

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Смоленский государственный университет»  
Кафедра прикладной математики и информатики

«Утверждаю»  
Проректор по учебно-  
методической работе  
\_\_\_\_\_ Ю.А. Устименко  
«23» июня 2022 г.

**Рабочая программа дисциплины  
Б1.В.01.06 Основы искусственного интеллекта**

Направление подготовки: **44.03.05 Педагогическое образование (с двумя профилями подготовки)**

Направленность (профиль): **Математика, Информатика**

Форма обучения: очная

Курс – 4

Семестр – 9

Всего зачетных единиц – 2 часов – 72

Форма отчетности: экзамен – 9 семестр

Программу разработал  
кандидат физ.-мат. наук, доцент Емельченков Е.П.

Одобрена на заседании кафедры  
«16» июня 2022 г., протокол № 10

Заведующий кафедрой \_\_\_\_\_ С.В. Козлов

Смоленск  
2022

## 1. Место дисциплины в структуре ОП

Дисциплина «Основы искусственного интеллекта» относится к обязательным дисциплинам вариативной части учебного плана. Данная дисциплина изучается в 8 семестре, и является логическим продолжением курсов «Операционные системы», «Информационные системы», «Администрирование информационных систем», «Администрирование облачных технологий» и др.

Курс построен так, чтобы сформировать у студентов целостное представление об основных понятиях Интернета вещей и методах его использования.

Изучение курса основано на традиционных методах высшей школы, тесной взаимосвязи со смежными курсами, а также на использовании современных информационных систем и технологий.

## 2. Планируемые результаты обучения по дисциплине

Компетенция	Индикаторы достижения
<b>ПК-6.</b> Способен использовать научные знания в предметной области (информатика) в процессе формирования предметной компетенции обучающихся в рамках реализации основной общеобразовательной программы	<b>Знать:</b> назначение, структуру и содержание курса информатики, современное состояние и перспективы развития информатики как учебной дисциплины, ее место и роль, фундаментальное ядро современного школьного курса информатики, принципы построения методической системы обучения информатике, ее основных компонентов. <b>Уметь:</b> анализировать цели и содержание школьного курса информатики, проектировать образовательный процесс, использовать дидактический потенциал средств информационных технологий в реализации образовательного процесса по курсу информатики; <b>Владеть:</b> основными видами профессиональной деятельности учителя информатики, профессиональными навыками реализации методики обучения основным разделам курса информатики, современными информационно-коммуникационными средствами для эффективного осуществления профессиональной деятельности.
<b>ПК-8.</b> Способен использовать современные системные программные средства, разрабатывать моделирующие алгоритмы и реализовывать их на базе языков и пакетов прикладных программ моделирования	<b>Знать:</b> основные принципы и методики создания алгоритмов и программ для решения прикладных задач, основные среды для разработки программного обеспечения, базовые информационные технологии программные средства; <b>Уметь:</b> корректно использовать современные информационные технологии и программные средства, внедрять и адаптировать прикладное программное обеспечение; <b>Владеть:</b> навыками решения прикладных задач с применением современных программных средств, владеть современными языками программирования и методиками разработки и внедрения прикладного программного обеспечения.

### 3. Содержание дисциплины

1. **История искусственного интеллекта (ИИ).** Основные разделы, на которых базируется ИИ: статистика, базы данных, машинное обучение, нейронные сети, генетические алгоритмы. Сферы применения ИИ: логистика, финансы и банковское дело, маркетинг, медицина, биоинформатика, интернет, компьютерные игры. Большие данные в СИИ.
2. **Машинное обучение (МО).** Обучение с учителем. Обучение без учителя. Постановка задачи обучения по прецедентам. Типы задач: классификация, регрессия, прогнозирование, кластеризация. Эвристики. Подготовка данных для обучения; построение моделей; проверка и оценка моделей для тестовых выборок. Оценки точности и полноты моделей. Примеры прикладных задач.
3. **Методы классификации данных.** Деревья решений. Методы построения деревьев решений. Жадный алгоритм ID3. Критерии качества дерева решений (критерий Джини, энтропийный и регуляризирующий критерии). Метод ближайших соседей (kNN) и его обобщения. Подбор числа k по критерию скользящего контроля.
4. **Кривая ошибок (ROC curve).** Алгоритм эффективного построения ROC-кривой. Градиентный метод максимизации площади под ROC-кривой. Пример оценки качества бинарной классификации методом ROC-анализа.
5. **Наивный байесовский классификатор.** Преимущества и недостатки. Теорема об оптимальности байесовского классификатора. Задача категоризации текстовых документов. Байесовские сети. Метод опорных векторов. Понятие опорных векторов. Оптимальная разделяющая гиперплоскость. Понятие зазора между классами (margin).
6. **Логистическая регрессия.** Принцип максимума правдоподобия и логарифмическая функция потерь. Предсказание зависимостей. Оценивание рисков. Пример прикладной задачи: кредитный скоринг. Бинаризация признаков. Скоринговые карты и оценивание вероятности дефолта. Риск кредитного портфеля банка.
7. **Задача регрессии.** Исправление грубых ошибок. Зашумленные данные и контрольные выборки. Многомерная линейная регрессия. Проблемы мультиколлинеарности и переобучения. Нелинейная параметрическая регрессия. Регрессия методом опорных векторов. Пример прикладной задачи: прогнозирование потребительского спроса.
8. **Методы кластеризации.** Постановка задачи кластеризации. Примеры прикладных задач. Типы кластерных структур. Графовые алгоритмы кластеризации. Функционалы качества кластеризации. Статистические алгоритмы: EM-алгоритм и Алгоритм k средних (k-means). Методы кластеризации с нечетким разбиением исходных данных. Алгоритм Fuzzy c-means и его использование в компьютерном зрении.
9. **Нейросетевая парадигма в искусственном интеллекте.** Нейронные сети (НС), принцип построения. Архитектура НС: однослойные, многослойные, с обратной связью. Искусственный нейрон. Активационные функции. Методы обучения нейронных сетей. Решение задач классификации с помощью нейронных сетей. Применения нейросетей в задачах распознавания и прогнозирования. Пример: разработка модуля распознавания графических данных.
10. **Эволюционные и генетические алгоритмы.** Этапы работы генетического алгоритма. Кодирование информации и формирование популяции. Селекция. Скрещивание и формирование нового поколения. Мутация. Настройка параметров генетического алгоритма. Применение генетического алгоритма для решения задач оптимизации и аппроксимации. Генетический алгоритм в решении задачи коммивояжера.
11. **Решение проблем ИИ методами поиска.** Поиск на основе логики. Шахматные программы. Вероятностные рассуждения. Нечеткая логика. Обход препятствий. Перемещение в случайном направлении. Методы полного перебора, поиск в ширину, поиск в глубину, двунаправленный поиск. Эвристический поиск, требования к эвристическим функциям. Оценочные функции и их использование. Алгоритм A\*. Пример реализации эвристического алгоритма A\*. Natural Language Processing и информационно - поисковые системы.

12. **Методы представления и обработки нечеткой информации в интеллектуальных системах.** Применение нечеткой логики в СИИ. Понятие о нечетких множествах и их связь с теорией построения СИИ. Функция принадлежности элемента подмножеству. Операции над нечеткими множествами. Нечеткая логика для управления безопасностью автомобиля.
13. **Интеллектуальные многоагентные системы (МАС).** Архитектура МАС. Определение "умного" агента, принципы его мышления. Методологии, применяемые интеллектуальными агентами в задачах поиска. Международные стандарты создания агентов и платформ МАС. Платформы для построения МАС: Java Agent Development Framework (Jade), ABLE, REPAST. Средство мультиагентного моделирования MASON. Пример проектирования МАС для распределения заказов такси.
14. **Применения СИИ в обработке изображений и компьютерном зрении.** Распознавание графических объектов методом Виолы-Джонса. СИИ в робототехнике. Планирование движения робота.

#### 4. Тематический план

№ п/п	Разделы и темы	Всего часов	Формы занятий			
			лекции	практические занятия	лабораторные занятия	самостоятельная работа
1	История искусственного интеллекта (ИИ).	4	1	–	1	2
2	Машинное обучение (МО).	4	1	–	1	2
3	Методы классификации данных.	4	1	–	1	2
4	Кривая ошибок (ROC curve).	4	1	–	1	2
5	Наивный байесовский классификатор.	6	2	–	2	2
6	Логистическая регрессия.	4	1	–	1	2
7	Задача регрессии.	4	1	–	1	2
8	Методы кластеризации.	6	2	-	2	2
9	Нейросетевая парадигма в искусственном интеллекте.	6	1	-	1	4
10	Эволюционные и генетические алгоритмы.	6	1	-	1	4
11	Решение проблем ИИ методами поиска.	6	1	-	1	4
12	Методы представления и обработки нечеткой информации в интеллектуальных системах.	6	1	-	1	4
13	Интеллектуальные многоагентные системы (МАС).	6	1	-	1	4
14	Применения СИИ в	6	1	-	1	4

	обработке изображений и компьютерном зрении.					
	<b>ИТОГО</b>	<b>72</b>	<b>16</b>	<b>–</b>	<b>16</b>	<b>13+27</b>

## 5. Виды образовательной деятельности

### Занятия лекционного типа

**1. История искусственного интеллекта. Машинное обучение (МО).** Основные разделы, на которых базируется ИИ: статистика, базы данных, машинное обучение, нейронные сети, генетические алгоритмы. Сферы применения ИИ: логистика, финансы и банковское дело, маркетинг, медицина, биоинформатика, интернет, компьютерные игры. Большие данные в СИИ.

Обучение с учителем. Обучение без учителя. Постановка задачи обучения по прецедентам. Типы задач: классификация, регрессия, прогнозирование, кластеризация. Эвристики. Подготовка данных для обучения; построение моделей; проверка и оценка моделей для тестовых выборок. Оценки точности и полноты моделей. Примеры прикладных задач.

**2. Методы классификации данных. Кривая ошибок (ROC curve).** Деревья решений. Методы построения деревьев решений. Жадный алгоритм ID3. Критерии качества дерева решений (критерий Джини, энтропийный и регуляризирующий критерии). Метод ближайших соседей (kNN) и его обобщения. Подбор числа k по критерию скользящего контроля.

Алгоритм эффективного построения ROC-кривой. Градиентный метод максимизации площади под ROC-кривой. Пример оценки качества бинарной классификации методом ROC-анализа.

**3. Наивный байесовский классификатор.** Преимущества и недостатки. Теорема об оптимальности байесовского классификатора. Задача категоризации текстовых документов. Байесовские сети. Метод опорных векторов. Понятие опорных векторов. Оптимальная разделяющая гиперплоскость. Понятие зазора между классами (margin).

**4. Логистическая регрессия. Задача регрессии.** Принцип максимума правдоподобия и логарифмическая функция потерь. Предсказание зависимостей. Оценивание рисков. Пример прикладной задачи: кредитный скоринг. Бинаризация признаков. Скоринговые карты и оценивание вероятности дефолта. Риск кредитного портфеля банка. Задача регрессии. Исправление грубых ошибок. Зашумленные данные и контрольные выборки. Многомерная линейная регрессия. Проблемы мультиколлинеарности и переобучения. Нелинейная параметрическая регрессия. Регрессия методом опорных векторов. Пример прикладной задачи: прогнозирование потребительского спроса.

**5. Методы кластеризации.** Постановка задачи кластеризации. Примеры прикладных задач. Типы кластерных структур. Графовые алгоритмы кластеризации. Функционалы качества кластеризации. Статистические алгоритмы: EM-алгоритм и Алгоритм k средних (k-means). Методы кластеризации с нечетким разбиением исходных данных. Алгоритм Fuzzy c-means и его использование в компьютерном зрении.

**6. Нейросетевая парадигма в искусственном интеллекте. Эволюционные и генетические алгоритмы.** Нейронные сети (НС), принцип построения. Архитектура НС: однослойные, многослойные, с обратной связью. Искусственный нейрон. Активационные функции. Методы обучения нейронных сетей. Решение задач классификации с помощью нейронных сетей. Применения нейросетей в задачах распознавания и прогнозирования. Пример: разработка модуля распознавания графических данных. Этапы работы генетического алгоритма. Кодирование информации и формирование популяции. Селекция. Скрещивание и формирование нового поколения. Мутация. Настройка параметров генетического алгоритма. Применение генетического алгоритма для решения задач оптимизации и аппроксимации. Генетический алгоритм в решении задачи коммивояжера.

**7. Решение проблем ИИ методами поиска. Методы представления и обработки нечеткой информации в интеллектуальных системах.** Поиск на основе логики. Шахматные программы. Вероятностные рассуждения. Нечеткая логика. Обход препятствий. Перемещение в

случайном направлении. Методы полного перебора, поиск в ширину, поиск в глубину, двунаправленный поиск. Эвристический поиск, требования к эвристическим функциям. Оценочные функции и их использование. Алгоритм A\*. Пример реализации эвристического алгоритма A\*. Natural Language Processing и информационно - поисковые системы. Применение нечеткой логики в СИИ. Понятие о нечетких множествах и их связь с теорией построения СИИ. Функция принадлежности элемента подмножеству. Операции над нечеткими множествами. Нечеткая логика для управления безопасностью автомобиля.

**8. Интеллектуальные многоагентные системы (МАС). Применения СИИ в обработке изображений и компьютерном зрении.** Архитектура МАС. Определение "умного" агента, принципы его мышления. Методологии, применяемые интеллектуальными агентами в задачах поиска. Международные стандарты создания агентов и платформ МАС. Платформы для построения МАС: Java Agent Development Framework (Jade), ABLE, REPAST. Средство мультиагентного моделирования MASON. Пример проектирования МАС для распределения заказов такси. Распознавание графических объектов методом Виолы-Джонса. СИИ в робототехнике. Планирование движения робота.

#### **Занятия семинарского типа**

Не предусмотрены.

#### **Практические занятия**

Не предусмотрены.

#### **Лабораторные работы**

ЛР 1 «Обработка исходных данных для машинного обучения. Методы классификации данных».

ЛР 2 «Машинное обучение методом наивного Байеса. Оценка качества машинного обучения методом ROC – анализа».

ЛР 3 «Машинное обучение методом логистической регрессии. Алгоритмы регрессии в машинном обучении».

ЛР 4 «Алгоритмы кластеризации в машинном обучении».

ЛР 5 «Нейронные сети в машинном обучении. Генетические алгоритмы».

ЛР 6 «Нечеткие методы машинного обучения».

ЛР 7 «Разработка интеллектуального агента».

ЛР 8 «Компьютерное зрение».

Задания для лабораторных работ по дисциплине «Основы искусственного интеллекта» предоставляется студентам на занятиях в электронном виде.

#### **ЛР 1 «Обработка исходных данных для машинного обучения. Методы классификации данных».**

Гибель Титаника является одним из самых печально известных кораблекрушений в истории. 15 апреля 1912 года во время своего первого плавания, Титаник затонул после столкновения с айсбергом, при этом погибло 1502 из 2224 пассажиров и членов экипажа. Некоторые группы людей имели больше шансов выжить, по сравнению с другими. Например, женщины, дети, пассажиры высшего класса.

В этом примере применяются методы машинного обучения для предсказания того, какие пассажиры могли выжить в этой трагедии. Для обучения применяются данные, размещенные на ресурсе: Machine Learning - **dataset Titanic:** test.csv и train.csv (<https://russianblogs.com/article/49401398867/>).

**Задание 1.** Используя библиотеки машинного обучения, выполнить импорт наборов данных test.csv и train.csv содержащий пропущенные данные. Выполнить:

- замену отсутствующих средними (максимальными) по столбцу;

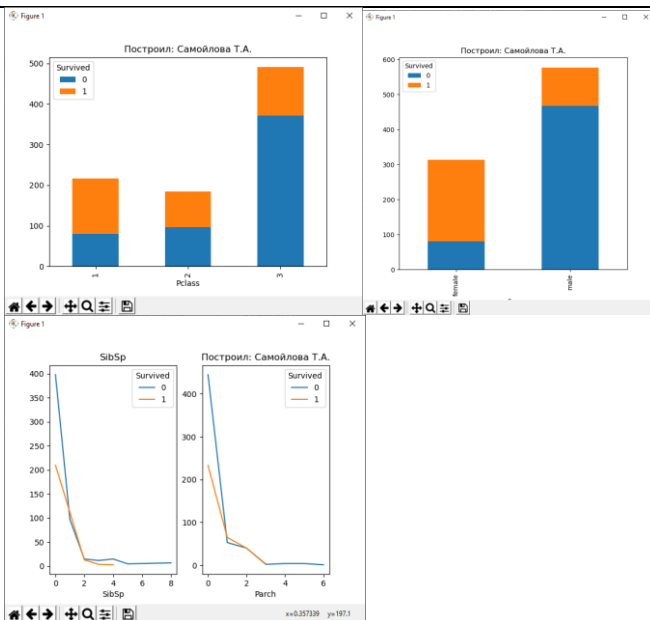
- замену категориальных признаков числовыми;
- вывод диаграмм: 1) зависимость числа умерших от класса каюты, 2) зависимость числа умерших от пола;
- вывод графиков зависимости числа умерших 1) от количества братьев (сестер), 2) от наличия родителей;

Результаты выполнения:

	Survived	Pclass	Sex	SibSp	Parch	Fare	Embarked
0	0	3	male	1	0	7.2500	S
1	1	1	female	1	0	71.2833	C
2	1	3	female	0	0	7.9250	S
3	1	1	female	1	0	53.1000	S
4	0	3	male	0	0	8.0500	S

	Survived	Pclass	Sex	SibSp	Parch	Fare	Embarked
0	0	3	1	1	0	7.2500	2
1	1	1	0	1	0	71.2833	0
2	1	3	0	0	0	7.9250	2
3	1	1	0	1	0	53.1000	2
4	0	3	1	0	0	8.0500	2



**Задание 2.** Разработайте python-программу, в которой методом ближайшего соседа выполните обучение классификационной модели на обучающих данных и определите значения классов тестовых данных для разных значений параметров модели. Организуйте ввод обучающих данных из текстового файла. Фрагмент обучающих данных "Риски выдачи кредита":

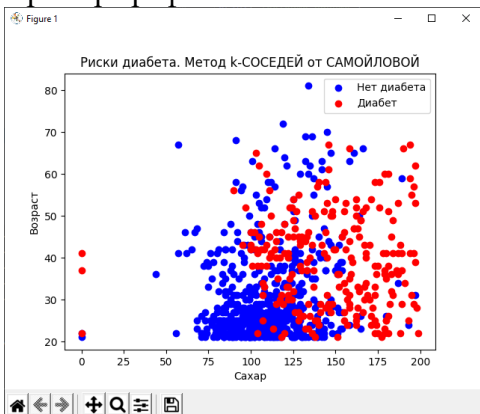
```
age,salary,house,class
18,25,1,1
22,100,1,1
30,10,0,0
32,120,0,1
24,15,1,0
25,22,1,1
32,20,0,0
19,15,1,0
52,135,0,1
.....
```

**Задание 3.** Выполните сохранение обученной классификационной модели Задания 1 прогнозирования в файл. Используя файл обученной модели, выполните прогноз, не используя этап обучения.

**Задание 4.** Примените алгоритм классификации для новых данных (выберите предметную область самостоятельно). Оцените качество прогноза (аккуратность, точности, полноту, F-меру) для этих данных. Выведите зависимость класса от параметров модели.

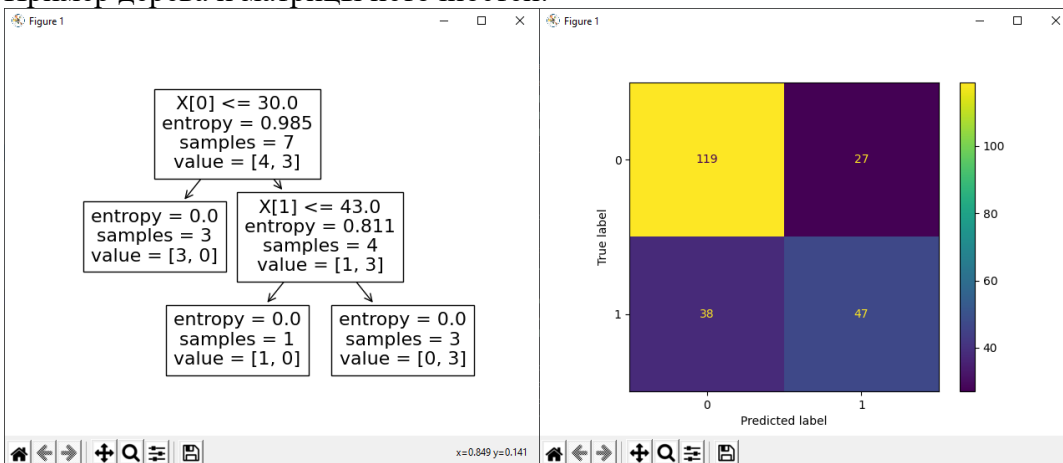
**Данные для экспериментов:** UCI Machine Learning Repository <http://archive.ics.uci.edu/ml/>

**Пример графика зависимости:**

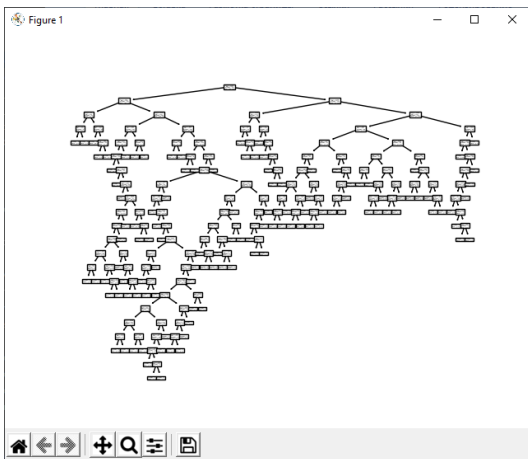


**Задание 5.** Методом деревьев решений выполните обучение модели классификации на алгоритмом ID3. Постройте дерево решений и соответствующую ему матрицу неточностей.

**Пример дерева и матрицы неточностей:**



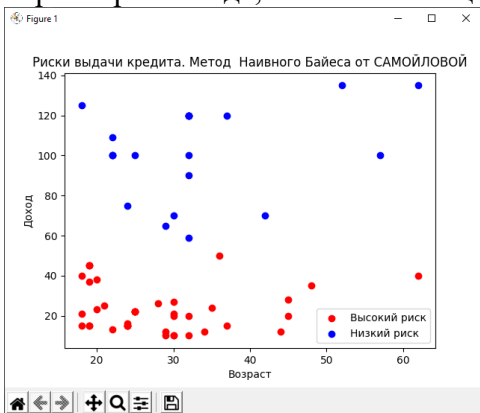
**Задание 6.** Обучите модель методом деревьев решений по вашим данным. Бинарная классификация. Постройте матрицу неточностей и дерево. Поясните содержимое матрицы. Сравните результаты и точность моделей, используя методы C4.5 и CART. Результаты:



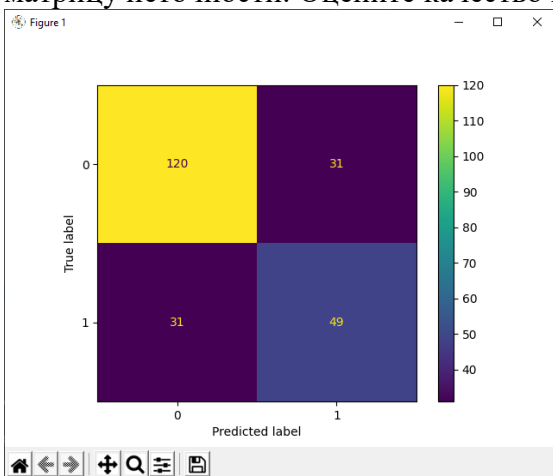


## ЛР 2 «Машинное обучение методом наивного Байеса. Оценка качества машинного обучения методом ROC – анализа».

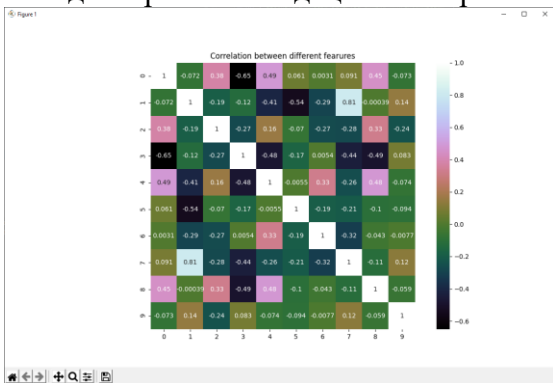
**Задание 1.** Методом наивного Байеса выполните обучение модели классификации на конкретных данных. Оцените качество модели на тестовых данных, эквивалентных обучающим. Результаты: Оценки качества точность, полнота, f-мера. Постройте график зависимости класса от двух входных параметров (например, возраст, доход). Подберите параметры метода, обеспечивающие лучшее качество. Результат:

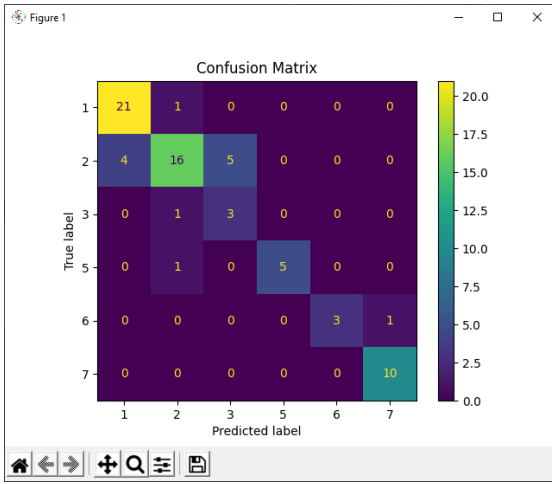


**Задание 2.** Обучите модель наивного Байеса по демонстрационным данным "Диабет". Выборка из двух классов. Тестовая выборка формируется методом кросс - валидации. Постройте матрицу неточности. Оцените качество модели. Пример матрицы неточности:

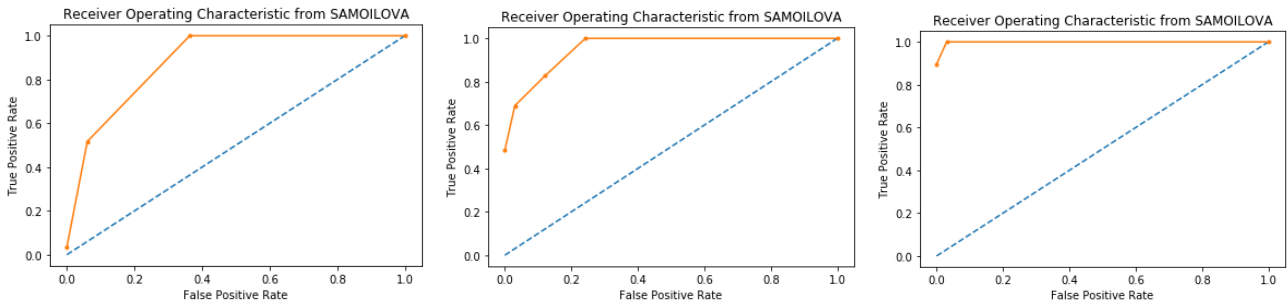


**Задание 3.** Обучите модель методом наивного Байеса по данным "Виды стекол". Выборка из семи классов. Параметры модели: 10 характеристик содержания в стекле определенного вещества. До обучения модели, постройте корреляционную матрицу параметров, которая показывает, как параметры будут влиять на результат. Тестовую выборку сформируйте методом кросс - валидации. Постройте матрицу неточности.

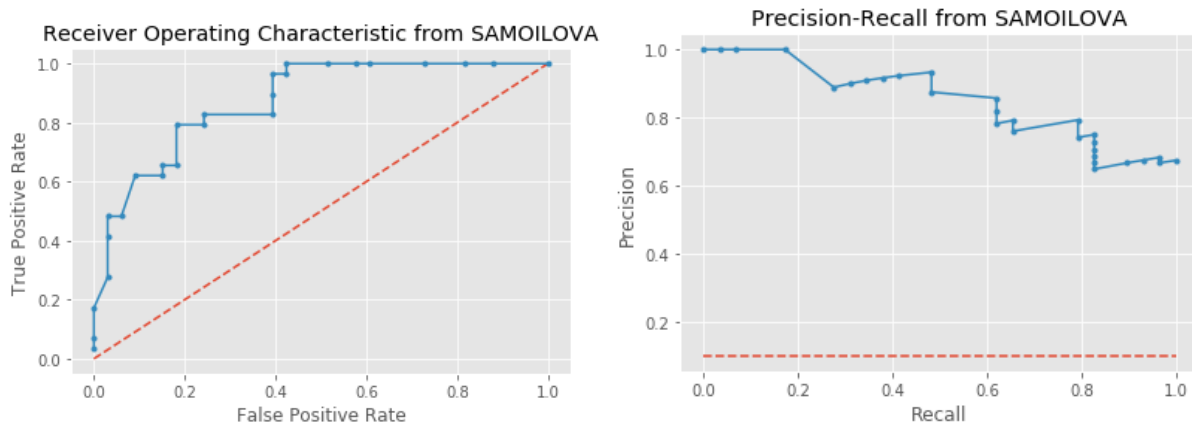


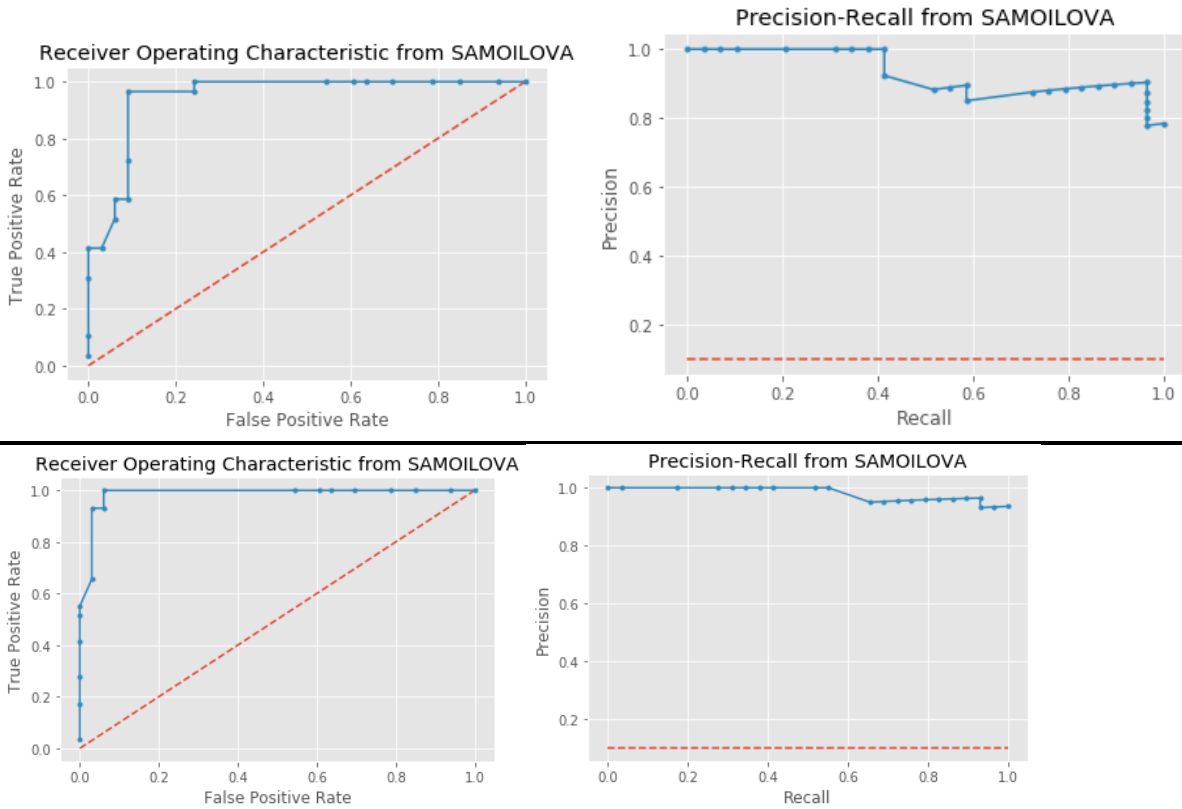


**Задание 4.** Методом деревьев решений выполните обучение модели классификации на конкретных данных. Оцените качество модели на тестовых данных, эквивалентных обучающим. Результаты: Оценки качества точность, полнота, f-мера. По данным ROC-анализа оцените зависимость AUC от значений глубины дерева. Для приведенных ROC- кривых  $AUC = 0.883, 0.991$  и  $0.998$

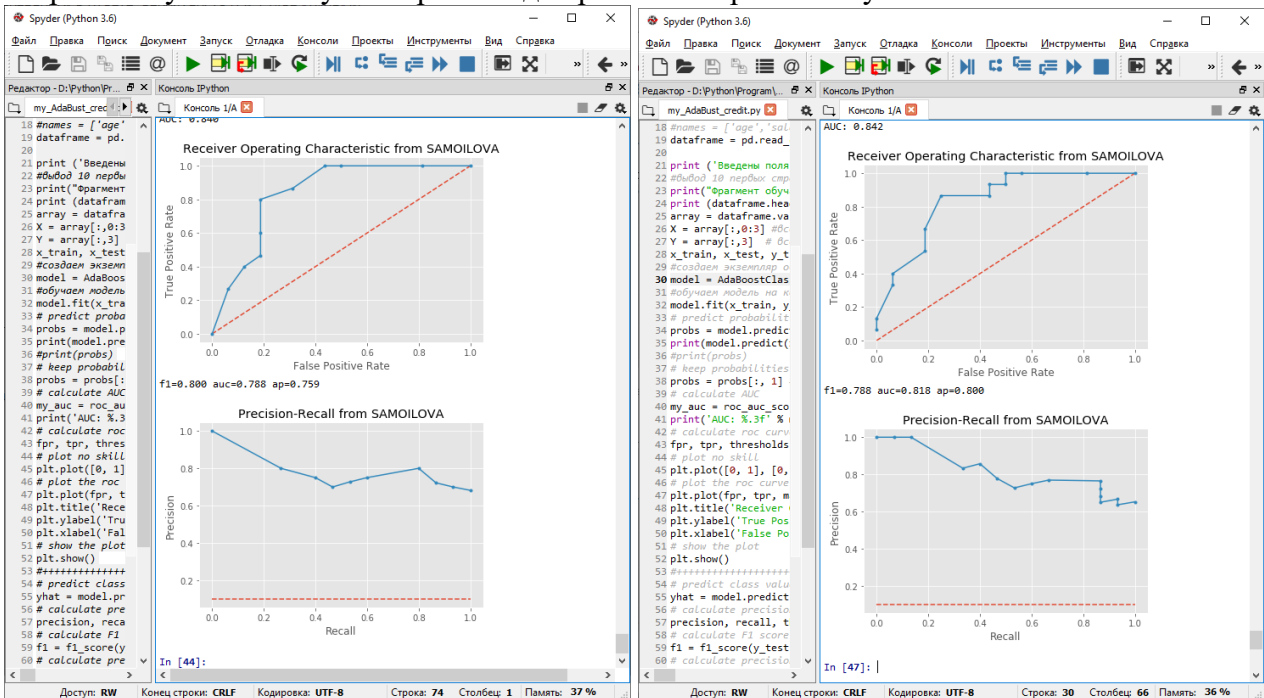


**Задание 5.** Методом случайного леса решений выполните обучение модели классификации на конкретных данных. Оцените качество модели методами ROC - анализа и PR-анализа на тестовых данных, эквивалентных обучающим. Примеры оценок качества для **разных значений глубины дерева:**





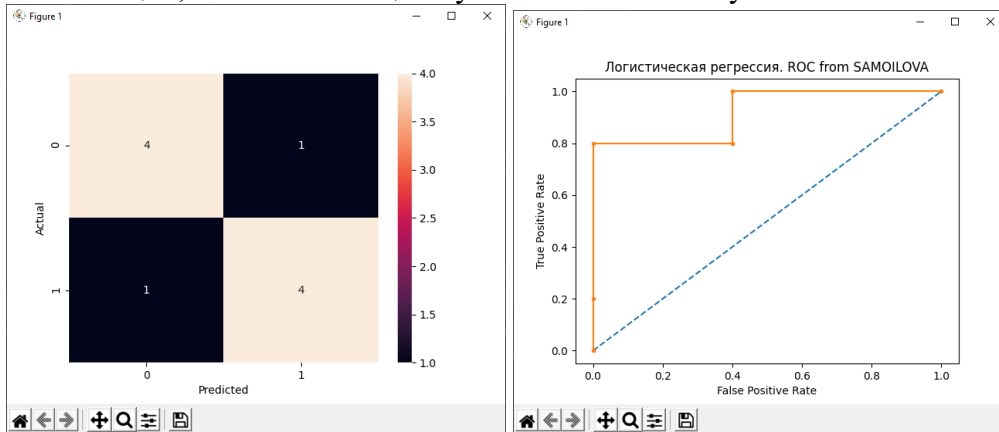
**Задание 6.** Методом адаптивного бустинга деревьев решений `AdaBoostClassifier` выполните обучение модели классификации на данных по кредитам ( в csv –файле) Оцените качество модели на тестовых данных. Результаты: Оценки качества точность, полнота, f-мера, ROC – кривая, AUC. Подберите алгоритм оптимизации, обеспечивающий лучшую скорость обучения. Результат работы для разных скоростей обучения:



### ЛР 3 «Машинное обучение методом логистической регрессии. Алгоритмы регрессии в машинном обучении».

**Задание 1.** Методом логистической регрессии выполните обучение модели классификации на данных для поступления в престижный западный университет. Класс – поступит, не поступит. Постройте матрицу неточности. Оцените качество модели на тестовых данных. Результаты:

Оценки качества точность, полнота, f-мера, ROC – кривая, AUC. Подберите алгоритм оптимизации, обеспечивающий лучшее качество. Результат:



предсказала классы для тестовой выборки: [1 0 1 1 0 0 1 1 0 1]

Вероятности предсказания классов 0 и 1 - две колонки для тестовой выборки:

```
[[0.3018346 0.6981654 ]
 [0.69176147 0.30823853]
 [0.05322031 0.94677969]
 [0.20417163 0.79582837]
 [0.58169393 0.41830607]
 [0.88906961 0.11093039]
 [0.02920068 0.97079932]
 [0.11033348 0.88966652]
 [0.67029328 0.32970672]
 [0.12638422 0.87361578]]
```

Accuracy: 0.8

	precision	recall	f1-score	support
0	0.80	0.80	0.80	5
1	0.80	0.80	0.80	5
accuracy			0.80	10
macro avg	0.80	0.80	0.80	10
weighted avg	0.80	0.80	0.80	10

AUC: 0.920

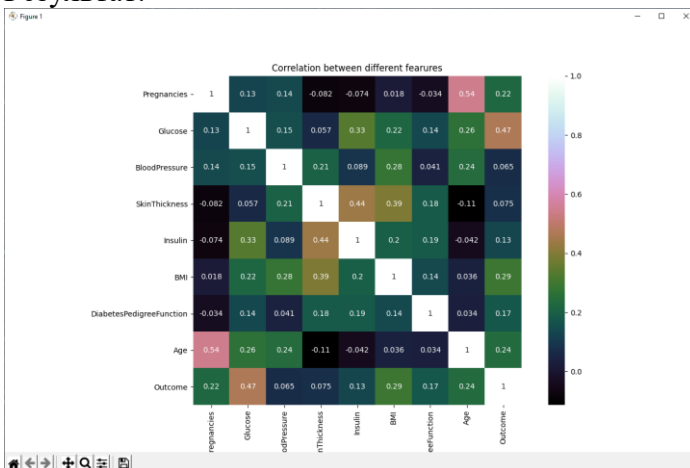
**Задание 2.** Обучите модель логистической регрессии по демонстрационным данным "Диабет".

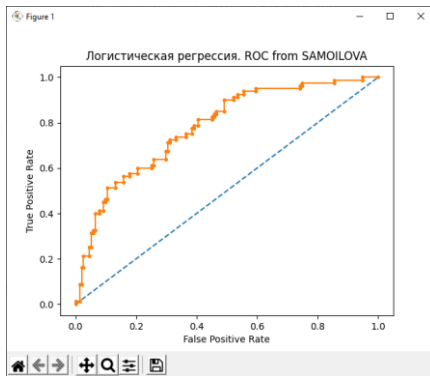
Выборка из двух классов. Тестовая выборка формируется методом кросс - валидации.

Сформируйте таблицу вероятности предсказания классов 0 и 1. Оцените качество модели.

Постройте корреляционную матрицу. Постройте ROC-кривую.

Результат:





Name: Outcome, Length: 537, dtype: int64

	Pregnancies	Glucose	Insulin	BMI	Age
334	1	95	58	23.9	22
139	5	105	325	36.9	28
485	0	135	250	42.3	24
547	4	131	166	33.1	28
18	1	103	83	43.3	33
..	...	...	...	...	...
71	5	139	140	28.6	26
106	1	96	0	22.4	27
270	10	101	0	45.6	38
435	0	141	0	42.4	29
102	0	125	0	22.5	21

[537 rows x 5 columns]

Точность предсказания составила: 0.7359307359307359

	precision	recall	f1-score	support
0	0.76	0.87	0.81	151
1	0.66	0.49	0.56	80
accuracy			0.74	231
macro avg	0.71	0.68	0.69	231
weighted avg	0.73	0.74	0.72	231

Прогноз диабета у КЛИЕНТА = [0]

**Задание 3.** Обучите модель логистической регрессии по своим данным. Выборка из двух классов. Тестовая выборка формируется методом кросс - валидации. Сформируйте таблицу вероятности предсказания классов 0 и 1. Оцените качество модели. Постройте ROC-кривую.

**Задание 4.** Разработайте программу линейной многофакторной регрессии методом LinearRegression. Выполните прогноз пробега автомобиля. Факторы: вес, год выпуска, страна-производитель. Результаты: Оценки качества для тестовой выборки + Прогноз пробега как функция от веса, года выпуска, страны-производителя.

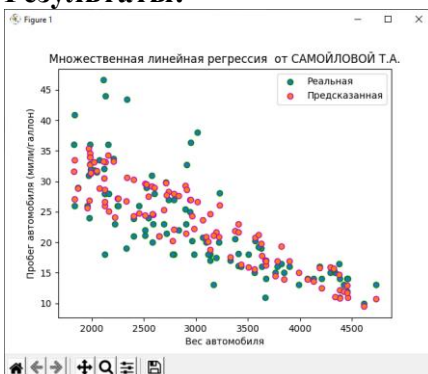
Ссылка на исходный набор данных: <https://archive.ics.uci.edu/ml/datasets/Auto+MPG>

Характеристики ('cylinders', 'displacement', 'horsepower', 'weight', 'acceleration', 'model\_year', 'origin', 'car\_name'):

1. цилиндры: многозначные дискретные
2. смещение: непрерывный
3. мощность: непрерывная
4. вес: непрерывный
5. ускорение: непрерывное
6. модельный год: многозначный дискретный
7. Происхождение: многозначный дискретный
8. имя машины: строка (уникальная для каждого экземпляра)

Количество образцов: 392. Целевая переменная (непрерывная): mpg - пробег в милях на галлон. Оценки точности выполните методом кросс – валидации. Рассчитайте коэффициент детерминации для тестовых данных, коэффициент детерминации для обучающих данных.

### Результаты:



G:\PyCharm\_tatsamoilova\sklearn\_simple\REGR\My\_Auto\_Lin\_Regr.py

RMSE= 4.069215579662049

коэффициент детерминации(тестовые)  $r^2 = 0.7507898901990242$

коэффициент детерминации(обучающие)  $r^2 = 0.8022913548564652$

Предскажем!!!Пробег автомобиля для Самойловой = [33.1146191]

Process finished with exit code 0

**Задание 5.** Разработать программу линейной регрессии, используя модель **LinearRegression**, для прогноза цены квартиры. Сохранить модель на диске. Вычислить цену своей (виртуальной) квартиры.

Вычислить цену своей квартиры, используя только сохраненную модель.

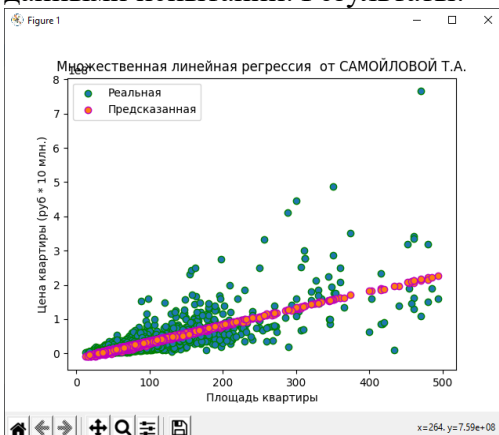
**Выборка исследования:** архив объявлений о продаже квартир в Москве за несколько лет - данные сервиса Яндекс.Недвижимость (31191 записей). Данные в csv-файле содержат: материал стен, этажность, этажность общая, общая площадь, площадь кухни, широта, долгота, цена. Соответствующие имена полей для DataFrame: wallsMaterial, floorNumber, floorsTotal, totalArea, kitchenArea, latitude, longitude, price

**Факторы регрессии:** этажность, этажность общая, общая площадь, площадь кухни. Соответствующие имена полей для DataFrame: floorNumber, floorsTotal, totalArea, kitchenArea.

Целевая переменная (непрерывная): price – цена квартиры.

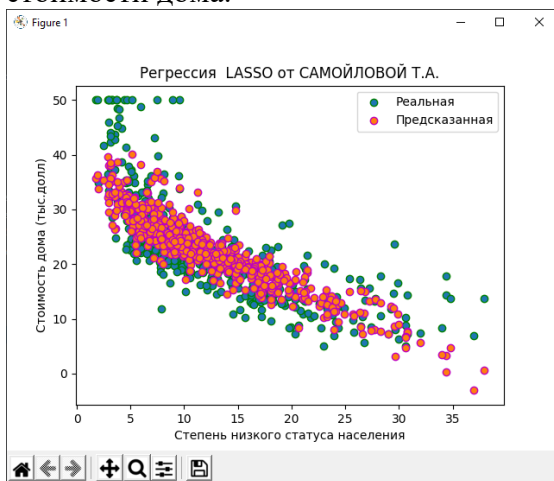
В программе используется классический набор данных **moscow dataset 2020.csv** и она строит модель для прогнозирования цены на квартиру. Для этого предоставим модели описание многих квартир периодов 2018-2019.

Модель регрессии пытается предсказать цену на квартиру на основе четырех предикторов - этажность, этажность общая, общая площадь, площадь кухни. Мы используем простой метод перекрестной проверки с удержанием: 75% данных являются данными обучения, а 25% - данными испытаний. Результаты:

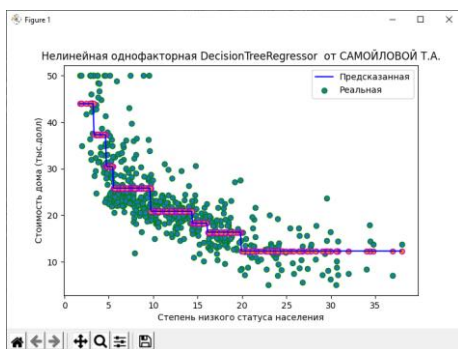


```
wallsMaterial floorNumber floorsTotal ... latitude longitude price
0 brick 1 5.0 ... 55.723379 37.628577 5600000
1 brick 1 5.0 ... 55.725980 37.671031 4650000
2 brick 1 5.0 ... 55.735976 37.657817 2990000
3 brick 1 7.0 ... 55.786698 37.595321 4390000
4 brick 2 5.0 ... 55.767894 37.665920 4890000
[5 rows x 8 columns]
RMSE= 16059592.911492676
```

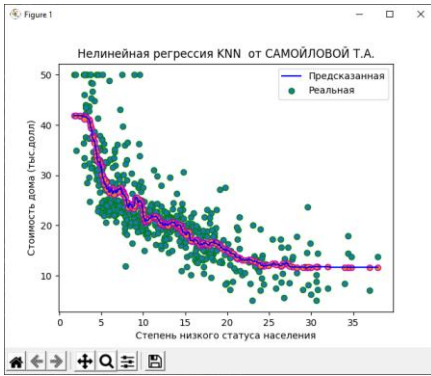
**Задание 6.** Разработайте python-программу многофакторной линейной регрессии, используя модель **LassoCV**, созданную библиотекой **sklearn**. Программа должна выполнять прогноз стоимости дома.



**Задание 7.** Разработайте программу однофакторной нелинейной регрессии, используя модель **DecisionTreeRegressor**, созданную библиотекой **sklearn**. Программа должна выполнять прогноз стоимости дома (задания 1,2). Входные и выходные факторы см задания 1, 2. Единственный входной фактор – степень низкого статуса населения ( или среднее число комнат в доме на 1 квартиру). Результат:



**Задание 8.** Разработайте программу однофакторной нелинейной регрессии, используя модель **KNeighborsRegressor**, созданную библиотекой **sklearn**. Программа должна выполнять прогноз стоимости дома (задания 1,2). Входные и выходные факторы см задания 1, 2. Единственный входной фактор – степень низкого статуса населения ( или среднее число комнат в доме на 1 квартиру). Результат:

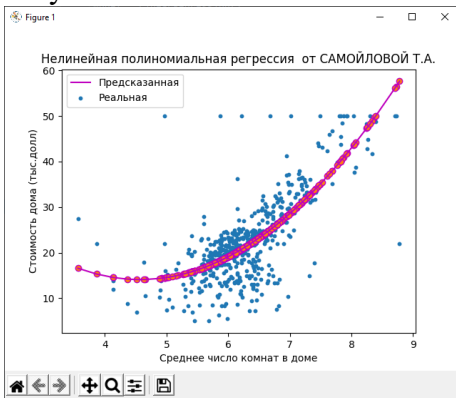


rmse= 5.089057644349284

Точность модели - r2 0.6932172724449991

**Задание 9.** Разработайте программу однофакторной нелинейной регрессии, используя модель **LinearRegression**, предварительно обученную полиномиальным функциям. Программа должна выполнять прогноз стоимости дома. Входные и выходные факторы см задания 1, 2. Единственный входной фактор – среднее число комнат в доме на 1 квартиру

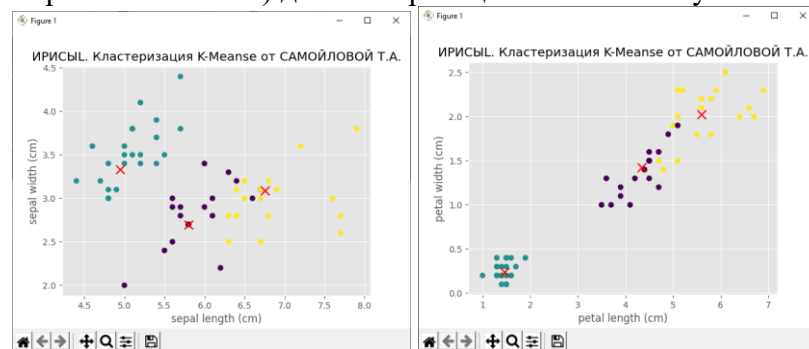
Результат:



**Задание 10.** Разработайте программы регрессии методом опорных векторов с использованием линейных, полиномиальных и RBF-ядер для индивидуальных данных (UCI Machine Learning Repository - <http://archive.ics.uci.edu/ml/>). Вычислите оценки качества для тестовой и обучающей выборки: среднеквадратичную ошибку регрессии - mean\_squared\_error (MSE), коэффициент детерминации r2\_score. Сравните результаты для разных типов ядер. Постройте графики зависимости целевой функции от признаков выборки.

**ЛР 4 «Алгоритмы кластеризации в машинном обучении».**

**Задание 1.** Методом **K-Means** выполните обучение без учителя модели кластеризации (число кластеров = 3) на конкретных данных (ИРИСЫ). Оцените точность модели методом кросс-валидации. Постройте графики зависимости двух входных параметров (например, длины и ширины лепестков) до кластеризации и после. Результаты:



КЛАСТЕРИЗАЦИЯ ОТ САМОЙЛОВОЙ - ТРИ КЛАССА

напрогнозировала: [0 1 2 0 2 1 0 2 0 0 2 1 1 1 1 0 2 0 0 2 1 0 1 2 2 2 2 2 1 1 1 1 0 1 1 2 0



```

1 1 1 0 0 2 1 1 0 2 2 0 2 0 2 0 1 2 0 1 1 1 0]
а должно бы быть: [0 1 2 0 0 1 0 2 0 0 2 1 1 1 1 0 2 0 0 2 1 2 1 2 2 2 2 2 1 1 1 1 0 1 1 2 0
1 1 1 2 0 0 1 1 0 2 2 0 2 0 2 0 1 2 0 1 1 1 0]

```

размер выборки: 60  
из них корректных: 56

Кросс-валидация. Результаты оценки качества модели кластеризации:  
Общая точность кластеризации - accuracy\_score = 0.9333333333333333

По кластерам: Точность - Полнота - F-мера - Кол-во

	precision	recall	f1-score	support
0	0.89	0.89	0.89	19
1	1.00	1.00	1.00	23
2	0.89	0.89	0.89	18
accuracy			0.93	60
macro avg	0.93	0.93	0.93	60
weighted avg	0.93	0.93	0.93	60

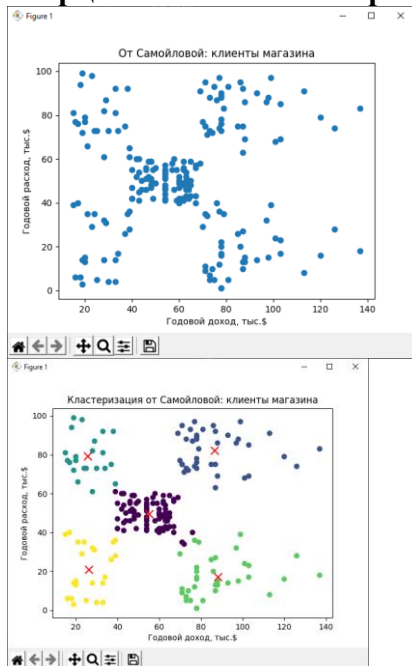
**Задание 2.** Методом K-Means выполните обучение без учителя модели кластеризации на данных о клиентах торгового центра [параметры: CustomerID, Genre, Age, Annual Income (k\$), Spending Score (1-100)]. Фрагмент данных для кластеризации:

```

CustomerID,Genre,Age,Annual Income (k$),Spending Score (1-100)
0001,Male,19,15,39
0002,Male,21,15,81
0003,Female,20,16,6
0004,Female,23,16,77

```

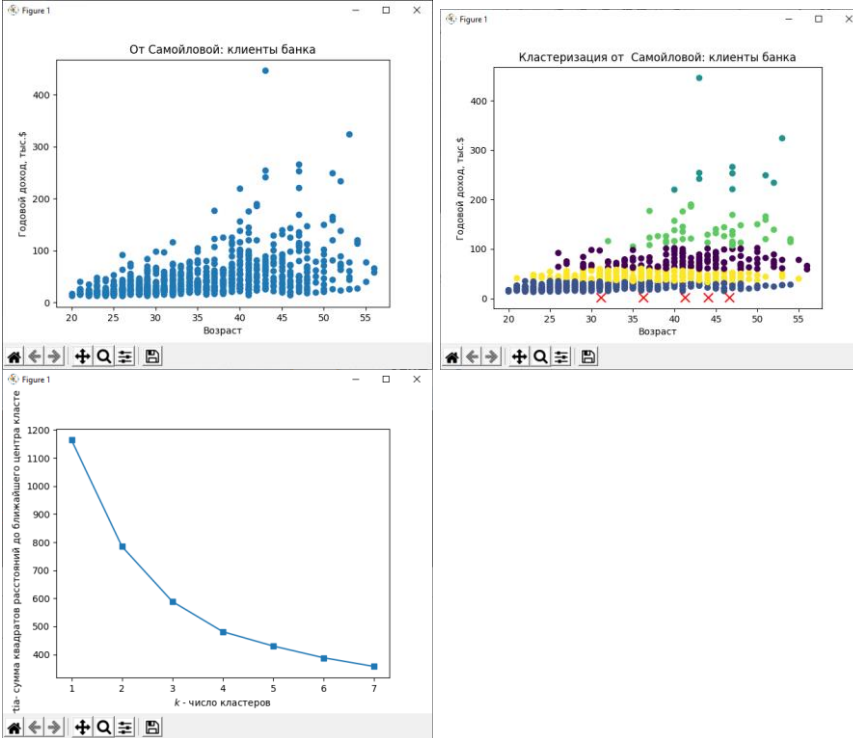
Постройте графики зависимости двух входных параметров (например, доход - расход) до кластеризации и после. Оцените качество модели путем вычисления инерции - суммы квадратов расстояний до ближайшего центра кластера. Постройте график зависимости инерции от числа кластеров. Результаты:



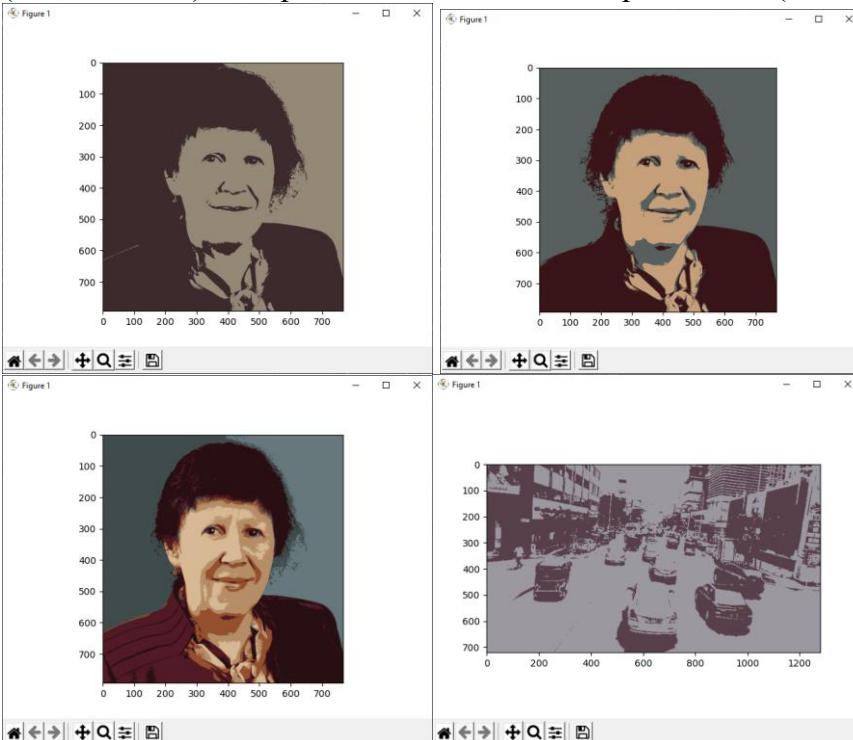


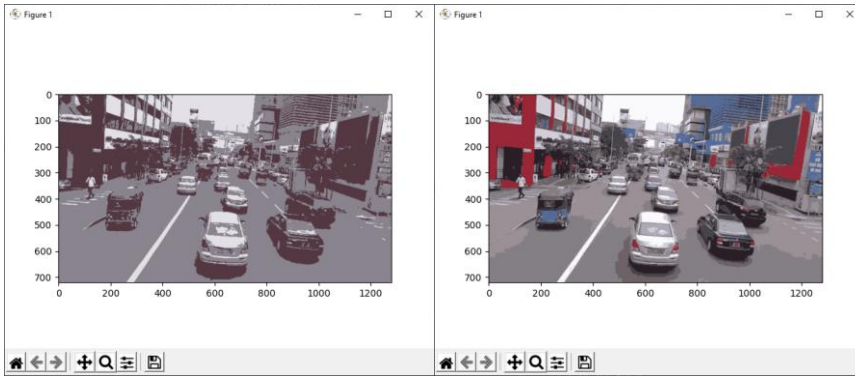
1,2,47,1,26,100,4.582,8.218,0.0,12.8  
 2,3,33,2,10,57,6.1110000000000001,5.8020000000000005,1.0,20.9  
 3,4,29,2,4,19,0.6809999999999999,0.516,0.0,6.3  
 4,5,47,1,31,253,9.308,8.908,0.0,7.2

Постройте графики зависимости двух входных параметров (например, доход - расход) до кластеризации и после. Оцените качество модели путем вычисления инерции - суммы квадратов расстояний до ближайшего центра кластера. Постройте график зависимости инерции от числа кластеров. Результаты:

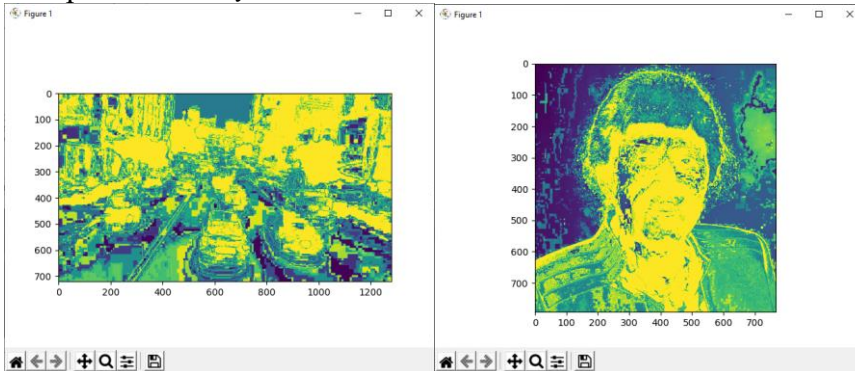


**Задание 5.** Методом K-Meanse пакета компьютерного зрения CV2 выполните кластеризацию (сегментацию) изображения с числом кластеров = 2,3,7 (подберите сами). Результаты:





**Задание 6.** Методом DBSCAN пакета `sklearn.cluster` выполните кластеризацию (сегментацию) изображения. Результат:



**Задание 7.** Методом K-Means выполните обучение без учителя модели кластеризации на ваших данных. Постройте графики зависимости двух входных параметров (например, доход - расход) до кластеризации и после. Оцените качество модели путем вычисления инерции - суммы квадратов расстояний до ближайшего центра кластера. Постройте график зависимости инерции от числа кластеров.

**Задание 8.** Методом агломеративной кластеризации постройте дендограмму на ваших конкретных данных. В качестве меры расстояния между кластерами возьмите метод Уорда.

### ЛР 5 «Нейронные сети в машинном обучении. Генетические алгоритмы».

**Задание 1.** Разработать python-программу классификации  $(0,1)$ , используя ИНС, созданную библиотекой `keras` из `TensorFlow`.

Ссылка на исходный набор данных: <https://archive.ics.uci.edu/ml/datasets/diabetas>

Данные:

1. Количество беременностей
2. Концентрация глюкозы в плазме через 2 часа в оральном тесте на толерантность к глюкозе
3. Диастолическое артериальное давление (мм рт.ст.)
4. Толщина кожной складки трицепса (мм)
5. 2-часовой сывороточный инсулин (м.е. / мл)
6. Индекс массы тела (вес в кг / (рост в м) <sup>2</sup>)
7. Диабет родословной
8. Возраст (лет)
9. Переменная класса (0 или 1)

Функция активации `relu` (прямая) на первых двух слоях и сигмовидная функция в выходном слое. Первый слой имеет 12 нейронов и ожидает 8 входных переменных. Второй скрытый слой (сеть глубокая, только у них есть скрытые слои) имеет 8 нейронов. Выходной слой имеет 1 нейрон, чтобы предсказать класс (диабет или нет). Тренировка модели: обучается модель на 150 эпохах.

**Задание 2.** Разработать python-программу регрессии, используя ИНС, созданную библиотекой `keras` из `TensorFlow`. Выполните прогноз пробега автомобиля. Факторы: вес, год выпуска,

страна-производитель. Результаты: Оценки качества для тестовой выборки + Прогноз пробега как функция от веса, года выпуска, страны-производителя.

Ссылка на исходный набор данных: <https://archive.ics.uci.edu/ml/datasets/Auto+MPG>

Характеристики ('cylinders', 'displacement', 'horsepower', 'weight', 'acceleration', 'model\_year', 'origin', 'car\_name'):

1. цилиндры: многозначные дискретные
2. смещение: непрерывный
3. мощность: непрерывная
4. вес: непрерывный
5. ускорение: непрерывное
6. модельный год: многозначный дискретный
7. Происхождение: многозначный дискретный
8. имя машины: строка (уникальная для каждого экземпляра)

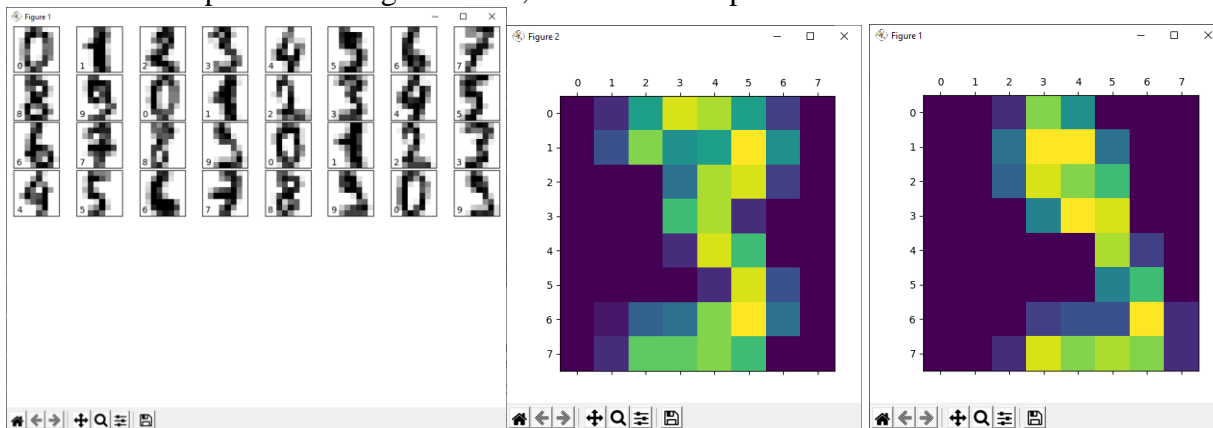
Количество образцов: 392. Целевая переменная (непрерывная): mpg - пробег в милях на галлон.

**Задание 3.** Разработайте модель ИНС алгоритмом MLPClassifier для реализации многоклассовой классификации. Исходные обучающие данные - изображения цифр библиотеки MNIST. Выполнить распознавание 1-2 цифр по обученной модели

Результат обучения и распознавания:

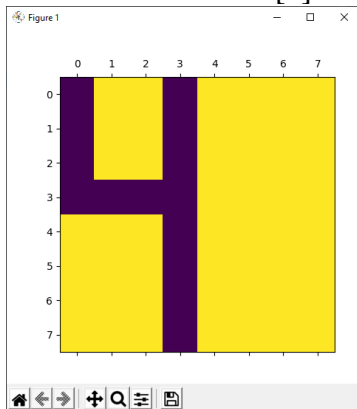
We have 1797 samples

Number of samples in training set: 1437, number of samples in test set: 360



Распознала!! Это - [3]

Распознала!! Это - [9]



Распознала рисунок Самойловой!! Это - [4]

**Задание 4.** Разработайте модель ИНС алгоритмом MLPClassifier для реализации регрессии.

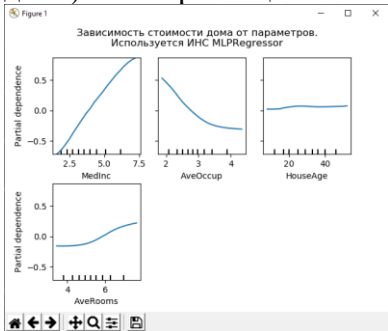
Обучающие данные – файл цен на жилье в Калифорнии и значение влияющих на цену параметров. [https://scikit-learn.org/stable/auto\\_examples/inspection/plot\\_partial\\_dependence.html#california-housing-data-preprocessing](https://scikit-learn.org/stable/auto_examples/inspection/plot_partial_dependence.html#california-housing-data-preprocessing)

Файл содержит все переменные в масштабируемой форме. В частности, он содержит медианную стоимость дома, средний возраст жилья, общее количество комнат, общее количество спален, население, домохозяйства, широту и долготу в указанном порядке. Поля: MedInc, HouseAge, AveRooms, AveBedrms, Population, AveOccup, Latitude Longitude.

Target - переменные содержат цены на дома в ln форме (медианной). Мы будем использовать функцию  $e^x$ , псевдоним, exp чтобы инвертировать ln и получить цену дома. Вычисление цены дома по медианной форме:

Примерные результаты:

Графики частичной зависимости показывают зависимость между целевой функцией (стоимость дома) и набором «целевых» признаков:



Целевыми переменными для ИНС являются: средний доход ( MedInc), средний процент жителей на домохозяйство ( AveOccup), средний возраст дома ( HouseAge) и средний номер на домохозяйство ( AveRooms).

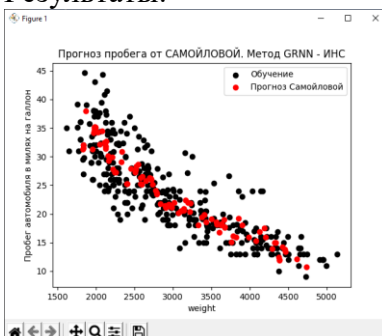
**Анализ графиков.** Мы можем ясно видеть, что медианная цена дома показывает линейную зависимость со средним доходом (вверху слева) и что цена дома падает, когда увеличивается среднее количество жителей на домохозяйство (вверху посередине). Верхний правый график показывает, что возраст дома в районе не оказывает сильного влияния на (среднюю) цену дома; так же, как и в среднем на домохозяйство.

Размер нейронной сети и скорость обучения настроены так, чтобы получить разумный компромисс между временем обучения и прогнозирующей эффективностью в тестовом наборе.

**Задание 5.** Разработайте модель ИНС алгоритмом GRNN.(Neural Networks with Radial Basis Functions) для реализации регрессии. Обучающие данные – файл пробега автомобилей.

Результаты: Оценки качества прогнозирования пробега автомобиля - MSE. Зависимость пробега от параметра – веса автомобиля.

Результаты:



Среднеквадратичная ошибка: 113.87132157993446

Предсказываю пробег для тестовой выборки:

[[11.86674329]

[15.98118317]

.....

[21.34850449]

[12.45185296]]

ОК!!

Предскажем!!!Пробег автомобиля для Самойловой = [[32.61753309]]

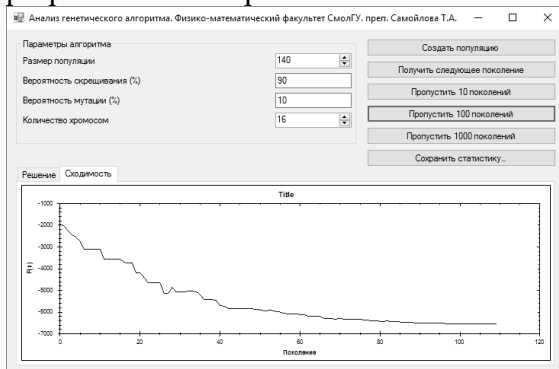
**Задание 6:** Выполните разработку модели ИНС для данных другой предметной области, например, прогноз цены недвижимости.



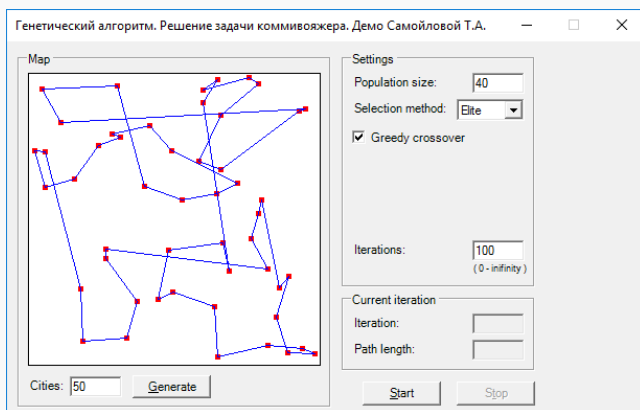
**Задание 6.** Установите библиотеку AForge .NET Используя средства AForge разработать **WindowsForm** - приложение для вычисления максимального значения функции средствами ГА. Построить график функции, используя объект Chart. Вариант функции взять у преподавателя.

**Задание 7.** Установите библиотеку Accord .NET Используя средства Accord.Genetic разработать консольное приложение для вычисления максимального значения функции от двух переменных средствами ГА. Построить график функции, используя средства Excel. Вариант функции взять у преподавателя.

**Задание 8.** Используя библиотеку AForge, в среде Visual Studio.NET разработать программу анализа возможностей генетического алгоритма (создать популяцию, получить поколение). Размер популяции, количество хромосом и т.п. - входные данные. Примерный вид формы разработанного приложения:



**Задание 9.** Используя библиотеку AForge, в среде Visual Studio.NET разработать программу решения задачи коммивояжера средствами генетического алгоритма. Примерный вид формы разработанного приложения:



## ЛР 6 «Нечеткие методы машинного обучения».

**Задание 1.** Используя пакеи scikit-fuzzy выполните нечеткую кластеризацию данных алгоритмом Fuzzy C-Means (данные – демонстрационный пример и ваша выборка). Выполните вычисление коэффициента нечеткого распределения Fuzzy partition coefficient (FPC). Выполните 5 этапов обработки данных:

- Этап 1: Генерация тестовых данных для кластеризации (каждый элемент имеет два random - параметра: x,y)
- Этап 2: Кластеризация тестовых данных по моделям с разным числом центров (2-9) и вычисление коэффициентов нечеткого распределения FPC
- Этап 3: Выбор модели по построенной зависимости коэфф. нечеткого распределения от числа центров кластеризации (Выбираем модель с максимальным значением FPC)
- Этап 4: Кластеризация тестовых данных по выбранной модели.

- Этап 5: Кластеризация новых данных (из диапазона тестовых) по выбранной модели. Расчет FPC для новых данных.

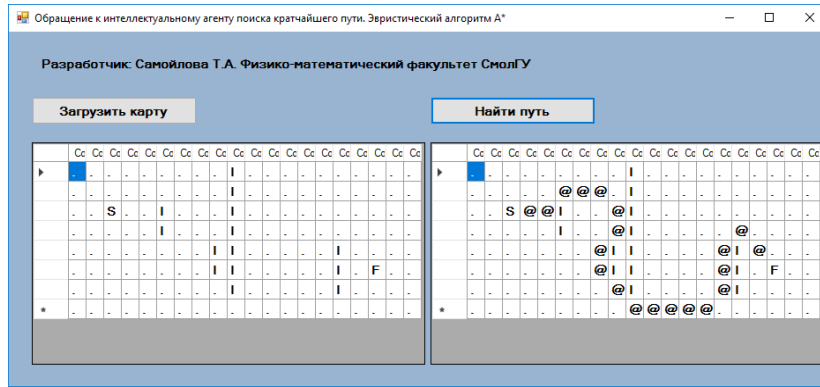
**Задание 2.** Методом **K-Means** пакета алгоритмов нечеткой логики **scikit-fuzzy** выполните кластеризацию (сегментацию) изображения с числом кластеров = 2,3,7 (подберите сами).  
**Результаты:**



### ЛР 7 «Разработка интеллектуального агента».

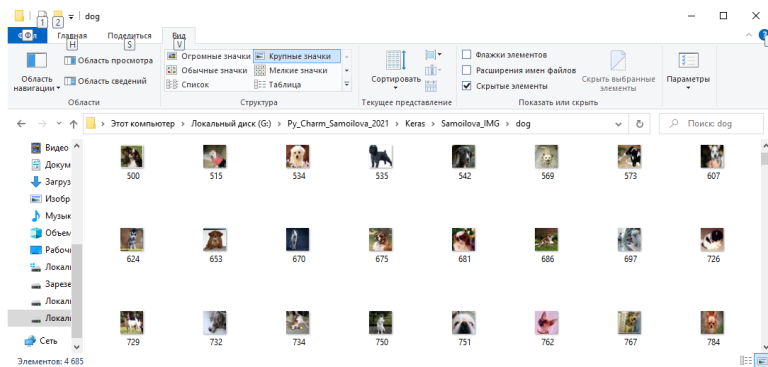
**Задание.** Используя **#**-библиотеки машинного обучения, в среде Visual Studio.NET разработать **WEBAPI**-сервис, реализующий модель интеллектуального агента поиска кратчайшего пути эвристическим алгоритмом **A\***. Входные данные сервиса - карта местности, выходные данные - матрица пути. Обмен данными с сервисом выполнить в формате **JSON**. Обратиться к агенту из **Windows Form** - приложения. Примерный вид формы разработанного приложения:



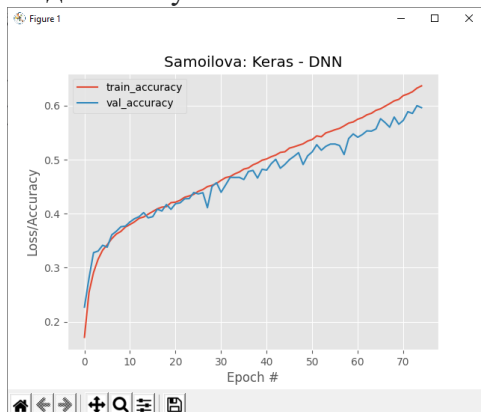


## ЛР 8 «Компьютерное зрение».

**Задание 1.** Разработать программу для обучения нейронной сети распознаванию изображений 10 классов. Сеть имеет 4 слоя – входной, два скрытых и выходной. Сохраните обученную модель на диск. Изображения для классов скачать из интернет. В каждом классе не менее 100-1000 изображений (например, легковые автомобили). Один из классов можете заполнить своими изображениями (например, ваша собака в разных видах). Пример фрагмента набора класса «собаки»:



Сеть должна иметь 4 слоя – входной, два скрытых и выходной. Сохраните обученную модель на диск. Результат:



```
[Loading images...
Загрузила рисунки
Epoch 1/75
1172/1172 [=====] - 17s 14ms/step - loss: 2.2387 - accuracy: 0.1713 -
val loss: 2.1468 - val accuracy: 0.2271
Epoch 2/75
1172/1172 [=====] - 18s 15ms/step - loss: 2.0790 - accuracy: 0.2540 -
val loss: 2.0104 - val accuracy: 0.2804
Epoch 3/75
1172/1172 [=====] - 17s 14ms/step - loss: 1.9818 - accuracy: 0.2912 -
val loss: 1.9385 - val accuracy: 0.3279
Epoch 4/75
1172/1172 [=====] - 17s 14ms/step - loss: 1.9275 - accuracy: 0.3157 -
val loss: 1.9023 - val accuracy: 0.3309
Epoch 5/75
```

```

Epoch 72/75
1172/1172 [=====] - 17s 14ms/step - loss: 1.0821 - accuracy: 0.6217 -
val_loss: 1.1715 - val_accuracy: 0.5888
Epoch 73/75
1172/1172 [=====] - 17s 14ms/step - loss: 1.0719 - accuracy: 0.6258 -
val_loss: 1.1702 - val_accuracy: 0.5856
Epoch 74/75
1172/1172 [=====] - 17s 14ms/step - loss: 1.0566 - accuracy: 0.6325 -
val_loss: 1.1427 - val_accuracy: 0.6000
Epoch 75/75
1172/1172 [=====] - 17s 14ms/step - loss: 1.0427 - accuracy: 0.6367 -
val_loss: 1.1614 - val_accuracy: 0.5962
[Evaluating network...]

```

	precision	recall	f1-score	support
Samoilova_IMG/airplane	0.69	0.57	0.62	1250
Samoilova_IMG/bird	0.44	0.71	0.54	1317
Samoilova_IMG/car	0.83	0.57	0.68	1274
Samoilova_IMG/cat	0.56	0.39	0.46	1240
Samoilova_IMG/deer	0.51	0.52	0.51	1254
Samoilova_IMG/dog	0.59	0.45	0.51	1146
Samoilova_IMG/frog	0.61	0.65	0.63	1277
Samoilova_IMG/horse	0.79	0.57	0.66	1235
Samoilova_IMG/ship	0.56	0.82	0.66	1281
Samoilova_IMG/truck	0.65	0.69	0.67	1226
accuracy			0.60	12500
macro avg	0.62	0.59	0.59	12500
weighted avg	0.62	0.60	0.60	12500

```

Serializing network and label binarizer...
Process finished with exit code 0

```

**Задание 2.** Разработать программу, которая по обученной и сохраненной модели (задание 5) распознает изображения. Распознайте ваше изображение размера 32\*32. Пример распознавания:

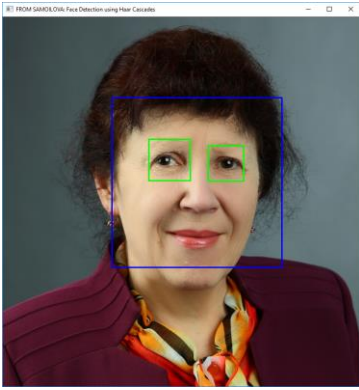


```

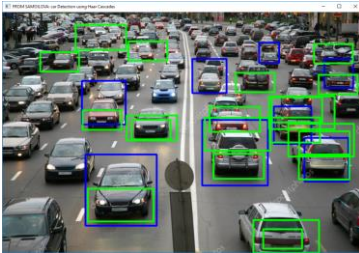
[INFO] loading network and label binarizer...
Samoilova_IMG/dog

```

**Задание 3.** Используя Python-библиотеки распознавания образов (OpenCV), разработать модель распознавания элементов изображений алгоритмом Виолы-Джонса. XML - файл диаграммы Хаара должен соответствовать вашему варианту распознаваемого элемента (автомобиль, пешеход, номер автомобиля, лицо, нос, уши, голова, глаза, торс, полный рост и т.п.). Примеры распознавания:  
Лицо-глаза:



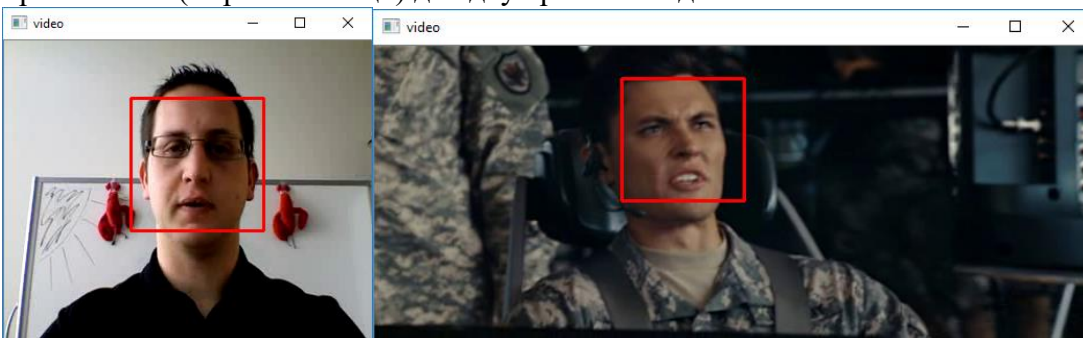
Автомобили:



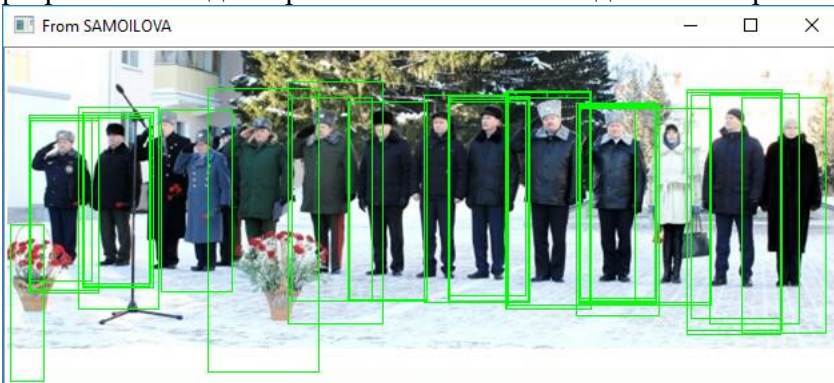
Номера автомобилей:



**Задание 4.** Используя Python-библиотеки, разработать модель распознавания кадров видеопотока алгоритмом Виолы-Джонса. XML - файл диаграммы Хаара должен соответствовать вашему варианту распознаваемого элемента (нос, уши, голова, глаза, торс, полный рост и т.п.). Далее представлен примерный вид кадров распознавания разработанного приложения (Вариант - лицо) для двух разных видео.



**Задание 5.** Используя python-библиотеки opencv распознавания образов и алгоритм HOG, разработать модель распознавания пешеходов на изображении. Пример распознавания:

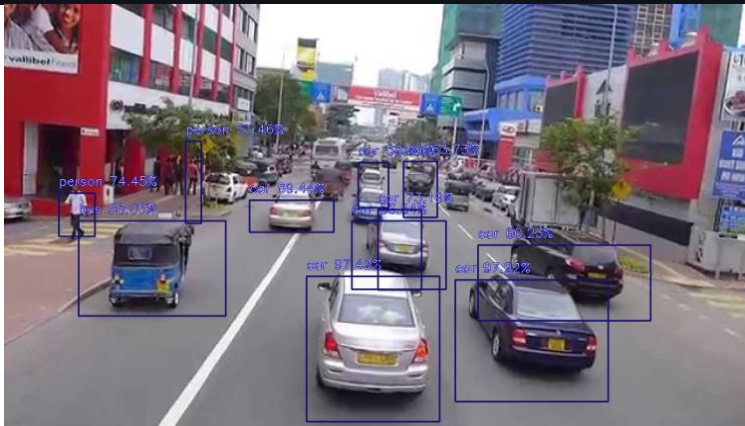


**Задание 6.** Используя Python-библиотеку ImageAI, разработать программу обращения к обученной модели обнаружения элементов изображения алгоритмом глубокого обучения YOLO - yolo-tiny.h5. Результат 1:

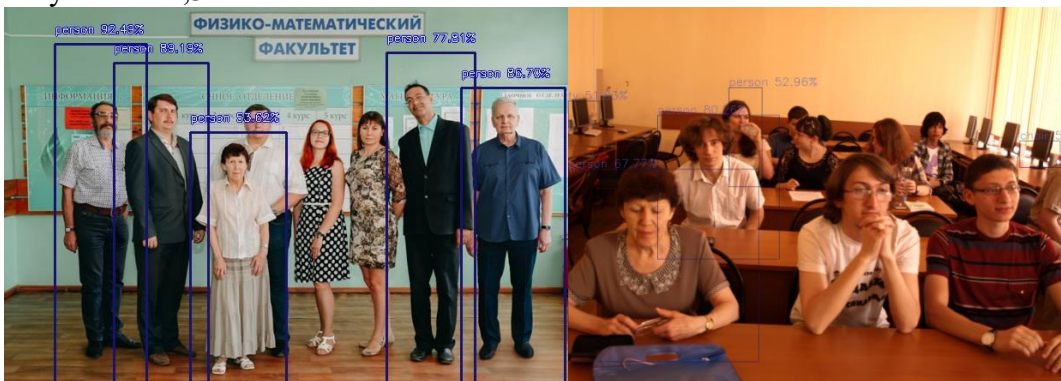
```

car : 56.54352307319641
bus : 53.0504584312439
car : 60.22621989250183
car : 97.22371101379395
car : 97.42370843887329
person : 51.46095156669617
car : 58.49268436431885
car : 63.7304425239563
person : 74.44759607315063
car : 89.44361805915833
car : 73.18670153617859

```



Результаты 2,3:



### Самостоятельная работа

Текущая самостоятельная работа направлена на углубление и закрепление знаний студентов и развитие их практических умений. Она заключается в работе с лекционными материалами, поиске и обзоре литературы и электронных источников, информации по заданным темам курса, опережающей самостоятельной работе, в изучении тем, вынесенных на самостоятельную проработку, подготовке к лабораторным занятиям.

Самостоятельная внеаудиторная работа студентов состоит в:

- проработке лекционного материала, составлении конспекта лекций по темам, вынесенным на самостоятельное изучение;
- выполнении домашних заданий.

### Темы для самостоятельного изучения

1. История развития СИИ.
2. Платформы разработки СИИ.
3. Методики экспериментального исследования СИИ.
4. Big Data в СИИ.
5. Применение алгоритмов ИИ в системах видеоанализа.

Консультирование студентов осуществляется в индивидуальном порядке на занятиях и во внеурочное время. Выполнение самостоятельной работы оценивается по электронным материалам, подготовленным студентами. Результаты деятельности накапливаются в индивидуальных портфолио студентов.

## 6. Критерии оценивания результатов освоения дисциплины (модуля)

### 6.1. Оценочные средства и критерии оценивания для текущей аттестации

#### Текущая контрольная работа

1. Используя эвристический алгоритм, решите задачу «Тур коня на шахматной доске».
2. Опишите алгоритм классификации методом ближайшего соседа. Проиллюстрируйте метод на простом примере.

#### Критерии оценивания текущей контрольной работы

##### 1. Нормы оценивания работы

№ п/п	Структурная часть контрольной работы	Количество баллов (*)
1	Правильно реализован метод решения	3 балла
2	Правильно описан алгоритм, приведен пример.	2 балла

(\*) Возможна градация в 0,25 балла.

##### 2. Шкала оценивания работы:

п/п	Оценка	Количество баллов
1	Отлично	4,75-5
2	Хорошо	3,75-4,5
3	Удовлетворительно	3-3,5
4	Неудовлетворительно	менее 3

#### Задания для лабораторных занятий

Задачи по темам курса предложены к каждому лабораторному занятию.

#### Критерии оценивания выполнения лабораторных работ

##### 1. Нормы оценивания каждой лабораторной работы:

№п/п	Структурная часть работы	Количество баллов (*)
1	Ответ на теоретические вопросы по теме лабораторной работы	1 балл
2	Демонстрация выполнения конкретного задания, предложенного для самостоятельного решения к лабораторной работе	2 балла

(\*) с возможностью градации до 0,25 балла.

2. Шкала оценивания. Оценка «зачтено» за лабораторную работу выставляется, если набрано не менее 2 баллов, в противном случае за работу выставляется «не зачтено».

### 6.2. Оценочные средства и критерии оценивания для промежуточной аттестации

#### Вопросы к экзамену

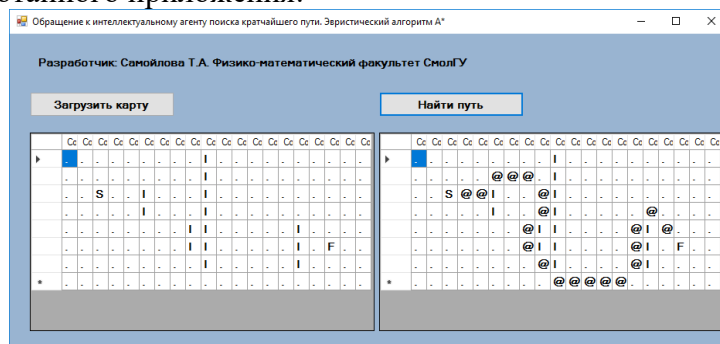
1. История искусственного интеллекта. Сферы применения. Большие данные в СИИ.
2. Машинное обучение (МО). Постановка задачи обучения по прецедентам.
3. Методы классификации данных.
4. Метод ближайших соседей (kNN) и его обобщения. Подбор числа k по критерию скользящего контроля.



5. Кривая ошибок (ROC curve). Пример оценки качества бинарной классификации методом ROC-анализа.
6. Наивный байесовский классификатор.
7. Метод опорных векторов. Особенности применения в задаче классификации.
8. Логистическая регрессия. Риск кредитного портфеля банка.
9. Задача регрессии. Регрессия методом опорных векторов.
10. Методы кластеризации. Алгоритм k средних (k-means). Методы кластеризации с нечетким разбиением исходных данных.
11. Нейросетевая парадигма в искусственном интеллекте. Методы обучения нейронных сетей.
12. Решение задач классификации с помощью нейронных сетей
13. Эволюционные и генетические алгоритмы. Генетический алгоритм в решении задачи коммивояжера.
14. Решение проблем ИИ методами поиска. Пример реализации эвристического алгоритма A\*.
15. Методы представления и обработки нечеткой информации в интеллектуальных системах.
16. Интеллектуальные многоагентные системы.
17. Платформы для построения MAC.
18. Применения СИИ в обработке изображений и компьютерном зрении.
19. Распознавание графических объектов методом Виолы-Джонса.

### Образец экзаменационного билета

1. История искусственного интеллекта. Сферы применения. Большие данные в СИИ.
2. Задание. Используя с#-библиотеки машинного обучения, в среде Visual Studio.NET разработать WEBAPI-сервис, реализующий модель интеллектуального агента поиска кратчайшего пути эвристическим алгоритмом A\*. Входные данные сервиса - карта местности, выходные данные - матрица пути. Обмен данными с сервисом выполнить в формате JSON. Обратиться к агенту из Windows Form - приложения. Примерный вид формы разработанного приложения:



### Критерии оценивания ответа на экзамене

1. Нормы оценивания ответа

№п/п	Структурная часть билета	Количество баллов
1	Теоретический вопрос	2 балла
2	Математическая модель	1 балл
3	Реализация решения задачи	2 балла

(\*) Возможна градация в 0,25 балла.

2. Шкала оценивания работы:

п/п	Оценка	Количество баллов
1	Отлично	4,75-5
2	Хорошо	3,75-4,5
3	Удовлетворительно	3-3,5
4	Неудовлетворительно	менее 3

## 7. Перечень основной и дополнительной учебной литературы

### 7.1. Основная литература

1. Бессмертный И. А. Интеллектуальные системы: учебник и практикум для вузов / И. А. Бессмертный, А. Б. Нугуманова, А. В. Платонов. – Москва : Издательство Юрайт, 2022. – 243 с. – (Высшее образование). – ISBN 978-5-534-01042-8. – URL: <https://urait.ru/bcode/490020>
2. Бессмертный И. А. Интеллектуальные системы: учебник и практикум для среднего профессионального образования / И. А. Бессмертный, А. Б. Нугуманова, А. В. Платонов. – Москва: Издательство Юрайт, 2022. – 243 с. – (Профессиональное образование). – ISBN 978-5-534-07818-3. – URL: <https://urait.ru/bcode/494506>
3. Бессмертный И. А. Системы искусственного интеллекта : учебное пособие для среднего профессионального образования / И. А. Бессмертный. – 2-е изд., испр. и доп. – Москва : Издательство Юрайт, 2022. – 157 с. – (Профессиональное образование). – ISBN 978-5-534-11361-7. – URL: <https://urait.ru/bcode/494434>
4. Бессмертный И. А. Системы искусственного интеллекта: учебное пособие для вузов / И. А. Бессмертный. – 2-е изд., испр. и доп. – Москва: Издательство Юрайт, 2022. – 157 с. – (Высшее образование). – ISBN 978-5-534-07467-3. – URL : <https://urait.ru/bcode/490657>
5. Болотова Л. С. Системы поддержки принятия решений в 2 ч. Часть 1: учебник и практикум для вузов / Л. С. Болотова ; ответственные редакторы В. Н. Волкова, Э. С. Болотов. – Москва: Издательство Юрайт, 2022. – 257 с. – (Высшее образование). – ISBN 978-5-9916-8250-3. – URL: <https://urait.ru/bcode/490259>
6. Болотова Л. С. Системы поддержки принятия решений в 2 ч. Часть 2: учебник и практикум для вузов / Л. С. Болотова; ответственные редакторы В. Н. Волкова, Э. С. Болотов. – Москва: Издательство Юрайт, 2022. – 250 с. – (Высшее образование). – ISBN 978-5-9916-8251-0. – URL: <https://urait.ru/bcode/471000>
7. Воронов М. В. Системы искусственного интеллекта: учебник и практикум для вузов / М. В. Воронов, В. И. Пименов, И. А. Небаев. – Москва: Издательство Юрайт, 2022. – 256 с. – (Высшее образование). – ISBN 978-5-534-14916-6. – URL: <https://urait.ru/bcode/485440>
8. Иванов В. М. Интеллектуальные системы : учебное пособие для среднего профессионального образования / В. М. Иванов ; под научной редакцией А. Н. Сесекина. – Москва : Издательство Юрайт, 2022. – 93 с. – (Профессиональное образование). – ISBN 978-5-534-07819-0. – URL : <https://urait.ru/bcode/494505>
9. Иванов В. М. Интеллектуальные системы: учебное пособие для вузов / В. М. Иванов; под научной редакцией А. Н. Сесекина. – Москва : Издательство Юрайт, 2022. – 91 с. – (Высшее образование). – ISBN 978-5-534-00551-6. – URL : <https://urait.ru/bcode/492094>
10. Новиков Ф. А. Символический искусственный интеллект: математические основы представления знаний: учебное пособие для вузов / Ф. А. Новиков. – Москва: Издательство Юрайт, 2022. – 278 с. – (Высшее образование). – ISBN 978-5-534-00734-3. – URL: <https://urait.ru/bcode/490386>
11. Платонов А. В. Машинное обучение: учебное пособие для вузов / А. В. Платонов. – Москва: Издательство Юрайт, 2022. – 85 с. – (Высшее образование). – ISBN 978-5-534-15561-7. – URL: <https://urait.ru/bcode/508804>
12. Сергеев Л. И. Цифровая экономика: учебник для вузов / Л. И. Сергеев, А. Л. Юданова; под редакцией Л. И. Сергеева. – Москва: Издательство Юрайт, 2022. – 332 с. – (Высшее образование). – ISBN 978-5-534-13619-7. – URL: <https://urait.ru/bcode/497448>
13. Станкевич Л. А. Интеллектуальные системы и технологии: учебник и практикум для среднего профессионального образования / Л. А. Станкевич. – Москва: Издательство Юрайт, 2022. – 397 с. – (Профессиональное образование). – ISBN 978-5-534-11659-5. – URL: <https://urait.ru/bcode/495988>

### 7.2. Дополнительная литература

1. Mitchell T. Machine learning. McGraw-Hill, 1997. 414 p.

2. Воронцов К.В. Машинное обучение, курс лекций. [Электронный ресурс] – Режим доступа. – URL: [http:// www. machinelearning.ru/ wiki/images/6/6d/Voron-ML-1.pdf](http://www.machinelearning.ru/wiki/images/6/6d/Voron-ML-1.pdf)
3. Джонс М. Т. Программирование искусственного интеллекта в приложениях; Пер. с англ. Осипов А. И. – М.: ДМК Пресс, 2016. – 312 с.
4. Луис Педро Коэльо, Вилли Ричард. Построение систем машинного обучения на языке Python. ДМК Пресс , 2016, 302 с.
5. Макконелл Дж. Основы современных алгоритмов: Учеб. пособие . – М.: Техносфера, 2014. – 366 с.
6. Петер Флах. Машинное обучение. Пер. с англ. ДМК Пресс, 2015, 400 с.
7. Рассел С. Искусственный интеллект. Современный подход. – М.: Вильямс, 2007. – 1410 с.
8. Рутковская Д. Нейронные сети, генетические алгоритмы и нечёткие системы. – М.: Горячая линия – Телеком, 2016. – 452 с.

### **7.3. Перечень ресурсов информационно-телекоммуникационной сети «Интернет»**

1. Система дистанционного обучения СмолГУ ([moodle.smolgu.ru](http://moodle.smolgu.ru)).
2. Национальный открытый университет ([intuit.ru](http://intuit.ru)).
3. Национальная платформа открытого образования ([opened.ru](http://opened.ru)).
4. Ресурс Сетевой Академии Cisco (учебный курс IoE).
5. Каталог образовательных Internet-ресурсов (<http://window.edu.ru>).
6. Библиотека разработчика Microsoft (<http://msdn.microsoft.com>).

### **8. Материально-техническое обеспечение**

Для проведения занятий лекционного типа предлагаются наборы демонстрационного оборудования и учебно-наглядных пособий, обеспечивающие тематические иллюстрации, соответствующие программе дисциплины (модулей), учебная ауд. 224 на 12 посадочных мест.

Перечень материально-технического обеспечения, необходимого для реализации курса, включает в себя лабораторию, оснащенную персональными компьютерами, объединенные в сеть с выходом в Интернет, проектором и интерактивной доской, ауд.224 на 12 посадочных мест и 6 парт (12 посадочных мест).

Помещение для самостоятельной работы обучающихся оснащено компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду университета, ауд.224 на 12 посадочных мест и 6 парт (12 посадочных мест).

### **9. Программное обеспечение**

1. Операционная система MS Windows
2. Система программирования MS Visual Studio 19 (язык программирования C#).
3. Python 3.9
4. PyCharm Pro
5. Поисковые системы сети Интернет.

ДОКУМЕНТ ПОДПИСАН  
ЭЛЕКТРОННОЙ ПОДПИСЬЮ

Сертификат: 03B6A3C600B7ADA9B742A1E041DE7D81B0  
Владелец: Артеменков Михаил Николаевич  
Действителен: с 04.10.2021 до 07.10.2022