

Приложение 2 к рабочей программе дисциплины
«Системы искусственного интеллекта»

МУНИЦИПАЛЬНОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«ВОЛЖСКИЙ ИНСТИТУТ ЭКОНОМИКИ, ПЕДАГОГИКИ И ПРАВА»

Факультет экономики и управления

Фонд оценочных средств
по дисциплине
«Системы искусственного интеллекта»

Специальность:
38.05.01 Экономическая безопасность

Специализация образовательной программы:
Экономико-правовое обеспечение экономической безопасности

Уровень высшего образования:
специалитет

Квалификация выпускника:
«экономист»

Оглавление

1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы, описание показателей, критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания.....	3
2. Методические материалы и типовые контрольные задания, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы.....	5
2.1. Типовые контрольные задания, используемые для проведения входного контроля.....	5
2.2. Типовые контрольные задания, используемые для промежуточной аттестации по дисциплине.....	5
2.2.1. Примерный перечень вопросов к зачету (зачету с оценкой).....	5
2.2.2. Примерное задание на зачет (зачет с оценкой).....	6
2.3. Методические материалы и типовые контрольные задания, используемые для текущего контроля по дисциплине.....	8
2.3.1. Методические материалы, используемые для текущего контроля знаний по дисциплине.....	8
2.3.2. Вопросы, выносимые на самостоятельное изучение.....	9
2.3.3. Задания для самостоятельной работы.....	9
2.3.4. Тесты по дисциплине.....	14
2.3.5. Лабораторные работы.....	16
2.3.6. Методика проведения лекции-беседы.....	40
2.3.7. Методика проведения работы в малых группах.....	41
3. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков по дисциплине.....	43

1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы, описание показателей, критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания

Перечень компетенций	Показатели оценивания компетенций	Критерии оценивания компетенций	Этапы формирования компетенций	Шкала оценивания
Способен использовать современные информационные технологии и программные средства при решении профессиональных задач (ОПК-6)	<ul style="list-style-type: none"> – знает понятия современных информационных технологий и программных средств; – умеет использовать современные программные средства при решении профессиональных задач; – владеет навыками использования современных информационных технологий при решении профессиональных задач 	<ul style="list-style-type: none"> – знает некоторые понятия современных информационных технологий и программных средств; – умеет при помощи преподавателя использовать современные программные средства при решении профессиональных задач; – владеет отдельными навыками использования современных информационных технологий при решении профессиональных задач 	Начальный	удовлетворительно (61 – 75 баллов)
		<ul style="list-style-type: none"> – знает основные понятия современных информационных технологий и программных средств; – умеет в большинстве случаев самостоятельно использовать современные программные средства при решении профессиональных задач; – владеет основными навыками использования современных информационных технологий при решении профессиональных задач 	Основной	хорошо (76 – 90 баллов)
		<ul style="list-style-type: none"> – знает понятия современных информационных технологий и программных средств; – умеет использовать современные программные средства при решении профессиональных задач; – владеет навыками использования современ- 	Завершающий	отлично (91 – 100 баллов)

Перечень компетенций	Показатели оценивания компетенций	Критерии оценивания компетенций	Этапы формирования компетенций	Шкала оценивания
		менных информационных технологий при решении профессиональных задач		
Способен понимать принципы работы современных информационных технологий и использовать их для решения задач профессиональной деятельности (ОПК-7)	<ul style="list-style-type: none"> – знает принципы работы современных информационных технологий; – умеет использовать современные информационные технологии для решения задач профессиональной деятельности; – владеет навыками работы с современными информационными технологиями при решении задач профессиональной деятельности 	<ul style="list-style-type: none"> – знает отдельные принципы работы современных информационных технологий; – умеет при помощи преподавателя использовать современные информационные технологии для решения задач профессиональной деятельности; – владеет некоторыми навыками работы с современными информационными технологиями при решении задач профессиональной деятельности 	Начальный	удовлетворительно (61 – 75 баллов)
		<ul style="list-style-type: none"> – знает основные принципы работы современных информационных технологий; – умеет самостоятельно в большинстве случаев использовать современные информационные технологии для решения задач профессиональной деятельности; – владеет основными навыками работы с современными информационными технологиями при решении задач профессиональной деятельности 	Основной	хорошо (76 – 90 баллов)
		<ul style="list-style-type: none"> – знает принципы работы современных информационных технологий; – умеет использовать современные информационные технологии для решения задач профессиональной деятельности; – владеет навыками работы с современными информационными технологиями при решении задач профессиональной деятельности 	Завершающий	отлично (91 – 100 баллов)

2. Методические материалы и типовые контрольные задания, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы

2.1. Типовые контрольные задания, используемые для проведения входного контроля

1. Задан алфавит А из букв кириллицы и алфавит В из десятичных цифр. Запишите несколько слов над алфавитами А и В, а также над алфавитом С, полученным слиянием этих двух алфавитов.

2. Система может находиться в любом из 10 равновозможных состояний. Какое количество информации содержит сообщение о том или ином конкретном состоянии системы? Указание: использовать формулу Хартли.

3. По отношению к результату исследования системы, информация бывает ...

- 1) исходная, постоянная, результирующая
- 2) исходная, внутренняя, выходная
- 3) закрытая, открытая, смешанная
- 4) исходная, промежуточная, результирующая.

4. Какой минимальной длины должна быть битовая комбинация для кодировки всех различных 111 символов, расположенных на клавиатуре? Указание: $2^6 = 64 < 111 < 128 = 2^7$.

5. Чему будет равно значение переменной s после завершения фрагмента программы:

```
s := 0 ; i := 1 ;  
while (i < 4) do  
begin i := i + 1 ; s := s + 1  
end ; ?
```

6. Алгоритм не обладает свойством ...

- 1) формализованности
- 2) массовости
- 3) детерминированности
- 4) индивидуальности.

7. Опишите простой тип данных «стол». Укажите атрибуты и операции, определенные для этого типа данных. Можно ли отнести «приставить к столу другой стол» к операциям? Указание: например, к атрибутам можно отнести «рабочая поверхность стола», к операциям – «поворот стола».

8. При решении задачи правильно:

- 1) данные и алгоритмы разрабатывать независимо друг от друга
- 2) данные разрабатывать после разработки и уточнения алгоритма
- 3) уточнять данные после каждого теста к алгоритму
- 4) алгоритм разрабатывать после разработки и уточнения данных.

9. Скрытые ошибки в программе после начала ее эксплуатации устраняют в процессе ...

- 1) отладки
- 2) сопровождения
- 3) тестирования
- 4) трассировки.

10. Запишите этапы обработки информации человеком. Указание: использовать основные структуры – память, нервная система, рецепторы, эффекторы.

2.2. Типовые контрольные задания, используемые для промежуточной аттестации по дисциплине

2.2.1. Примерный перечень вопросов к зачету (зачету с оценкой)

1. Понятие искусственного интеллекта.
2. Этапы развития систем искусственного интеллекта.

3. Области искусственного интеллекта.
4. Программные среды для реализации СИИ.
5. Установка программного обеспечения.
6. Работа в среде notebook jupyter.
7. Классификация языков программирования.
8. Основные сведения о языке программирования Python.
9. Данные: типы, значения, переменные и имена.
10. Арифметические операции.
11. Реализация разветвляющихся алгоритмов.
12. Реализация циклических алгоритмов.
13. Строковые выражения.
14. Списки.
15. Кортежи.
16. Словари.
17. Множества.
18. Модули, библиотеки, фреймворки.
19. Функции.
20. Библиотека для анализа данных
21. Нейронные сети.
22. Задача кластеризации.

2.2.2. Примерное задание на зачет (зачет с оценкой)

1. Кто в 50-е годы XX века создал математическую модель функционирования сетей нервных клеток живого организма?
 - а) Алан Тьюринг
 - б) Фрэнк Розентблатт
 - в) Иван Павлов
 - г) Билл Гейтс
2. Часть методов искусственного интеллекта, означающая извлечение знаний из данных:
 - а) Data Science
 - б) Статистика
 - в) Data Mining
 - г) Big data
3. Нажатием каких клавиш можно добавить ячейки в Notebook Jupyter (Возможно более одного правильного ответа)?
 - а) M
 - б) A
 - в) Y
 - г) C
 - д) B
4. Языки программирования, которые относятся к объектно-ориентированной парадигме (Возможно более одного правильного ответа):
 - а) Python
 - б) Haskell
 - в) C#
 - г) C
5. Кто является создателем языка Python?
 - а) Гвидо ван Россум
 - б) Марк Цукерберг
 - в) Деннис Ритчи
 - г) Томас Курц
6. Число с плавающей точкой обозначается:

- a) list
- б) bool
- в) int
- г) float

7. S=list('12345') С помощью функции list() создается:

- a) Множество
- б) Кортеж
- в) Список
- г) Словарь

8. Каким будет результат:

```
In [1]: a={1, 2, 3}
        a.intersection_update({1, 5})
        print(a)
```

- a) {1}
- б) {2, 3, 5}
- в) {1, 2, 3, 5}
- г) {1, 1, 2, 3, 5}

9. Функция tuple() применяется для:

- a) Очистки кортежа
- б) Очистка списка
- в) Создания пустого кортежа
- г) Добавления элемента в конец списка

10. Какое назначение функций собранных в модуле random?

- a) Обработка статистических данных
- б) Математические расчеты
- в) Работа с базами данных
- г) Генерация случайных чисел

11. Какой массив был сгенерирован при выполнении инструкции data=np.linspace(10, 18, 5)?

- a) [10., 12., 14., 16., 18.]
- б) [10., 15.]
- в) [10., 18., 5]
- г) [10, 15]

12. Был создан двумерный массив mas. Какой результат будет выведен в консоль при выполнении инструкции print(mas[3, 3]):

```
mas=np.array([[ -8,  3, 10],
              [10,  2,  0],
              [ 7, 17,  4]])
print(mas[3, 3])
```

- a) 4
- б) -8
- в) 0
- г) Будет выведена ошибка

13. В машинном обучении данные делятся на:

- a) Обучающую и обучаемую выборку
- б) Большую и малую
- в) Основную и неосновную
- г) Обучающую и тестовую выборку

14. Как называется метод получения коэффициентов уравнения регрессии, при котором минимизируются квадраты ошибок?

- а) Метод максимального правдоподобия
- б) Метод минимизации средней квадратической ошибки
- в) Метод минимизации модулей ошибок
- г) Метод наименьших квадратов

15. Был создан двумерный массив `mas`. Какой результат будет выведен в консоль при выполнении инструкции `print(mas[3, 3])`:

```
mas=np.array([[ -8,  3, 10],
              [10,  2,  0],
              [ 7, 17,  4]])
print(mas[3, 3])
```

- а) -8
- б) Будет выведена ошибка
- в) 0
- г) 4

16. В нейронной сети вычислительная единица, которая получает информацию, производит над ней простые вычисления и передает ее дальше:

- а) Градиент
- б) Нейрон
- в) Аксон
- г) Синапс

17. Какой алгоритм используется в методе обратного распространения ошибок для поиска минимума?

- а) Метод дихотомии
- б) Алгоритм Дейкстры
- в) Метод наименьших квадратов
- г) Градиентный спуск

18. Как называется одна итерация в процессе обучения, включающая предъявление всех примеров из обучающего множества?

- а) Эпоха
- б) Пакет
- в) Сет
- г) Минипакет

2.3. Методические материалы и типовые контрольные задания, используемые для текущего контроля по дисциплине

2.3.1. Методические материалы, используемые для текущего контроля знаний по дисциплине

Наименование оценочного средства	Краткая характеристика оценочного средства	Представление оценочного средства в фонде
Контрольный опрос	Контрольный опрос – это метод оценки уровня освоения компетенций, основанный на непосредственном (беседа, интервью) или опосредованном (анкета) взаимодействии преподавателя и студента. Источником контроля знаний в данном случае служит словесное или письменное суждение студента	Примерный перечень вопросов к зачету и экзамену Вопросы, выносимые на самостоятельное изучение Задания для самостоятельной работы
Защита лабораторной работы	Лабораторные практикумы один из видов самостоятельной практической работы студентов. Целью лабораторных	Лабораторные работы

Наименование оценочного средства	Краткая характеристика оценочного средства	Представление оценочного средства в фонде
	практикумов является углубление и закрепление теоретических знаний и развитие профессиональных умений и навыков	
Домашнее задание	Домашние задания – одна из основных форм самостоятельной работы студентов, направленная на усвоение и закрепление полученных знаний на занятиях.	Домашние задания

2.3.2. Вопросы, выносимые на самостоятельное изучение

1. Работа в среде notebook jupyter.
2. Списки.
3. Кортежи.
4. Словари.
5. Множества.
6. Модули, библиотеки, фреймворки.
7. Функции.
8. Математические функции
9. Библиотека numpy.
10. Библиотека pandas.
11. Построение графиков matplotlib.
12. Библиотека SciPy.
13. Понятие машинного обучения.
14. Линейное уравнение регрессии.
15. Методы оценки параметров уравнения регрессии.
16. Метрики качества.
17. Методы обучения нейронных сетей.
18. Основные этапы создания нейронных сетей.
19. Гиперпараметры нейронных сетей.
20. Основные задачи, которые решают нейронные сети.

2.3.3. Задания для самостоятельной работы

ЛАБОРАТОРНАЯ РАБОТА №1 ОСНОВЫ ПРОГРАММИРОВАНИЯ НА ЯЗЫКЕ PYTHON

Программное обеспечение: Python 3.10, Notebook Jupyter.

Методические указания к выполнению работы:

1. Создайте файл Python.
2. Заполните общую информацию в первой ячейке, используя язык разметки markdown:

Дисциплина: Системы искусственного интеллекта

Лабораторная работа №1

Основы программирования на языке Python

Студент: Сидоров Петр Иванович

Группа: 1М-1

Дата выполнения: 02.02.2002 **Вариант:** 1

3. Решить задание в соответствии со своим вариантом. Выберите свой вариант задания согласно правилу: $1+(N-1) \% 5$, где N – номер студента по списку.

ЗАДАНИЕ

Вариант №1

1. Пользователь вводит два натуральных числа: a и b . Если произведение двух чисел больше 20 и оба числа четные, то вычислить натуральный логарифм их произведения, в противном случае первое число умножить на 3.

2. Дано n целых чисел. Найти количество отрицательных чисел.

Вариант №2

1. Пользователь вводит два действительных числа x и y . Написать программу, которая меньшее из этих двух чисел заменит половиной их суммы, а не меньшее – их удвоенным произведением.

2. Написать программу, которая считает факториал натурального числа.

Вариант №3

1. Преподаватель выставляет зачет по дисциплине «Математика», если количество пропусков у студента не более 5 и набрано баллов от 60 и выше. Написать программу, которая выводит сообщение «зачтено» или «незачтено».

2. Дана последовательность n целых чисел, где n – задано. Определить, все ли числа попадают в заданный интервал $[x; y]$.

Вариант №4

1. Пользователь вводит 2 действительных числа a и b . Если произведение двух чисел больше 50 или меньше 10, то вычислить удвоенный косинус первого числа, в противном случае второе число умножить на 3.

2. Для данного натурального числа n найти сумму n первых четных чисел.

Вариант №5

1. Преподаватель выставляет зачет по дисциплине «Математика», если количество пропусков у студента не более 5 и набрано баллов от 60 и выше. Написать программу, которая выводит сообщение «зачтено» или «незачтено».

2. Дано число. Посчитать сумму цифр в десятичной записи этого числа.

ЛАБОРАТОРНАЯ РАБОТА №2

ИЗУЧЕНИЕ ОСНОВ ЯЗЫКА PYTHON ДЛЯ АНАЛИЗА ДАННЫХ

Программное обеспечение: Python 3.10, Notebook Jupyter, numpy.

Методические указания к выполнению работы:

1. Создайте файл Python.

2. Заполните общую информацию в первой ячейке, используя язык разметки markdown:

Дисциплина: Системы искусственного интеллекта

Лабораторная работа №1

Основы программирования на языке Python

Студент: Сидоров Петр Иванович

Группа: 1М-1

Дата выполнения: 02.02.2002 **Вариант:** 1

3. Решить задание в соответствии со своим вариантом. Выберите свой вариант задания согласно правилу: $1+(N-1) \% 3$, где N – номер студента по списку.

ЗАДАНИЕ

Вариант №1

1. Создать массив из 10 чисел в диапазоне от 0 до 20.

2. Создать матрицы:

$A = (-8 \ 3 \ 2 \ 8 \ 4 \ 0)$ и $B = (14 \ 2 \ 1 \ 3 \ -7 \ 12)$.

Вычислить: 1) $2 \cdot A^T + B$; 2) AB .

3. Найти определитель $\det(A)$ и обратную матрицу A^{-1} , если $A = (1 \ 0 \ -2 \ 4 \ 3 \ -4 \ 2 \ 3 \ 2)$.

Вариант №2

1. Создать массив из 21 числа в диапазоне от -1 до 1.

2. Создать матрицы:

$A = (5 \ 2 \ 1 \ 10 \ 4 \ 6)$ и $B = (4 \ 12 \ 1 \ 3 \ 5 \ -1)$.

Вычислить: 1) $-2 \cdot A + B^T$; 2) AB .

3. Найти определитель $\det(A)$ и обратную матрицу A^{-1} , если $A = (1 \ 0 \ -2 \ 4 \ 3 \ -4 \ 2 \ 3 \ 2)$.

Вариант №3

1. Создать массив из 15 чисел в диапазоне от 10 до 20.

2. Создать матрицы:

$A = (3 \ 3 \ 2 \ 0 \ 4 \ 7)$ и $B = (0 \ 2 \ -1 \ 4 \ 7 \ 1)$.

Вычислить: 1) $A^T \cdot 3 \cdot B$; 2) AB .

3. Найти определитель $\det(A)$ и обратную матрицу A^{-1} , если $A = (-3 \ 7 \ 2 \ 9 \ 3 \ -4 \ 2 \ 5 \ 0)$.

ЛАБОРАТОРНАЯ РАБОТА №3

ПОСТРОЕНИЕ ГРАФИКОВ

Программное обеспечение: Python 3.10, Notebook Jupyter, библиотеки pandas и matplotlib, numpy.

Методические указания к выполнению работы:

1. Создайте файл Python.

2. Заполните общую информацию в первой ячейке, используя язык разметки markdown:

Дисциплина: Системы искусственного интеллекта

Лабораторная работа №3

Построение графиков

Студент: Сидоров Аристарх Венедиктович

Группа: 2Эв-1

Дата выполнения: 01.04.2022

Вариант: №1

3. Решить задание в соответствии со своим вариантом. Выберите свой вариант задания согласно правилу: $1 + (N - 1) \% 3$, где N – номер студента по списку, $\%$ - операция, результатом которой будет остаток при делении.

4. Теоретический материал по данной теме соответствует лекции №4.

ЗАДАНИЕ

Вариант №1

Задача №1

В таблице приведены данные о квартирах по Волгоградской области

Цена, тыс. руб.	Площадь, м ²	Жилая площадь, м ²
2300	44,2	27,2
2700	61	33
2900	57	24
4350	68	39,2
2190	52	34
2150	36	18
1980	47	31,7

На основе данной таблицы создать словарь, который затем преобразовать в DataFrame.

Найти максимальные значения по каждому показателю, среднее и медианное значение площади. Определить число квартир, у которых площадь больше 50 м².

Построить график зависимости стоимости квартиры от площади квартиры.

Вариант №2

Задача №1

В таблице приведены данные о квартирах по Волгоградской области

Цена, тыс. руб.	Площадь, м ²	Жилая площадь, м ²
2300	44,2	27,2
2700	61	33

2900	57	24
4350	68	39,2
2190	52	34
2150	36	18
1980	47	31,7

На основе данной таблицы создать словарь, который затем преобразовать в DataFrame.

Найти минимальные значения по каждому показателю, среднее и медианное значение площади. Определить число квартир, у которых жилая площадь меньше 40 м².

Построить график зависимости стоимости квартиры от площади квартиры.

Вариант №3

Задача №1

В таблице приведены данные о квартирах по Волгоградской области

Цена, тыс. руб.	Площадь, м ²	Жилая площадь, м ²
2300	44,2	27,2
2700	61	33
2900	57	24
4350	68	39,2
2190	52	34
2150	36	18
1980	47	31,7

На основе данной таблицы создать словарь, который затем преобразовать в DataFrame.

Найти минимальные значения по каждому показателю, среднее и медианное значения жилой площади. Определить число квартир, у которых цена не превышает 2 млн. руб.

Построить график зависимости стоимости квартиры от жилой площади квартиры.

ПРИМЕРЫ РЕШЕНИЯ ЗАДАЧ

Задача №1. Создать таблицу DataFrame на основе словаря. Найти максимальное значение по каждому показателю. Найти средний рост ребенка. Определить число детей, рост которых не превышает 86 см. Построить график зависимости массы ребенка от его роста.

```
In [3]: #Создаем словарь, в котором приведены данные о физическом развитии детей 2 года
map={'Рост, см':[90, 87, 86, 85, 84], 'Масса, кг':[10.4, 9.8, 9.6, 9.3, 9.4], 'Обхват головы, см':[48, 48, 47, 46, 46]}
print(map)
data=pd.DataFrame(map, index=[1, 2, 3, 4, 5])
```

```
{'Рост, см': [90, 87, 86, 85, 84], 'Масса, кг': [10.4, 9.8, 9.6, 9.3, 9.4], 'Обхват головы, см': [48, 48, 47, 46, 46]}
```

```
In [4]: print(data)
```

	Рост, см	Масса, кг	Обхват головы, см
1	90	10.4	48
2	87	9.8	48
3	86	9.6	47
4	85	9.3	46
5	84	9.4	46

```
In [5]: #Вывод статистики
data.describe()
```

Out[5]:

	Рост, см	Масса, кг	Обхват головы, см
count	5.000000	5.000000	5.0
mean	86.400000	9.700000	47.0
std	2.302173	0.43589	1.0
min	84.000000	9.300000	46.0
25%	85.000000	9.400000	46.0
50%	86.000000	9.600000	47.0
75%	87.000000	9.800000	48.0
max	90.000000	10.400000	48.0

Вывод:

Максимальный рост 90 см, максимальная масса ребенка 10.4 кг, максимальный обхват головы 48 см. Средний рост ребенка составляет 86,4 см.

```
In [6]: #Мальчики, у которых рост не выше 86 см
data.loc[data['Рост, см']<=86]
```

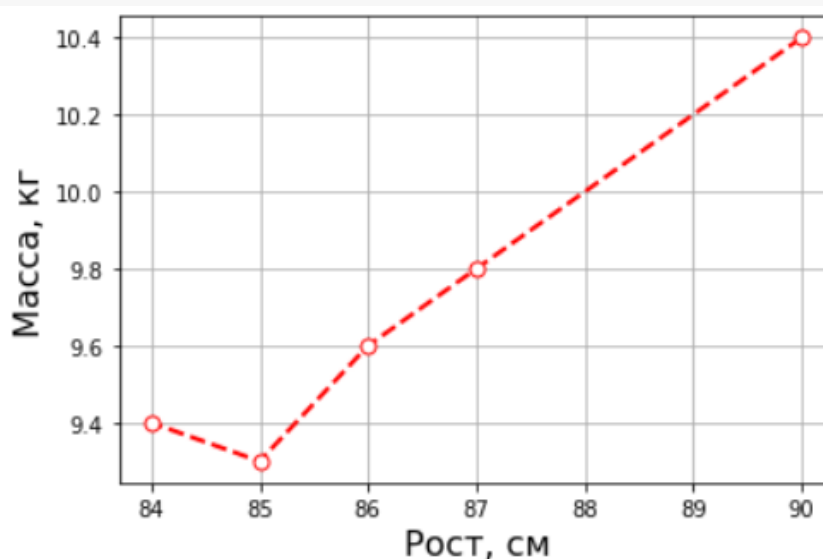
Out[6]:

	Рост, см	Масса, кг	Обхват головы, см
3	86	9.6	47
4	85	9.3	46
5	84	9.4	46

Вывод:

В данной выборке число детей, рост которых не превышает 86 см равно 3.

```
In [12]: #Построение графика зависимости массы ребенка от его роста
plt.figure(figsize=(6, 4))
plt.grid(True)
plt.plot(data['Рост, см'], data['Масса, кг'], color='r', marker='o', markersize=7, markerfacecolor='w',
         linestyle='--', linewidth=2)
plt.xlabel('Рост, см', fontsize=16)
plt.ylabel('Масса, кг', fontsize=16)
plt.show()
```



2.3.4. Тесты по дисциплине

1. Шахматный суперкомпьютер, который выиграл матч у чемпиона мира по шахматам Гарри Каспарова:
 - а) AlphaGo
 - б) Фугаку
 - в) Deep Blue
 - г) Ломоносов-2
2. С помощью какого символа можно создать заголовок на языке разметки Markdown?
 - а) *
 - б) #
 - в) %
 - г) &
3. Какие теги используются для создания жирного шрифта на языке разметки HTML?
 - а) <I>...</I>
 - б) <P>...</P>
 - в) _{...}
 - г) ...
4. Какой символ используется для набора формулы с помощью LaTeX в Notebook Jupyter?
 - а) \$
 - б) %
 - в) //
 - г) &
5. Какие варианты имен переменных допустимы в Python?
 - а) x1
 - б) k&
 - в) _main
 - г) 1x
 - д) new_переменная
6. С помощью какой операции над двумя целыми числами можно получить остаток?
 - а) %
 - б) /
 - в) //
 - г) *
7. Пусть переменной a присвоено значение 5, т. е. $a = 5$. Какие операции возможны?
 - а) $a+=1$
 - б) $a/=3$
 - в) $a=a+1$
 - г) $a//=2$
8. Для чего применяется функция `input()`?
 - а) Для определения типа данных
 - б) Для возведения числа в степень
 - в) Для вывода данных в консоль
 - г) Для ввода данных с клавиатуры
9. Какие операторы используются для реализации циклических алгоритмов?
 - а) `for`
 - б) `while`
 - в) `elif`
 - г) `switch`
10. $C=(1, 2, 3, 4, 5)$. Возможно ли в данном кортеже заменить 5 на 6?
 - а) Нет
 - б) Да
11. Чтобы подключить библиотеку необходимо использовать ключевое слово:

- a) import
 - б) is
 - в) from
 - г) install
12. Для создания функции используется ключевое слово
- a) func
 - б) lambda
 - в) class
 - г) def
13. Какой метод используется при создании словаря?
- a) dict()
 - б) tupe()
 - в) list()
 - г) set()
14. Какую операцию можно выполнить с помощью метода union()?
- a) Симметричная разность
 - б) Объединение
 - в) Пересечение
 - г) Дополнение
15. Какая функция из библиотеки numpy служит для умножения двух матриц?
- a) det()
 - б) dot()
 - в) len()
 - г) mult()
16. Какой тип данных в библиотеке pandas представляет собой проиндексированный многомерный массив?
- a) DataFrame
 - б) list
 - в) Series
 - г) Array
17. Какие функции из библиотеки matplotlib служат для построения графика?
- a) figure()
 - б) hist()
 - в) plot()
 - г) func()
18. Для чего служит функция head из библиотеки pandas?
- a) Для подсчета числа столбцов и строк таблицы
 - б) Для вывода любых пяти строк таблицы
 - в) Для вывода последних пяти строк таблицы
 - г) Для вывод первых пяти строк таблицы
19. С помощью какой функции из библиотеки matplotlib можно сохранить рисунок с графиком?
- a) save()
 - б) fig()
 - в) show()
 - г) savefig()
20. В каком диапазоне значений изменяется коэффициент детерминации?
- a) $[-1; 1]$
 - б) $[0, 1]$
 - в) $[0, +\infty]$
 - г) $[-\infty; +\infty]$
21. Какие показатели используются для оценки качества линейной модели?

- а) Коэффициент детерминации
 - б) Коэффициент чистой регрессии
 - в) Root Mean Squared Error
 - г) Mean Absolute Error
22. Какие значения может принимать коэффициент корреляции?
23. -1,2
24. 2
25. -0,9
26. 0
27. Библиотека, которая предназначена для машинного обучения:
- а) random
 - б) sklearn
 - в) pandas
 - г) numpy
28. С помощью каких показателей можно выявить наличие взаимосвязи между двумя случайными величинами?
- а) Дисперсия
 - б) Коэффициент корреляции
 - в) Ковариация
 - г) Среднее квадратическое отклонение
29. Выберите верные утверждения (Возможно более одного правильного ответа):
- а) Перцептрон Розенблатта – это пример нейронной сети с учителем
 - б) Нейронные сети можно применять только для решения задач прогнозирования.
 - в) Нейроны соединяются синапсами только с теми нейронами, которые находятся в одном слое
 - г) Вес синапса – это число в диапазоне $[-1; +1]$
30. Какие типы нейронов могут быть в перцептроне Розенблатта?
- а) Нормированные
 - б) Открытые
 - в) Входные
 - г) Скрытые
31. Какие функции из перечисленных ниже применяются в качестве функций активации в нейронных сетях?
- а) Сигмоида
 - б) Сигнум
 - в) Логарифмическая функция
 - г) Экспонента
 - д) Полином второй степени
32. Какая библиотека используется для работы с нейронными сетями?
- а) keras
 - б) numpy
 - в) matplotlib
 - г) seaborn
33. Какой показатель используется для оценки качества нейронной сети в задаче классификации?
- а) Root Mean Squared Error
 - б) Коэффициент детерминации
 - в) accuracy
 - г) Mean Squared Error

2.3.5. Лабораторные работы

ОСНОВЫ ПРОГРАММИРОВАНИЯ НА ЯЗЫКЕ PYTHON

Программное обеспечение: Python 3.10, Notebook Jupyter.

Методические указания к выполнению работы:

1. Создайте файл Python.
2. Заполните общую информацию в первой ячейке, используя язык разметки markdown:

Дисциплина: Системы искусственного интеллекта

Лабораторная работа №1

Основы программирования на языке Python

Студент: Сидоров Петр Иванович

Группа: 1М-1

Дата выполнения: 02.02.2002 **Вариант:** 1

3. Решить задание в соответствии со своим вариантом. Выберите свой вариант задания согласно правилу: $1+(N-1) \% 5$, где N – номер студента по списку.

ЗАДАНИЕ

Вариант №1

1. Написать программу, которая выводит на экран ваше имя, возраст и учебную группу.
2. Составить программу, находящую площадь кольца, заключенного между двумя окружностями с общим центром и радиусами R_1 и R_2 (R_1 и R_2 – вещественные, $R_1 > R_2$). Значения переменных R_1 и R_2 вводит пользователь с клавиатуры.

Вариант №2

1. Написать программу, которая выводит на экран ваше имя, возраст и учебную группу.
2. Дано a и b – стороны прямоугольного треугольника. Найти площадь прямоугольного треугольника, построенного на катетах a и b . Найти периметр треугольника. Значения переменных a и b вводит пользователь с клавиатуры

Вариант №3

1. Написать программу, которая выводит на экран ваше имя, возраст и учебную группу.
2. Составить программу, находящую величину угла в радианах R , если дана его величина D в градусах (D – вещественное число, $0 < D < 360$). Значение переменной D вводит пользователь с клавиатуры.

Вариант №4

1. Написать программу, которая выводит на экран ваше имя, возраст и учебную группу.
2. Приобрели A шт. книг по цене B руб. за шт. и C шт. тетрадей по цене D руб. за шт. Определить стоимость всего товара

Вариант №5

1. Написать программу, которая выводит на экран ваше имя, возраст и учебную группу.
2. Составить программу, находящую объем шара радиуса R (R – вещественное число) и площадь его поверхности. Значение переменной R вводит пользователь с клавиатуры.

ЛАБОРАТОРНАЯ РАБОТА №2

ИЗУЧЕНИЕ ОСНОВ ЯЗЫКА PYTHON ДЛЯ АНАЛИЗА ДАННЫХ

Программное обеспечение: Python 3.10, Notebook Jupyter, numpy.

Методические указания к выполнению работы:

1. Создайте файл Python.
2. Заполните общую информацию в первой ячейке, используя язык разметки markdown:

Лабораторная работа №1

Основы программирования на языке Python

Студент: Сидоров Петр Иванович

Группа: 1М-1

Дата выполнения: 02.02.2002 Вариант: 1

3. Решить задание в соответствии со своим вариантом. Выберите свой вариант задания согласно правилу: $1+(N-1) \% 3$, где N – номер студента по списку.

4. Выполнить не менее шести задач.

ЗАДАНИЕ

Вариант №1

1. Создайте список, состоящий из следующих элементов: -10, -8, -6, 0, 2, 3.

Выполните следующие задания: добавьте в конец данного списка число 5. Добавьте на элемент -7 после элемента -8. Удалите элемент, который находится под индексом 4. Обратите список.

2. Составьте двумя способами русско-английский словарь, в котором будут присутствовать не менее 8 слов.

3. Вычислите:

$$\frac{\sin \sin \frac{2\pi}{3} + 2}{40\sqrt{2}} - \sqrt[3]{15 - \lg \lg 15}.$$

4. Даны два множества: $A = \{2.3, 11.5, 31.2, 32.0\}$ и $B = \{0, 8.5, 11.5\}$. Найдите: $A \cup B$, $A \cap B$, A/B , $A \Delta B$.

Вариант №2

1. Создайте список, состоящий из следующих элементов: -12, -5, 4, 10, 2, 5.

Выполните следующие задания: добавьте в конец данного списка число 8. Добавьте на элемент -7 после элемента 4. Удалите элемент, который находится под индексом 4. Отсортируйте список по возрастанию элементов.

2. Составьте двумя способами русско-немецкий словарь, в котором будут присутствовать не менее 8 слов.

3. Вычислите:

$$\frac{\cos \cos \frac{\pi}{4} + 2}{-10 + \sin \sin \frac{5\pi}{3}} - \sqrt[3]{5 - \lg \lg 15}.$$

4. Даны два множества: $A = \{10, 20, 30\}$ и $B = \{-23, 0, 80, 12, 25\}$. Найдите: $A \cup B$, $A \cap B$, A/B , $A \Delta B$.

Вариант №3

1. Создайте два списка. Первый из них состоит из отрицательных элементов: -10, -9, -6, -5. Второй – из положительных: 2, 10, 12.

Выполните следующие задания: объедините оба списка в один. Удалите элемент, который находится под индексом 5. Подсчитайте количество элементов в списке. Обратите список.

2. Составьте двумя способами англо-русский словарь, в котором будут присутствовать не менее 8 слов.

3. Вычислите:

$$\frac{3 \sqrt[4]{95} + \ln \ln 14}{e^{-2} + \lg \lg 20}.$$

4. Даны два множества: $A = \{a, b, c, d\}$ и $B = \{-b, c, e, f\}$. Найдите: $A \cup B$, $A \cap B$, A/B , $A \Delta B$.

ПРИМЕРЫ РЕШЕНИЯ ЗАДАЧ

Пример 1.

```
In [54]: #Даны два множества
A={-5, -12, 3, 4}
B={0, 3, 4, 5, 8}
#Найдем пересечение двух множеств
C=A.intersection(B)
print(C)

{3, 4}
```

Пример 2. Вычислить:

$$\cos \cos \frac{7\pi}{4} + \frac{\ln \ln 12 - 10}{\sqrt{5}}$$

```
In [55]: import math
```

```
In [62]: x=math.cos(7*math.pi/4)+(math.log1p(12)-math.log(10, 3))/math.sqrt(5)
print(round(x, 3))
```

0.917

Пример 3.

```
massiv=['f', 'g', 'd', 'd', 'y']
print(massiv)
#Добавим элемента 'a' в конец списка
massiv.append('a')
print(massiv)
#Подсчет числа элементов 'd'
print('Число элементов "d" в массиве равно: ', massiv.count('d'))
#развернуть список
massiv.reverse()
print(massiv)
```

```
['f', 'g', 'd', 'd', 'y']
['f', 'g', 'd', 'd', 'y', 'a']
Число элементов "d" в массиве равно: 2
['a', 'y', 'd', 'd', 'g', 'f']
```

ЛАБОРАТОРНАЯ РАБОТА №3 ПОСТРОЕНИЕ ГРАФИКОВ

Программное обеспечение: Python 3.10, Notebook Jupyter, библиотеки pandas и matplotlib, numpy.

Методические указания к выполнению работы:

1. Создайте файл Python.
2. Заполните общую информацию в первой ячейке, используя язык разметки markdown:

Дисциплина: Системы искусственного интеллекта

Лабораторная работа №3

Построение графиков

Студент: Сидоров Аристарх Венедиктович

Группа: 2Эв-1

Дата выполнения: 01.04.2022

Вариант: №1

3. Решить задание в соответствии со своим вариантом. Выберите свой вариант задания согласно правилу: $1+(N-1) \% 3$, где N – номер студента по списку, $\%$ - операция, результатом которой будет остаток при делении.

4. Теоретический материал по данной теме соответствует лекции №4.

ЗАДАНИЕ

Вариант №1

Задача №1

Постройте графики функций: а) $y = 1 - 2\sin x$, б) $y = 0,5\cos 3x$.

Вариант №2

Задача №1

Постройте графики функций: а) $y = -3\sin x$, б) $y = 2 + 0,5\cos 3x$.

Вариант №3

Задача №1

Постройте графики функций: а) $y = x - \sqrt{x+3}$, б) $y = x^2 + 8x$.

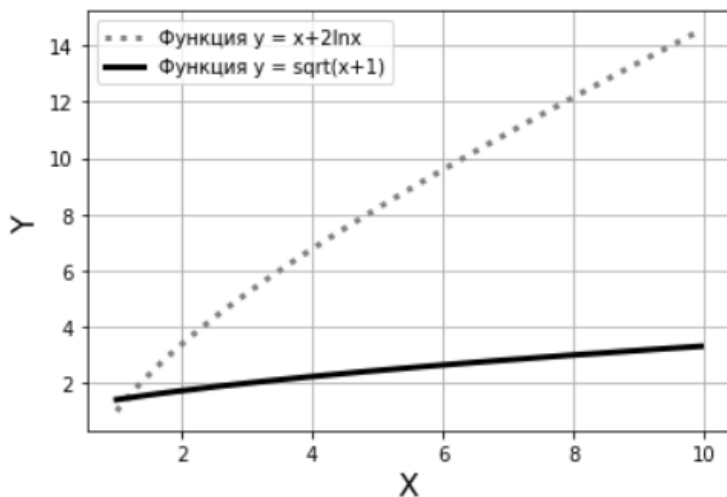
ПРИМЕРЫ РЕШЕНИЯ ЗАДАЧ

Задача №1. Постройте графики функций: а) $y = x + 2\ln x$, б) $y = \sqrt{x+1}$.

```
In [1]: #Подключение необходимых библиотек
import math
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
```

```
In [2]: #Магическая команда для отображения в блокноте графики matplotlib (другой вариант notebook,
%matplotlib inline
#Формирования массивов данных
X=np.arange(1, 10, 0.05)
Y1=[x+2*math.log(x) for x in X]
Y2=[math.sqrt(x+1) for x in X]

# Установка размеров графика в дюймах
plt.figure(figsize=(6, 4))
#Создание сетки
plt.grid(True)
#Построение графика
plt.plot(X, Y1,label='Функция y = x+2lnx', color='gray', linestyle=':', linewidth=3)
plt.plot(X, Y2, label='Функция y = sqrt(x+1)', color='black', linestyle='-', linewidth=3)
#Добавление подписи осей
plt.xlabel('X', fontsize=16)
plt.ylabel('Y', fontsize=16)
#Добавление легенды
plt.legend()
#Показ графика
plt.show()
```



ЛАБОРАТОРНАЯ РАБОТА №4 ЛИНЕЙНОЕ УРАВНЕНИЕ РЕГРЕССИИ

Цель работы: построить и проанализировать множественное уравнение регрессии с помощью библиотеки для машинного обучения sklearn.

Программное обеспечение: Python 3.10, Notebook Jupyter, библиотеки pandas и matplotlib, numpy, sklearn.

СТАТИСТИЧЕСКИЕ ДАННЫЕ

В лабораторной работе анализируются данные по квартирам на вторичном рынке недвижимости города Волгограда (данные за январь 2022 года).

Данные находятся в файле «квартиры.csv».

В таблице приведены пояснения к переменным:

Выходные данные		
price	float	Цена квартиры, тыс. руб.
Входные данные		
Переменные, характеризующие положение дома в зависимости от района		
R_1	int	1 – если Кировский район, 0 – иначе
R_2	int	1 – если Советский район, 0 – иначе
R_3	int	1 – если Красноармейский район, 0 – иначе
R_4	int	1 – если Тракторозаводский район, 0 – иначе
R_5	int	1 – если Краснооктябрьский район, 0 – иначе
R_6	int	1 – если Дзержинский район, 0 – иначе
R_7	int	1 – если Ворошиловский район, 0 – иначе
lat C	int	Географическая широта
log C	int	Географическая долгота
при $R_1, R_2, R_3, R_4, R_5, R_6, R_7 = 0$ – Центральный район		
Переменные, характеризующие количество комнат		
C_1	int	1 – если 1-комнатная квартира, 0 – иначе
C_2	int	1 – если 2-комнатная квартира, 0 – иначе
при $C_1, C_2 = 0$ – 3 и более комнаты		
Переменные, характеризующие состояние ремонта		
P_1	int	1 – если дизайнерский ремонт, 0 – иначе
P_2	int	1 – если евро ремонт, 0 – иначе
P_3	int	1 – если косметический ремонт, 0 – иначе
при $P_1, P_2, P_3 = 0$ – требует ремонта		
Остальные переменные, характеризующие квартиру		
area_t	float	общая площадь объекта недвижимости, m^2
area_k	float	жилая площадь объекта недвижимости, m^2
area_l	float	общая площадь объекта недвижимости, m^2
floor	int	отношение текущего этажа к общему количеству этажей
type_house	int	показатель элитной недвижимости
B_1	int	1 – если в квартире есть балкон, 0 – иначе
B_2	int	1 – если в квартире есть лоджия, 0 – иначе
при $B_1, B_2 = 0$ – нет ни балкона, ни лоджии		
age	int	год постройки объекта недвижимости
Переменные, которые не учитывались при расчете		
adress	varchar	адрес расположения объекта недвижимости

ЗАДАНИЕ

1. Изучить теоретический материал к лекциям №4 и №5.
2. Необходимо оценить множественное линейное уравнение регрессии зависимости стоимости квартиры price от ее параметров на вторичном рынке недвижимости, используя библиотеку sklearn для машинного обучения. В качестве параметров использовать такие переменные как

area_t, area_k, C₁, C₂, age, floor, R₁, ..., R₇.

3. Рассчитать среднюю площадь квартир, представленных в выборке. Найти квартиры с наибольшей и наименьшей площадью. Найти медианное значение для общей площади и для стоимости. Подсчитать число квартир из Дзержинского и Ворошиловского района.

4. Построить гистограмму для стоимости квартиры и для логарифма стоимости квартиры. Сделать вывод о симметричности распределений.

5. Рассчитайте корреляционную матрицу. С какими показателями переменная price сильнее всего связана. Чем это можно объяснить?

6. Найти коэффициенты множественного уравнения регрессии методом наименьших квадратов. Выборку необходимо разделить на обучающую (80%) и тестовую (20%).

7. Рассчитайте коэффициент детерминации для обучающей и тестовой выборки. Сделайте вывод (имеет ли место переобучение или недообучение регрессионной модели).

8. Запишите множественное уравнение регрессии (округление коэффициентов при регрессорах – до сотых). Что показывают коэффициенты при переменных area_t и floor? Вывод запишите в тетрадь.

Методические указания к выполнению работы:

1. Подключите необходимые библиотеки.

```
In [46]: #Загрузка необходимых библиотек
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import matplotlib
import seaborn as sns
%matplotlib inline
```

2. Подключите необходимые модули и функции из библиотеки sklearn.

```
In [75]: # Загрузка модулей из библиотеки sklearn
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn import metrics
from sklearn.metrics import r2_score
```

3. Загрузите данные из файла:

```
In [55]: #Загрузка данных
dt=pd.read_csv('Квартиры.csv', sep=';', usecols=['price', 'floor', 'area_t', 'area_k', 'age', 'R1', 'R2', 'R3', 'R4', 'R5', 'R6', 'R7', 'c1', 'c2'])
dt.head(5)
```

4. Рассчитайте требуемые показатели из пункта 3 задания.

Основные статистические показатели можно найти, используя метод: dt.describe()

Для поиска числа квартир можно использовать операторы if и while:

```
In [134]: 1 #Поиск числа квартир Ворошиловского района
2 data=dt['R7']
3 i=0
4 s=0
5 while i<len(data):
6     if data[i]==1:
7         s=s+1
8     i+=1
9
10 print(s)
```

5. Постройте гистограмму для стоимости квартиру:

```
In [99]: #стиль диаграммы
matplotlib.style.use('classic')
#Размер диаграммы
plt.figure(figsize=(7,4))
#Построение гистограммы
dt['price'].hist(bins=30, color='gray', linewidth=2)
plt.xlabel("Стоимость квартиры")
```

Можно задать другие стили. Перечень всех стилей гистограммы можно вызвать следующим образом: print(matplotlib.style.available).

6. Рассчитайте натуральный логарифм от переменной price и сделайте вывод о том, как изменилась симметричность гистограммы после логарифмирования.

```
In [82]: #логарифмируем стоимость квартиры
dt['ln_price']=np.log(dt.price)
dt.head(4)
```

7. Рассчитайте матрицу коэффициентов корреляции. Сделайте вывод в соответствии с пунктом 5 задания.

```
In [56]: #Матрица коэффициентов корреляции
dt.corr()
```

8. Разделите выборку на обучающую и тестовую. Найдите коэффициенты при регрессорах.

```
In [58]: # Определяем регрессоры и зависимую переменную
x = dt.iloc[:, 1:].values.reshape(-1, 13) #Включаем с 2 по 7 столбец
y = dt.iloc[:, 0].values.reshape(-1, 1)#включаем только один столбец

In [60]: # С помощью встроенного метода Scikit-Learn train_test_split() разделим выборку на две обучающую и тестовую
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

In [61]: #обучаем модель
regressor = LinearRegression()
regressor.fit(X_train, y_train)

In [62]: #Свободный коэффициент
print('свободный коэффициент:', regressor.intercept_)

In [63]: #Коэффициенты регрессии
print(regressor.coef_)
```

9. Рассчитайте коэффициент детерминации для обучающей и тестовой выборки. Сделайте вывод в соответствии с пунктом 7 задания.

```
In [73]: #Расчет коэффициента детерминации на обучающей выборке
r_sq = regressor.score(X, y)
print('коэффициент детерминации равен:', round(r_sq, 3))

In [74]: #Предсказываем значения для выборки
y_LR = regressor.predict(X_test)
#Расчет коэффициента детерминации на тестовой выборке
print ('R2_score:', round (r2_score(y_test, y_LR),3))
```

10. Запишите уравнение регрессии. Сделайте выводы в соответствии с пунктом 8 задания.

ЛАБОРАТОРНАЯ РАБОТА №5

ПРОГНОЗИРОВАНИЕ НА ОСНОВЕ НЕЙРОСЕТЕВЫХ МОДЕЛЕЙ

Цель: изучить методы прогнозирования на основе нейронных сетей с учителем.

Программное обеспечение: Deductor.

О ПРОГРАММЕ DEDUCTOR

Аналитическая платформа платформа Deductor – это отечественный программный продукт, разработанный компанией BaseGroup Labs. С его помощью возможно решение самого широкого спектра задач, начиная от создания систем корпоративной отчетности и до решения задач Data Mining. Это специализированное программное решение выполнено на базе единой платформы и содержит в себе необходимые инструменты для извлечения закономерностей из «сырых» данных. Оно позволяет пройти все этапы построения аналитической системы: от создания хранилища данных до автоматического подбора моделей и визуализации полученных результатов.

Система позволяет анализировать любые табличные данные и для решения аналитических задач. В ней предусмотрена возможность использования следующих специальных технологий:

- Data Warehouse (хранилища данных) – консолидация данных и обеспечение быстрого и понятного для аналитика доступа к ним – OLAP (многомерный анализ) – визуализация, отчетность и удобное манипулирование большими объемами данных
- Data Mining (моделирование, интеллектуальный анализ данных) – поиск скрытых закономерностей, выявление причинно-следственных связей, анализ рисков
- KDD – Knowledge Discovery in Databases (обнаружение, извлечение знаний) – построение сценариев обработки от очистки и предобработки данных до моделирования.

Последовательность действий (импорт, экспорт, обработка, визуализация) при решении конкретных задач в Deductor задается сценарием обработки. Сами же сценарии формируются специальным приложением Deductor Studio, которое является рабочим местом аналитика.

В профессиональной версии системы для импорта и анализа пригодны разнообразные табличные данные из стороннего источника (Oracle, MS SQL, Sybase, MS Access, Excel, 1С и др.). В учебной версии Deductor Academic – импортируются только данные в формате текстовых файлов с разделителями в виде табуляции.

Под обработкой данных в системе подразумевается любые действия, связанные с их преобразованием, такие как очистка данных, их трансформация и построение разнообразных моделей Data Mining. При визуализации производится отображение полученных и обработанных данных.

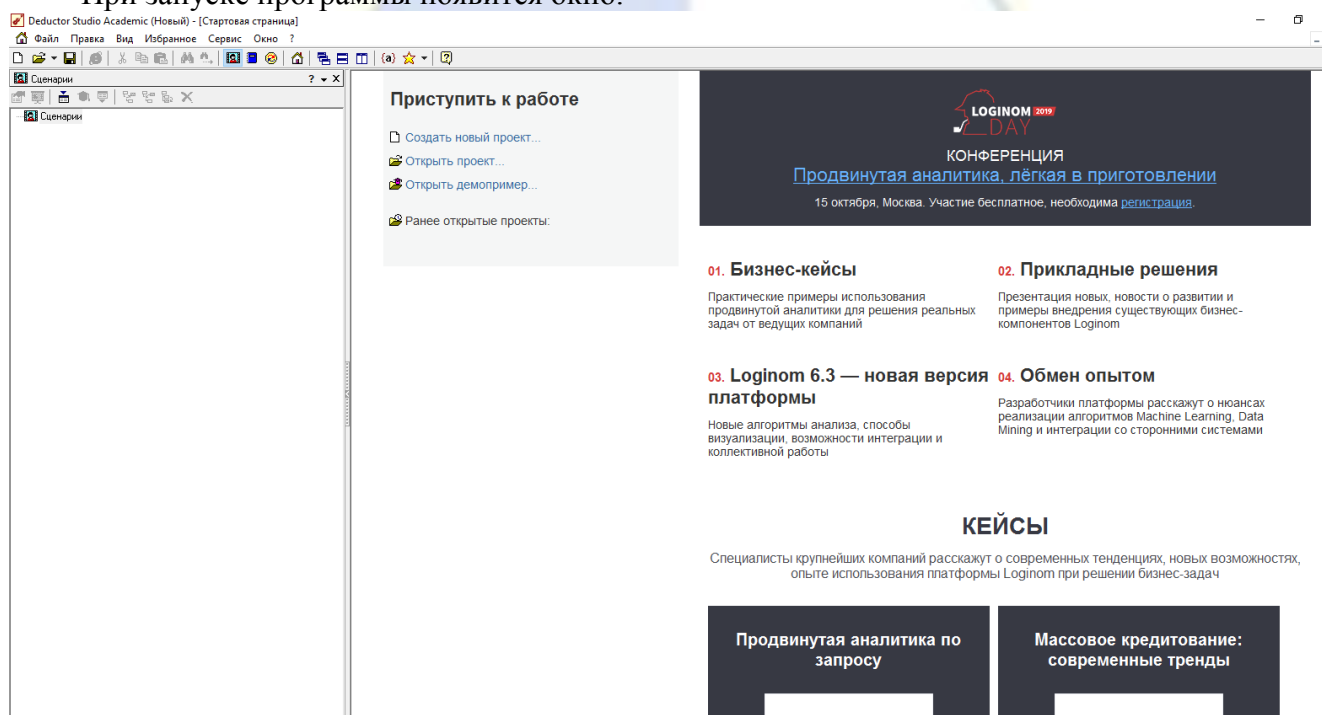
Аналитическая платформа самостоятельно анализирует формат отображения, предоставляя пользователю возможность выбора необходимого варианта.

В профессиональной версии системы предусмотрен экспорт (вывод) результатов обработки в виде файлов для последующего использования. В учебной версии Deductor Academic данная опция отсутствует.

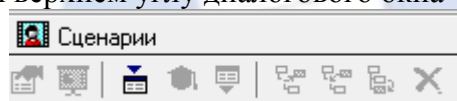
В платформе Deductor представлено большинство основных технологий анализа, позволяющих достаточно быстро проектировать законченные аналитические решения, охватывающие весь цикл обработки данных. Это – многомерный анализ, нейронные сети, деревья решений, самоорганизующиеся карты, спектральный анализ и ряд других.

Применение подобных самообучающиеся методов и машинного обучения дает возможность создавать адаптивные информационные системы. Во многом такой подход делает более мягкими требования к квалификации персонала, приближая современные информационные технологии к более широкому кругу пользователей.

При запуске программы появится окно:

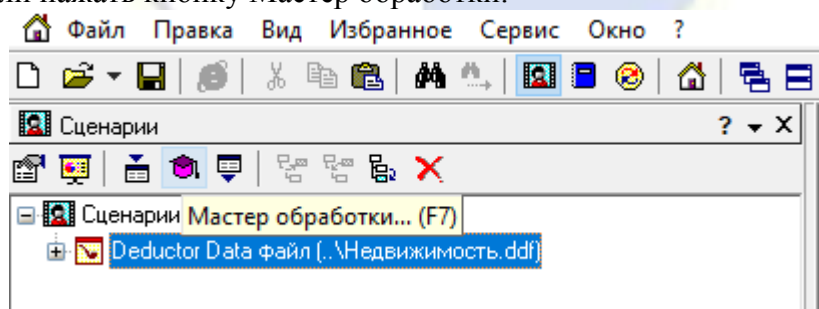


С помощью мастера импорта можно только загрузить в Deductor текстовый файл. Для этого необходимо нажать иконку в левом верхнем углу диалогового окна



или нажать клавишу F6. Для загрузки данных необходимо перейти Файл – Открыть.

Чтобы построить нейросетевую модель на базе загруженных данных есть два варианта: нажать клавишу F7 или нажать кнопку Мастер обработки:



Откроется окно с возможностями обработки данных. Необходимо выбрать «Нейросеть»

В этом режиме можно построить нейронную сеть с заданной структурой, определить ее параметры и обучить с помощью одного из доступных в системе алгоритмов обучения. В результате

будет получен эмулятор нейронной сети, который может быть использован для решения задач прогнозирования, классификации, поиска скрытых закономерностей, сжатия данных и многих других приложений.

Настройка и обучение нейронной сети состоит из следующих шагов:

1. Настройка назначений полей.
2. Настройка нормализации полей.
3. Настройка обучающей выборки.
4. Настройка структуры нейронной сети.
5. Выбор алгоритма и настройка параметров обучения.
6. Настройка условий остановки обучения.
7. Запуск процесса обучения.
8. Выбор способа отображения данных.

Настройка назначения полей

Здесь необходимо определить, как будут использоваться поля исходного набора данных при обучении нейронной сети и практической работе с ней.

Мастер обработки - Нейросеть (2 из 9)

Настройка назначений столбцов
Задайте назначения исходных столбцов данных

ID объекта	Имя столбца	ID
Район	Тип данных	Целый
Тип планировки	Назначение	Входное
Количество комнат	Вид данных	Непрерывный
Первый/Последний этаж	Статистика	
Общая площадь (м2)	Минимум	1
Жилая площадь (м2)	Максимум	2134
Площадь кухни (м2)	Среднее	1067,5
Наличие агентства	Стандартное откл.	616,177057887314
Состояние		
Стоимость (т.руб.)		

Настройка нормализации...

< Назад Далее > Отмена

Имя столбца – идентификатор поля, определенный для него в источнике данных. Изменить его здесь нельзя.

Тип данных – тип данных, содержащихся в поле (вещественный, строковый, дата). Он также задается в источнике данных и здесь изменен быть не может.

Назначение – здесь необходимо выбрать порядок использования данного поля при обучении и работе нейронной сети. Выбор производится с помощью списка, открываемого кнопкой и содержащего следующие варианты: входное – значения поля будут подаваться на соответствующие входы нейронной сети, т. е. являться исходными данными для обучения и дальнейшей практической работы обученной нейронной сети; выходное – значения данного поля будут являться эталонными при обучении нейронной сети, а при практической работе с ней будут содержать результаты обработки входных полей; информационное – поле не будет использоваться при обучении сети, но будет помещено в результирующий набор в исходном состоянии; неиспользуемое – поле не будет использоваться при обучении и работе нейронной сети и будет исключено из результирующей выборки. В отличие от непригодного такое поле может быть использовано, если в этом возникнет необходимость.

Вид данных – указывает на характер данных, содержащихся в поле (непрерывный или дискретный). Изменить это свойство здесь нельзя.

При выборе входных и выходных полей необходимо принимать во внимание следующее:

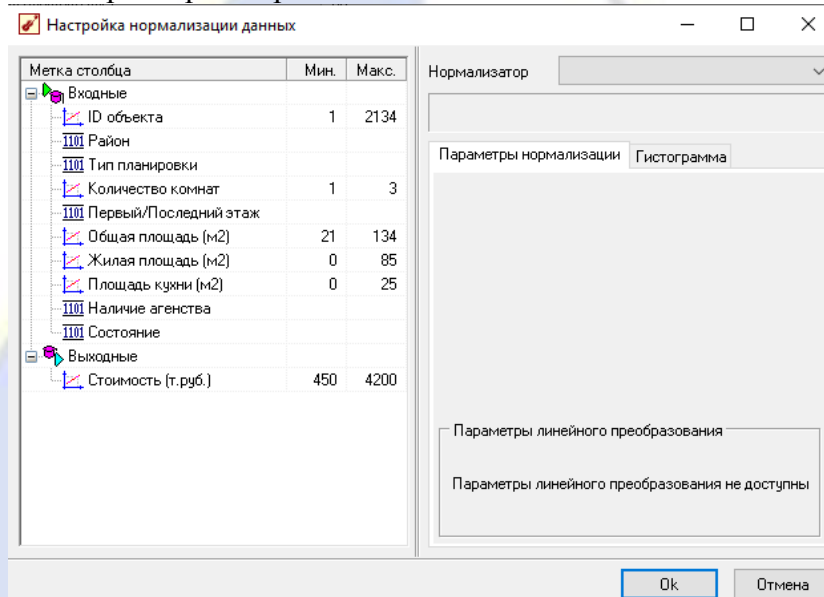
Входные и выходные поля определяются характером решаемой задачи: входные поля содержат исходные данные, подлежащие обработке, а Выходные – результаты обработки.

Использование полей, данные в которых не имеют отношения к решаемой задаче, крайне нежелательно, поскольку они только загромождают исходную и результирующую выборки и замедляют работу сети.

В случае, если текущее поле содержит непрерывные (числовые) данные, в правой нижней части окна появляется секция «Статистика», в которой отображаются максимальное и минимальное значения поля, его среднее значение и стандартное отклонение. Если выделенное поле содержит дискретные (строковые) данные, то для него открывается секция «Уникальные значения», в которой отображается общее число уникальных значений в поле, а также список самих уникальных значений.

Настройка нормализации полей.

С помощью кнопки «Настройка нормализации» можно вызвать окно настройки нормализации полей для указания параметров нормализации исходных данных.



Целью нормализации значений полей является преобразование данных к виду, наиболее подходящему для обработки средствами пакета Deductor. Для таких обработчиков, как нейронная сеть, линейная модель прогнозирования, самоорганизующиеся карты, данные, поступающие на вход, должны иметь числовой тип (т. е. иметь непрерывный характер), а их значения должны быть распределены в определенном диапазоне. В этом случае нормализатор может преобразовать дискретные данные к набору уникальных индексов или непрерывным значениям.

В окне настройки нормализации полей слева приведен полный список входных и выходных полей. При этом каждое поле помечено значком, обозначающим вид нормализации поля: Линейная – линейная нормализация исходных значений; Уникальные значения – преобразование уникальных значений в их индексы; Битовая маска – преобразование дискретных значений в битовую маску.

Параметры нормализации: В правой части окна для выделенного поля отображаются параметры нормализации. Правила нормализации и представляемые пакетом Deductor значения отражены в техдокументации.

Гистограмма: На закладке «Гистограмма» отображается гистограмма значений выделенного поля независимо от способа нормализации. Высота столбца показывает число значений, попадающих в определенный интервал. Границы интервала подписаны под столбцами гистограммы. Для отображения на гистограмме значений из нескольких полей следует выделить в списке доступных полей нужные с помощью мыши при нажатых клавишах Shift или Ctrl.

При работе с гистограммой предусмотрена возможность увеличения масштаба просмотра произвольной области гистограммы. Для этого следует, удерживая левую кнопку мыши нажатой, выделить ту область гистограммы, которую нужно просмотреть более детально, при этом двигая мышь слева направо.

Выделение левой части гистограммы отмечается белым прямоугольником. Как только кнопка мыши будет отпущена, масштаб просмотра выделенной области будет увеличен. Для дальнейшего увеличения масштаба данную процедуру можно повторить. При выделении области гистограммы движением мыши справа налево масштаб просмотра гистограммы будет возвращен к ис-

ходному независимо от размера выделенной области. Направив указатель мыши в произвольную точку гистограммы и перемещая ее с нажатой правой кнопкой, можно перемещать диаграмму по экрану, делая доступными для просмотра различные ее части.

Настройка обучающей выборки.

Здесь можно разбить обучающую выборку для на два множества – обучающее и тестовое.

Мастер обработки - Нейросеть (3 из 9)

Разбиение исходного набора данных на подмножества
Настройте разбиение исходного множества данных на обучающее и тестовое множества

Способ разделения исходного множества данных: Случайно

Столбец для разделения исходного множества: [выпадающий список]

Множество	Размер		Порядок сортировки
	В процентах	В строках	
<input checked="" type="checkbox"/> Обучающее	95,00	2027	По возрастанию
<input checked="" type="checkbox"/> Тестовое	5,00	107	По возрастанию
ИТОГО:	100,00	2134	

Количество строк (всего): 2134

< Назад **Далее >** Отмена

Обучающее множество включает записи (примеры), которые будут использоваться в качестве входных данных, а также соответствующие желаемые выходные значения.

Тестовое множество также включает записи, содержащие входные и желаемые выходные значения, но используемое не для обучения модели, а для проверки его результатов.

Для разбиения исходного множества на обучающее и тестовое необходимо настроить несколько параметров. Из списка «Способ разделения исходного множества» выбирается порядок отбора записей во все множества. Если выбран вариант «по порядку», то порядок следования записей при их разделении не меняется. Множества последовательно формируются в соответствии с определенным для них числом записей. Если выбран вариант «случайно», то отбор записей происходит случайным образом. Затем необходимо указать, какие множества будут использоваться. Для того чтобы множество было сформировано, нужно установить флажок слева от его названия. Если флажок сброшен, то множество использовано не будет. Обучающее множество используется всегда, поэтому сбросить флажок для него нельзя. Для каждого из используемых множеств необходимо задать его размер. Размер может быть задан непосредственно количеством записей или в процентах от объема исходной выборки. Для этого достаточно дважды щелкнуть мышью в соответствующей клетке и ввести нужное значение с клавиатуры. При этом размер, введенный в процентах, автоматически пересчитывается в количество строк и наоборот. В поле «Количество строк (всего)» отображается общее количество записей в исходной выборке данных, которое может быть задействовано для формирования множеств. Если суммарное число строк для всех используемых множеств меньше полного числа строк исходной выборки, то размеры множеств можно задавать произвольно. Можно, например, использовать не все записи, а только часть из них. Если же суммарное указанное число строк превышает максимальное для данной исходной выборки, то автоматически включается баланс множеств. Т.е. при указании для одного из множеств размера, в результате которого будет превышено максимальное число записей в исходной выборке, размер остальных множеств будет автоматически уменьшен таким образом, чтобы суммарный размер множеств не превышал доступного числа записей. В строке «Итого» указывается количество записей, задействованных во всех используемых множествах, а также процент от полного числа записей исходной выборки, который они составляют. В столбце "Порядок сортировки" можно определить порядок следования записей внутри каждого множества. Для этого дважды щелкнуть мышью в столбце «Порядок сортировки» для соответствующего множества и с помощью появившейся кнопки выбора открыть список, в котором выбрать один из возможных вариантов:

- По возрастанию – записи в данном множестве будут следовать в порядке возрастания.
- По убыванию – записи в данном множестве будут следовать в порядке убывания.
- Случайно – записи в данном множестве будут следовать в случайном порядке.

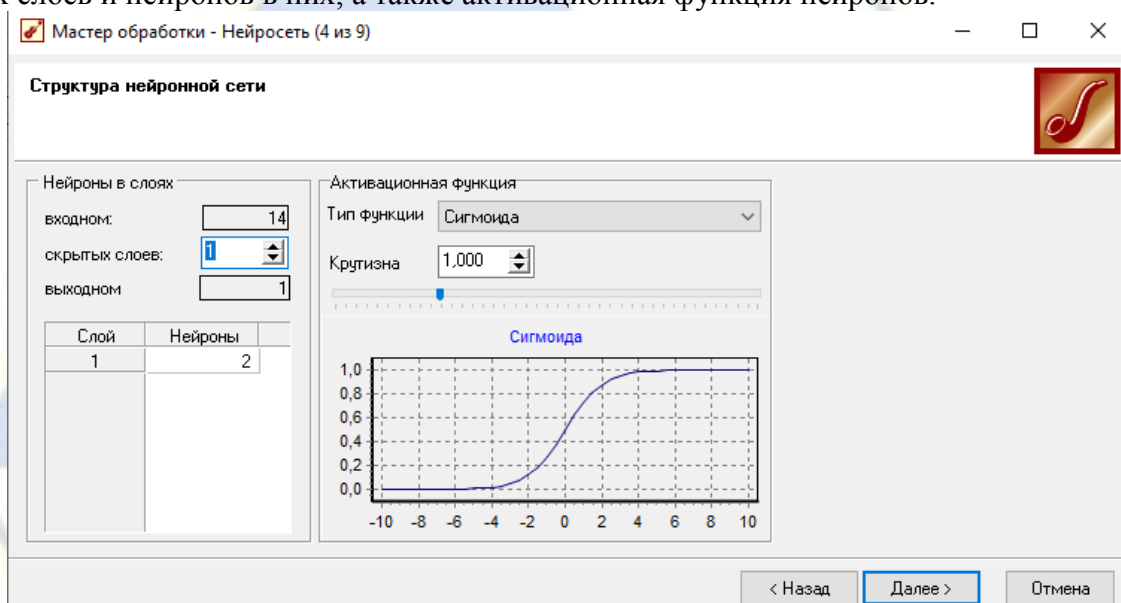
В правой части окна расположены кнопки, которые позволяют изменять порядок применения множеств в процессе обучения:

- По возрастанию – располагает уникальные значения по возрастанию (по алфавиту).
- По убыванию – располагает уникальные значения по убыванию (против алфавита).
- Вверх – перемещает выделенное значение на одну позицию вверх по списку.
- Вниз – перемещает выделенное значение на одну позицию вниз по списку.

Для того чтобы обучающее множество было репрезентативным необходимо, чтобы все уникальные значения всех дискретных столбцов содержались в данном наборе данных.

Настройка структуры сети.

На этом шаге задаются параметры, определяющие структуру нейронной сети, количество скрытых слоев и нейронов в них, а также активационная функция нейронов.



В секции «Нейроны в слоях» необходимо указать количество скрытых слоев, т. е. слоев нейронной сети, расположенных между входным и выходным слоями. Для этого в поле «скрытых слоев» указать соответствующее значение. Это можно сделать с клавиатуры, предварительно вызвав щелчком мыши курсор, или с помощью стрелочек в правой части поля. Число нейронов во входном и выходном слоях автоматически устанавливается в соответствии с числом входных и выходных полей обучающей выборки и здесь изменить его нельзя.

К выбору количества скрытых слоев и количества нейронов для каждого скрытого слоя нужно подходить осторожно. Хотя до сих пор не выработаны четкие критерии выбора, дать некоторые общие рекомендации все же возможно. Считается, что задачу любой сложности можно решить при помощи двухслойной нейросети, поэтому конфигурация с количеством скрытых слоев, превышающих 2, вряд ли оправдана. Для решения многих задач вполне подойдет однослойная нейронная сеть. При выборе количества нейронов следует руководствоваться следующим правилом: «количество связей между нейронами должно быть примерно на порядок меньше количества примеров в обучающем множестве». Количество связей рассчитывается как связь каждого нейрона со всеми нейронами соседних слоев, включая связи на входном и выходном слоях. Слишком большое количество нейронов может привести к так называемому «переобучению» сети, когда она выдает хорошие результаты на примерах, входящих в обучающую выборку, но практически не работает на других примерах.

В секции «Активационная функция» необходимо определить тип функции активации нейронов и ее крутизну. Для этого в списке «Тип функции» следует выбрать нужную функцию активации, а в поле «Крутизна» – задать ее крутизну (также крутизну можно задать с помощью ползунка,

расположенного ниже). В нижней части окна отображается график выбранной функции в соответствии с установленной крутизной.

Выбор алгоритма и настройка параметров обучения.

На данном шаге можно выбрать алгоритм обучения нейронной сети и задать его параметры. Чтобы выбрать один из доступных в системе алгоритмов, нужно активизировать соответствующий пункт в секции «Алгоритм». При этом в секции «Параметры» отобразится набор настроек, соответствующих выбранному алгоритму.

Мастер обработки - Нейросеть (5 из 9)

Настройка процесса обучения нейронной сети
Выбор алгоритма и задание параметров обучения

Алгоритм

Back - Propagation
Обучение в режиме "онлайн". Коррекция весов производится после предъявления каждого примера обучающего множества.

Resilient Propagation (RPROP)
Обучение в режиме "оффлайн". Коррекция весов производится после предъявления всех примеров обучающего множества. Учитывается только знак градиента по каждому весу.

Параметры

Шаг спуска
В случае изменения знака градиентной составляющей ошибки для данного веса задает величину следующей коррекции веса.

Шаг подъема
В случае сохранения знака градиентной составляющей ошибки для данного веса задает величину следующей коррекции веса.

< Назад **Далее >** Отмена

Для алгоритма обратного распространения ошибки (Back-Propagation) задаются два параметра:

Скорость обучения – определяет величину шага при итерационной коррекции весов в нейронной сети (рекомендуется задавать в интервале 0...1).

Момент – учитывает величину последнего изменения веса при коррекции весов (задается в интервале 0...1).

Для алгоритма Resilient Propagation указываются следующие параметры:

Шаг спуска – коэффициент увеличения скорости обучения, который определяет шаг увеличения скорости обучения при не достижении алгоритмом оптимального результата.

Шаг подъема – коэффициент уменьшения скорости обучения. Задается шаг уменьшения скорости обучения в случае пропуска алгоритмом оптимального результата.

Настройка условий остановки обучения.

На данном шаге необходимо задать условия, при выполнении которых обучение будет прекращено.

Мастер обработки - Нейросеть (6 из 9)

Настройка параметров остановки обучения

Укажите условия прекращения обучения. Обучение будет остановлено при выполнении одного из условий.

Считать пример распознанным, если ошибка меньше

По достижению эпохи

Обучающее множество

Средняя ошибка меньше

Максимальная ошибка меньше

Распознано примеров (%)

Тестовое множество

Средняя ошибка меньше

Максимальная ошибка меньше

Распознано примеров (%)

< Назад **Далее >** Отмена

«Считать пример распознанным, если ошибка меньше» – критерием остановки в данном случае является условие, что рассогласование между эталонным и реальным выходом сети становится меньше заданного значения.

«По достижении эпохи» – установка данного флажка позволяет задать число эпох (циклов обучения) по достижении которого обучение останавливается независимо от величины ошибки. Если флажок сброшен, то обучение будет продолжаться, пока ошибка не станет меньше заданного значения.

Кроме этого для обучающего и тестового множества в соответствующих секциях окна могут независимо устанавливаться следующие критерии остановки обучения:

«Средняя ошибка меньше» – средняя квадратичная ошибка на обучающем множестве или тестовом множестве меньше заданного значения.

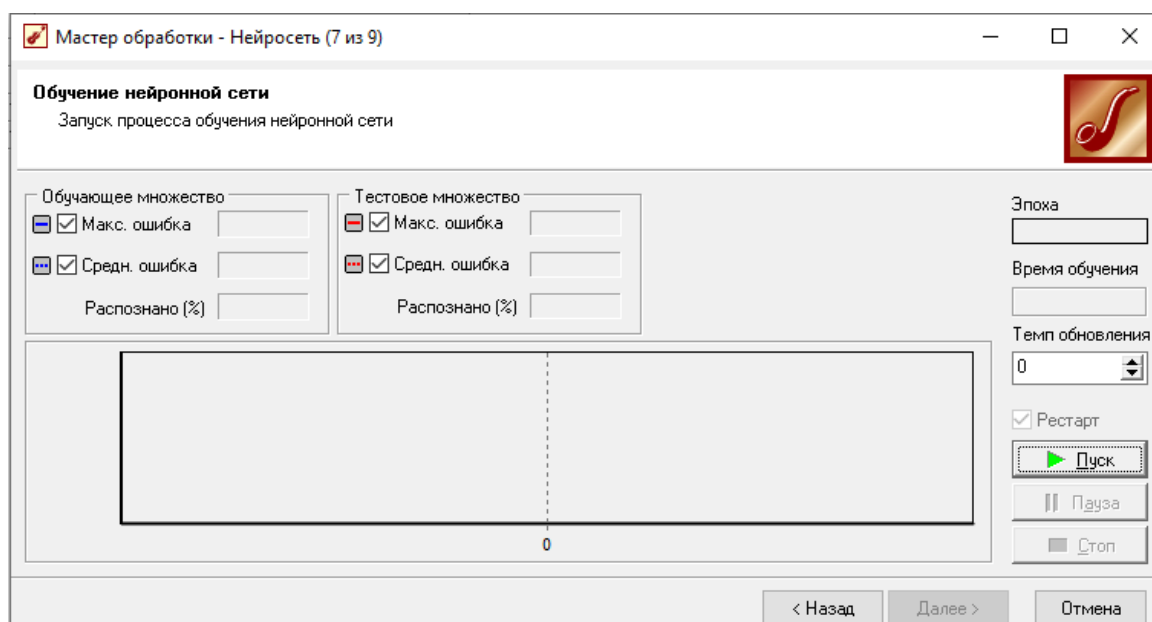
«Максимальная ошибка меньше» – максимальная квадратичная ошибка на обучающем множестве или тестовом множестве меньше заданного значения.

«Распознано примеров (%)» – количество распознанных примеров на обучающем множестве или тестовом множестве больше заданного процента.

При выборе нескольких условий остановки процесса обучения происходит по достижении хотя бы одного из них.

Запуск процесса обучения.

На данном шаге производится собственно процесс обучения нейронной сети. В зависимости от объема обрабатываемых данных и быстродействия компьютера, он может занять определенное время.



Остановить процесс обучения можно с помощью кнопки «Стоп». После этого появляется возможность вернуться на предыдущие шаги Мастера обработки для проверки и изменения параметров. Далее процесс обучения может быть запущен заново, продолжен или отменен. Если обучение было завершено успешно, то в секции «Название текущего процесса» появится сообщение «Успешное завершение».

Для управления процессом обучения предусмотрены следующие кнопки:

«Пуск» – запускает процесс в первый раз или возобновляет после паузы.

«Пауза» – временно приостанавливает обработку. Временная приостановка процесса имеет смысл в двух случаях: для оценки текущих результатов процесса обучения, например, просмотра графиков динамики ошибок обучения; при необходимости освободить ресурсы процессора для других приложений.

«Стоп» – останавливает процесс без возможности его продолжения. Практически, останавливать процесс обучения модели принудительно имеет смысл если: значения ошибок длительное время не уменьшаются; процент распознанных примеров не увеличивается.

В процессе обучения в секциях «Обучающее множество» и «Тестовое множество» отображаются максимальная квадратичная ошибка и средняя квадратичная ошибка на обучающем множестве и тестовом множестве соответственно, а также процент распознанных примеров.

Обучение может с большой долей вероятности считаться успешным, если процент распознанных примеров на обучающем и тестовом множествах достаточно велик (близок к 100%). В процессе обучения в окне отображаются следующие графики хода обучения для обучающего (синяя линия) и тестового (красная линия) множеств:

Максимальная квадратичная ошибка – на обучающем множестве и тестовом множестве (сплошная линия).

Средняя квадратичная ошибка – на обучающем множестве и тестовом множестве (пунктирная линия).

Любой из графиков можно скрыть (показать). Для этого достаточно сбросить (установить) соответствующий флажок. Если какой-либо участок графика требуется рассмотреть более подробно, можно увеличить масштаб просмотра данного участка. Для этого следует выделить прямоугольную область с нужным участком, двигая указатель с нажатой левой кнопкой мыши от левого верхнего к правому нижнему углу прямоугольника. Повторение данного действия приведет к дальнейшему увеличению масштаба. Чтобы вернуть исходный масштаб графика, достаточно выделить любой участок графика, двигая при этом мышь с нажатой левой кнопкой от правого нижнего угла прямоугольника к левому верхнему.

В правой части окна постоянно отображаются номер текущей эпохи и время, прошедшее с начала обучения. В поле «Темп обновления» можно задать число эпох обучения сети, через которое будет происходить обновление графика. Флажок «Рестарт» позволяет включить режим ини-

циализации начальных весов сети случайными значениями. Если флажок сброшен, то при повторном запуске обучения после остановки будет иметь место так называемое «дообучение сети», когда обучение будет начато с текущими весами.

Выбор способа отображения.

На данном шаге пользователь должен выбрать, в каком виде будут отображены результаты обработки данных. Для этого достаточно пометить нужные виды отображения флажками и щелкнуть по кнопке «Далее». Например: для выборки данных, полученных в результате обработки данных с помощью нейронной сети, доступны следующие виды отображения: Обучающий набор; Диаграмма рассеяния; Граф нейросети; Что-если; Таблица сопряженности. Таблица; Статистика; Диаграмма; Гистограмма; Куб; Сведения.

ЗАДАНИЕ

Для встроенных в deductor данных «Оценка стоимости недвижимости» постройте нейронную сеть. Гиперпараметры приведены в табл. 1.

Таблица 1.

Параметр	Варианты									
	1	2	3	4	5	6	7	8	9	10
Число скрытых слоев нейронов	1	1	1	1	1	1	1	1	1	1
Число нейронов в слое	2	3	4	2	3	4	2	3	4	2
Функция активации	Сигмоида	Сигмоида	Сигмоида	Сигмоида	Сигмоида	Сигмоида	Сигмоида	Сигмоида	Сигмоида	Сигмоида
Крутизна	0,9	0,9	0,9	1,0	1,0	1,0	1,1	1,1	1,0	0,8
Алгоритм обучения	RPRO P	RPRO P	RPRO P	RPRO P	RPRO P	RPRO P	RPRO P	RPRO P	RPRO P	RPRO P
Шаг спуска	0,5	0,5	0,5	0,5	0,5	0,4	0,4	0,4	0,4	0,5
Шаг подъема	1,2	1,2	1,2	1,2	1,2	1,2	1,1	1,1	1,1	1,2
Ошибка меньше	0,05	0,05	0,05	0,05	0,05	0,01	0,01	0,01	0,01	0,01
Эпоха	500	1000	600	1500	1000	400	800	900	600	700

Методические указания

1. После запуска пакета deductor откроется окно. Загрузите данные «Оценка стоимости недвижимости». Для этого перейдите по вкладке Открыть демопример / samples / Предсказательная аналитика / Оценка стоимости недвижимости (нейросеть).

2. В левой части окна в списке сценариев появится файл с данными. Чтобы посмотреть кликайте дважды мышкой по файлу. Вызовите мастер обработки данных (например, при помощи клавиши F7). Выберите «Нейросеть».

3. На первом этапе необходимо выбрать входные и выходные нейроны. Установите переменную ID – Информационное, стоимость – выходное; остальные переменные – входное. Нажмите Далее.

4. Следующий этап – разбиение исходного набора данных на обучающую и тестирующую. Оставьте настройки по умолчанию. То есть, установите размер обучающего множества 95%, а тестового 5%. Укажите способ разбиения исходного множества данных «Случайно». Это означает, что из всего набора данных пять процентов случайным образом выбранных значений используется для контроля работоспособности построенной нейронной сети, остальные 95% данных используются для обучения нейросети подбором синоптических весов межнейронных связей с помощью специальных алгоритмов.

5. Нажмите Далее. Остальные параметры сети настройте в соответствии с вашим вариантом.

6. Для начала обучения нажмите кнопку пуск в диалоговом окне Запуск процесса обучения нейронной сети. Наблюдайте изменение величины ошибки, а также процента распознанных примеров. Во время процесса обучения нейросети происходит автоматический подбор весовых коэффициентов связи между ее отдельными нейронами, согласно штатному алгоритму, реализованному в аналитической платформе Deductor (в системе используется алгоритм обратного распространения ошибки). Как правило, обучение занимает несколько минут. При необходимости пользуйтесь кнопками «Стоп» и «Пауза».

После остановки обучения запишите в бланк отчета значения средней ошибки для обучающего и тестируемого множества. Затем нажмите «Далее».

7. После окончания обучения выберите визуализаторы Data Minig («Граф нейросети», «Диаграмма рассеяния», «Что-если») и Табличные данные («Обучающий набор», «Таблица» и «Статистика»).

Граф нейросети.

Структура полученной нейронной сети представлена на ее графе. На нем приведена схема связей между отдельными нейронами. Нейроны (они представлены на рисунке красными точками) взаимодействуют при помощи синаптических связей (представлены в виде линий) с весовыми коэффициентами передачи сигнала, найденными в процессе обучения. Цвет найденной связи соответствует ее величине. Красный – сильная связь, синий – слабая.

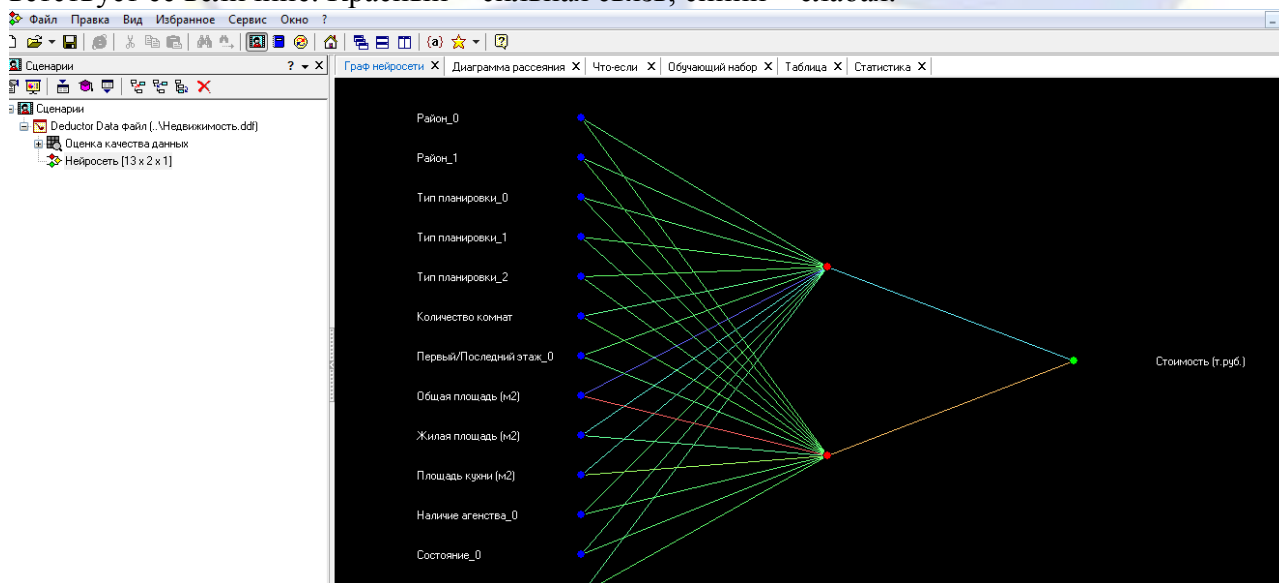


Диаграмма рассеяния показывает расхождение прогнозируемых данных относительно эталонных.

Таблица результатов, указываются исходные входные значения, эталонное значение стоимости квартиры, стоимость квартиры полученная от нейронной сети, значение абсолютной ошибки.

Статистические характеристики: в данном примере наибольший интерес представляют характеристики ошибки моделирования D.

«Что-если» – инструмент для прогнозирования.

8. Сохраните полученный сценарий в файле (имя файла произвольно) нажав Файл → Сохранить как → Сохранить.

9. Перерисуйте граф в бланк отчета.

10. Сделайте прогноз стоимости жилья в соответствии со своим вариантом. Для этого перейдите на вкладку Что-если и введите необходимые параметры. Результат запишите в бланк отчета.

Таблица №2

Параметры квартиры	Номер варианта									
	1	2	3	4	5	6	7	8	9	10
Район	Орджоник.	Правоб.	Лен.	Лен.	Орджоник.	Правоб.	Лен.	Орджоник.	Правоб.	Лен.
Тип планировки	Улучш.	Брежн.	Хрущ.	Улучш.	Брежн.	Улучш.	Брежн.	Улучш.	Брежн.	Хрущ.
Количество комнат	2	2	2	2	2	1	1	1	1	3
Первый/последний этаж	False	False	False	False	True	False	False	False	False	True
Общая площадь	65	43	44	55	45	38	32	39	31	58
Жилая площадь	40	31	31	34	33	20	20	18	20	43
Площадь кухни	8	5	5	9	5	9	6	11	6	5
Наличие агентства	True	True	True	True	False	True	True	True	True	True
Состояние	2	3	4	3	5	5	3	4	3	4

ЛАБОРАТОРНАЯ РАБОТА №6

ПРОГНОЗИРОВАНИЕ ВРЕМЕННЫХ РЯДОВ НА ОСНОВЕ НЕЙРОННЫХ СЕТЕЙ

Цель: изучить возможности нейронных сетей для прогнозирования временных рядов.

Программное обеспечение: Deductor, MS Excel.

Краткая теория

Искусственная нейронная сеть – математическая модель, а также её программное или аппаратное воплощение, построенная по принципу организации и функционирования биологических нейронных сетей – сетей нервных клеток живого организма.

Нейронные сети используются для решения сложных задач, которые требуют аналитических вычислений подобных тем, что делает человеческий мозг. Самыми распространенными применениями нейронных сетей является:

Классификация – распределение данных по параметрам. Например, на вход дается набор людей и нужно решить, кому из них давать кредит, а кому нет. Эту работу может сделать нейронная сеть, анализируя такую информацию как: возраст, пол, семейное положение, кредитная история и т. д.

Прогнозирование – возможность предсказывать следующий шаг. Например, рост или падение акций, основываясь на ситуации на фондовом рынке.

Распознавание (образов, речи, рукописного текста) – в настоящее время, самое широкое применение нейронных сетей. Используется в Google, когда вы ищете фото или в камерах телефонов, когда оно определяет положение вашего лица и выделяет его и многое другое

Выявление закономерностей.

Нейронные сети активно применяются в экономике и бизнесе. Можно привести несколько примеров удачного применения нейросетей:

1. Отслеживание операций с краденными кредитными картами и поддельными чеками.
2. Прогнозирование объема закупок.
3. Прогнозирование изменений биржевых котировок.
4. Управление ценами и производством. Разработаны нейросетевые системы планирования, которые применяются совместно со стандартными методами исследования операций, динамического прогнозирования и т. д. Создана система планирования цен и затрат, основанная на нейросети с использованием генетических алгоритмов. На основе исторических данных система обнаруживает сложные зависимости между затратами на рекламу, объемом продаж, ценой продукции, ценами конкуренции, днем недели и т. д. После этого возможен подбор оптимальной стратегии с точки зрения максимизации объема продаж или прибыли.
5. Оценка стоимости недвижимости.
6. Прогнозирование потребления энергии.
7. Прогнозирование банкротств.
8. Построение рейтингов. В США был предложен следующий способ построения рейтингов: данные сотен фирм, представленных в многомерном пространстве признаков, были обработаны с помощью самообучающихся нейронных сетей (сетей Кохонена), которые произвели классификацию на основе стандартного кластерного анализа. Иными словами, всё множество компаний, представленных в многомерном пространстве, было разложено на группы (кластеры) и тем самым проранжировано в определенном порядке.

Широкое использование нейросетевого подхода обусловлено следующими причинами:

- нейронная сеть способна обучаться на примерах в случаях, когда неизвестен вид и структура взаимосвязей между входными и выходными данными (зависимыми и независимыми переменными изучаемого процесса);
- с помощью нейронных сетей можно получать приемлемое решение даже в случае неполной, искаженной и зашумленной информации;
- нейронная сеть позволяет быстро обработать огромный объем информации и выявить в наблюдаемых данных скрытые закономерности;
- с помощью нейронной сети можно аппроксимировать любую нелинейную зависимость.

Процесс создания нейросетевых моделей включает следующие этапы:

- 1) Определение признаков. Сбор данных.
- 2) Подготовка к автоматизированной обработке.
- 3) Задание гиперпараметров нейронной сети.
- 4) Обучение.
- 5) Тестирование обученной модели.
- 6) Применение.

Гиперпараметры – это значения, которые нужно подбирать вручную и зачастую методом проб и ошибок. Среди таких значений можно выделить:

- Момент и скорость обучения
- Количество скрытых слоев
- Количество нейронов в каждом слое
- Наличие или отсутствие нейронов смещения.
- Тип обучения – онлайн/офлайн
- Функция активации

Временные ряды – статистические данные, отображающие развитие во времени изучаемого явления.

Считают, что значения уровней временных рядов экономических показателей складываются из следующих составляющих (компонент): тренда, сезонной, циклической и случайной:

- длительные, постоянно действующие факторы оказывают на изучаемое явление определяющее влияние и формируют основную тенденцию ряда, называемую трендом $T(t)$;

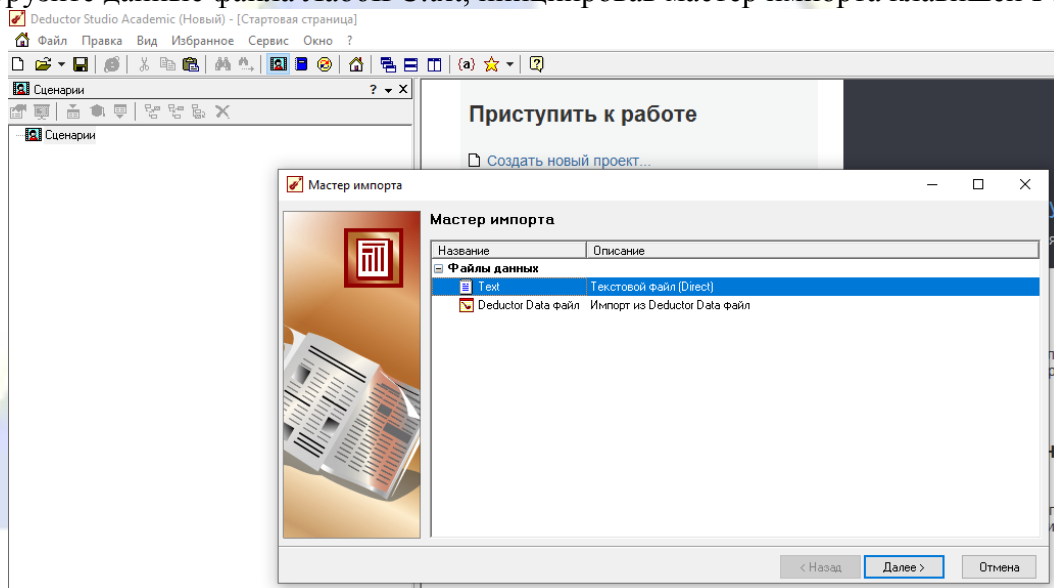
- кратковременные, периодические факторы формируют сезонные колебания ряда $S(t)$, период сезонных колебаний не превышает одного года; если период более одного года, то говорят о наличии циклической составляющей $BP V(t)$;
- случайная компонента $\varepsilon(t)$ отражает влияние не поддающихся учету и регистрации случайных факторов.

Задание №1

Сделать точечный прогноз индекса потребительских цен IPC в России на 6 месяцев. Построить график фактических и прогнозных данных.

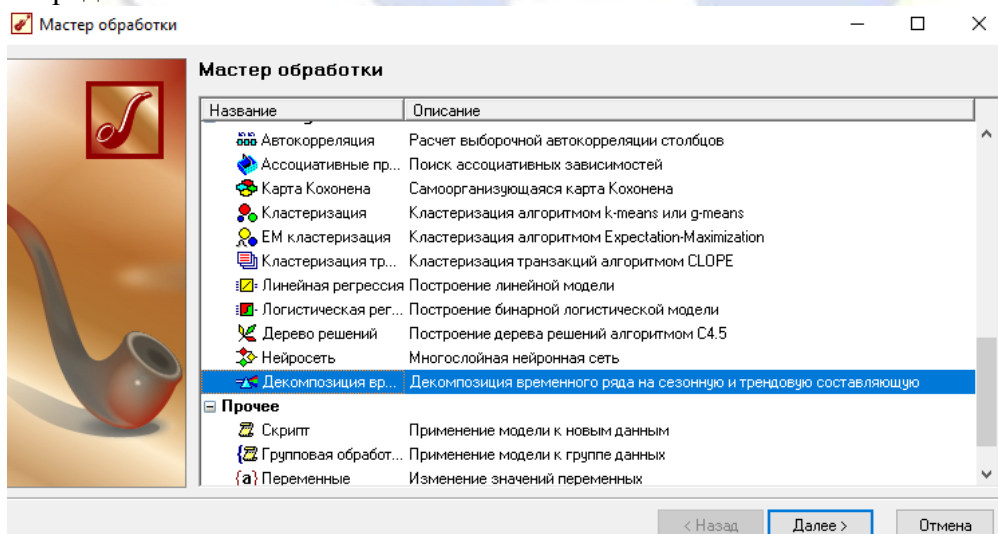
Методические указания

1. Запустите программу Deductor Studio Academic и создайте новый проект. Для этого перейдите в главное меню на вкладку Файл / Создать.
2. Загрузите данные файла ЛаббIPC.txt, инициировав мастер импорта клавишей F6.



Выберите импорт данных из текстового файла. Нажмите Далее. Укажите путь к нужному текстовому файлу. Установите форматы импорта и разделители по умолчанию. Установите тип данных для переменных IPC , N – вещественный, t – дата/время. Запустите кнопкой «Пуск» процесс импорта данных в аналитическую систему. Определите отображение данных в виде таблицы и завершите импорт, нажав кнопку «Готово». В результате проделанных действий обучающий массив будет введен в буфер аналитической системы Deductor.

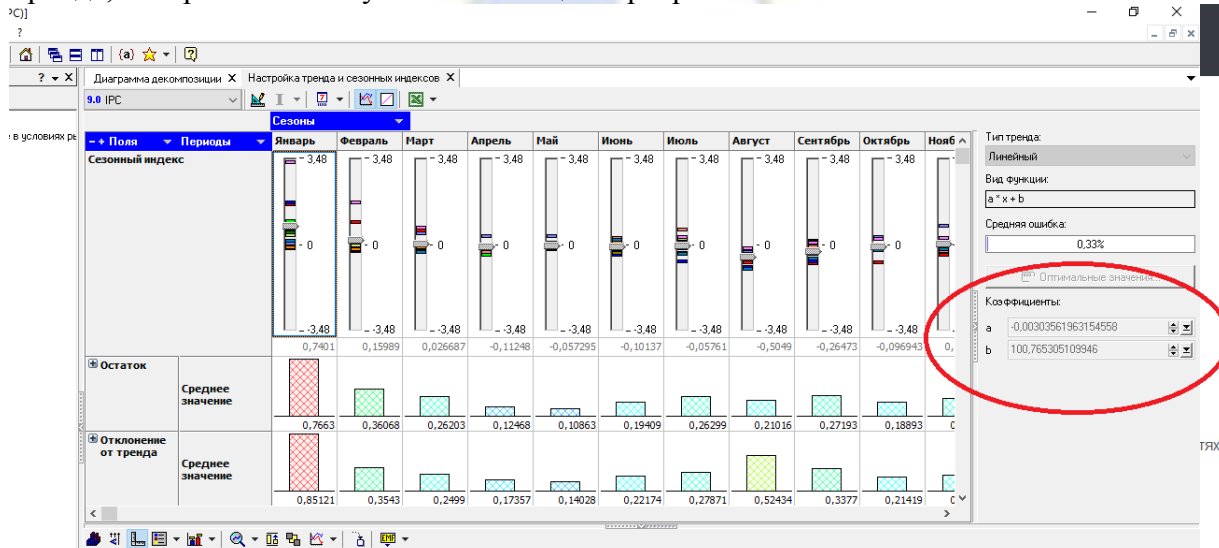
3. Активизируйте мастер обработки данных (клавиша F7). Иницируйте режим «Декомпозиция временного ряда».



Установите для переменной «IPC» назначение «Используемое», тренд «Линейный». Прделайте пункт 3 «Мастера обработки» по умолчанию. Установите способы отображения Диаграмма декомпозиции ряда и Настройка тренда и сезонных индексов.

Прделайте пункты 5 и 6 «Мастера обработки» по умолчанию.

4. Откройте файл ЛаббIPC.xlsx. Добавьте справа от данных два столбца T – трендовая составляющая и S – сезонная составляющая. Для выделения трендовой составляющей внесите формулу тренда, которая была получена с помощью программы Deductor:

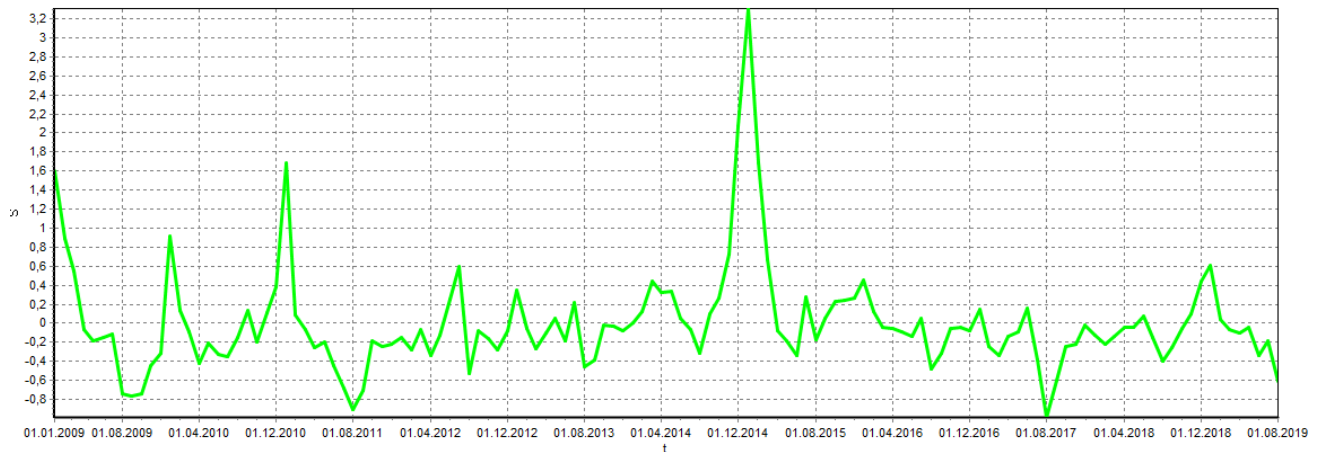


В качестве аргумента X используйте данные в столбце N . Сезонную составляющую можно оценить как разность между столбцами IPC и T (фактически в S будет входить и случайная составляющая).

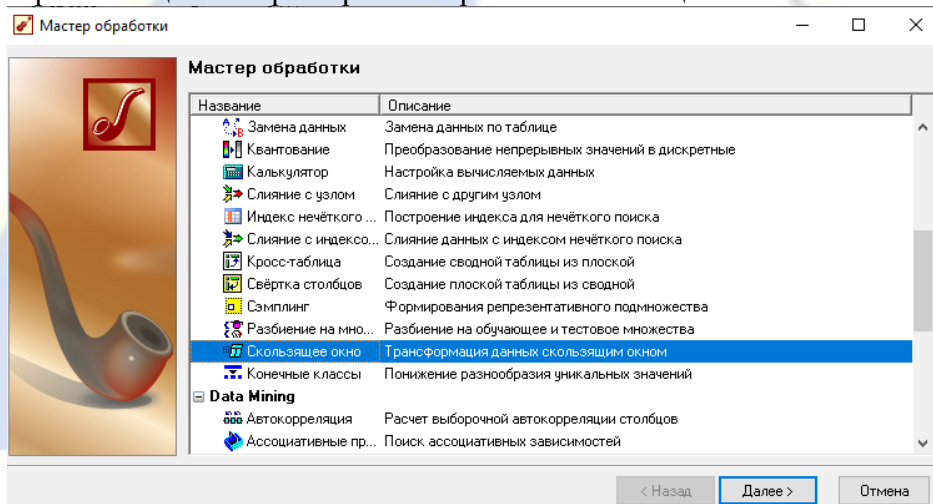
5. Скопируйте содержимое листа «Данные» на новый лист «Сезонная компонента» программы Excel как «Значения». Удалите с листа «Сезонная компонента» столбцы «IPC» и «Т». Сохраните лист как файл «Сезонная компонента. txt» как текстовый файл с разделителями табуляция.

	A	B	C
1	t	N	S
2	01.01.2009	1	1,6077
3	01.02.2009	2	0,8908
4	01.03.2009	3	0,5538
5	01.04.2009	4	-0,0631
6	01.05.2009	5	-0,1801
7	01.06.2009	6	-0,1471
8	01.07.2009	7	-0,1140
9	01.08.2009	8	-0,7410
10	01.09.2009	9	-0,7679
11	01.10.2009	10	-0,7349
12	01.11.2009	11	-0,4419
13	01.12.2009	12	-0,3188
14	01.01.2010	13	0,9142

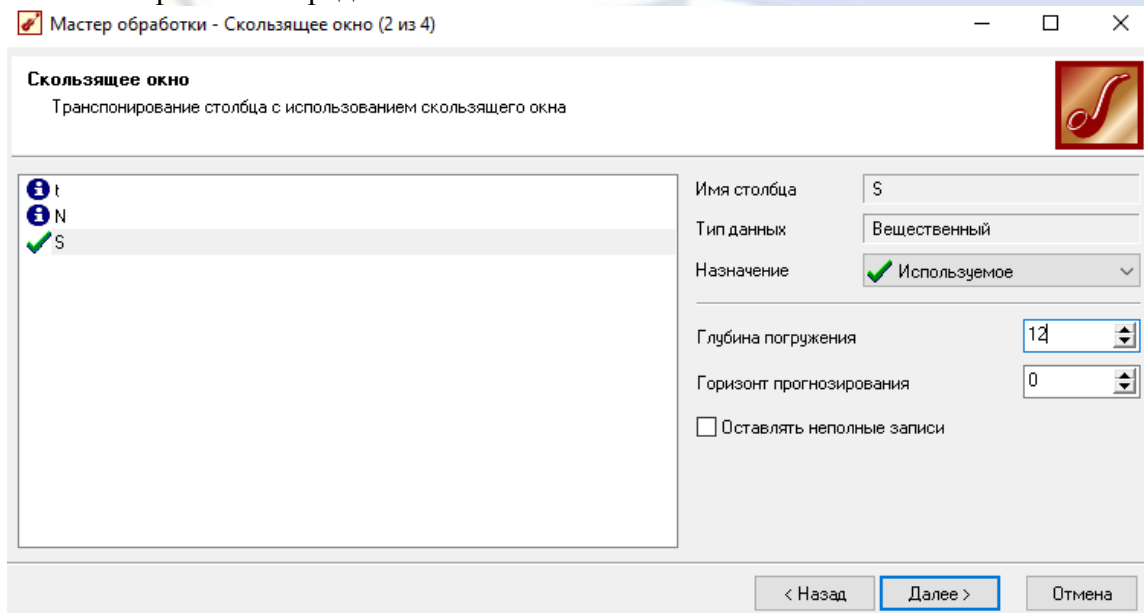
6. Введите файл Сезонная компонента.txt в аналитическую платформу Deductor. Установите типы входных данных и способы отображения – Типы данных – t —> Дата/время, N —> Целый и IPC —> Вещественный; – Способы отображения – Таблица и Диаграмма. В результате должна быть получена следующая диаграмма:



7. Задайте при помощи мастера обработки режим скользящее окно:



Для переменной S установите назначение – Используемое, глубину погружения – 12. Данное преобразование позволяет получить определенное количество (в данном случае 12) предыдущих значений временного ряда.



Нажмите Далее. Установите способ отображения «Таблица».

8. На основе данной таблицы постройте нейронную сеть.

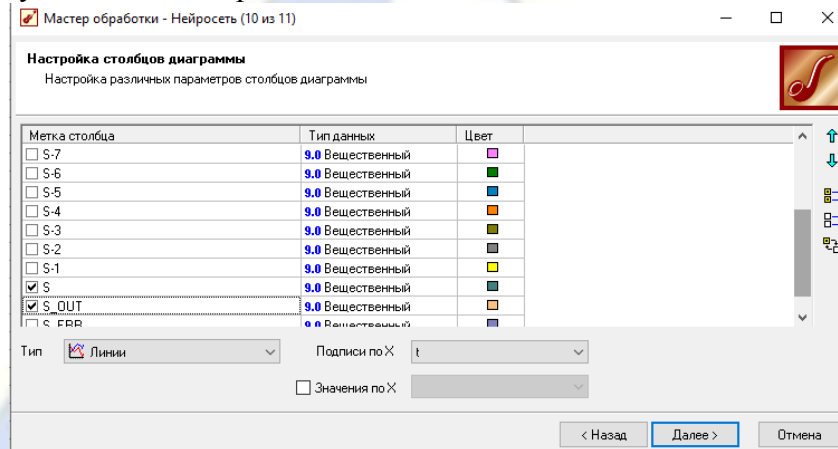
- количество входов – 12 ($S-1, S-2, S-3, \dots, S-12$).
- количество выходов – 1 (S – сезонная компонента)
- количество нейронов – 4

- приемлемая ошибка менее – 0,01
- количество эпох обучения – 5000.

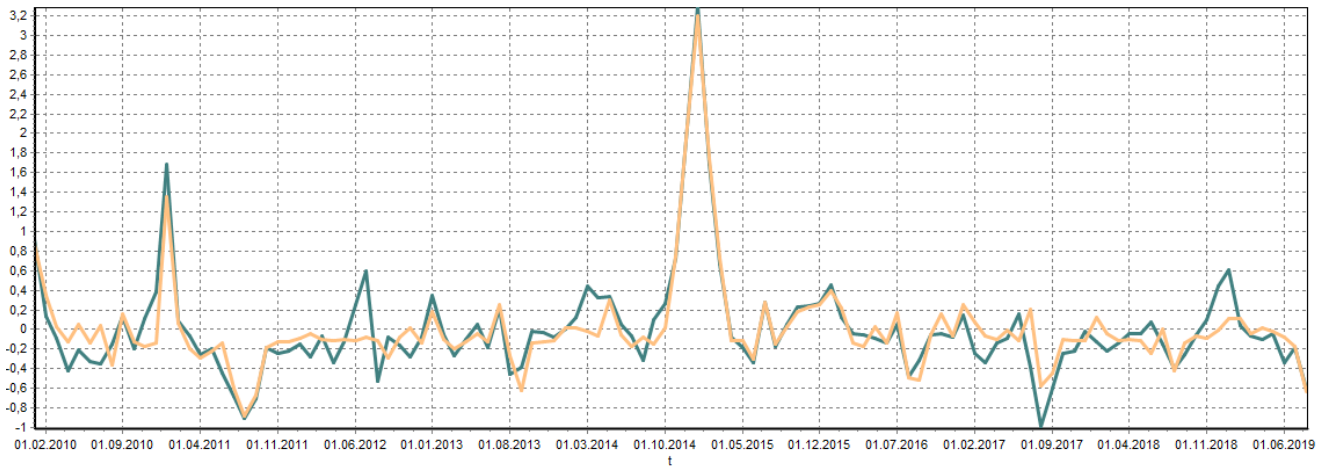
Остальные параметры – по умолчанию.

Произведите обучение нейронной сети (нажав кнопку «Пуск»). Задайте способы отображения результатов построения: граф нейросети, диаграмма рассеивания, диаграмма.

Произведите настройку столбцов диаграммы

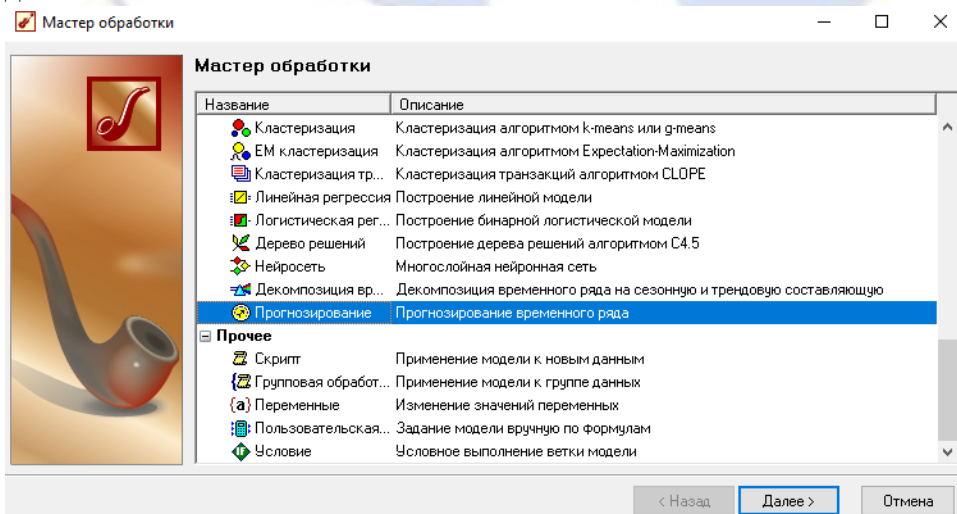


В результате должна быть получена следующая диаграмма:



На ней двумя различными цветами показаны исходная кривая временного ряда и значения, получаемая в результате моделирования с использованием нейронной сети.

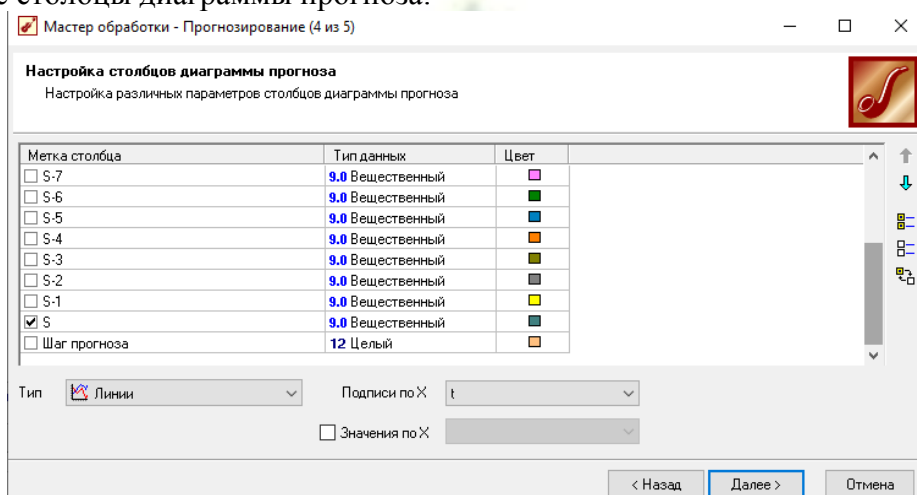
9. Находясь в строке «Нейросеть» мастера обработки запустите режим прогнозирования временного ряда.



Установите горизонт прогноза, равный 6.

Определите способы отображения: диаграмма прогноза, таблица.

Настройте столбцы диаграммы прогноза:



Нажмите далее, затем Готово и произведите прогноз на 6 месяцев.

10. В результате проделанных действий будут получены диаграмма и таблица прогноза.

S	Шаг прогноза
-0,0495	
-0,0365	
-0,0735	
0,1496	
-0,2474	
-0,3343	
-0,1313	
-0,0883	
0,1548	
-0,3822	
-0,9891	
-0,5961	
-0,2431	
-0,0983	
-0,0453	
-0,3423	
-0,1792	
-0,6162	
-0,907603198461056	1
-0,212728850519361	2
-0,0579475447120438	3
-0,0689470859567514	4
-0,00990548253616544	5
0,0895256394413844	6

11. Перенесите полученные в п.10 значения в файл Лаб4РС.xls с исходными данными в столбец S. Соответственно необходимо заполнить другие столбцы N, T и S. Прогнозное значение индекса потребительских цен найдите, сложив два столбца T и S. Прогнозные значения выделите красным цветом.

12. Прогнозные значения перенесите в бланк отчета.

13. В пакете MS Excel постройте график фактических данных и прогнозных значений. Тип диаграммы выберите Точечная с гладкими кривыми; цвет ряда фактических данных сделайте синим, прогнозных данных – красным. График распечатайте и вклейте в бланк отчета.

2.3.6. Методика проведения лекции-беседы

Лекция-беседа – диалогический метод изложения и усвоения учебного материала. Лекция-беседа позволяет с помощью системы вопросов, умелой их постановки и искусного поддержания диалога воздействовать как на сознание, так и на подсознание обучающихся, научить их самокоррекции. Проведение лекции-беседы предполагает наличие определенного объема знаний об изучаемом материале и связи с ним. Лекция-беседа помогает побудить обучающихся к актуализации имеющихся знаний, вовлечь их в процесс самостоятельных размышлений, в эвристический, творческий процесс получения новых знаний; способствует активизации познавательной деятельности, вовлекает в максимальный мыслительный поиск, с целью разрешения противоречий, подводит к самостоятельному формированию выводов и обобщений, создает условия для оперативного

управления процессом познания.

По назначению в учебном процессе выделяют следующие виды лекции-беседы:

- вводные или вступительные (организующие);
- сообщения новых знаний;
- закрепляющие.

Вводная лекция-беседа проводится в начале лекционного занятия. С ее помощью обеспечивается психологическая настройка обучающихся на восприятие и усвоение нового материала. Беседа способствует пониманию значения предстоящей работы, формирует представления о ее содержании, специфике и особенностях.

Сообщение новых знаний строится в форме вопросов и ответов преимущественно при анализе прочитанных текстов, запоминании ответов; способствует подведению обучающихся за счет умело поставленных вопросов, имеющихся знаний и жизненного опыта, к усвоению новых знаний, формулированию понятий, решению задач; создает субъективное впечатление, что обучающийся сам сделал открытие, проделал путь от практики к научной истине.

Закрепляющие лекции-беседы применяются для закрепления, обобщения и систематизации знаний.

Эффективность беседы зависит от тщательной подготовки преподавателя, продуманности и профессиональной формулировки вопросов в четкой постановке, их логической последовательности. Вопросы должны развивать все виды мышления, обеспечивать логическую форму мышления (весь спектр мыслительных действий), соответствовать уровню развития обучающихся; со стороны обучающихся ответы должны быть осознанными и аргументированными, полными, точными, ясными, правильно сформулированными.

2.3.7. Методика проведения работы в малых группах

Работа в малых группах – это одна из самых популярных стратегий, так как она дает всем обучающимся (в том числе и стеснительным) возможность участвовать в работе, практиковать навыки сотрудничества, межличностного общения (в частности, умение активно слушать, вырабатывать общее мнение, разрешать возникающие разногласия). Все это часто бывает невозможно в большом коллективе. При организации групповой работы, следует обращать внимание на следующие ее аспекты. Нужно убедиться, что учащиеся обладают знаниями и умениями, необходимыми для выполнения группового задания. Нехватка знаний очень скоро даст о себе знать - учащиеся не станут прилагать усилий для выполнения задания. Надо стараться сделать свои инструкции максимально четкими. Надо предоставлять группе достаточно времени на выполнение задания.

Деление обучающихся на группы – это важный момент в организации работы в малых группах. Способов деления обучающихся на группы существует множество, и они в значительной степени определяют то, как будет протекать дальнейшая работа в группе, и на какой результат эта группа выйдет.

Способы деления обучающихся на группы:

1) По желанию.

Объединение в группы происходит по взаимному выбору. Задание на формирование группы по желанию может даваться, как минимум, в двух вариантах:

Разделитесь на группы по ... человек. Разделитесь на ... равные группы.

2) Случайным образом.

Группа, формируемая по признаку случайности, характеризуется тем, что в ней могут объединяться (правда, не по взаимному желанию, а волей случая) дети, которые в иных условиях никак не взаимодействуют между собой либо даже враждуют. Работа в такой группе развивает у участников способность приспосабливаться к различным условиям деятельности и к разным деловым партнерам.

Этот метод формирования групп полезен в тех случаях, когда перед преподавателем стоит задача научить обучающихся сотрудничеству. В этом случае преподаватель должен обладать достаточной компетентностью в работе с межличностными конфликтами.

Способы формирования «случайной» группы: жребий; объединение тех, кто сидит рядом (в одном ряду, в одной половине аудитории); с помощью импровизированных «фантов» (один из обучающихся с закрытыми глазами называет номер группы, куда отправится обучающийся, на которого указывает в данный момент преподаватель) и т.п.

3) По определенному признаку.

Такой признак задается либо преподавателем, либо любым обучающимся. Так, можно разделить по первой букве имени (гласная – согласная), в соответствии с тем, в какое время года родился (на четыре группы), по цвету глаз (карие, серо-голубые, зеленые) и так далее.

Этот способ деления интересен тем, что, с одной стороны, может объединить обучающихся, которые либо редко взаимодействуют друг с другом, либо вообще испытывают эмоциональную неприязнь, а с другой – изначально задает некоторый общий признак, который сближает объединившихся. Есть нечто, что их роднит и одновременно отделяет от других. Это создает основу для эмоционального принятия друг друга в группе и некоторого отдаления от других (по сути дела – конкуренции).

4) По выбору «лидера».

«Лидер» в данном случае может либо назначаться преподавателем (в соответствии с целью, поэтому в качестве лидера может выступать любой обучающийся), либо выбираться обучающимися. Формирование групп осуществляется самими «лидерами». Например, они по очереди называют имена тех, кого они хотели бы взять в свою группу. Наблюдения показывают, что в первую очередь «лидеры» выбирают тех, кто действительно способен работать и достигать результата. Иногда даже дружба и личные симпатии отходят на второй план.

В том случае, если в аудитории есть явные аутсайдеры, для которых ситуация набора в команду может быть чрезвычайно болезненной, лучше или не применять этот способ, или сделать их «лидерами».

5) По выбору преподавателя.

В этом случае преподаватель создает группы по некоторому важному для него признаку, решая тем самым определенные педагогические задачи. Он может объединить обучающихся с близкими интеллектуальными возможностями, со схожим темпом работы, а может, напротив, создать равные по силе команды. При этом организатор групповой работы может объяснить принцип объединения, а может уйти от ответа на вопросы участников по этому поводу.

При работе в малой группе учащиеся могут выполнять следующие роли:

- фасилитатор (посредник-организатор деятельности группы);
- регистратор (записывает результаты работы);
- докладчик (докладывает результаты работы группы);
- журналист (задает уточняющие вопросы, которые помогают группе лучше выполнить задание, например те вопросы, которая могла бы задать другая сторона в дискуссии);
- активный слушатель (старается пересказать своими словами то, о чем только что говорил кто-либо из членов группы, помогая сформулировать мысль);
- наблюдатель (должен отмечать признаки определенного поведения, заранее описанного преподавателем, и определять, как члены группы справляются с возникающими по ходу работы проблемами. Отчитываясь перед группой, наблюдатели обязаны представлять свои заметки в максимально описательной и объективной форме);
- хронометрист (следит за временем, отпущенным на выполнение задания) и другие.

Рекомендации по организации работы в малых группах:

- нельзя принуждать к общей работе обучающихся, которые не хотят вместе работать;
- следует разрешить работать индивидуально обучающемуся, который хочет работать один;
- нельзя требовать в аудитории абсолютной тишины, так как обучающиеся должны обмениваться мнениями, прежде чем представить «продукт» совместного труда.

3. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков по дисциплине

Обучающемуся в ходе изучения дисциплины предоставляется возможность набрать не более 60 баллов за текущую работу в семестре и до 40 баллов, включительно, на зачёте (зачете с оценкой).

Результирующая оценка по дисциплине складывается из суммы баллов текущего контроля и промежуточной аттестации в форме зачёта. К зачёту допускаются обучающиеся, посетившие не менее 30% лекционных и лабораторных занятий, с рейтингом не менее 30 баллов по результатам текущего контроля успеваемости. Текущий контроль по лабораторным занятиям выполняет преподаватель, проводивший эти занятия.

Баллы за текущую работу по дисциплине складываются из следующих видов деятельности студента.

Нормативы реализации балльно-рейтинговой системы

№ п/п	Деятельность студента для начисления баллов	Количество баллов
1	Посещение лекций и ведение конспектов	9
2	Выполнение лабораторных работ	45
3	Самостоятельная работа студентов	6
	Всего за семестр (не более)	60