

УДК 004.8
05.13.18 - Математическое моделирование,
численные методы и комплексы программ
(технические науки)

ИДЕНТИФИКАЦИЯ ЯЗЫКОВ ПРОГРАММИРОВАНИЯ НА ОСНОВЕ АСК- АНАЛИЗА

Луценко Евгений Вениаминович
д.э.н., к.т.н., профессор
Scopus Author ID: 57188763047
РИНЦ SPIN-код: 9523-7101
prof.lutsenko@gmail.com <http://lc.kubagro.ru>

Черкашин Андрей Романович
a.cherkashin43@gmail.com

*Кубанский государственный аграрный
университет имени И.Т.Трубилина, Краснодар,
Россия*

Целью данной работы является идентификация
языков программирования по наиболее часто
встречающимся в них командам, а также анализ их
сходств и различий.

КЛЮЧЕВЫЕ СЛОВА: АСК-АНАЛИЗ,
АВТОМАТИЗИРОВАННЫЙ СИСТЕМНО-
КОГНИТИВНЫЙ АНАЛИЗ, СИСТЕМА
«ЭЙДОС», ЯЗЫКИ ПРОГРАММИРОВАНИЯ

UDC 004.8
05.13.18 - Mathematical modeling, numerical methods
and program complexes (technical sciences)

IDENTIFICATION OF PROGRAMMING LANGUAGES BASED ON ASK ANALYSIS

Lutsenko Evgeniy Veniaminovich
Dr.Sci.Econ., Cand.Tech.Sci., professor
Scopus Author ID: 57188763047
RSCI SPIN-code: 9523-7101
prof.lutsenko@gmail.com <http://lc.kubagro.ru>

Cherkashin Andrey Romanovich
a.cherkashin43@gmail.com

*Kuban State Agrarian University named after
I.T.Trubilin, Krasnodar, Russia*

The aim of this work is to identify programming
languages by the most common commands in them, as
well as to analyze their similarities and differences.

KEYWORDS: ASK-ANALYSIS, AUTOMATED
SYSTEM-COGNITIVE ANALYSIS, "EIDOS"
SYSTEM, PROGRAMMING LANGUAGES

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	2
ЗАДАЧА 1: КОГНИТИВНАЯ СТРУКТУРИЗАЦИЯ ПРЕДМЕТНОЙ ОБЛАСТИ.....	6
ЗАДАЧА 2: ПОДГОТОВКА ИСХОДНЫХ ДАННЫХ И ФОРМАЛИЗАЦИЯ ПРЕДМЕТНОЙ ОБЛАСТИ	7
ЗАДАЧА 3: СИНТЕЗ И ВЕРИФИКАЦИЯ СТАТИСТИЧЕСКИХ И СИСТЕМНО-КОГНИТИВНЫХ МОДЕЛЕЙ И ВЫБОР НАИБОЛЕЕ ДОСТОВЕРНОЙ ИЗ НИХ ДЛЯ РЕШЕНИЯ ЗАДАЧ	13
ЗАДАЧА 4: РЕШЕНИЕ РАЗЛИЧНЫХ ЗАДАЧ В НАИБОЛЕЕ ДОСТОВЕРНОЙ МОДЕЛИ	20
Подзадача 4.1. Прогнозирование (диагностика, классификация, распознавание, идентификация).....	20
Подзадача 4.2. Поддержка принятия решений (SWOT-анализ)	22
Подзадача 4.3. Исследование моделируемой предметной области путем исследования ее модели.....	26
4.3.1. Когнитивные диаграммы классов	26
4.3.2. Агломеративная когнитивная кластеризация классов	27
4.3.3. Нелокальные нейроны и нелокальные нейронные сети	29
4.3.4. 3d-интегральные когнитивные карты	31
4.3.5. Когнитивные функции.....	31
4.3.6. Сила и направление влияния различных команд на идентификацию языков программирования	34
4.3.7. Степень детерминированности идентификации языков программирований различными командами	37
7. ВЫВОДЫ.....	39
ЛИТЕРАТУРА.....	39

Введение

Целью данной работы является идентификация языков программирования по наиболее часто встречающимся в них командам, а также анализ их сходств и различий.

Для достижения поставленной цели в данной работе предлагается применить Автоматизированный системно-когнитивный анализ (АСК-анализ) и его программный инструментарий – интеллектуальная система «Эйдос».

АСК-анализ предполагает, что для достижения поставленной цели необходимо решить следующие задачи, которые получаются путем декомпозиции цели и являются этапами ее достижения:

Задача 1: когнитивная структуризация предметной области.

Задача 2: подготовка исходных данных и формализация предметной области.

Задача 3: синтез и верификация статистических и системно-когнитивных моделей и выбор наиболее достоверной модели.

Задача 4: решение различных задач в наиболее достоверной модели:

- подзадача 4.1. Прогнозирование (диагностика, классификация, распознавание, идентификация);
- подзадача 4.2. Поддержка принятия решений;
- подзадача 4.3. Исследование моделируемой предметной области путем исследования ее модели (когнитивные диаграммы классов и значений факторов, агломеративная когнитивная кластеризация классов и значений факторов, нелокальные нейроны и нейронные сети, 3d-интегральные когнитивные карты, когнитивные функции и др.).

Эти задачи по сути представляют собой **этапы** Автоматизированного системно-когнитивный анализа (АСК-анализ), который и поэтому и предлагается применить для их решения.

АСК-анализ представляет собой метод искусственного интеллекта, разработанный проф. Е.В. Луценко в 2002 году для решения широкого класса задач идентификации, прогнозирования, классификации, диагностики, поддержки принятия решений и исследования моделируемой предметной области путем исследования ее модели. АСК-анализ доведен до **инновационного** уровня благодаря тому, что имеет свой программный инструментарий – универсальную когнитивную аналитическую систему «Эйдос-Х++» (система «Эйдос») [7-17].

Система «Эйдос» выгодно отличается от других интеллектуальных систем следующими параметрами:

- разработана в универсальной постановке, не зависящей от предметной области. Поэтому она является универсальной и может быть применена во многих предметных областях (<http://lc.kubagro.ru/aidos/index.htm>);

- находится в полном открытом бесплатном доступе (http://lc.kubagro.ru/aidos/_Aidos-X.htm), причем с актуальными исходными текстами (http://lc.kubagro.ru/_AIDOS-X.txt);

- является одной из первых отечественных систем искусственного интеллекта персонального уровня, т.е. она не требует от пользователя специальной подготовки в области технологий искусственного интеллекта (есть акт внедрения системы «Эйдос» 1987 года) (<http://lc.kubagro.ru/aidos/aidos02/PR-4.htm>);

- обеспечивает устойчивое выявление в сопоставимой форме силы и направления причинно-следственных зависимостей в неполных зашумленных взаимозависимых (нелинейных) данных очень большой размерности числовой и не числовой природы, измеряемых в различных типах шкал (номинальных, порядковых и числовых) и в различных единицах измерения (т.е. не предъявляет жестких требований к данным, которые невозможно выполнить, а обрабатывает те данные, которые есть);

- содержит большое количество локальных (поставляемых с инсталляцией) и облачных учебных и научных приложений (в настоящее время их 31 и 154, соответственно) (http://lc.kubagro.ru/aidos/Presentation_Aidos-online.pdf);

- обеспечивает мультиязычную поддержку интерфейса на 44 языках. Языковые базы входят в инсталляцию и могут пополняться в автоматическом режиме;

- поддерживает on-line среду накопления знаний и широко используется во всем мире (<http://aidos.byethost5.com/map5.php>);

- наиболее трудоемкие в вычислительном отношении операции синтеза моделей и распознавания реализует с помощью графического процессора (GPU), что на некоторых задачах обеспечивает ускорение решения этих задач в несколько тысяч раз, что реально обеспечивает интеллектуальную обработку больших данных, большой информации и больших знаний;

- обеспечивает преобразование исходных эмпирических данных в информацию, а ее в знания и решение с использованием этих знаний задач классификации, поддержки принятия решений и исследования предметной области путем исследования ее системно-когнитивной модели, генерируя при этом очень большое количество табличных и графических выходных форм (развития когнитивная графика), у многих из которых нет никаких аналогов в других системах (примеры форм можно посмотреть в работе: http://lc.kubagro.ru/aidos/aidos18_LLS/aidos18_LLS.pdf);

- хорошо имитирует человеческий стиль мышления: дает результаты анализа, понятные экспертам на основе их опыта, интуиции и профессиональной компетенции.

Всем этим и обусловлен выбор АСК-анализа и его программного инструментария – интеллектуальной системы «Эйдос» в качестве метода и инструмента решения поставленной проблемы и достижения цели работы (рисунок 1).

Последовательность обработки данных, информации и знаний в системе «Эйдос-X++»

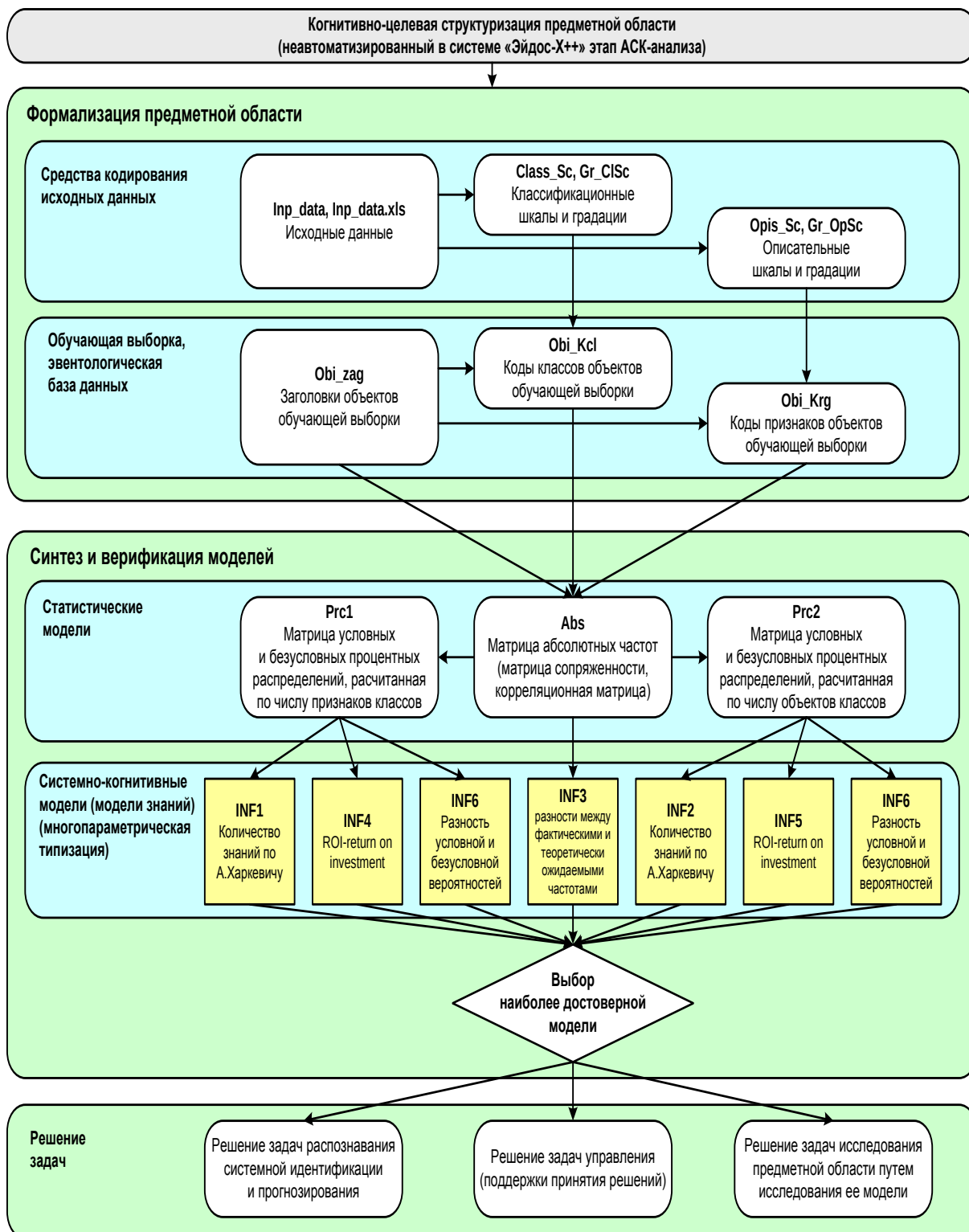


Рисунок 1. Последовательность решения задач в АСК-анализе и системе «Эйдос»

Задача 1: когнитивная структуризация предметной области

На этапе когнитивно-целевой структуризации предметной области мы неформализуемым путем решаем на качественном уровне, что будем рассматривать в качестве факторов, действующих на моделируемый объект (причин), а что в качестве результатов действия этих факторов (последствий).

При этом необходимо отметить, что системно-когнитивные модели (СК-модели) отражают лишь сам факт наличия зависимостей между значениями факторов и результатами их действия. Но они не отражают причин и механизмов такого влияния.

Это значит:

– во-первых, что содержательная интерпретация СК-моделей – это компетенция специалистов-экспертов, хорошо разбирающихся в данной предметной области. Иногда встречается ситуация, когда и то, что на первый взгляд является причинами, и то, что, казалось бы, является их последствиями, на самом деле является последствиями неких глубинных причин, которых мы не видим и никоим образом непосредственно не отражаем в модели [17];

– во-вторых, даже если содержательной интерпретации не разработано, то в принципе это не исключает возможности пользоваться ими на практике для достижения заданных результатов и поставленных целей, т.е. для управления [17].

В данной работе в качестве классификационной шкалы выберем языки программирования (таблица 1), а в качестве описательной шкалы, – команды, которые наиболее часто встречаются в этих языках (таблица 2):

Таблица 1 – Классификационные
шкалы

Код	Наименование
1	ЯЗЫК ПРОГРАММИРОВАНИЯ

Таблица 2 – Описательные шкалы

Код	Наименование
1	КОМАНДЫ

Задача 2: подготовка исходных данных и формализация предметной области

Команды (а также типы данных и модификаторы) были взяты из примеров программ, написанных на данных языках. Исходные данные приведены полностью (таблица 3).

Таблица 3 – Исходные данных
для ввода в систему «Эйдос»¹

Название	Язык программирования	Команды
Bash	Bash	function local array length while do if then done echo
Basic	Basic	do for to if then end next loop while not
C	C	#include void int while for if main sizeof printf return
C#	C#	using namespace public static class bool count do for compare to while
C++	C++	#include template typename void bool while for auto if int main copy cout
D	D	import pure nothrow bool foreach if if break return void main auto writeln
Dart	Dart	list var new from do for length if while return
Delphi	Delphi	program type integer array of procedure var integer begin repeat for downto do
Fortran	Fortran	begin if then end until array setlength random write writeln readln
Go	Go	subroutine real integer logical do size if then end not exit end
Groovy	Groovy	package import func println for if break
Haskell	Haskell	def boolean while
Java	Java	ord a case of otherwise where
JavaScript	JavaScript	public static extends comparable super void boolean do for int length compare to while
Kotlin	Kotlin	array prototype function var while for this length return
Lua	Lua	fun var boolean do for if val while
Objective-C	Objective-C	function local repeat for do if then until end
Pascal	Pascal	bool while for if return
Perl	Perl	procedure var array of real integer begin length for downto do for to if then end
PHP	PHP	sub for and
PowerShell	PowerShell	function array count do for if list while return
Python	Python	function length while for if
Ruby	Ruby	def while for xrange len if return
Scala	Scala	class array def do for in if self end downto
Swift	Swift	def import val var do foreach if while
VBScript	VBScript	func inout var while for count if
Visual Basic .NET	Visual Basic .NET	sub end dim function do for to if then next loop until not
Ада	Ада	do for to if then end next loop
Ассемблер	Ассемблер	generic type is private with return boolean array range of procedure in out element begin loop for if then end exit when

Затем с параметрами, показанными на рисунке 2, запустим режим 2.3.2.2 системы «Эйдос», представляющий собой автоматизированный программный интерфейс (API) с внешними данными табличного типа. На рисунке 2 приведены реально использованные параметры.

¹ Таблицу исходных данных 1 можно скачать с FTP-сервера системы «Эйдос» по ссылке:

2.3.2.2. Универсальный программный интерфейс импорта данных в систему "ЭЙДОС-Х++"

Автоматическая формализация предметной области: генерация классификационных и описательных шкал и градаций, а также обучающей и распознаваемой выборки на основе базы исходных данных: "Inp_data"

Задайте тип файла исходных данных: "Inp_data":

☒ XLS - MS Excel-2003 Стандарт XLS-файла
☐ XLSX - MS Excel-2007(2010) Стандарт XLSX-файла
☐ DBF - DBASE IV (DBF/NTX) Стандарт DBF-файла
☐ CSV - Comma-Separated Values Стандарт CSV-файла

Задайте диапазон столбцов классификационных шкал:

Начальный столбец классификационных шкал:
 Конечный столбец классификационных шкал:

Задайте диапазон столбцов описательных шкал:

Начальный столбец описательных шкал:
 Конечный столбец описательных шкал:

Задайте режим:

☒ Формализации предметной области (на основе "Inp_data")
☐ Генерации распознаваемой выборки (на основе "Inp_rasp")

Задайте способ выбора размера интервалов:

☒ Равные интервалы с разным числом наблюдений
☐ Разные интервалы с равным числом наблюдений

Задание параметров формирования сценариев или способа интерпретации текстовых полей "Inp_data":

☒ Не применять сценарный метод АСК-анализа ☐ Применить сценарный метод АСК-анализа
☐ Применить спец. интерпретацию текстовых полей классов ☒ Применить спец. интерпретацию текстовых полей признаков

Параметры интерпретации значений текстовых полей "Inp_data":

Интерпретация TXT-полей классов:

Значения полей текстовых классификационных шкал файла исходных данных "Inp_data" рассматриваются как целое

В качестве признаков рассматриваются:

☐ Значения полей целиком
☒ Элементы значений полей - слова > символов:
☐ Элементы значений полей - символы

☐ Проводить лемматизацию
☒ Не проводить лемматизацию

Какие наименования ГРАДАЦИЙ числовых шкал использовать:

☒ Только интервальные числовые значения (например: "1/3-{59873.0000000, 178545.6666667}")
☐ Только наименования интервальных числовых значений (например: "Минимальное")
☐ И интервальные числовые значения, и их наименования (например: "Минимальное: 1/3-{59873.0000000, 178545.6666667}")

Ok Cancel

2.3.2.2. Задание размерности модели системы "ЭЙДОС-Х++"

ИНФОРМАЦИЯ О РАЗМЕРНОСТИ МОДЕЛИ

Суммарное количество градаций классификационных и описательных шкал: [29 x 29]

Тип шкалы	Количество классификационных шкал	Количество градаций классификационных	Среднее количество градаций на класс. шкалу	Количество описательных шкал	Количество градаций описательных шкал	Среднее количество градаций на опис. шкалу
Числовые	0	0	0,00	0	0	0,00
Текстовые	1	29	29,00	1	29	29,00
ВСЕГО:	1	29	29,00	1	29	29,00

Задайте число интервалов (градаций) в шкале:

Пересчитать шкалы и градации Выйти на создание модели

2.3.2.2. Процесс импорта данных из внешней БД "Inp_data" в систему "ЭЙДОС-Х++"

Стадии исполнения процесса:

1/3: Формирование классификационных и описательных шкал и градаций на основе БД "Inp_data"- Готово
 2/3: Генерация обучающей выборки и базы событий "EventsKO" на основе БД "Inp_data"- Готово
 3/3: Переиндексация всех баз данных нового приложения- Готово

ПРОЦЕСС ФОРМАЛИЗАЦИИ ПРЕДМЕТНОЙ ОБЛАСТИ ЗАВЕРШЕН УСПЕШНО !!!

Прогноз времени исполнения

Начало: 23:46:39 Окончание: 23:46:42
 Прошло: 0:00:03 Осталось: 0:00:00

Ok

Рисунок 2. Экранные форма программного интерфейса (API) 2.3.2.2 системы «Эйдос» с внешними данными табличного типа

Обратим внимание на то, что текстовые поля признаков подвергаются специальной интерпретации. Каждое слово рассматривается в качестве отдельного признака.

На рисунке 3 приведен Help данного режима, в котором объясняется принцип организации таблицы исходных данных для данного режима. Здесь же обратим внимание на то, что в таблице 3 как значения параметров представлены только текстовыми значениями.

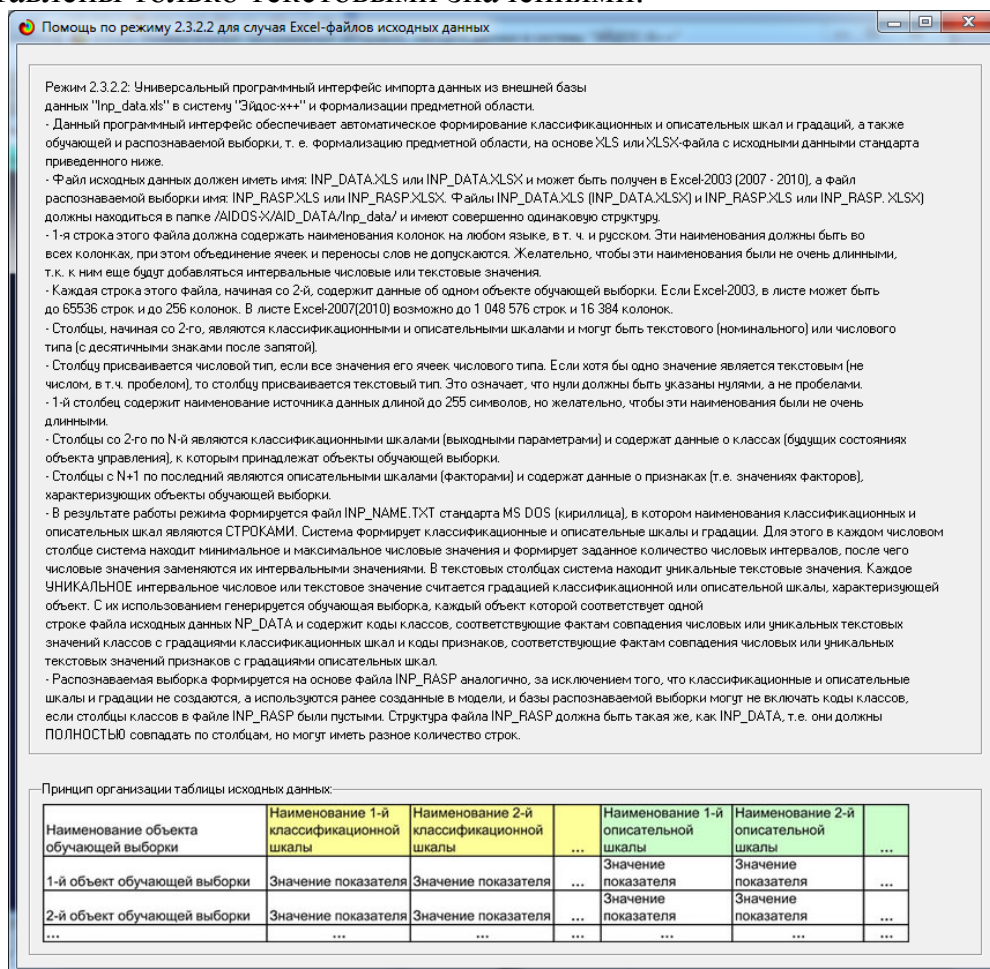


Рисунок 3. Экранная форма HELP программного интерфейса (API) 2.3.2.2²

В результате работы режима сформирована одна классификационная шкала с количеством градаций (классов) 29 (таблица 4) и одна описательная шкала с количеством градаций 110 (таблица 5). С использованием классификационных и описательных шкал и градаций (таблицы 4 и 5) исходные данные (таблица 3) были закодированы и в результате получена обучающая выборка (таблица 6):

² Все рисунки в статье приведены с достаточно высоким разрешением и при увеличении масштаба просмотра вполне читабельны

Таблица 4 – Классификационные шкалы и градации
(названия языков программирования)

Код	Наименование
1	ЯЗЫК ПРОГРАММИРОВАНИЯ-Bash
2	ЯЗЫК ПРОГРАММИРОВАНИЯ-Basic
3	ЯЗЫК ПРОГРАММИРОВАНИЯ-C
4	ЯЗЫК ПРОГРАММИРОВАНИЯ-C#
5	ЯЗЫК ПРОГРАММИРОВАНИЯ-C++
6	ЯЗЫК ПРОГРАММИРОВАНИЯ-D
7	ЯЗЫК ПРОГРАММИРОВАНИЯ-Dart
8	ЯЗЫК ПРОГРАММИРОВАНИЯ-Delphi
9	ЯЗЫК ПРОГРАММИРОВАНИЯ-Fortran
10	ЯЗЫК ПРОГРАММИРОВАНИЯ-Go
11	ЯЗЫК ПРОГРАММИРОВАНИЯ-Groovy
12	ЯЗЫК ПРОГРАММИРОВАНИЯ-Haskell
13	ЯЗЫК ПРОГРАММИРОВАНИЯ-Java
14	ЯЗЫК ПРОГРАММИРОВАНИЯ-JavaScript
15	ЯЗЫК ПРОГРАММИРОВАНИЯ-Kotlin
16	ЯЗЫК ПРОГРАММИРОВАНИЯ-Lua
17	ЯЗЫК ПРОГРАММИРОВАНИЯ-Objective-C
18	ЯЗЫК ПРОГРАММИРОВАНИЯ-Pascal
19	ЯЗЫК ПРОГРАММИРОВАНИЯ-Perl
20	ЯЗЫК ПРОГРАММИРОВАНИЯ-PHP
21	ЯЗЫК ПРОГРАММИРОВАНИЯ-PowerShell
22	ЯЗЫК ПРОГРАММИРОВАНИЯ-Python
23	ЯЗЫК ПРОГРАММИРОВАНИЯ-Ruby
24	ЯЗЫК ПРОГРАММИРОВАНИЯ-Scala
25	ЯЗЫК ПРОГРАММИРОВАНИЯ-Swift
26	ЯЗЫК ПРОГРАММИРОВАНИЯ-VBScript
27	ЯЗЫК ПРОГРАММИРОВАНИЯ-Visual Basic .NET
28	ЯЗЫК ПРОГРАММИРОВАНИЯ-Ада
29	ЯЗЫК ПРОГРАММИРОВАНИЯ-Ассемблер

Таблица 5 – Описательные шкалы и градации
(команды языков программирования)

Код	Наименование
1	КОМАНДЫ-and
2	КОМАНДЫ-array
3	КОМАНДЫ-auto
4	КОМАНДЫ-begin
5	КОМАНДЫ-bool
6	КОМАНДЫ-boolean
7	КОМАНДЫ-break
8	КОМАНДЫ-bx
9	КОМАНДЫ-call

10	КОМАНДЫ-case
11	КОМАНДЫ-class
12	КОМАНДЫ-cmp
13	КОМАНДЫ-comparable
14	КОМАНДЫ-compareto
15	КОМАНДЫ-copy
16	КОМАНДЫ-count
17	КОМАНДЫ-cout
18	КОМАНДЫ-db
19	КОМАНДЫ-dec
20	КОМАНДЫ-def
21	КОМАНДЫ-dim
22	КОМАНДЫ-do
23	КОМАНДЫ-done
24	КОМАНДЫ-downto
25	КОМАНДЫ-echo
26	КОМАНДЫ-element
27	КОМАНДЫ-end
28	КОМАНДЫ-exit
29	КОМАНДЫ-extends
30	КОМАНДЫ-for
31	КОМАНДЫ-foreach
32	КОМАНДЫ-from
33	КОМАНДЫ-fun
34	КОМАНДЫ-func
35	КОМАНДЫ-function
36	КОМАНДЫ-generic
37	КОМАНДЫ-if
38	КОМАНДЫ-import
39	КОМАНДЫ-in
40	КОМАНДЫ-inc
41	КОМАНДЫ-include
42	КОМАНДЫ-inout
43	КОМАНДЫ-int
44	КОМАНДЫ-integer
45	КОМАНДЫ-is
46	КОМАНДЫ-jb
47	КОМАНДЫ-jbe
48	КОМАНДЫ-len
49	КОМАНДЫ-length
50	КОМАНДЫ-list
51	КОМАНДЫ-local
52	КОМАНДЫ-logical
53	КОМАНДЫ-loop
54	КОМАНДЫ-main
55	КОМАНДЫ-mov
56	КОМАНДЫ-namespace
57	КОМАНДЫ-new
58	КОМАНДЫ-next
59	КОМАНДЫ-not
60	КОМАНДЫ-nothrow

61	КОМАНДЫ-of
62	КОМАНДЫ-ord
63	КОМАНДЫ-otherwise
64	КОМАНДЫ-out
65	КОМАНДЫ-package
66	КОМАНДЫ-popn
67	КОМАНДЫ-printf
68	КОМАНДЫ-println
69	КОМАНДЫ-private
70	КОМАНДЫ-procedure
71	КОМАНДЫ-program
72	КОМАНДЫ-prototype
73	КОМАНДЫ-public
74	КОМАНДЫ-pure
75	КОМАНДЫ-random
76	КОМАНДЫ-range
77	КОМАНДЫ-readln
78	КОМАНДЫ-real
79	КОМАНДЫ-repeat
80	КОМАНДЫ-ret
81	КОМАНДЫ-return
82	КОМАНДЫ-self
83	КОМАНДЫ-setlength
84	КОМАНДЫ-size
85	КОМАНДЫ-sizeof
86	КОМАНДЫ-start
87	КОМАНДЫ-static
88	КОМАНДЫ-sub
89	КОМАНДЫ-subroutine
90	КОМАНДЫ-super
91	КОМАНДЫ-template
92	КОМАНДЫ-then
93	КОМАНДЫ-this
94	КОМАНДЫ-to
95	КОМАНДЫ-type
96	КОМАНДЫ-typename
97	КОМАНДЫ-until
98	КОМАНДЫ-using
99	КОМАНДЫ-val
100	КОМАНДЫ-var
101	КОМАНДЫ-void
102	КОМАНДЫ-when
103	КОМАНДЫ-where
104	КОМАНДЫ-while
105	КОМАНДЫ-with
106	КОМАНДЫ-write
107	КОМАНДЫ-writeln
108	КОМАНДЫ-xchg
109	КОМАНДЫ-xor
110	КОМАНДЫ-xrange

Таблица 6 – Обучающая выборка

2.3.1. Ручной ввод-корректировка обучающей выборки. Текущая модель: "INF1"

Код объекта	Наименование объекта	Дата	Время
1	Bash		
2	Basic		
3	C		
4	C#		
5	C++		
6	D		
7	Dart		
8	Delphi		
9	Fortran		
10	C++		

Код объекта	Класс 1	Класс 2	Класс 3	Класс 4
1	1	0	0	0

Код объекта	Признак 1	Признак 2	Признак 3	Признак 4	Признак 5	Признак 6	Признак 7
1	35	51	2	49	104	22	37
1	92	23	25	0	0	0	0

Помощь Скопировать обуч.выб.в расп. Добавить объект Добавить классы Добавить признаки Удалить объект Удалить классы Удалить признаки Очистить БД

Обучающая выборка (таблица 6), по сути, представляет собой нормализованные исходные данные, т.е. таблицу исходных данных (таблица 3), закодированную с помощью классификационных и описательных шкал и градаций (таблицы 4 и 5). Желтым фоном выделены классификационные шкалы. Таким образом, созданы все необходимые и достаточные условия для выполнения следующего этапа АСК-анализа: т.е. для синтеза и верификации моделей.

Задача 3: синтез и верификация статистических и системно-когнитивных моделей и выбор наиболее достоверной из них для решения задач

Синтез и верификация статистических и системно-когнитивных моделей (СК-моделей) моделей осуществляется в режиме 3.5 системы «Эйдос» (рисунок 4). Сами эти модели описаны в ряде работ [7-16].

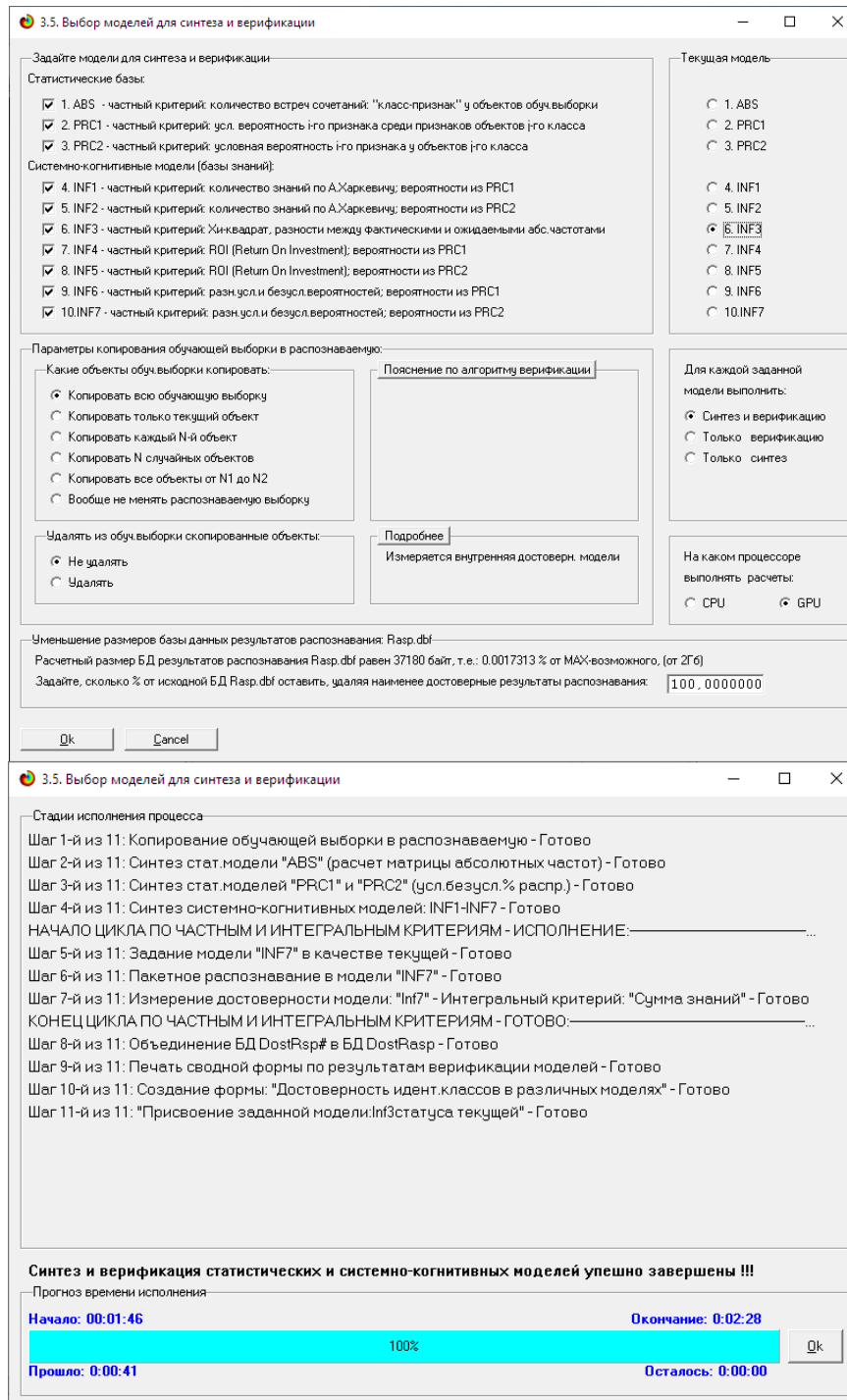


Рисунок 4. Экранные формы режима синтеза и верификации статистических и системно-когнитивных моделей системы «Эйдос»

Обратим внимание на то, что на рисунке 4 в правом нижнем углу окна задана опция: «Расчеты проводить на графическом процессоре (GPU)».

Из рисунка 4 видно, что весь процесс синтеза и верификации моделей занял 41 секунду. Отметим, что при синтезе и верификации моделей использовался графический процессор (GPU) видеокарты. На центральном процессоре (CPU) выполнение этих операций занимает

значительно большее время (на некоторых задачах это происходит в десятки, сотни и даже тысячи раз дольше). Таким образом, неграфические вычисления на графических процессорах видеокарты делает возможной обработку больших объемов исходных данных за разумное время. В процесс синтеза и верификации моделей осуществляется также расчет 10 выходных форм, на что уходит более 99% времени исполнения.

Фрагменты самих созданных статистических и системно-когнитивных моделей (СК-модели) приведены на рисунках 5, 6, 7:

5.5. Модель: "1. ABS - частный критерий: количество встреч сочетаний "Класс-признак" у объектов обучающей выборки"

Код признака	Наименование описательной шкалы и градаций	1. ЯЗЫК ПРОГРАММИР... BASH	2. ЯЗЫК ПРОГРАММИР... BASIC	3. ЯЗЫК ПРОГРАММИР... C	4. ЯЗЫК ПРОГРАММИР... C#	5. ЯЗЫК ПРОГРАММИР... C++	6. ЯЗЫК ПРОГРАММИР... D	7. ЯЗЫК ПРОГРАММИР... DART	8. ЯЗЫК ПРОГРАММИР... DELPHI	9. ЯЗЫК ПРОГРАММИР... FORTRAN	10. ЯЗЫК ПРОГРАММИР... GO	11. ЯЗЫК ПРОГРАММИР... GROOVY
1	КОМАНДЫ-and											
2	КОМАНДЫ-array	1							2			
3	КОМАНДЫ-auto					1	1					
4	КОМАНДЫ-begin								2			
5	КОМАНДЫ-bool				1	1	1					
6	КОМАНДЫ-boolean											1
7	КОМАНДЫ-break						1				1	
8	КОМАНДЫ-bx											
9	КОМАНДЫ-call											
10	КОМАНДЫ-case											
11	КОМАНДЫ-class				1							
12	КОМАНДЫ-cmp											
13	КОМАНДЫ-comparable											
14	КОМАНДЫ-compareto				1							
15	КОМАНДЫ-copy					1						
16	КОМАНДЫ-count				1							
17	КОМАНДЫ-cout					1						
18	КОМАНДЫ-db											
19	КОМАНДЫ-dec											
20	КОМАНДЫ-def											1
21	КОМАНДЫ-dim											
22	КОМАНДЫ-do	1	1		1			1	1	1		

Рисунок 5. Матрица абсолютных частот (фрагмент)

5.5. Модель: "4. INF1 - частный критерий: количество знаний по А.Харкевичу; вероятности из PRC1"

Код признака	Наименование описательной шкалы и градаций	1. ЯЗЫК ПРОГРАММИР... BASH	2. ЯЗЫК ПРОГРАММИР... BASIC	3. ЯЗЫК ПРОГРАММИР... C	4. ЯЗЫК ПРОГРАММИР... C#	5. ЯЗЫК ПРОГРАММИР... C++	6. ЯЗЫК ПРОГРАММИР... D	7. ЯЗЫК ПРОГРАММИР... DART	8. ЯЗЫК ПРОГРАММИР... DELPHI	9. ЯЗЫК ПРОГРАММИР... FORTRAN	10. ЯЗЫК ПРОГРАММИР... GO	11. ЯЗЫК ПРОГРАММИР... GROOVY
1	КОМАНДЫ-and											
2	КОМАНДЫ-array	1.023							0.868			
3	КОМАНДЫ-auto					2.081	2.081					
4	КОМАНДЫ-begin								1.559			
5	КОМАНДЫ-bool				1.633	1.491	1.491					
6	КОМАНДЫ-boolean											
7	КОМАНДЫ-break						2.081				2.609	
8	КОМАНДЫ-bx											
9	КОМАНДЫ-call											
10	КОМАНДЫ-case											
11	КОМАНДЫ-class				2.224							
12	КОМАНДЫ-cmp											
13	КОМАНДЫ-comparable											
14	КОМАНДЫ-compareto				2.224							
15	КОМАНДЫ-copy					2.672						
16	КОМАНДЫ-count				1.878							
17	КОМАНДЫ-cout					2.672						
18	КОМАНДЫ-db											
19	КОМАНДЫ-dec											
20	КОМАНДЫ-def											
21	КОМАНДЫ-dim											
22	КОМАНДЫ-do	0.588	0.588		0.507			0.588	-0.158	0.432		

Рисунок 6. Матрица информативностей INF1 (фрагмент)

5.5. Модель: "6. INF3 - частный критерий: Хи-квадрат, разности между фактическими и ожидаемыми абс-частотами"

Код признака	Наименование описательной шкалы и градаций	1. ЯЗЫК ПРОГРАММИРОВАНИЯ BASH	2. ЯЗЫК ПРОГРАММИРОВАНИЯ BASIC	3. ЯЗЫК ПРОГРАММИРОВАНИЯ C	4. ЯЗЫК ПРОГРАММИРОВАНИЯ C#	5. ЯЗЫК ПРОГРАММИРОВАНИЯ C++	6. ЯЗЫК ПРОГРАММИРОВАНИЯ D	7. ЯЗЫК ПРОГРАММИРОВАНИЯ DART	8. ЯЗЫК ПРОГРАММИРОВАНИЯ DELPHI	9. ЯЗЫК ПРОГРАММИРОВАНИЯ FORTRAN	10. ЯЗЫК ПРОГРАММИРОВАНИЯ GO	11. ЯЗЫК ПРОГРАММИРОВАНИЯ GPG
1	КОМАНДЫ-and	-0.033	-0.033	-0.033	-0.037	-0.043	-0.043	-0.033	-0.080	-0.040	-0.023	
2	КОМАНДЫ-array	0.699	-0.301	-0.301	-0.331	-0.391	-0.391	-0.301	1.278	-0.361	-0.211	
3	КОМАНДЫ-auto	-0.067	-0.067	-0.067	-0.074	0.913	0.913	-0.067	-0.161	-0.080	-0.047	
4	КОМАНДЫ-begin	-0.134	-0.134	-0.134	-0.147	-0.174	-0.174	-0.134	1.679	-0.161	-0.094	
5	КОМАНДЫ-bool	-0.134	-0.134	-0.134	0.853	0.826	0.826	-0.134	-0.321	-0.161	-0.094	
6	КОМАНДЫ-boolean	-0.134	-0.134	-0.134	-0.147	-0.174	-0.174	-0.134	-0.321	-0.161	-0.094	
7	КОМАНДЫ-break	-0.067	-0.067	-0.067	-0.074	-0.087	0.913	-0.067	-0.161	-0.080	0.953	
8	КОМАНДЫ-bx	-0.033	-0.033	-0.033	-0.037	-0.043	-0.043	-0.033	-0.080	-0.040	-0.023	
9	КОМАНДЫ-call	-0.033	-0.033	-0.033	-0.037	-0.043	-0.043	-0.033	-0.080	-0.040	-0.023	
10	КОМАНДЫ-case	-0.033	-0.033	-0.033	-0.037	-0.043	-0.043	-0.033	-0.080	-0.040	-0.023	
11	КОМАНДЫ-class	-0.067	-0.067	-0.067	0.926	-0.087	-0.087	-0.067	-0.161	-0.080	-0.047	
12	КОМАНДЫ-cmp	-0.067	-0.067	-0.067	-0.074	-0.087	-0.087	-0.067	-0.161	-0.080	-0.047	
13	КОМАНДЫ-comparable	-0.033	-0.033	-0.033	-0.037	-0.043	-0.043	-0.033	-0.080	-0.040	-0.023	
14	КОМАНДЫ-compareto	-0.067	-0.067	-0.067	0.926	-0.087	-0.087	-0.067	-0.161	-0.080	-0.047	
15	КОМАНДЫ-copy	-0.033	-0.033	-0.033	-0.037	0.957	-0.043	-0.033	-0.080	-0.040	-0.023	
16	КОМАНДЫ-count	-0.100	-0.100	-0.100	0.890	-0.130	-0.130	-0.100	-0.241	-0.120	-0.070	
17	КОМАНДЫ-cout	-0.033	-0.033	-0.033	-0.037	0.957	-0.043	-0.033	-0.080	-0.040	-0.023	
18	КОМАНДЫ-db	-0.033	-0.033	-0.033	-0.037	-0.043	-0.043	-0.033	-0.080	-0.040	-0.023	
19	КОМАНДЫ-dec	-0.033	-0.033	-0.033	-0.037	-0.043	-0.043	-0.033	-0.080	-0.040	-0.023	
20	КОМАНДЫ-def	-0.134	-0.134	-0.134	-0.147	-0.174	-0.174	-0.134	-0.321	-0.161	-0.094	
21	КОМАНДЫ-dim	-0.033	-0.033	-0.033	-0.037	-0.043	-0.043	-0.033	-0.080	-0.040	-0.023	
22	КОМАНДЫ-do	0.498	0.498	-0.502	0.448	-0.652	-0.652	0.498	-0.204	0.398	-0.351	

Рисунок 7. Модель INF3 (фрагмент)

Отметим, что в АСК-анализе и СК-моделях степень выраженности различных свойств объектов наблюдения рассматривается с единственной точки зрения: с точки зрения того, какое **количество информации** содержится в них о том, к каким обобщающим категориям (классам) будут принадлежать или не принадлежать эти объекты [9]. Поэтому не играет никакой роли в каких единицах измерения измеряются те или иные свойства объектов наблюдения, а также в каких единицах измеряются результаты влияния этих свойств, натуральных, в процентах или стоимостных [9]. Это и есть решение проблемы сопоставимости в АСК-анализе и системе «Эйдос», отличающее их от других интеллектуальных технологий.

Верификация статистических и системно-когнитивных моделей

Оценка достоверности моделей в системе «Эйдос» осуществляется путем решения задачи классификации объектов обучающей выборки по обобщенным образам классов и подсчета количества истинных положительных и отрицательных, а также ложных положительных и отрицательных решений по F-мере Ван Ризбергера, а также по критериям L1- L2-мерам проф. Е.В.Луценко, которые предложены для того, чтобы смягчить или полностью преодолеть некоторые недостатки F-меры [10]. В режиме 3.4 системы «Эйдос» изучается достоверность каждой частной модели в соответствии с этими мерами достоверности (рисунок 8).

Из рисунка 8 мы видим, что в данном интеллектуальном приложении по F-критерию Ван Ризбергера наиболее достоверной является СК-модель

INF5 с интегральным критерием «Семантический резонанс знаний» ($F=0,252$ при максимуме 1,000), что является довольно слабым результатом, а по критерию L1 проф. Е.В.Луценко [10] наиболее достоверной также является СК-модель INF5 с интегральным критерием «Семантический резонанс знаний» ($L1=0,698$ при максимуме 1,000), что является неплохим результатом.

3.4. Обобщ. форма по достов. моделей при разн. инт. крит. Текущая модель: "INF3"

Наименование модели и частного критерия	Интегральный критерий	Всего логических объектов выборки	Число истинно-положительных решений (TP)	Число истинно-отрицательных решений (TN)	Число ложно-положительных решений (FP)	Число ложно-отрицательных решений (FN)	Точность модели	Полнота модели	F-мера Ван Ризбергера	Сумма модулей уровней сход. истинно-полож. решений (STP)	Сумма модулей уровней сход. истинно-отриц. решений (STN)	Сумма модулей уровней сход. ложно-полож. решений
1. ABS - частный критерий: количество встреч сометаний: "Улас..."	Корреляция абс частот с обр...	29	29	160	652		0.043	1.000	0.082	29.000	9.748	160
1. ABS - частный критерий: количество встреч сометаний: "Улас..."	Средн абс частот по признакам...	29	29	102	710		0.039	1.000	0.076	10.433		69
2. PRC1 - частный критерий: усл. вероятность его признака сред.	Корреляция усл.отн частот с о...	29	29	160	652		0.043	1.000	0.082	29.000	9.748	160
2. PRC1 - частный критерий: усл. вероятность его признака сред.	Средн усл.отн частот по признакам...	29	29	102	710		0.039	1.000	0.076	23.841		177
3. PRC2 - частный критерий: условная вероятность его признака...	Корреляция усл.отн частот с о...	29	29	160	652		0.043	1.000	0.082	29.000	9.748	160
3. PRC2 - частный критерий: условная вероятность его признака...	Средн усл.отн частот по признакам...	29	29	102	710		0.039	1.000	0.076	10.433		69
4. INF1 - частный критерий: количество знаний по А.Харкевичу: в...	Семантический резонанс зна...	29	29	410	402		0.067	1.000	0.126	25.166	26.762	51
4. INF1 - частный критерий: количество знаний по А.Харкевичу: в...	Средн знаний	29	29	152	660		0.042	1.000	0.081	11.727	0.531	34
5. INF2 - частный критерий: количество знаний по А.Харкевичу: в...	Семантический резонанс зна...	29	29	459	353		0.076	1.000	0.141	24.937	24.214	42
5. INF2 - частный критерий: количество знаний по А.Харкевичу: в...	Средн знаний	29	29	102	710		0.039	1.000	0.076	9.888		26
6. INF3 - частный критерий: Уинкватер, разности между фактич...	Семантический резонанс зна...	29	29	436	376		0.072	1.000	0.134	26.928	62.406	61
6. INF3 - частный критерий: Уинкватер, разности между фактич...	Средн знаний	29	29	436	376		0.072	1.000	0.134	12.107	33.912	21
7. INF4 - частный критерий: ROI (Return On Investment), вероятн...	Семантический резонанс зна...	29	29	621	191		0.132	1.000	0.233	21.967	35.371	21
7. INF4 - частный критерий: ROI (Return On Investment), вероятн...	Средн знаний	29	29	149	663		0.042	1.000	0.080	10.935	0.059	14
8. INF5 - частный критерий: ROI (Return On Investment), вероятн...	Семантический резонанс зна...	29	29	640	172		0.144	1.000	0.252	21.844	35.670	18
8. INF5 - частный критерий: ROI (Return On Investment), вероятн...	Средн знаний	29	29	102	710		0.039	1.000	0.076	6.819		7
9. INF6 - частный критерий: разн усл и безуслов вероятностей; вер...	Семантический резонанс зна...	29	29	303	509		0.054	1.000	0.102	27.134	23.836	89
9. INF6 - частный критерий: разн усл и безуслов вероятностей; вер...	Средн знаний	29	29	160	652		0.043	1.000	0.082	22.410	1.735	115
10. INF7 - частный критерий: разн усл и безуслов вероятностей; ве...	Семантический резонанс зна...	29	29	294	518		0.053	1.000	0.101	27.040	15.353	76
10. INF7 - частный критерий: разн усл и безуслов вероятностей; ве...	Средн знаний	29	29	102	710		0.039	1.000	0.076	9.238		37

Помощь по меркам достоверности Помощь по частотным распределениям TP, TN, FP, FN (TP, FP), (TN, FN) (T-F)/(T+F)*100 Задать интервал сглаживания

Рисунок 8. Экранная форма с информацией о достоверности моделей по F-критерию Ван Ризбергера и L1- и L2-критериям проф.Е.В.Луценко [10]

На рисунке 9 приведено частотное распределения числа истинных и ложных положительных и отрицательных решений по результатам идентификации языков программирования по их командам в СК-модели INF5 по данным обучающей выборки:

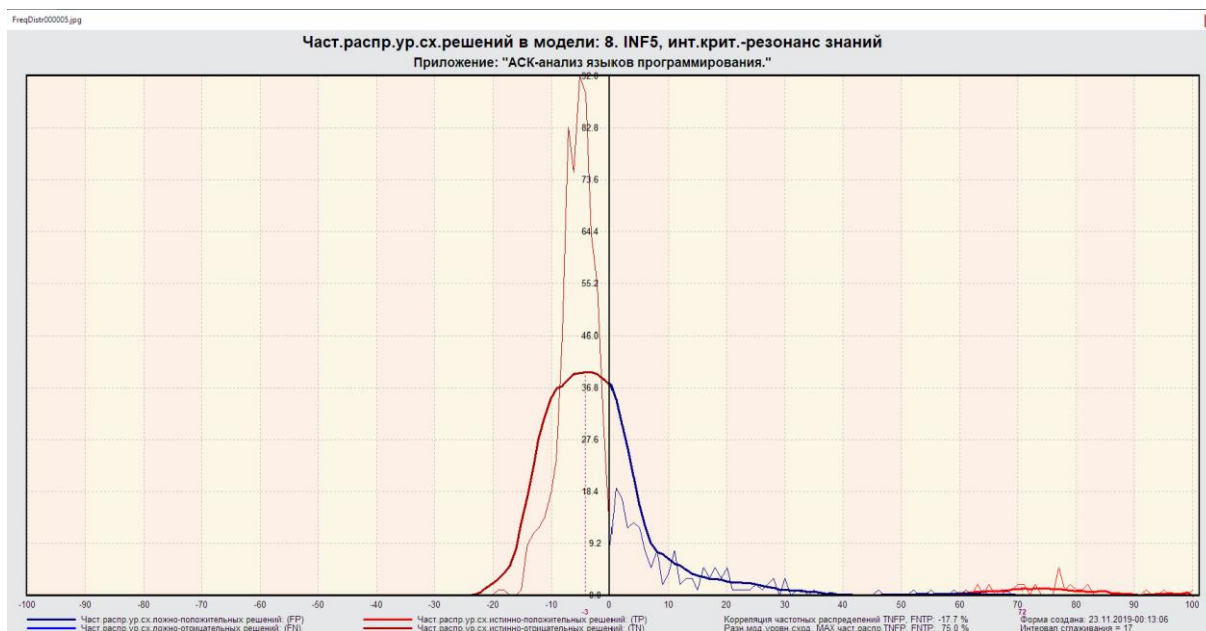


Рисунок 9. Частотные распределения числа

модели. Для этого запустим режим 5.6 с параметрами, приведенными на экранной форме (рисунок 11):

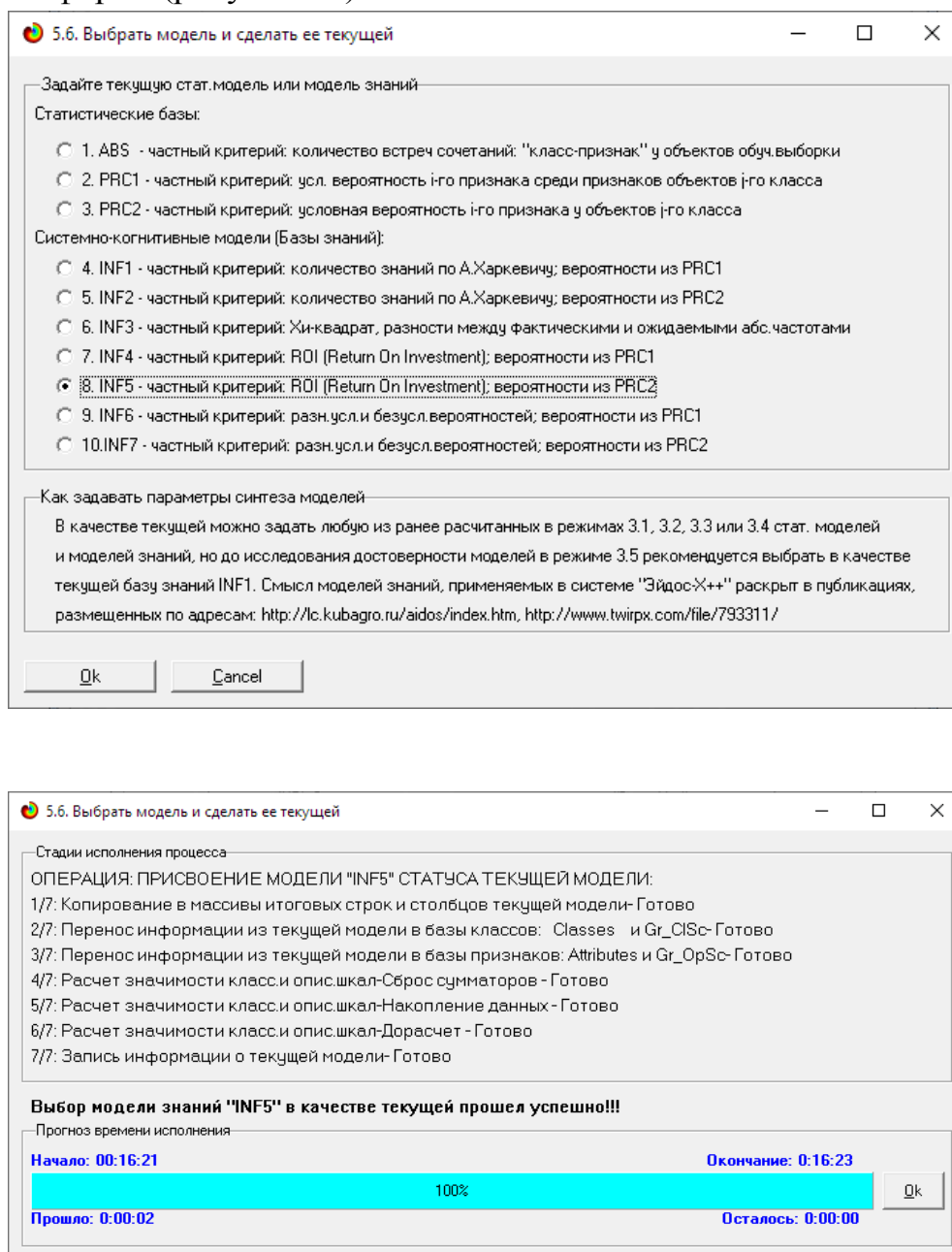


Рисунок 11. Экранные формы придания наиболее достоверной по F-критерию СК-модели Inf5 статуса текущей модели

Задача 4: решение различных задач в наиболее достоверной модели

Подзадача 4.1. Прогнозирование (диагностика, классификация, распознавание, идентификация)

Решим задачу системной идентификации, т.е. измерения содержания жира и белка в коровьем молоке на основе обучающей выборки в наиболее достоверной СК-модели INF1 на GPU (рисунок 11).

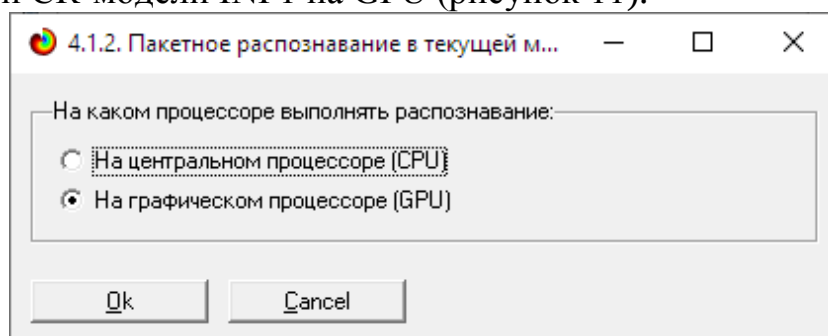
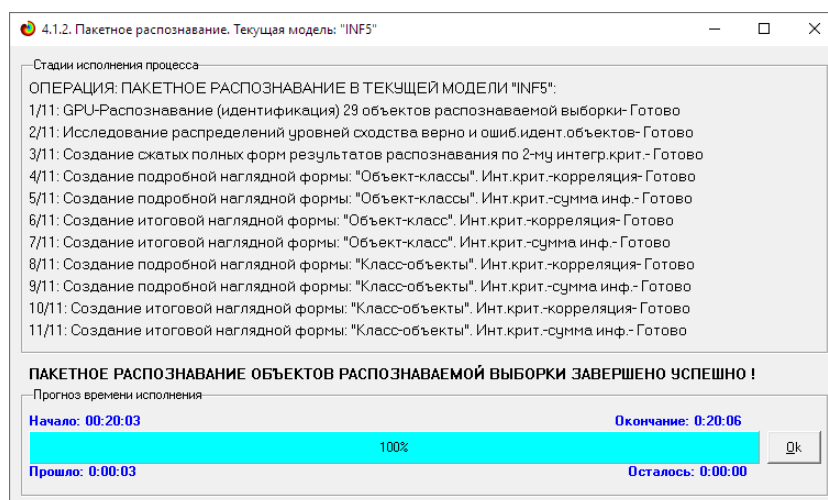


Рисунок 11. Экранные формы отображения процесса решения задачи системной идентификации в текущей модели



4.1.3. Вывод результатов распознавания

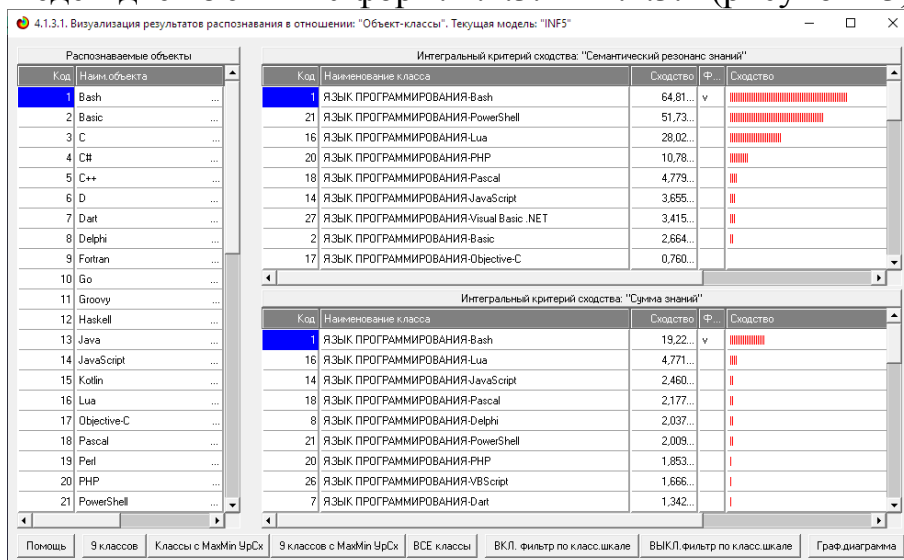
4.1.3.1. Подробно наглядно: "Объект - классы"	
4.1.3.2. Подробно наглядно: "Класс - объекты"	
4.1.3.3. Итоги наглядно: "Объект - класс"	
4.1.3.4. Итоги наглядно: "Класс - объект"	
4.1.3.5. Подробно сжато: "Объекты - классы"	
4.1.3.6. Обобщ. форма по достов.моделей при разных интегральных крит.	
4.1.3.7. Обобщ.стат.анализ результатов идент. по моделям и инт.крит.	
4.1.3.8. Стат.анализ результ. идент. по классам, моделям и инт.крит.	
4.1.3.9. Достоверность идент.объектов при разных моделях и инт.крит.	
4.1.3.10.Достоверность идент.классов при разных моделях и инт.крит.	
4.1.3.11.Распределения уровн.сходства при разных моделях и инт.крит.	
4.1.3.12.Объединение в одной БД строк по самым достоверным моделям	

Рисунок 12. Экранные формы отображения процесса решения задачи системной идентификации в текущей модели

Из рисунка 12 видно, что процесс идентификации занял 2 секунды.

Отметим, что 99,999% этого времени заняла не сама идентификация на GPU, а создание 10 выходных форм на основе результатов этого прогнозирования. Эти формы отражают результаты прогнозирования в различных разрезах и обобщениях:

Приведем две из этих 10 форм: 4.1.3.1 и 4.1.3.2 (рисунок 13).



4.1.3.2. Визуализация результатов распознавания в отношении: "Класс-объекты". Текущая модель: "INF5"

Классы		
Код	Наим. класса	
1	ЯЗЫК ПРОГРАММИРОВАНИЯ-Bash	...
2	ЯЗЫК ПРОГРАММИРОВАНИЯ-Basic	...
3	ЯЗЫК ПРОГРАММИРОВАНИЯ-C	...
4	ЯЗЫК ПРОГРАММИРОВАНИЯ-C#	...
5	ЯЗЫК ПРОГРАММИРОВАНИЯ-C++	...
6	ЯЗЫК ПРОГРАММИРОВАНИЯ-D	...
7	ЯЗЫК ПРОГРАММИРОВАНИЯ-Dart	...
8	ЯЗЫК ПРОГРАММИРОВАНИЯ-Delphi	...
9	ЯЗЫК ПРОГРАММИРОВАНИЯ-Fortran	...
10	ЯЗЫК ПРОГРАММИРОВАНИЯ-Go	...
11	ЯЗЫК ПРОГРАММИРОВАНИЯ-Groovy	...
12	ЯЗЫК ПРОГРАММИРОВАНИЯ-Haskell	...
13	ЯЗЫК ПРОГРАММИРОВАНИЯ-Java	...
14	ЯЗЫК ПРОГРАММИРОВАНИЯ-JavaScript	...
15	ЯЗЫК ПРОГРАММИРОВАНИЯ-Kotlin	...
16	ЯЗЫК ПРОГРАММИРОВАНИЯ-Lua	...
17	ЯЗЫК ПРОГРАММИРОВАНИЯ-Objective-C	...
18	ЯЗЫК ПРОГРАММИРОВАНИЯ-Pascal	...
19	ЯЗЫК ПРОГРАММИРОВАНИЯ-Perl	...
20	ЯЗЫК ПРОГРАММИРОВАНИЯ-PHP	...
21	ЯЗЫК ПРОГРАММИРОВАНИЯ-PowerShell	...

Интегральный критерий сходства: "Семантический резонанс знаний"				
Код	Наименование объекта	Сходство	Ф...	Сходство
1	Bash	64.81...	v	
16	Lua	12.43...		
21	PowerShell	5.812...		
14	JavaScript	3.451...		
20	PHP	1.091...		
7	Dart	-1.502...		
18	Pascal	-1.747...		
26	VBScript	-1.998...		
11	Groovy	-2.209...		

Интегральный критерий сходства: "Сумма знаний"				
Код	Наименование объекта	Сходство	Ф...	Сходство
1	Bash	19.22...	v	
16	Lua	4.771...		
14	JavaScript	2.460...		
18	Pascal	2.177...		
21	PowerShell	2.009...		
20	PHP	1.853...		
8	Delphi	1.807...		
26	VBScript	1.666...		
7	Dart	1.342...		

Помощь

Поиск объекта

В начало БД

В конец БД

Предыдущая

Следующая

9 записей

Все записи

Печать XLS

Печать TXT

Печать ALL

Рисунок 13. Выходные формы по результатам идентификации языков программирования по их командам

Символ «√» стоит против тех результатов идентификации, которые подтвердились на опыте, т.е. соответствуют факту. Из рисунка 13 видно, что результаты идентификации являются хорошими.

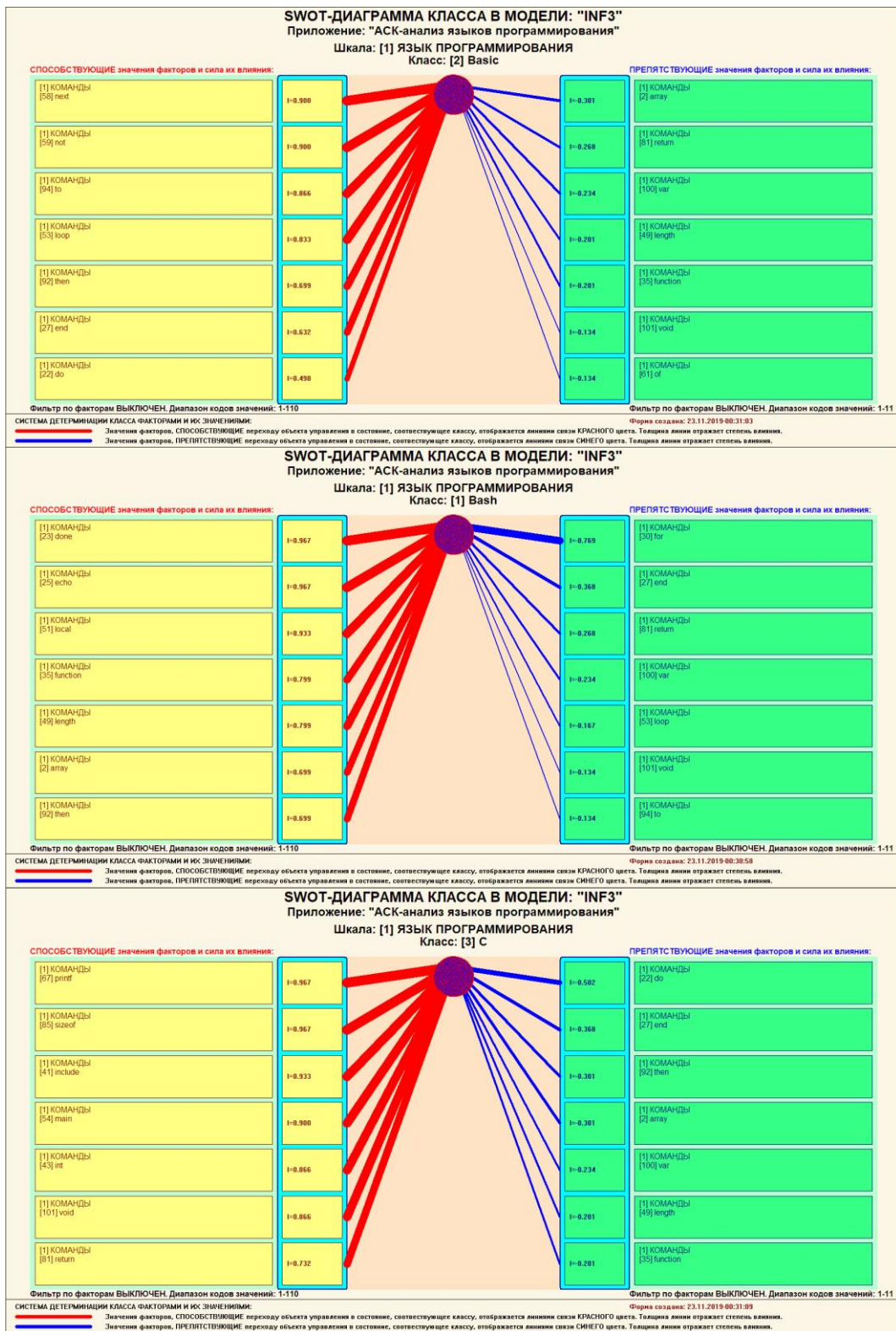
Подзадача 4.2. Поддержка принятия решений (SWOT-анализ)

При принятии решений определяется сила и направление влияния значений факторов на принадлежность состояний объекта моделирования к тем или иным классам, соответствующим различным будущим состояниям. По сути это решение задачи SWOT-анализа [11].

Применительно к задаче, решаемой в данной работе, SWOT-анализ показывает степень влияния различных команд на идентификацию языков программирования.

В системе «Эйдос» в режиме 4.4.8 поддерживается решение этой задачи. При этом **выявляется система детерминации заданного класса**, т.е. система значений факторов, обуславливающих переход объекта моделирования и управления в состояние, соответствующее данному классу, а также препятствующих этому переходу.

На рисунках 14 приведены SWOT-диаграммы, наглядно отражающие силу и направление влияния различных команд на идентификацию языка программирования.



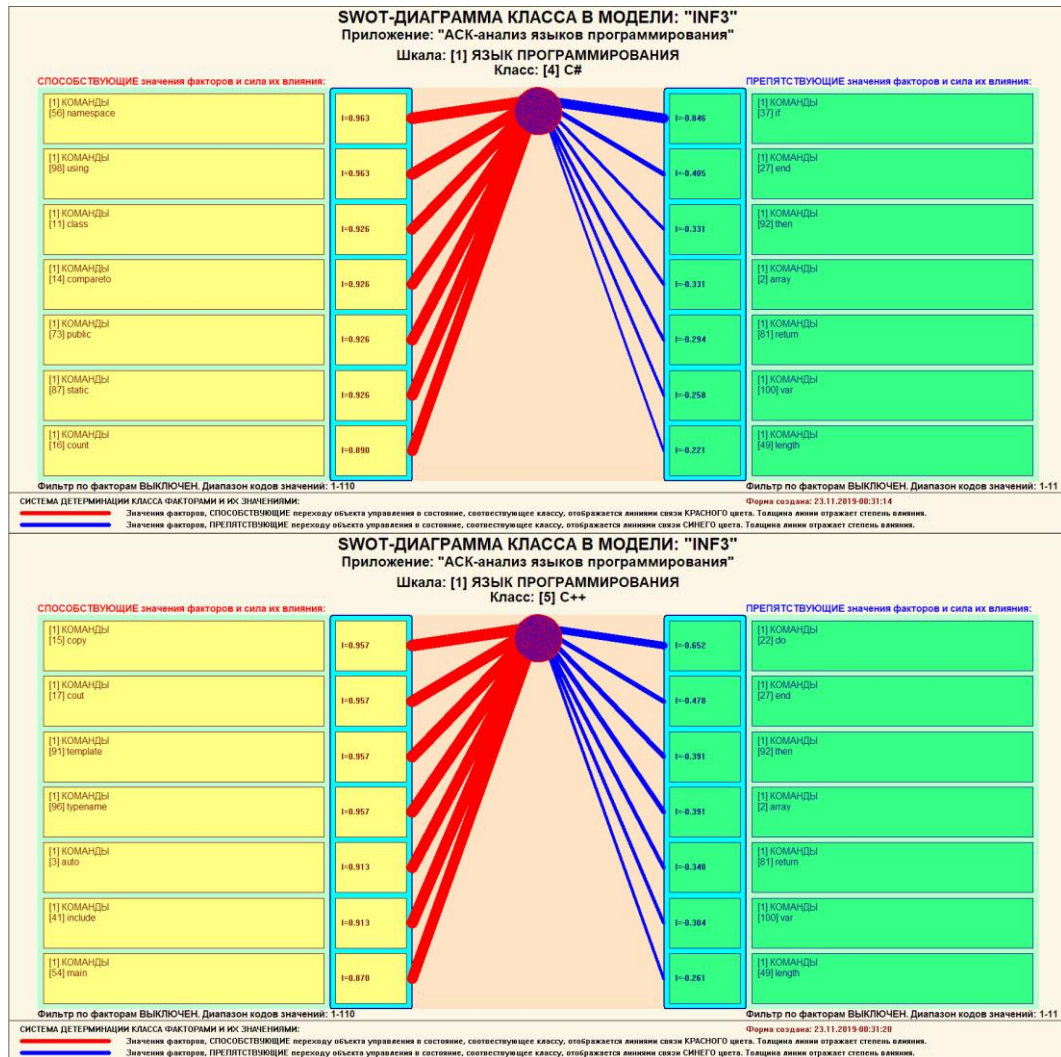


Рисунок 14. SWOT-диаграммы идентификации языка программирования

Эти SWOT-диаграммы наглядно отражают силу и направление влияния различных команд на идентификацию языка программирования.

Отметим также, что система «Эйдос» обеспечивала решение этой *всегда*, т.е. даже в самых ранних DOS-версиях и в реализациях системы «Эйдос» на других языках и типах компьютеров. Например, первый акт внедрения системы «Эйдос», где об этом упоминается в явном виде, датируется 1987 годом, а первый подобный расчет относится к 1981 году.

УТВЕРЖДАЮ
Заведующий Краснодарским
сектором ИСИ АН СССР, к.ф.н.
А.А. Хагуров
1987г.



А К Т

УТВЕРЖДАЮ
Директор Северо-Кавказского филиала
ВНИИ "АИУС-агроресурсы", к.э.н.
Э.М. Трахов
1987г.

Настоящий акт составлен комиссией в составе: Кириченко М.М., Ляшко Г.А., Самсонов Г.А., Коренец В.И., Луценко Е.В. в том, что в соответствии с договором о научно-техническом сотрудничестве между Северо-Кавказским филиалом ВНИИ "АИУС-агроресурсы" и Краснодарским сектором Института социологических исследований АН СССР Северо-Кавказским филиалом ВНИИ "АИУС-агроресурсы" выполнены следующие работы:

- осуществлена постановка задачи: "Обработка на ЭВМ социологических анкет Крайагропрома";
- разработаны математическая модель и программное обеспечение подсистемы распознавания образов, позволяющие решать данную задачу в среде персональной технологической системы ВЕГА-М;
- на профессиональной персональной ЭВМ "Искра-226" осуществлены расчёты по задаче в объёме:

Входная информация составила 425 анкет по 9-ти предприятиям.
Выходная информация - 4 вида выходных форм объёмом 90 листов формата А3 и 20 листов формата А4 содержит:

- процентное распределение ответов в разрезе по социальным типам корреспондентов;
- распределение информативностей признаков (в битах) для распознавания социальных типов корреспондентов;
- позитивные и негативные информационные портреты 30-ти социальных типов на языке 212 признаков;
- обобщённая характеристика информативности признаков для выбора такого минимального набора признаков, который содержит максимум информации о распознаваемых объектах (оптимизация анкет).

Работы выполнены на высоком научно-методическом уровне и в срок.

От ИСИ АН СССР:
Мл.научный сотрудник
Кириченко М.М.
19.05.1987г.

Мл.научный сотрудник
Ляшко Г.А.
19.05.1987г.

От СКФ ВНИИ "АИУС-агроресурсы":
Зав.отделом аэрокосмических и
тематических изысканий №4, к.э.н.
Самсонов Г.А.
19.05.1987г.

Главный конструктор проекта
Коренец В.И.
19.05.87.1987г.

Главный конструктор проекта
Луценко Е.В.
19.05.87.1987г.

Но тогда SWOT-диаграммы назывались позитивным и негативным информационными портретами классов.

Информация о системе значений факторов, обуславливающих переход объекта моделирования в различные будущие состояния, соответствующие классам, может быть приведена не только в диаграммах, показанных на рисунках 14, но и во многих других табличных и графических формах, которые в данной работе не приводятся только из-за ограниченности ее объема. В частности, в этих формах может быть

выведена значительно более полная информация (в т.ч. вообще вся имеющая в модели). Подобная подробная информация содержится в базах данных, расположенных по пути: c:\Aidos-X\AID_DATA\A0000001\System\SWOTCls####Infl.DBF, где: «####» – код класса с ведущими нулями. Эти базы открываются в MS Excel.

В заключение отметим, что SWOT-анализ является широко известным и общепризнанным методом стратегического планирования. Однако это не мешает тому, что он подвергается критике, часто вполне справедливой, обоснованной и хорошо аргументированной. В результате критического рассмотрения SWOT-анализа выявлено довольно много его слабых сторон (недостатков), источником которых чаще всего является необходимость привлечения экспертов, в частности для оценки силы и направления влияния факторов. Ясно, что эксперты это делают неформализуемым путем (интуитивно), на основе своего опыта и профессиональной компетенции. Но возможности экспертов имеют свои ограничения и часто по различным причинам они не могут и не хотят это сделать. Таким образом, возникает проблема проведения SWOT-анализа без привлечения экспертов. Эта проблема может решаться путем автоматизации функций экспертов, т.е. путем измерения силы и направления влияния факторов непосредственно на основе эмпирических данных. Подобная технология разработана давно, ей уже более 30 лет, но к сожалению, она сравнительно малоизвестна – это интеллектуальная система «Эйдос» [11, 18, 19].

Подзадача 4.3. Исследование моделируемой предметной области путем исследования ее модели

Если модель предметной области достоверна, то исследование модели можно считать исследованием самого моделируемого объекта, т.е. результаты исследования модели корректно относить к самому объекту моделирования, «переносить на него».

В системе «Эйдос» есть довольно много возможностей для такого исследования, но в данной работе из-за ограничений на ее объем мы рассмотрим лишь результаты кластерно-конструктивного анализа классов и признаков (когнитивные диаграммы и дендрограммы), а также нелокальные нейроны, нелокальные нейронные сети, 3d-интегральные когнитивные карты и когнитивные функции.

4.3.1. Когнитивные диаграммы классов

Эти диаграммы отражают сходство/различие классов. Мы получаем их в режимах 4.2.2.1 и 4.2.2.2 (рисунок 15).

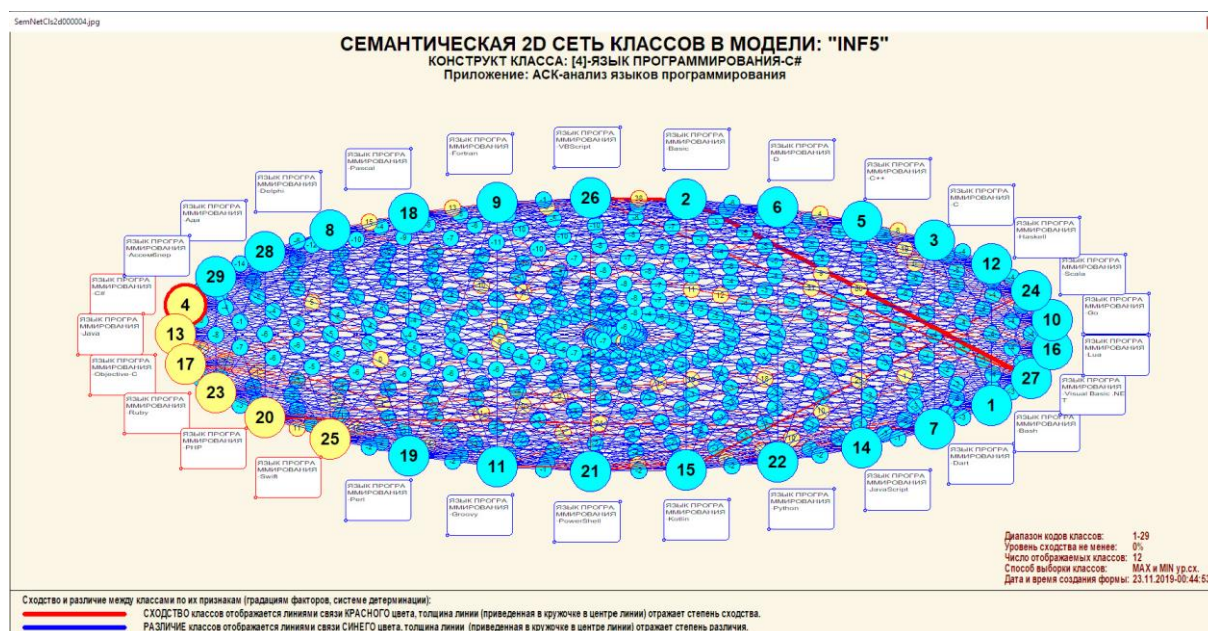


Рисунок 15. Когнитивная диаграмма классов, отражающая сходство/различие языков программирования, по используемым в них командам

В системе «Эйдос» есть возможность управлять параметрами формирования и вывода изображения, приведенного на рисунке 15. Для этого используется диалоговое окно, приведенное на рисунке 16.

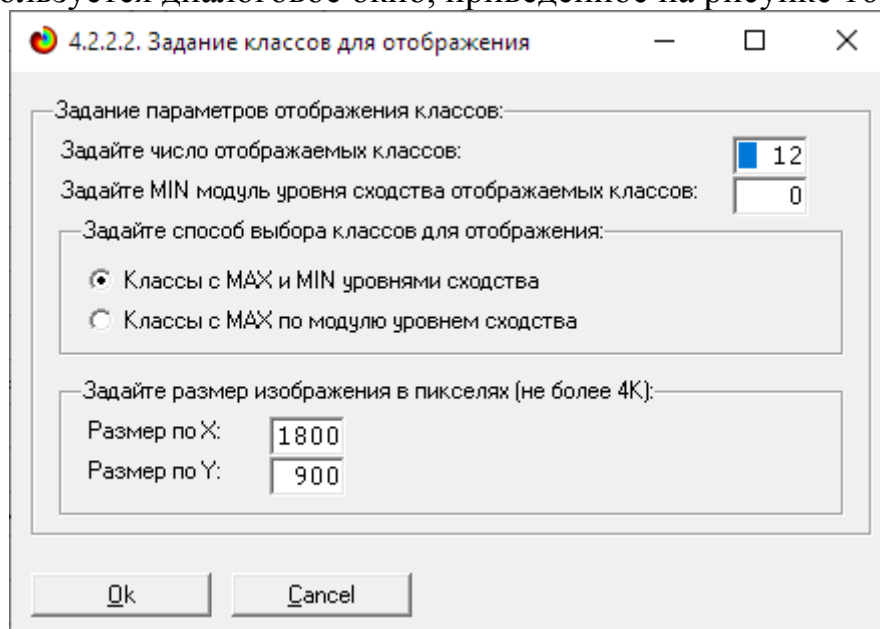


Рисунок 156. Диалоговое окно управления параметрами формирования и вывода изображения когнитивной диаграммы классов

4.3.2. Агломеративная когнитивная кластеризация классов

Информация о сходстве/различии классов, содержащаяся в матрице сходства, может быть визуализирована не только в форме, когнитивных диаграмм, пример которой приведен на рисунке 15, но и в форме

агломеративных дендрограмм, полученных в результате *когнитивной кластеризации* [5] (рисунок 17):

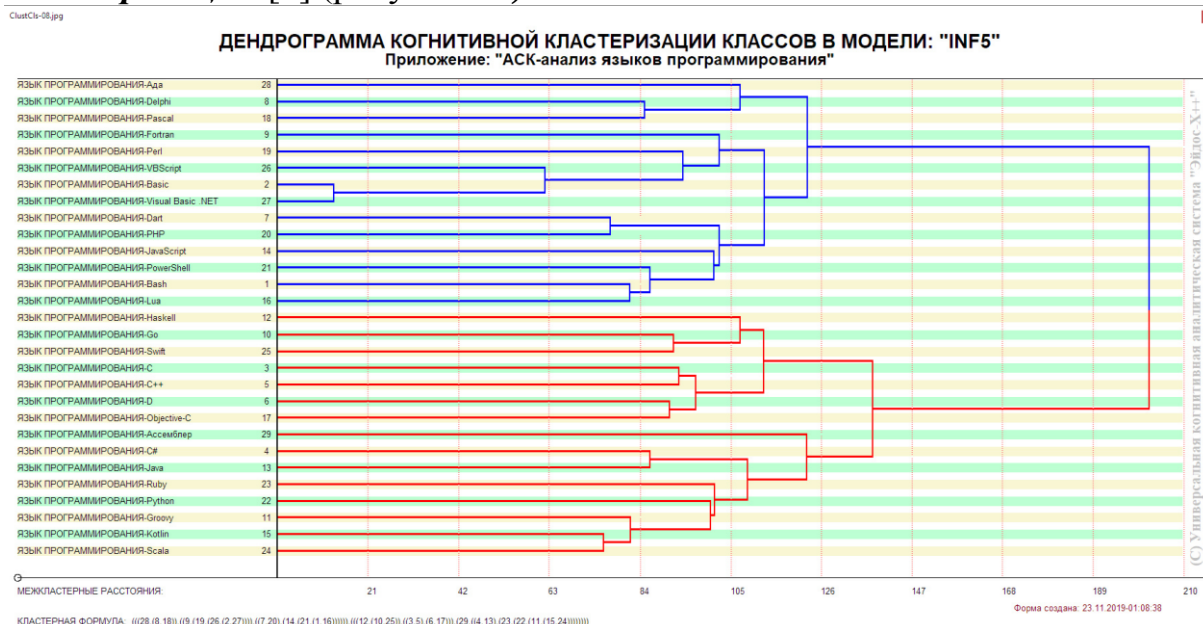


Рисунок 167. Дендрограмма когнитивной агломеративной кластеризации, отражающая сходство/различие различных языков программирования

Из рисунка 17 мы видим, что языки программирования VBScript, Basic и Visual Basic .NET объединены в один кластер, что не удивительно, так как VBScript и Visual Basic .NET являются потомками языка Basic и перенимают большую часть его синтаксиса.

В один кластер объединены языки Lua, Bash и PowerShell, все они являются языками для написания скриптов.

Так же неудивительно, что в один кластер попали языки C, C++ и Objective-C.

Высокий уровень сходства наблюдается между языками Java и C#, что соответствует действительности, так как C# был разработан компанией Microsoft для конкуренции с Java, вследствие чего имеет похожий синтаксис.

Можно заметить сходство языков Delphi и Pascal. Это объясняется тем, что Delphi является диалектом языка программирования Object Pascal, разработанного Apple на основе Pascal.

На рисунке 18 мы видим график изменения межкластерных расстояний:

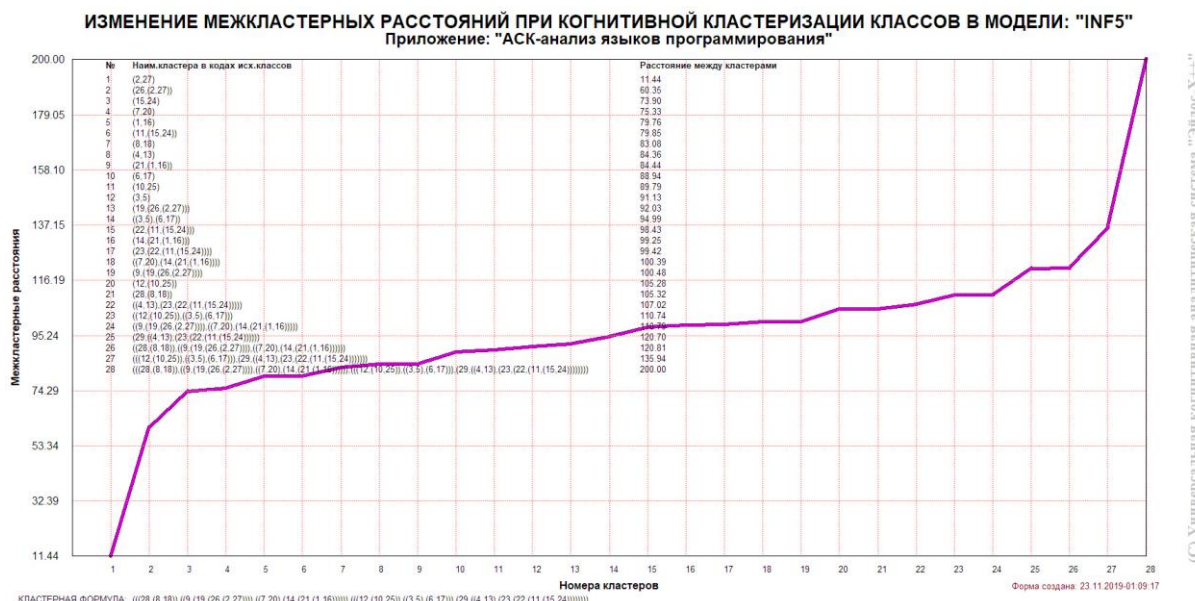


Рисунок 178. График изменения межкластерных расстояний

4.3.3. Нелокальные нейроны и нелокальные нейронные сети

На рисунке 19 приведены пример нелокального нейрона, а на рисунке 20 и фрагмент одного слоя нелокальной нейронной сети:

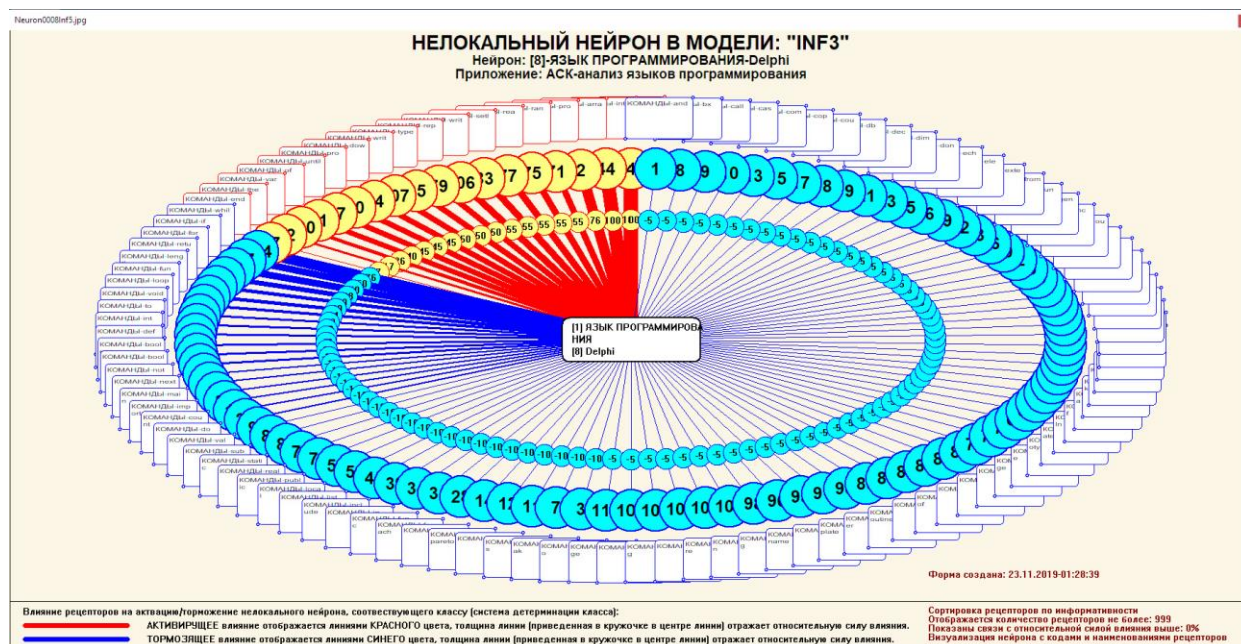


Рисунок 1918. Пример нелокального нейрона, отражающего команды, помогающие идентифицировать язык программирования Delphi

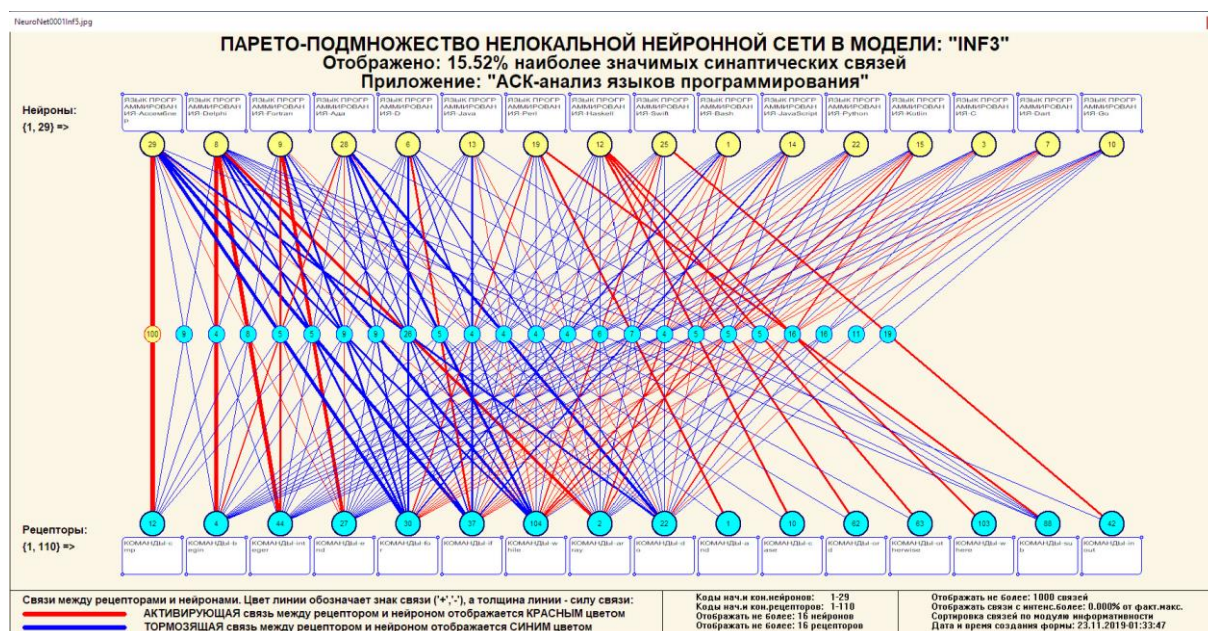


Рисунок 20. Один слой нелокальной нейронной сети, отражающий силу и направление влияния команд на идентификацию языков программирования (фрагмент около 15,5% сети)

В приведенном фрагменте слоя нейронной сети нейрон соответствует языку программирования, а рецепторы – командам. Нейроны расположены слева направо в порядке убывания силы детерминации, т.е. слева находятся результаты, наиболее жестко обусловленные обуславливающими их значениями факторами, а с права – менее жестко обусловленные.

Модель знаний системы «Эйдос» относится к **нечетким декларативным** гибридным моделям и объединяет в себе некоторые особенности нейросетевой [13, 20] и фреймовой моделей представления знаний [20]. Классы в этой модели соответствуют нейронам и фреймам, а признаки рецепторам и шпациям (описательные шкалы – слотам).

От фреймовой модели представления знаний модель системы «Эйдос» отличается своей эффективной и простой программной реализацией, полученной за счет того, что разные фреймы отличаются друг от друга не набором слотов и шпаций, а лишь информацией в них. Поэтому в системе «Эйдос» при увеличении числа фреймов само количество баз данных не увеличивается, а увеличивается лишь их размерность.

От нейросетевой модели представления знаний модель системы «Эйдос» отличается тем, что [13, 20]:

1) весовые коэффициенты на рецепторах не подбираются итерационным методом обратного распространения ошибки, а считаются прямым счетом на основе хорошо теоретически обоснованной модели, основанной на теории информации (это напоминает байесовские сети);

2) весовые коэффициенты имеют хорошо теоретически обоснованную содержательную интерпретацию, основанную на теории информации;

3) нейросеть является нелокальной, как сейчас говорят «полносвязной».

4.3.4. 3d-интегральные когнитивные карты

На рисунке 21 приведен фрагмент 3d-интегральной когнитивной карты, отражающая фрагмент около 15,5% СК-модели Inf3.

3d-интегральная когнитивная карта является отображением на одном рисунке когнитивных диаграмм классов и значений факторов, и одного слоя нейронной сети, приведенного на рисунке 20.



Рисунок 191. 3d-интегральная когнитивная карта в СК-модели Inf3

4.3.5. Когнитивные функции

Вместо описания того, что представляют собой когнитивные функции, приведем help соответствующего режима системы «Эйдос» (рисунок 22) и сошлемся на работу, в которой это довольно подробно описано [14].

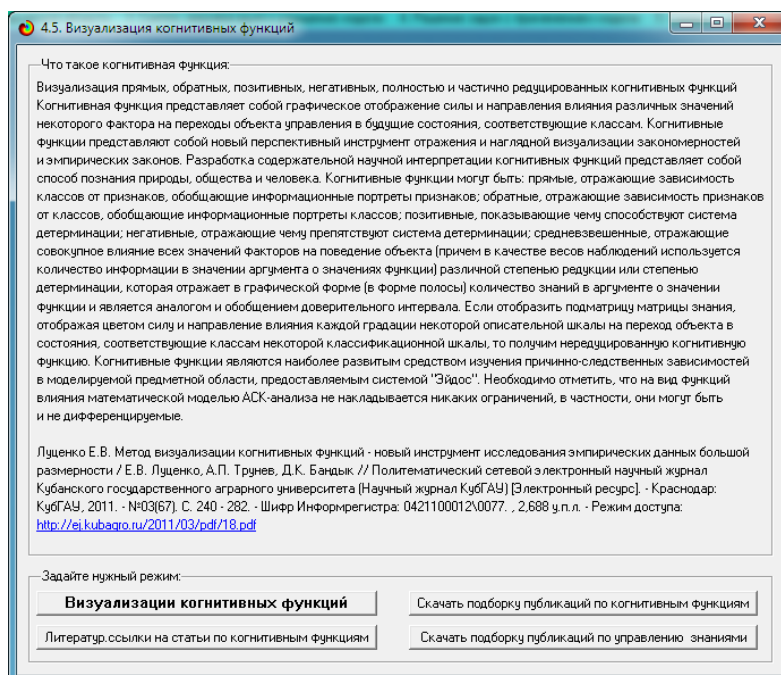


Рисунок 22. Help режима визуализации когнитивных функций

Когнитивная функция представляет собой графическое отображение силы и направления влияния различных значений некоторого фактора (признаков) на переходы объекта управления в будущие состояния, соответствующие классам. Классы являются градациями классификационных шкал.

Когнитивные функции представляют собой новый перспективный инструмент отражения и наглядной визуализации эмпирических закономерностей и эмпирических законов. Разработка содержательной научной интерпретации когнитивных функций представляет собой способ познания природы, общества и человека [17].

Когнитивные функции могут быть: прямые, отражающие зависимость классов от признаков, обобщающие информационные портреты признаков; обратные, отражающие зависимость признаков от классов, обобщающие информационные портреты классов; позитивные, показывающие чему способствуют система детерминации (обозначены белой линией); негативные, отражающие чему препятствуют система детерминации (обозначены черной линией); средневзвешенные, отражающие совокупное влияние всех значений факторов на поведение объекта (причем в качестве весов наблюдений используется количество информации в значении аргумента о значениях функции) различной степенью редукции или степенью детерминации, которая отражает в графической форме (в форме полосы разной толщины) количество знаний в аргументе о значении функции и является аналогом и обобщением доверительного интервала.

Если отобразить подматрицу матрицы знания, отображая цветом силу и направление влияния каждой градации некоторой описательной шкалы на переход объекта в состояния, соответствующие классам некоторой классификационной шкалы, то получим нередуцированную когнитивную функцию.

Когнитивные функции являются наиболее развитым средством изучения причинно-следственных зависимостей в моделируемой предметной области, предоставляемым системой "Эйдос".

Необходимо отметить, что *на вид функций влияния математической моделью АСК-анализа не накладывается никаких ограничений*, в частности, они могут быть и не дифференцируемые [14].

На рисунке 23 приведен примеры когнитивной функции, наглядно отражающей силу и направление влияния команд на идентификацию языков программирования.

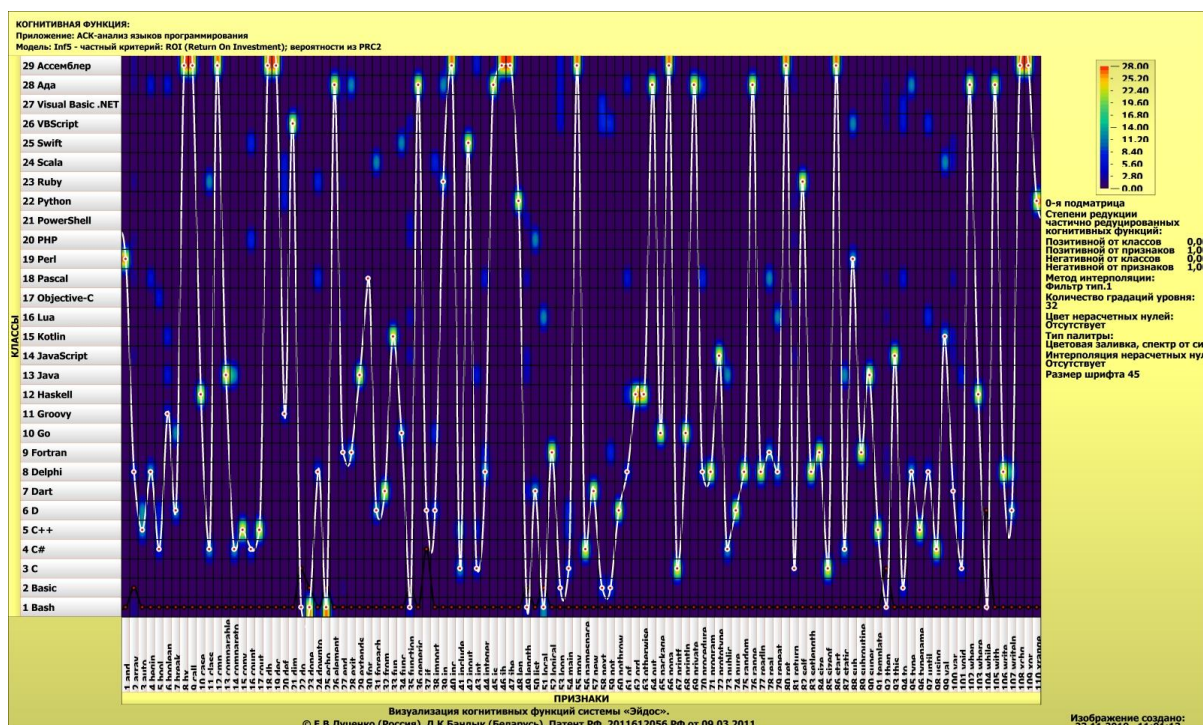


Рисунок 23. Пример когнитивной функций в СК-модели INF5, отражающей силу и направление влияния различных команд на идентификацию языков программирования.

Из когнитивной функции, приведенной на рисунке 23, хорошо видно, что проще всего идентифицировать низкоуровневый язык

программирования Ассемблер, так как его синтаксис значительно отличается от синтаксиса высокоуровневых языков программирования.

4.3.6. Сила и направление влияния различных команд на идентификацию языков программирования

На рисунках 5, 6, 7 приведены фрагменты некоторых статистических и системно-когнитивных моделей, отражающих моделируемую предметную область.

Строки матриц моделей соответствуют значениям признаков, т.е. командам языков программирования (градации описательных шкал).

Колонки матриц моделей соответствуют названиям языков программирования (градации классификационных шкал).

Числовые значения в ячейках матриц моделей, находящихся на пересечении строк и колонок, отражают направление (знак) и силу влияния конкретной команды на идентификацию конкретного языка программирования, соответствующего колонке.

Если какая-то команда слабо влияет на идентификацию языка программирования, то в соответствующей строке матрицы модели будут малые по модулю значения разных знаков, если же влияние сильное – то и значения будут большие по модулю разных знаков.

Если команда способствует идентификации некоторого определенного языка программирования, то в соответствующей этому результату ячейке матрицы модели будут положительные значения, если же понижает – то и значения будут отрицательные.

Из этого следует, что суммарную силу влияния той или иной команды на идентификацию языка программирования можно количественно оценивать *степенью вариабельности значений* в строке матрицы модели, соответствующей этой команде.

Существует много мер вариабельности значений: это и среднее модулей отклонения от среднего, и дисперсия, и среднеквадратичное отклонение, и другие. В АСК-анализе и системе «Эйдос» для этой цели принято использовать среднеквадратичное отклонение. Численно оно равно стандартному отклонению и вычисляется по той же формуле, но мы предпочитаем не использовать термин «стандартное отклонение», т.к. он предполагает нормальность распределения исследуемых последовательностей чисел, а значит и проверку соответствующих статистических гипотез.

Самая правая колонка в матрицах моделей на рисунках 5, 6, 7 содержит количественную оценку вариабельности значений строки модели (среднеквадратичное отклонение), которая и представляет собой ценность команды для решения задачи идентификации языков программирования.

Если рассортировать матрицу модели по этой самой правой колонке в порядке убывания, а потом просуммировать значения в ней

нарастающим итогом, то получим логистическую Парето-кривую, отражающую зависимость ценности модели от числа наиболее ценных признаков в ней (рисунок 24, таблица 7).

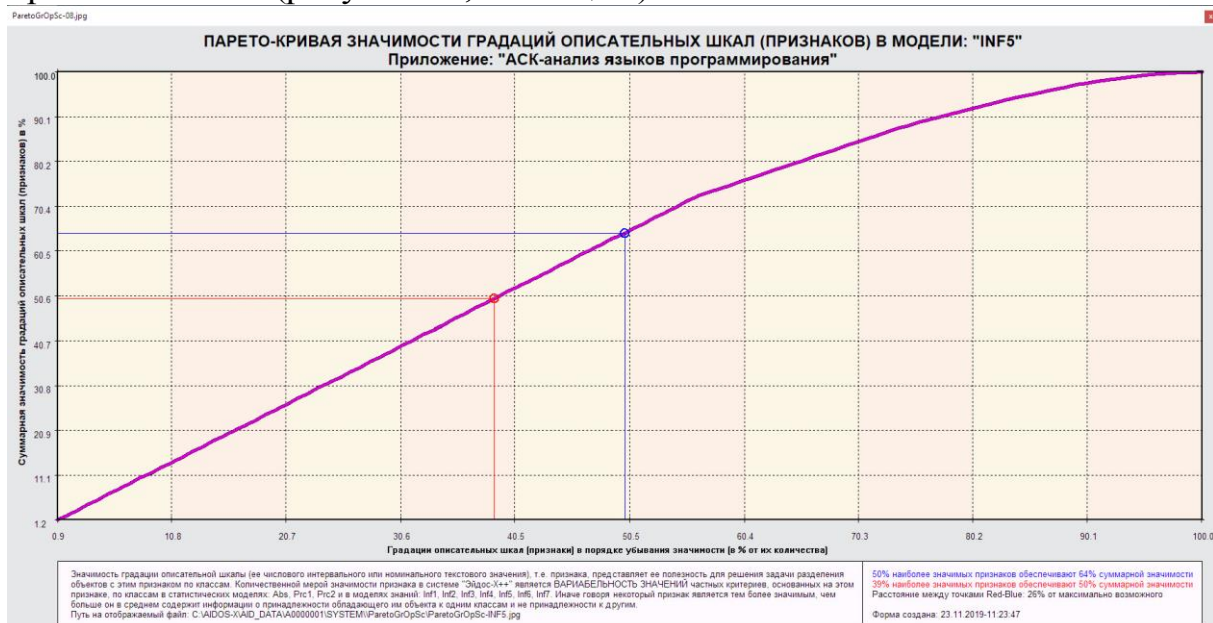


Рисунок 2420. Парето-кривая значимости градаций описательных шкал

Таблица 7 – Парето-таблица значимости градаций описательных шкал, т.е. сила влияния различных команд на идентификацию языков программирования в СК-модели INF5

№	Код значения параметра	Наименование параметра	Код параметра	Ценность значения параметра (%)	Ценность значения параметра нарастающим итогом (%)
1	20	КОМАНДЫ-def	1	6,6717449	6,6717449
2	6	КОМАНДЫ-boolean	1	6,2767639	12,9485088
3	104	КОМАНДЫ-while	1	6,2305898	19,1790986
4	88	КОМАНДЫ-sub	1	6,1585568	25,3376554
5	1	КОМАНДЫ-and	1	6,1277389	31,4653943
6	30	КОМАНДЫ-for	1	5,6855091	37,1509034
7	81	КОМАНДЫ-return	1	4,4594675	41,6103709
8	35	КОМАНДЫ-function	1	4,4266827	46,0370536
9	49	КОМАНДЫ-length	1	4,1741428	50,2111964
10	5	КОМАНДЫ-bool	1	3,9477174	54,1589138
11	37	КОМАНДЫ-if	1	3,9191369	58,0780507
12	100	КОМАНДЫ-var	1	3,7849212	61,8629719
13	10	КОМАНДЫ-case	1	3,6518012	65,5147731
14	62	КОМАНДЫ-ord	1	3,6518012	69,1665743
15	63	КОМАНДЫ-otherwise	1	3,6518012	72,8183755
16	103	КОМАНДЫ-where	1	3,6518012	76,4701767
17	61	КОМАНДЫ-of	1	3,5879988	80,0581755
18	34	КОМАНДЫ-func	1	3,5115148	83,5696903
19	38	КОМАНДЫ-import	1	3,3763391	86,9460294
20	27	КОМАНДЫ-end	1	3,3633229	90,3093523
21	16	КОМАНДЫ-count	1	3,3264625	93,6358148
22	99	КОМАНДЫ-val	1	3,0510136	96,6868284
23	92	КОМАНДЫ-then	1	2,9278849	99,6147133
24	58	КОМАНДЫ-next	1	2,8808273	102,4955406

25	94	КОМАНДЫ-to	1	2,8710705	105,3666111
26	22	КОМАНДЫ-do	1	2,8519644	108,2185755
27	7	КОМАНДЫ-break	1	2,8034146	111,0219901
28	2	КОМАНДЫ-array	1	2,7571937	113,7791838
29	53	КОМАНДЫ-loop	1	2,7351121	116,5142959
30	42	КОМАНДЫ-inout	1	2,5906850	119,1049809
31	48	КОМАНДЫ-len	1	2,5906850	121,6956659
32	65	КОМАНДЫ-package	1	2,5906850	124,2863509
33	68	КОМАНДЫ-println	1	2,5906850	126,8770359
34	110	КОМАНДЫ-xrange	1	2,5906850	129,4677209
35	50	КОМАНДЫ-list	1	2,5538984	132,0216193
36	51	КОМАНДЫ-local	1	2,5538984	134,5755177
37	31	КОМАНДЫ-foreach	1	2,5145644	137,0900821
38	101	КОМАНДЫ-void	1	2,4616047	139,5516868
39	59	КОМАНДЫ-not	1	2,3989925	141,9506793
40	54	КОМАНДЫ-main	1	2,3388141	144,2894934
41	43	КОМАНДЫ-int	1	2,3078741	146,5973675
42	11	КОМАНДЫ-class	1	2,2923144	148,8896819
43	97	КОМАНДЫ-until	1	2,2613028	151,1509847
44	33	КОМАНДЫ-fun	1	2,2590862	153,4100709
45	41	КОМАНДЫ-include	1	2,1311859	155,5412568
46	79	КОМАНДЫ-repeat	1	2,0228467	157,5641035
47	72	КОМАНДЫ-prototype	1	2,0011760	159,5652795
48	93	КОМАНДЫ-this	1	2,0011760	161,5664555
49	14	КОМАНДЫ-compareto	1	2,0002889	163,5667444
50	73	КОМАНДЫ-public	1	2,0002889	165,5670333
51	87	КОМАНДЫ-static	1	2,0002889	167,5673222
52	44	КОМАНДЫ-integer	1	1,9792052	169,5465274
53	24	КОМАНДЫ-downto	1	1,9641961	171,5107235
54	39	КОМАНДЫ-in	1	1,8524303	173,3631538
55	12	КОМАНДЫ-cmp	1	1,8304768	175,1936306
56	3	КОМАНДЫ-auto	1	1,8112023	177,0048329
57	23	КОМАНДЫ-done	1	1,7948479	178,7996808
58	25	КОМАНДЫ-echo	1	1,7948479	180,5945287
59	32	КОМАНДЫ-from	1	1,7948479	182,3893766
60	57	КОМАНДЫ-new	1	1,7948479	184,1842245
61	67	КОМАНДЫ-printf	1	1,7948479	185,9790724
62	82	КОМАНДЫ-self	1	1,7948479	187,7739203
63	85	КОМАНДЫ-sizeof	1	1,7948479	189,5687682
64	78	КОМАНДЫ-real	1	1,7304263	191,2991945
65	4	КОМАНДЫ-begin	1	1,6418413	192,9410358
66	56	КОМАНДЫ-namespace	1	1,6260340	194,5670698
67	98	КОМАНДЫ-using	1	1,6260340	196,1931038
68	28	КОМАНДЫ-exit	1	1,5718322	197,7649360
69	52	КОМАНДЫ-logical	1	1,4853557	199,2502917
70	84	КОМАНДЫ-size	1	1,4853557	200,7356474
71	89	КОМАНДЫ-subroutine	1	1,4853557	202,2210031
72	107	КОМАНДЫ-writeln	1	1,4360863	203,6570894
73	13	КОМАНДЫ-comparable	1	1,3663202	205,0234096
74	15	КОМАНДЫ-copy	1	1,3663202	206,3897298
75	17	КОМАНДЫ-cout	1	1,3663202	207,7560500
76	21	КОМАНДЫ-dim	1	1,3663202	209,1223702
77	29	КОМАНДЫ-extends	1	1,3663202	210,4886904
78	60	КОМАНДЫ-nothrow	1	1,3663202	211,8550106
79	74	КОМАНДЫ-pure	1	1,3663202	213,2213308
80	90	КОМАНДЫ-super	1	1,3663202	214,5876510
81	91	КОМАНДЫ-template	1	1,3663202	215,9539712
82	96	КОМАНДЫ-typename	1	1,3663202	217,3202914
83	70	КОМАНДЫ-procedure	1	1,2699072	218,5901986
84	95	КОМАНДЫ-type	1	0,9521949	219,5423935
85	8	КОМАНДЫ-bx	1	0,9152384	220,4576319
86	9	КОМАНДЫ-call	1	0,9152384	221,3728703
87	18	КОМАНДЫ-db	1	0,9152384	222,2881087

88	19	КОМАНДЫ-dec	1	0,9152384	223,2033471
89	40	КОМАНДЫ-inc	1	0,9152384	224,1185855
90	46	КОМАНДЫ-jb	1	0,9152384	225,0338239
91	47	КОМАНДЫ-jbe	1	0,9152384	225,9490623
92	55	КОМАНДЫ-mov	1	0,9152384	226,8643007
93	66	КОМАНДЫ-popa	1	0,9152384	227,7795391
94	80	КОМАНДЫ-ret	1	0,9152384	228,6947775
95	86	КОМАНДЫ-start	1	0,9152384	229,6100159
96	108	КОМАНДЫ-xchg	1	0,9152384	230,5252543
97	109	КОМАНДЫ-xor	1	0,9152384	231,4404927
98	26	КОМАНДЫ-element	1	0,7819642	232,2224569
99	36	КОМАНДЫ-generic	1	0,7819642	233,0044211
100	45	КОМАНДЫ-is	1	0,7819642	233,7863853
101	64	КОМАНДЫ-out	1	0,7819642	234,5683495
102	69	КОМАНДЫ-private	1	0,7819642	235,3503137
103	76	КОМАНДЫ-range	1	0,7819642	236,1322779
104	102	КОМАНДЫ-when	1	0,7819642	236,9142421
105	105	КОМАНДЫ-with	1	0,7819642	237,6962063
106	71	КОМАНДЫ-program	1	0,7116251	238,4078314
107	75	КОМАНДЫ-random	1	0,7116251	239,1194565
108	77	КОМАНДЫ-readln	1	0,7116251	239,8310816
109	83	КОМАНДЫ-setlength	1	0,7116251	240,5427067
110	106	КОМАНДЫ-write	1	0,7116251	241,2543318

Из рисунка 24 и таблицы 7 видно, что 50% наиболее ценных для решения задачи идентификации языков программирования команд обуславливают 64% суммарной ценности, а 50% суммарной ценности обеспечиваются 39% наиболее ценными командами.

4.3.7. Степень детерминированности идентификации языков программирований различными командами

Степень детерминированности (обусловленности) класса в системе «Эйдос» количественно оценивается **степенью вариабельности значений факторов** (градаций описательных шкал) в колонке матрицы модели, соответствующей данному классу (таблица 8).

В данной работе у нас классами являются языки программирования, а значениями градаций описательных шкал – различные команды этих языков.

На рисунке 25 мы видим Парето-кривую степени детерминированности классов нарастающим итогом.

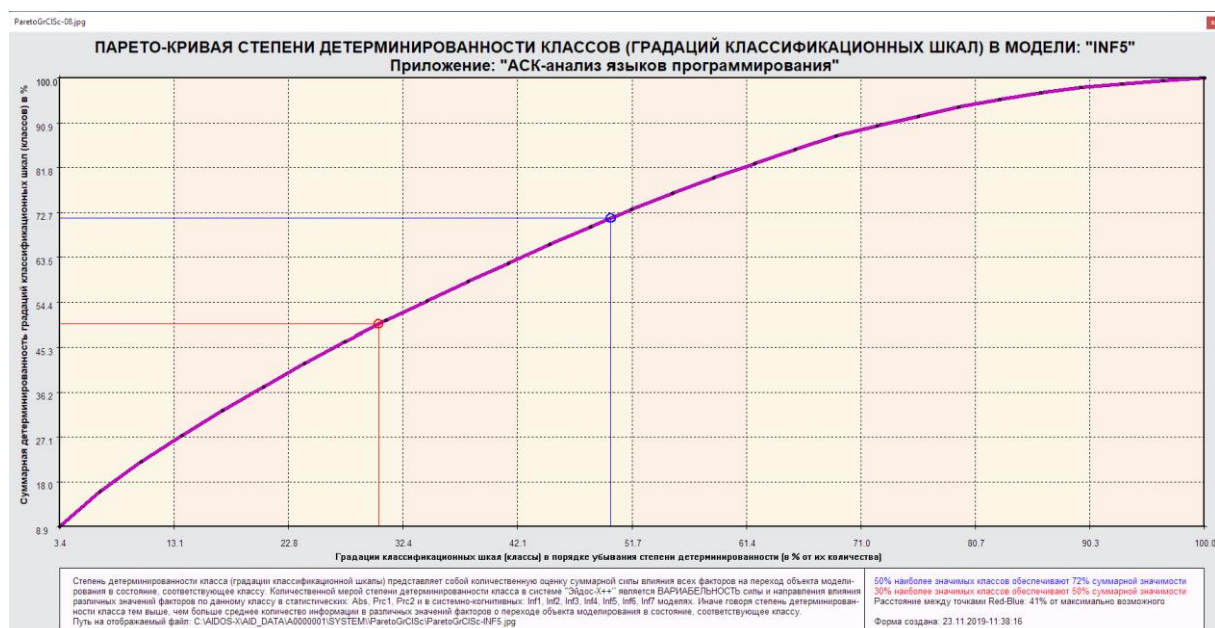


Рисунок 25. Парето-кривая степени детерминированности классов

Таблица 8 – Парето-таблица степеней детерминированности (обусловленности) классов (языков программирования) в СК-модели INF5

№	Код класса	Наименование класса	Код класс. шкалы	Степень детерминированности класса (%)	Степень детерминированности класса нарастающим итогом (%)
1	29	ЯЗЫК ПРОГРАММИРОВАНИЯ-Ассемблер	1	9,3556120	9,3556120
2	28	ЯЗЫК ПРОГРАММИРОВАНИЯ-Ада	1	7,5436002	16,8992122
3	8	ЯЗЫК ПРОГРАММИРОВАНИЯ-Delphi	1	6,4672004	23,3664126
4	5	ЯЗЫК ПРОГРАММИРОВАНИЯ-C++	1	5,6159777	28,9823903
5	12	ЯЗЫК ПРОГРАММИРОВАНИЯ-Haskell	1	5,2879851	34,2703754
6	13	ЯЗЫК ПРОГРАММИРОВАНИЯ-Java	1	5,0955638	39,3659392
7	9	ЯЗЫК ПРОГРАММИРОВАНИЯ-Fortran	1	4,9701216	44,3360608
8	6	ЯЗЫК ПРОГРАММИРОВАНИЯ-D	1	4,6393774	48,9754382
9	4	ЯЗЫК ПРОГРАММИРОВАНИЯ-C#	1	4,5598191	53,5352573
10	10	ЯЗЫК ПРОГРАММИРОВАНИЯ-Go	1	4,2084300	57,7436873
11	3	ЯЗЫК ПРОГРАММИРОВАНИЯ-C	1	4,0971823	61,8408696
12	1	ЯЗЫК ПРОГРАММИРОВАНИЯ-Bash	1	3,9810926	65,8219622
13	7	ЯЗЫК ПРОГРАММИРОВАНИЯ-Dart	1	3,9760640	69,7980262
14	22	ЯЗЫК ПРОГРАММИРОВАНИЯ-Python	1	3,8011320	73,5991582
15	14	ЯЗЫК ПРОГРАММИРОВАНИЯ-JavaScript	1	3,7973139	77,3964721
16	23	ЯЗЫК ПРОГРАММИРОВАНИЯ-Ruby	1	3,3453320	80,7418041
17	26	ЯЗЫК ПРОГРАММИРОВАНИЯ-VBScript	1	3,3277916	84,0695957
18	25	ЯЗЫК ПРОГРАММИРОВАНИЯ-Swift	1	3,0663758	87,1359715
19	15	ЯЗЫК ПРОГРАММИРОВАНИЯ-Kotlin	1	3,0155046	90,1514761
20	19	ЯЗЫК ПРОГРАММИРОВАНИЯ-Perl	1	2,9529394	93,1044155
21	18	ЯЗЫК ПРОГРАММИРОВАНИЯ-Pascal	1	2,1039283	95,2083438
22	24	ЯЗЫК ПРОГРАММИРОВАНИЯ-Scala	1	2,0765445	97,2848883
23	16	ЯЗЫК ПРОГРАММИРОВАНИЯ-Lua	1	2,0197915	99,3046798
24	20	ЯЗЫК ПРОГРАММИРОВАНИЯ-PHP	1	1,5900575	100,8947373

25	2	ЯЗЫК ПРОГРАММИРОВАНИЯ-Basic	1	1,3906030	102,2853403
26	27	ЯЗЫК ПРОГРАММИРОВАНИЯ-Visual Basic .NET	1	1,1338866	103,4192269
27	11	ЯЗЫК ПРОГРАММИРОВАНИЯ-Groovy	1	0,8414385	104,2606654
28	17	ЯЗЫК ПРОГРАММИРОВАНИЯ-Objective-C	1	0,6482287	104,9088941
29	21	ЯЗЫК ПРОГРАММИРОВАНИЯ-PowerShell	1	0,5197798	105,4286739

Чем выше степень детерминированности языка программирования, тем легче его идентифицировать. Как говорилось ранее, проще всего идентифицировать низкоуровневый язык программирования Ассемблер.

Выводы

Как показывает анализ результатов эксперимента по идентификации языков программирования, предложенное и реализованное в системе «Эйдос» решение поставленных задач является вполне эффективным, что позволяет обоснованно утверждать, что цель работы достигнута, поставленная проблема решена.

В результате проделанной работы, с помощью системы «Эйдос» были созданы 3 статистические и 5 системно-когнитивных моделей, в которых непосредственно сформированы обобщенные образы классов (идентифицированы языки программирования) по различным командам. Было изучено влияние различных команд на идентификацию языков программирования, и, на основе этого было проведено сравнение синтаксиса 29 языков программирования.

Со всеми моделями, созданными в данной статье, можно ознакомиться установив облачное Эйдос-приложение № 162 в режиме 1.3 системы «Эйдос».

Литература

1. Луценко Е.В. Синтез адаптивных интеллектуальных измерительных систем с применением АСК-анализа и системы «Эйдос» и системная идентификация в эконометрике, биометрии, экологии, педагогике, психологии и медицине / Е.В. Луценко // Политематический сетевой электронный научный журнал Кубанского государственного аграрного университета (Научный журнал КубГАУ) [Электронный ресурс]. – Краснодар: КубГАУ, 2016. – №02(116). С. 1 – 60. – IDA [article ID]: 1161602001. – Режим доступа: <http://ej.kubagro.ru/2016/02/pdf/01.pdf>, 3,75 у.п.л.
2. Луценко Е.В. Автоматизированный системно-когнитивный анализ в управлении активными объектами (системная теория информации и ее применение в исследовании экономических, социально-психологических, технологических и организационно-технических систем): Монография (научное издание). – Краснодар: КубГАУ. 2002. – 605 с. <http://elibrary.ru/item.asp?id=18632909>
3. Луценко Е.В. Метризация измерительных шкал различных типов и совместная сопоставимая количественная обработка разнородных факторов в системно-когнитивном анализе и системе «Эйдос» / Е.В. Луценко // Политематический сетевой

электронный научный журнал Кубанского государственного аграрного университета (Научный журнал КубГАУ) [Электронный ресурс]. – Краснодар: КубГАУ, 2013. – №08(092). С. 859 – 883. – IDA [article ID]: 0921308058. – Режим доступа: <http://ej.kubagro.ru/2013/08/pdf/58.pdf>, 1,562 у.п.л.

4. Луценко Е.В. Инвариантное относительно объемов данных нечеткое мультиклассовое обобщение F-меры достоверности моделей Ван Ризбергена в АСК-анализе и системе «Эйдос» / Е.В. Луценко // Политематический сетевой электронный научный журнал Кубанского государственного аграрного университета (Научный журнал КубГАУ) [Электронный ресурс]. – Краснодар: КубГАУ, 2017. – №02(126). С. 1 – 32. – IDA [article ID]: 1261702001. – Режим доступа: <http://ej.kubagro.ru/2017/02/pdf/01.pdf>, 2 у.п.л.

5. Луценко Е.В. Количественный автоматизированный SWOT- и PEST-анализ средствами АСК-анализа и интеллектуальной системы «Эйдос-X++» / Е.В. Луценко // Политематический сетевой электронный научный журнал Кубанского государственного аграрного университета (Научный журнал КубГАУ) [Электронный ресурс]. – Краснодар: КубГАУ, 2014. – №07(101). С. 1367 – 1409. – IDA [article ID]: 1011407090. – Режим доступа: <http://ej.kubagro.ru/2014/07/pdf/90.pdf>, 2,688 у.п.л.

6. Луценко Е.В. Метод когнитивной кластеризации или кластеризация на основе знаний (кластеризация в системно-когнитивном анализе и интеллектуальной системе «Эйдос») / Е.В. Луценко, В.Е. Коржаков // Политематический сетевой электронный научный журнал Кубанского государственного аграрного университета (Научный журнал КубГАУ) [Электронный ресурс]. – Краснодар: КубГАУ, 2011. – №07(071). С. 528 – 576. – Шифр Информрегистра: 0421100012\0253, IDA [article ID]: 0711107040. – Режим доступа: <http://ej.kubagro.ru/2011/07/pdf/40.pdf>, 3,062 у.п.л.

7. Луценко Е.В. Системная теория информации и нелокальные интерпретируемые нейронные сети прямого счета / Е.В. Луценко // Политематический сетевой электронный научный журнал Кубанского государственного аграрного университета (Научный журнал КубГАУ) [Электронный ресурс]. – Краснодар: КубГАУ, 2003. – №01(001). С. 79 – 91. – IDA [article ID]: 0010301011. – Режим доступа: <http://ej.kubagro.ru/2003/01/pdf/11.pdf>, 0,812 у.п.л.

8. Орлов А.И., Луценко Е.В. Системная нечеткая интервальная математика. Монография (научное издание). – Краснодар, КубГАУ. 2014. – 600 с. ISBN 978-5-94672-757-0. <http://elibrary.ru/item.asp?id=21358220>

9. Луценко Е.В., Открытая масштабируемая интерактивная интеллектуальная on-line среда «Эйдос» («Эйдос-online»). Свид. РосПатента РФ на программу для ЭВМ, Заявка № 2017618053 от 07.08.2017, Гос.рег.№ 2017661153, зарегистр. 04.10.2017. – Режим доступа: <http://lc.kubagro.ru/aidos/2017661153.jpg>, 2 у.п.л.

10. Луценко Е.В. Открытая масштабируемая интерактивная интеллектуальная on-line среда для обучения и научных исследований на базе АСК-анализа и системы «Эйдос» / Е.В. Луценко // Политематический сетевой электронный научный журнал Кубанского государственного аграрного университета (Научный журнал КубГАУ) [Электронный ресурс]. – Краснодар: КубГАУ, 2017. – №06(130). С. 1 – 55. – IDA [article ID]: 1301706001. – Режим доступа: <http://ej.kubagro.ru/2017/06/pdf/01.pdf>, 3,438 у.п.л. http://lc.kubagro.ru/aidos/Presentation_Aidos-online.pdf

11. Луценко Е.В. Проблемы и перспективы теории и методологии научного познания и автоматизированный системно-когнитивный анализ как автоматизированный метод научного познания, обеспечивающий содержательное феноменологическое моделирование / Е.В. Луценко // Политематический сетевой электронный научный журнал Кубанского государственного аграрного университета

(Научный журнал КубГАУ) [Электронный ресурс]. – Краснодар: КубГАУ, 2017. – №03(127). С. 1 – 60. – IDA [article ID]: 1271703001. – Режим доступа: <http://ej.kubagro.ru/2017/03/pdf/01.pdf>, 3,75 у.п.л.

12. Луценко Е.В. Универсальная автоматизированная система распознавания образов "Эйдос" (версия 4.1).-Краснодар: КЮИ МВД РФ, 1995.- 76с. <http://elibrary.ru/item.asp?id=18630282>

13. Луценко Е.В. Универсальная когнитивная аналитическая система «Эйдос». Монография (научное издание). – Краснодар, КубГАУ. 2014. – 600 с. ISBN 978-5-94672-830-0. <http://elibrary.ru/item.asp?id=22401787>

14. Луценко Е. В., Лойко В. И., Лаптев В. Н. Системы представления и приобретения знаний : учеб. пособие / Е. В. Луценко, В. И. Лойко, В. Н. Лаптев. – Краснодар : Экоинвест, 2018. – 513 с. ISBN 978-5-94215-415-8. <https://elibrary.ru/item.asp?id=35641755>

15. Луценко Е.В. Моделирование сложных многофакторных нелинейных объектов управления на основе фрагментированных зашумленных эмпирических данных большой размерности в системно-когнитивном анализе и интеллектуальной системе «Эйдос-Х++» / Е.В. Луценко, В.Е. Коржаков // Политематический сетевой электронный научный журнал Кубанского государственного аграрного университета (Научный журнал КубГАУ) [Электронный ресурс]. – Краснодар: КубГАУ, 2013. – №07(091). С. 164 – 188. – IDA [article ID]: 0911307012. – Режим доступа: <http://ej.kubagro.ru/2013/07/pdf/12.pdf>, 1,562 у.п.л.