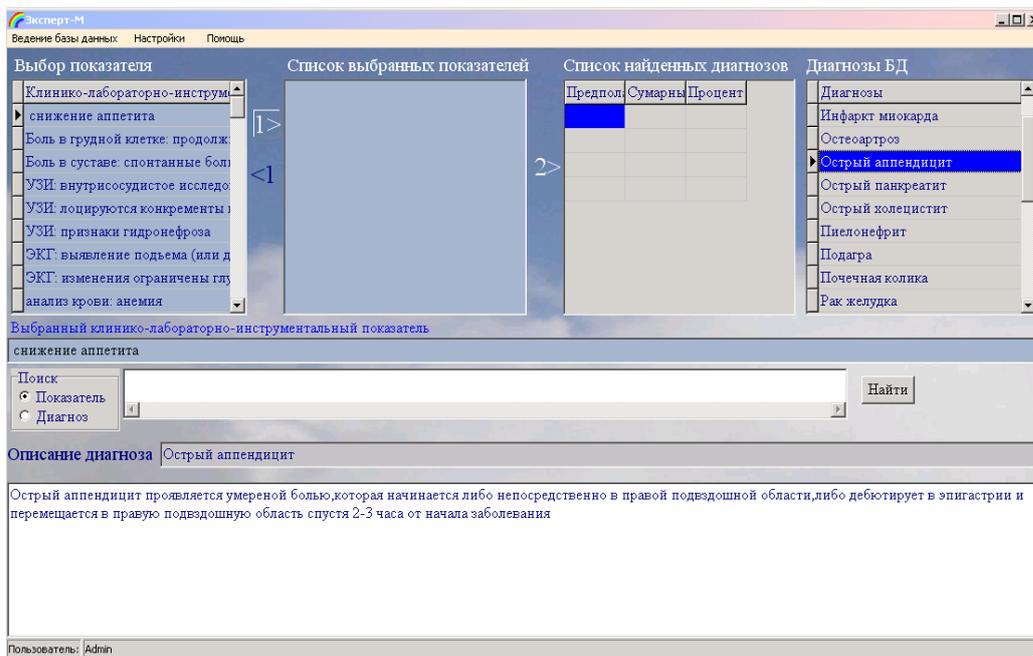


Рисунок 11 – Рабочее окно программы

В начале необходимо выбрать клиничко-лабораторно-инструментальный показатель таблицы «Выбор показателя» и нажать программную кнопку «1>». При этом выбранный показатель отобразится в списке выбранных показателей. Аналогичным образом следует выбрать все необходимые показатели, соответствующие собираемому анамнезу. Далее следует нажать программную кнопку «2>». При этом в списке найденных диагнозов отобразятся предполагаемые диагнозы, суммарный вес показателей в диагнозе и процентное соотношение суммарных весов показателей к максимальному суммарному показателю. Описание полученных предполагаемых диагнозов можно просмотреть при нажатии левой кнопки мыши при выделенном выбранном диагнозе таблицы «Диагнозы БД». Для удобства пользователя организован поиск выбранных диагнозов или показателей по базе данных. Для поиска необходимо ввести в поисковое окно искомый текст, пометить в окне поиска категорию поиска – диагноз/ показатель, и нажать программную кнопку «Найти». При этом найденная текстовая информация отразится в таблице диагнозов или симптомов в зависимости от категории поиска. Пользователь может изменить порог чувствительности, выбрав подпункт «Параметры системы» пункта меню «Настройки». При этом должно появиться окно «Параметры системы»,

представленное на рисунке 12. В открывшемся окне пользователь может изменить порог чувствительности, и нажать программную кнопку «Запомнить параметры».

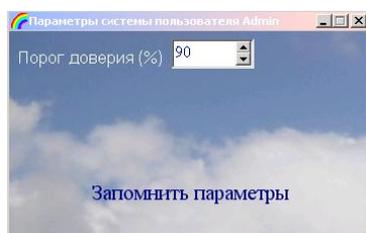


Рисунок 12 – Окно редактирования параметров системы

Для ведения базы данных необходимо входить в систему под именем администратора с паролем администратора и доступом администратора (полный доступ). В этом случае будут доступны пункты меню «Ведение база данных» и «Пользователи». Администратор при необходимости может добавить нового пользователя для работы с программой. Для этого необходимо выбрать подпункт «Пользователи» пункта меню «Настройки». При этом в открывшемся окне, представленном на рисунке 13, администратор может ввести в таблицу пользователей имя нового пользователя, пароль, доступ (полный доступ – 1, ограниченный – 0) и имя файла настроек очередного пользователя.

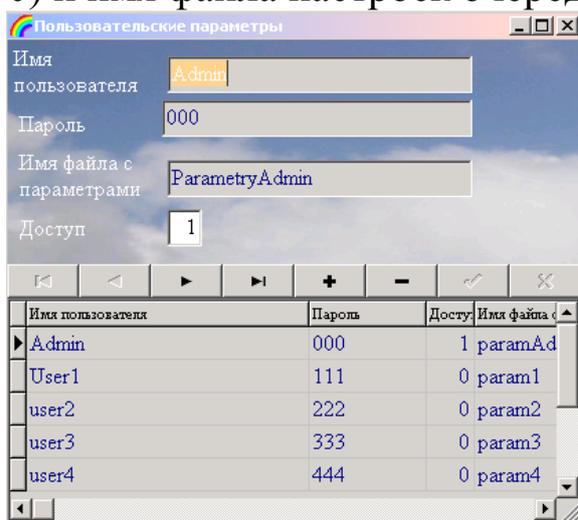


Рисунок 13 – Окно редактирования параметров пользователей

Обновлять базу данных рекомендуется эксперту. Для этого эксперт должен войти в программу под именем администратора с паролем администратора и доступом администратора. Далее следует

выбрать пункт меню «Ведение баз данных», после чего откроется окно, представленное на рисунке 14.

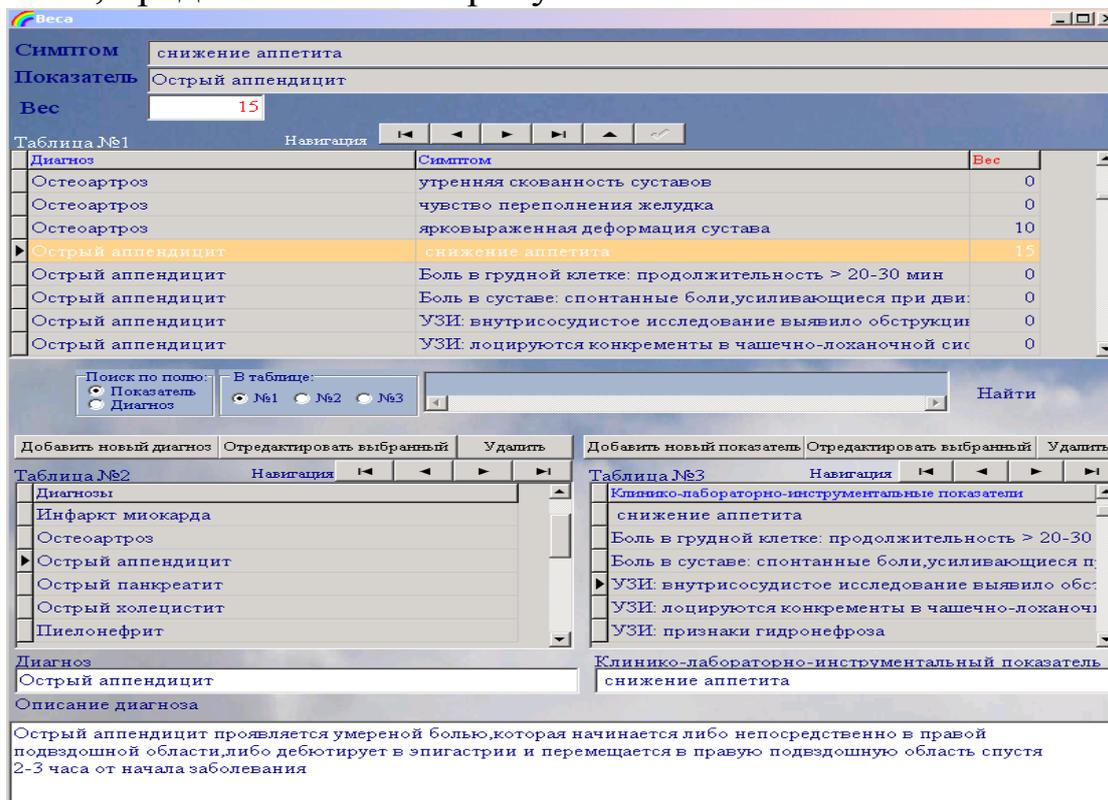


Рисунок 14 – Окно редактирования базы данных

Описываемую болезнь следует ввести в поле, озаглавленное «Диагноз», после чего нажать программную кнопку «Добавить новый диагноз». Введенный диагноз при этом отобразится в таблице №1 и таблице №2. Причем следует отметить, что в таблице №1 новый диагноз отобразится не один раз, а в комбинации со всеми клинико-лабораторными показателями, введенными ранее, для возможности последующей расстановки весов. Чтобы отредактировать диагноз, нужно выбрать его в таблице №2 путем нажатия левой кнопки мыши, произвести необходимые изменения в поле «Диагноз» и нажать программную кнопку «Отредактировать выбранный». Для удаления введенного ранее диагноза, его следует выбрать в таблице №2 и нажать программную кнопку «Удалить». Для краткого описания болезни, она выбирается в таблице №2, описание вводится в поле «Описание диагноза», после чего следует нажать программную кнопку «Отредактировать выбранный». Для введения симптомов болезней, а также результатов лабораторных исследований пациента следует использовать поле «Клинико-

лабораторно-инструментальный показатель», после чего нажать кнопку «Добавить новый показатель». При этом симптом отобразится в таблице №1 в комбинации с ранее введенными диагнозами и в таблице №3. Редактирование и удаление симптомов производится аналогично диагнозам. Для расстановки весов в таблице №1 нажатием левой кнопки мыши выбирается строка с нужной болезнью и симптомом, далее в поле «Вес» следует ввести число, соответствующее, по мнению специалиста, влиянию данного симптома на постановку рассматриваемого диагноза.

В экспертной подсистеме для дифференциальной диагностики внутренних болезней предусмотрена справочная система. Для ее вызова следует выбрать пункт меню «Помощь», после чего откроется окно, представленное на рисунке 15.

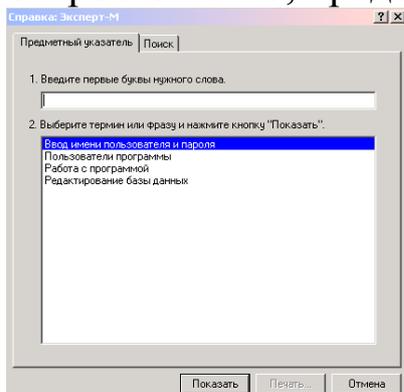


Рисунок 15 – Справочное окно

В открывшемся окне можно либо ввести искомый термин в поле под номером 1, либо выбрать из предложенных в поле под номером 2. После этого следует нажать программную кнопку «Показать». Например, при выборе пункта «Работа с программой» откроется справка, представленная на рисунке 16.

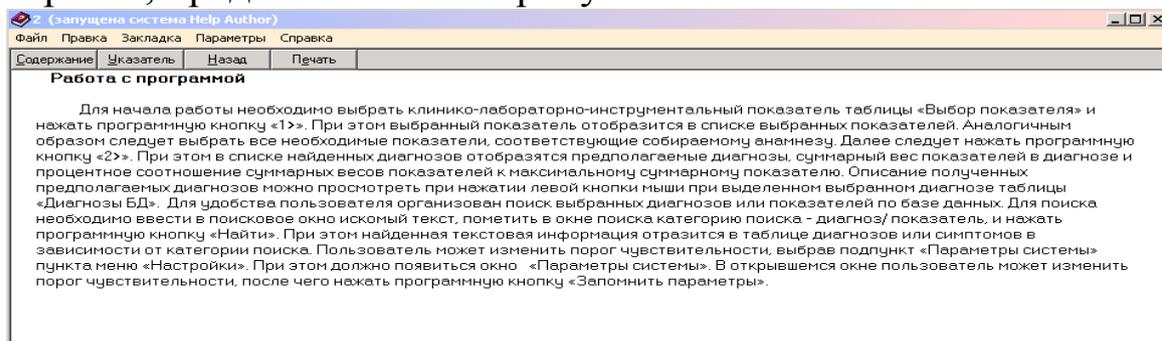


Рисунок 16 – Пример справочной информации.

Порядок выполнения работы:

1. Изучить теоретический материал.
2. С помощью поисковых систем найти любую медицинскую экспертную систему дифференциальной диагностики (например, по адресу: <http://ubertech.homedns.org:8083>).
3. В режиме опытной эксплуатации системы изучить: интерфейс системы, алгоритм эксплуатации системы, режимы функционирования, формы отображения выходной информации.
4. Ознакомиться через поисковые системы с существующими в настоящее время экспертными системами диагностического характера, построенными на принципах дифференциальной диагностики.
5. Под руководством преподавателя выбрать предметную область дифференциальной диагностики внутренних болезней.
6. Запустить оболочку экспертной системы дифференциальной диагностики внутренних болезней и экспериментальным путем освоить интерфейс системы.
7. Заполнить оболочку согласно выбранной предметной области.
8. Оценить качество созданной системы дифференциальной диагностики.
9. Оформить отчет (включая скрин-шоты интерфейсной части).

Контрольные вопросы:

1. В чем заключается функциональное назначение дифференциальной диагностики?
2. В чем заключаются основные принципы дифференциальной диагностики?
3. Перечислите наиболее используемые в настоящее время в России и за рубежом экспертные системы дифференциальной диагностики.
4. Какое влияние оказывают полисиндромальные процессы на формирование диагноза?
5. Как проверяется качество работы диагностической системы?
6. Что входит в типовой состав экспертной системы дифференциальной диагностики?

7. Какие функциональные модули (и в какой последовательности) входят в состав обобщенного алгоритма работы экспертной системы дифференциальной диагностики?

ПРАКТИЧЕСКАЯ РАБОТА №12. СРАВНИТЕЛЬНЫЙ АНАЛИЗ ИНФОРМАЦИОННЫХ ПОРТАЛОВ, ПОСВЯЩЕННЫХ СИСТЕМАМ АВТОМАТИЧЕСКОГО УПРАВЛЕНИЯ В МЕДИЦИНЕ

Цель: провести сравнительный анализ медицинских САУ, выделить недостатки и подчеркнуть достоинства.

Краткие теоретические сведения

Цели и задачи медицинского портала:

Построение единой базы данных по всем медицинским учреждениям, включая интеграцию с медицинскими информационными системами (МИС)

Обеспечение для пациентов доступа к информации о правах пациента, порядке оказания медицинской помощи, информация о клинике, услугах, специалистах и ценах, онлайн-запись на прием, и многое другое

Обеспечение качественной обратной связи с пациентами (сервис «обращения/ вопрос-ответ»)

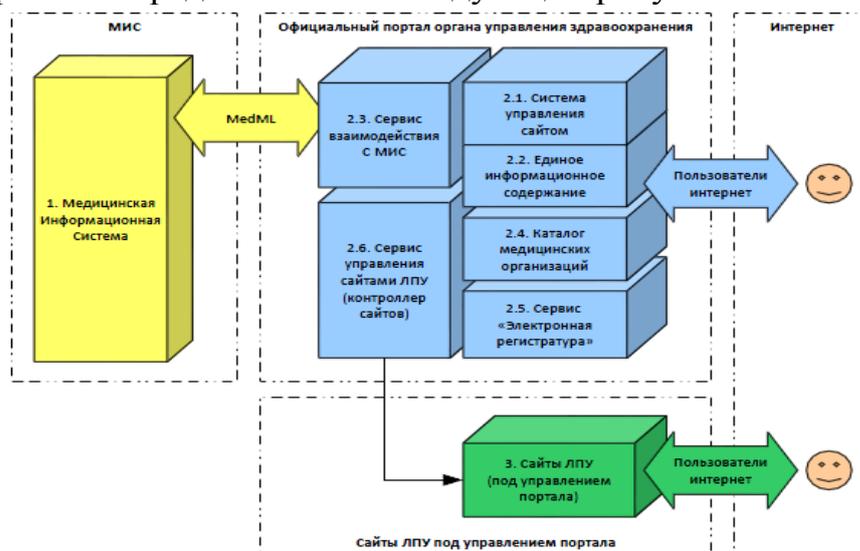
Автоматизация процесса записи на прием и управления расписанием врачей

Обеспечение работы сайта под высокими нагрузками

Создание мультисайтового проекта: основной сайт и подчиненные сайты медицинских учреждений или филиалов.

Архитектура медицинского портала

Архитектура портала как основного компонента комплексной веб-системы региона представлена на следующем рисунке:



На рисунке синим изображены составляющие решения «1С-Битрикс: Медицинский портал», желтым и зеленым - внешние по отношению к portalу подсистемы.

Сайты лечебно-профилактических учреждений

Концепция комплексной веб-системы здравоохранения предусматривает наличие центрального портала региона, а также сети сайтов подведомственных учреждений (ЛПУ). Согласно данной концепции портал обеспечивает возможность автоматизированного развертывания сайтов подведомственных ЛПУ. Общая логическая схема выглядит следующим образом:



Портал содержит каталог, в который могут быть включены все лечебно-профилактические учреждения и другие медицинские организации региона (государственные и коммерческие). В этом случае каталог сам по себе представляет большую ценность для населения, как источник актуальной информации об организациях

Каталог учреждений реализован с помощью модуля [информационные блоки](#), имеет рубрикатор учреждений с произвольной глубиной вложенности и возможность неограниченного количества атрибутов для описания каждой медицинской организации.

Базовые атрибуты организации (есть в типовой поставке):

наименование организации

адрес

контактные телефоны

контактный e-mail

адрес сайта (URL)

флаг участия в электронной регистратуре

подробное описание

Сайты подведомственных ЛПУ

Предоставить каждому ЛПУ возможность создавать себе сайт - означает потерять управление, получить множество сайтов на различных технологиях, с разными принципами подачи информации. Наиболее эффективно - централизованно предоставлять всем ЛПУ сайт на единой платформе с единой системой управления, системой разграничения прав доступа и обеспечению безопасности и централизованным хранилищем данных. В

таком случае центральный орган управления здравоохранением получит возможность контроля над любым сайтом ЛПУ, централизованное обновление и изменение информации на подчиненных сайтах, сбор необходимой информации со всей сети сайтов.

Можно выделить два способа развертывания и лицензирования подчиненных сайтов:

Самостоятельные сайты ЛПУ, размещенные на хостинге в сети Интернет и подключенные к порталу для управления.

Преимущества такого способа: полная изолированность и индивидуальность сайтов ЛПУ, сайты ЛПУ могут размещаться на разных хостингах

Из недостатков можно отметить сравнительно высокую стоимость, т.к. на каждый сайт требуется отдельная лицензия.

Сайты ЛПУ, размещенные на единой системе управления сайтом.

В этом случае все сайты будут иметь общую базу данных (отдельно от базы данных портала) и разделять ее между собой на базе прав доступа к контенту. Снаружи каждый сайт будет иметь собственное доменное имя (2-го или 3-го уровня), собственный шаблон дизайна, собственную структуру сайта - то есть для посетителя внутренняя архитектура остается прозрачной. Преимущества такого способа: простота развертывания каждого сайта (не нужен отдельный хостинг, не нужно с нуля разворачивать решение для каждого ЛПУ), низкая стоимость.

Недостатки: слабая изолированность сайтов (вследствие общей БД и "админки"), все сайты должны размещаться на едином хостинге, более сложное и ответственное управление (единая админка на все сайты, нужен специалист).

Управление сайтами ЛПУ (контроллер сайтов)

Подробная информация о функционале модуля "Контроллер сайтов" представлена в [соответствующем разделе](#).

Для обеспечения централизованного управления все сайты подключаются в контроллер сайтов. Для второго способа (см. выше) в контроллер подключается вся многосайтовая конфигурация 1С-Битрикс: Сайт медицинской организации (Базовый).

Контроллер обеспечивает: единую систему управления сайтами ЛПУ — управление контентом, правами доступа и пользователями на сайтах ЛПУ, выгрузка общих новостных материалов из портала на сайты ЛПУ, подключение неограниченного числа сайтов ЛПУ для управления, для каждого сайта могут быть установлены индивидуальные права доступа для управления информацией, позволяет выполнять управление любым подчиненным сайтом ЛПУ без дополнительной авторизации (при наличии авторизации с соответствующими правам на портале).

Сравнительная характеристика порталов, связанных с САУ в медицине

Наименование	Год создания	Краткая характеристика	Страна создания	Портал
1С	1991	<p>Фирма «1С» специализируется на разработке, дистрибьюции, издании и поддержке компьютерных программ делового и домашнего назначения.</p> <p>Комплексная автоматизация учреждений достигается использованием продуктов «1С:Медицина. Больница», «1С:Медицина. Поликлиника», «1С:Медицина. Больничная аптека», «1С:Медицина. Клиническая лаборатория» и др.</p>	Россия	1c.ru
Siemens Healthcare	1896	<p>Компания «Сименс», один из крупнейших в мире поставщиков энергоэффективных и экономящих ресурсы технологий, является главным разработчиком ветровых установок для других стран, ведущим поставщиком парогазовых турбин для выработки электроэнергии, лидером в поставке решений для передачи электроэнергии и первопроходцем в области инфраструктурных решений, средств автоматизации и программного обеспечения для промышленности. Кроме того, компания является ведущим поставщиком медицинского визуализационного оборудования (например, компьютерных и магнитно-резонансных томографов) и лидером в лабораторной диагностике и ИТ-решениях для лечебных учреждений.</p>	Германия	www.siemens.ru

Ал тэй	199 6	Флагманским продуктом ООО «Группа АЛТЭЙ» является Лабораторная информационная система АЛТЭЙ, в которой многолетний опыт в области автоматизации лабораторий реализован в новом качестве на передовых средствах разработки. Лабораторная информационная система (ЛИС) АЛТЭЙ предназначена для автоматизации технологического процесса и учета в лаборатории, включая штриховое кодирование образцов, автоматизацию хранения образцов, обмен информацией с анализаторами, внутрिलाбораторный контроль качества.	Россия	www.altey.ru
Ви так ор	200 4	ЗАО «Витакор» — авторитетная российская компания, с 2004 года осуществляющая широкий спектр услуг по информатизации регионального здравоохранения. Компания является лидером в разработке и внедрении информационных систем для фондов обязательного медицинского страхования и страховых медицинских компаний, а также медицинских организаций любой формы собственности.	Россия	www.vitacore.ru

Индивидуальное задание

Сравните указанные системы автоматизированного управления и сделайте вывод о приоритетности одной из указанных над другой (если это будет целесообразно).

№ варианта	Сравниваемые САУ
1	Нетрика, ООО и Flipbox
2	Polymedia и ВРМ «Робомед»
3	ХОСТ, Группа Компаний и САС Институт, ООО
4	InterSystems и АГФА, ООО

5	Брегис и ИНТЕРИН, группа компаний
6	Нетрика, ООО и Комплексные медицинские информационные системы, ООО
7	Лаборатория «Акросс-Инжиниринг», ООО и Polymedia
8	Мастер Лаб, ООО и Мик-Информ, НТЦ
9	Медицинские Информационные Технологии и САМСОН Групп, ООО
10	Комплексные медицинские информационные системы, ООО и Лаборатория «Акросс-Инжиниринг», ООО
11	ИНТЕРИН, группа компаний и Мастер Лаб, ООО
12	АГФА, ООО и Брегис
13	Медицинские Информационные Технологии и САМСОН Групп, ООО
14	Smart Delta Systems и СофтТраст, ООО
15	СП.АРМ и Мастер ЛАБ, ООО
16	СофтТраст, ООО и СП.АРМ
17	Smart Delta Systems и Брегис
18	ФОРС и СП.АРМ
19	АГФА, ООО и Алтэй
20	СофтТранс, ООО и ФОРС

Контрольные вопросы

Что такое портал?

Какие задачи преследуют, создавая медицинские порталы?

Опишите архитектуру портала медицинского САУ.

Какова роль блока «Сервис взаимодействия с МИС»?

Для чего используется блок «Сервис управления сайтами ЛПУ (контроллер сайтов)»?

Медицинская Информационная Система (МИС) является:

- а) внутренней системой по отношению к portalу, взаимодействие с которой осуществляется по стандарту MedML;
- б) внешней системой по отношению к portalу, взаимодействие с которой осуществляется по стандарту MedML;
- в) внутренней системой по отношению к portalу, взаимодействие с которой осуществляется по стандарту [Medtime](#);
- г) внешней системой по отношению к portalу, взаимодействие с которой осуществляется по стандарту [Medtime](#).

7. Одним из преимуществ сайтов ЛПУ, размещенные на единой системе управления сайтом является:

- а) простота развертывания каждого сайта (не нужен отдельный хостинг, не нужно с нуля разворачивать решение для каждого ЛПУ);
- б) полная изолированность и индивидуальность сайтов ЛПУ;
- в) сайты ЛПУ могут размещаться на разных хостингах.

8. Контроллер обеспечивает (выберите ответ, содержащий ошибочное утверждение):

- а) единую систему управления сайтами ЛПУ — управление контентом, правами доступа и пользователями на сайтах ЛПУ;
- б) выгрузку общих новостных материалов из портала на сайты ЛПУ;
- в) подключение ограниченного числа сайтов ЛПУ для управления;
- г) позволяет выполнять управление любым подчиненным сайтом ЛПУ без дополнительной авторизации (при наличии авторизации с соответствующими правам на портале).

ПРАКТИЧЕСКАЯ РАБОТА № 13. ИЗУЧЕНИЕ СТРУКТУРЫ МНОГОМЕРНЫХ МЕДИЦИНСКИХ ДАННЫХ

Цель работы: изучение методов анализа структуры многомерных данных обследования пациента.

Краткие теоретические сведения:

При анализе и прогнозировании состояния поведения объектов различной природы в условия изменяющегося внешнего воздействия и эволюционирования (в том числе внутреннего) окружающей среды часто сталкиваются с многомерностью их описания. Методы многомерного анализа - наиболее действенный количественный инструмент исследования процессов, описываемых большим числом характеристик [1, 4].

Классически выделяют две основные группы методов анализа структуры многомерных данных [3]:

1. визуализация данных: линейные методы снижения размерности, нелинейные отображения, многомерное шкалирование, заполняющие пространство кривые;
2. автоматическое группирование: факторный и кластерный анализ объектов и признаков, иерархическое группирование, определение «точек сгущения».

Наиболее часто структурный анализ многомерных экспериментальных данных применяется в слабоструктурированных проблемных областях (медицина, экология, экономика, педагогика и социология и т.п.).

Применение графики в исследовательских работах увеличивает скорость передачи информации и повышает уровень ее понимания, способствуя развитию таких важных для специалиста любой отрасли качеств, как интуиция, образное мышление. Воздействие интерактивной компьютерной графики (ИКГ) привело к возникновению нового направления в проблематике искусственного интеллекта, названного когнитивной компьютерной графикой.

Когнитивная графика - это совокупность приемов и методов образного представления условий задачи, которое позволяет либо сразу увидеть решение, либо получить подсказку для его нахождения [4]. Использование когнитивной графики дает

возможность пользователю, не анализируя большого количества информации, сделать определенные выводы. Информация может быть представлена когнитивным образом: сектором, гистограммой, крестом, кругом и т. д., цветами в качестве дополнительной когнитивной информации.

При автоматической группировки данных с целью дальнейшей классификации объектов применяется кластерный анализ. Кластер - объединение нескольких однородных элементов, которое может рассматриваться как самостоятельная единица, обладающая определёнными свойствами. Главное назначение кластерного анализа – разбиение множества исследуемых объектов и признаков на однородные, в соответствующем понимании группы или кластеры. Тем самым решается задача классификации данных и выявления соответствующей структуры в ней.

Методы кластерного анализа можно применять в самых различных случаях, начиная с простой группировкой согласно выбранному критерию. Основное достоинство кластерного анализа в том, что он позволяет производить разбиение объектов не по одному параметру, а по целому набору признаков. Пакета Statistica позволяет автоматизировать процесс кластеризации стандартными способами.

Порядок выполнения работы:

1. Изучить теоретические сведения о методах анализа структуры данных согласно указанным литературным источникам и самостоятельно на различных порталах.
2. В электронной таблице Excel сформировать таблицу данных. (Таблицы 4 или 5).
3. В электронной таблице Excel получить графики, отражающие: диаграммы каждого регистрируемого показателя в различных классах (состояниях); получить графики, отражающие зависимости между всеми возможными парами сочетаний показателей;
4. В электронной таблице Excel рассчитать матрицы парной корреляционной связи между показателями, построить граф, отражающий корреляцию между показателями в каждом классе, выявить системообразующий показатель (ядро графа) и рассчитать множественный коэффициент функциональных различий.

5. Используя возможности пакета Statistica (любой версии) получить: гистограммы изучаемых показателей, рассчитать доверительные интервалы, по которым проанализировать классификационную информативность показателя путем вычитания из единицы отношения пересечения значений показателя в различных классах к их объединению.

6. Сделать выводы по итогам разведочного анализа структуры данных.

В отчете должно быть представлено: таблица исходных данных, краткое описание основных методов анализа структуры данных, полученный количественный и графический материал проведенного анализа с указанием выполненных действий, выводы о классификационных возможностях анализируемых показателей, краткие ответы на вопросы.

Контрольные вопросы:

1. Что такое структура данных?
2. Какие используются формы и модальности для представления данных?
3. Что характеризуют статистические оценки в разведочном анализе?
4. Какие графические формы используются для представления структуры данных?
5. Какие универсальные программные инструментарии используются при анализе структуры данных?