

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.ДВ.01.01.03 Разработка web-сервисов для машинного обучения

Направление подготовки: **09.03.03 Прикладная информатика**

Направленность (профиль): **Информационные системы организаций и предприятий**

Форма обучения: очная

Курс – 3

Семестр – 6

Всего зачетных единиц – 4, часов–144

Форма отчетности: зачет – 6 семестр

Программу разработал
кандидат технических наук, доцент Т.А.Самойлова

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

Заведующий кафедрой

С.В. Козлов

Смоленск
2022

1. Место дисциплины в структуре ОП

Дисциплина «Разработка web – сервисов для машинного обучения» относится к дисциплинам по выбору учебного плана направления подготовки 09.03.03 Прикладная информатика. Она изучается на 3 курсе в 6 семестре и посвящена разработке web – сервисов, содержащих алгоритмы машинного обучения для использования в банках, финансовых и страховых компаниях, компаниях розничной торговли и других сфер бизнеса в целях анализа и повышения объема продаж и услуг. Для изучения данной дисциплины необходимы компетенции студентов, сформированные при изучении таких дисциплин, как «Разработка web – приложений», «Основы проектирования сайтов», «Языки и методы программирования». Рассматриваются возможные подходы к разработке систем машинного обучения и web-сервисов, предназначенных для построения корпоративных информационных систем современных организаций и предприятий, обеспечивающих возможность удаленного доступа к данным и коллективной работы сотрудников. Дисциплина «Разработка web – сервисов для машинного обучения» позволит студентам приобрести теоретические и практические знания по разработке моделей машинного обучения, а также обеспечит освоение новых технологий, таких, как применение методов машинного обучения в веб - проектировании.

Изучение курса основано на традиционных методах высшей школы, тесной взаимосвязи со смежными курсами, обобщающими методологию исследований и проектирования корпоративных информационных систем.

2. Планируемые результаты обучения по дисциплине

| Компетенция | Индикаторы достижения |
|--|---|
| ПК-3. Способен создавать прототипы автоматизированных информационных систем (ERP-систем), разрабатывать программный код информационной системы и баз данных информационной системы для управления бизнес-процессами, создавать прикладное программное обеспечение | Знать: современные языки программирования, их синтаксис, языки программирования и работы с базами данных, теорию баз данных, инструменты и методы тестирования характеристик ИС и прототипирования пользовательского интерфейса, возможности типовой ИС, ее устройство и функционирование, основы современных операционных систем, систем управления базами данных. Уметь: кодировать на языках программирования, тестировать результаты прототипирования, тестировать разрабатываемую ИС (модульное, интеграционное тестирование), обнаруживать и устранять несоответствия и дефекты. Владеть: навыками по созданию прикладного программного обеспечения, разработке прототипов ИС, разработке кода ИС и баз данных ИС, тестирования ИС, устранения обнаруженных несоответствий и дефектов. |

3. Содержание дисциплины

В дисциплине «Разработка web – сервисов для машинного обучения» рассматриваются следующие темы.

Тема 1.Машинное обучение (МО) и области его применения. Примеры задач МО: поиск информации в интернете, распознавание изображений, лиц, эмоций, пола, возраста, распознавание речи, языка, эмоциональной окраски текстов, прогнозирование продаж, прогнозирование оттока клиентов, кредитный скоринг, рекомендательные системы. Сервисы, основанные на обработке данных, машинном обучении и их место в финансовых услугах.

Большие данные и МО. Критерии аналитических задач, решение которых предпочтительно с использованием технологий BigData. Процесс решения задач методами МО. Подходы и методы МО: обучение с учителем и обучение без учителя. Типология задач МО, выбор алгоритма в зависимости от типа решаемой задачи.

Тема 2. Средства Python для работы с алгоритмами МО. Библиотека Scikit-learn. NumPy как средство эффективной и SciPy как средство интеллектуальной обработки данных. Предварительная обработка и очистка данных. Алгоритмы классификации: логистическая регрессия (Logistic Regression), метод ближайших соседей (kNearest Neighbor, kNN), наивный байесовский классификатор (Naive Bayes Classifier, NBC), дерево принятия решений (Decision Tree). Фреймворк Django для создания веб – сервисов. Разработка веб – сервиса классификации, прогнозирующего отток клиентов банка.

Тема 3. Алгоритмы регрессии. Постановка задачи. Алгоритмы регрессии: линейная и нелинейная регрессии (Linear Regression, Nonlinear Regression), метод опорных векторов (Support Vector Machine, SVM-Regression), обобщенная линейная модель (Generalized Linear Model), дерево принятия решений (Regression Tree). Множественная линейная регрессия, ее преимущества и недостатки. Проверка регрессионных моделей. Ошибка обучения и ошибка прогноза. Оценка качества регрессионных моделей. Разработка веб – сервиса регрессии, прогнозирующего стоимость квартир.

Тема 4. Алгоритмы кластеризации. Обучение без учителя. Постановка задачи. Типы кластерных структур. Меры сходства в задачах кластеризации. Типы метрик. Жесткая и мягкая кластеризации. Алгоритмы МО для задач кластеризации. Нейронные сети Кохонена, архитектура и обучение. Слой Кохонена. Самоорганизующаяся карта Кохонена, архитектура и правила обучения.

Тема 5. Поиск ассоциативных правил в анализе данных. Понятия и методы выявления закономерностей алгоритмами ассоциаций. Использование выявленных закономерностей для предсказания неизвестных значений. Класс «Задача маркетинговой корзины». Поддержка, достоверность, лифт. Алгоритмы Apriori, FP-Growth, Eclat. Анализ исключений, предназначенный для выявления и толкования аномалий в найденных закономерностях. Функция ошибок. Функционал качества.

Тема 6. Искусственные нейронные сети (ИНС). Основные понятия и определения теории ИНС. Однослойные и многослойные ИНС. Неглубокие (Shallow Networks) и глубокие (Deep Networks) нейронные сети. Области применения ИНС. Обучение глубокой нейронной сети. Сверточные нейронные сети (СНС). Функции активации. Рекуррентные нейронные сети (РНС). Назначение, структура РНС. Проблема долгосрочных зависимостей. Сети LSTM. Архитектура LSTM-сетей в зависимости от типа решаемых задач (классификации, регрессии, видео-классификации).

Тема 7. Компьютерное зрение. Введение в компьютерное зрение и устройство зрительной системы человека. Обработка изображений. Простые методы анализа изображений. Представление изображений. Оценка параметров моделей. Машинное обучение и классификация изображений. Сегментация изображений. Детектирование объектов на изображениях. Отслеживание объектов на видео. Атаки на обученные модели нейронных сетей. Поиск изображений по содержанию. Распознавание событий в видео. Компьютерное зрение в реальном времени. Разработка веб – сервиса распознавания изображений.

Тема 8. Распознавание и синтез речи. Понятия звука, речи. Виды и структура звуковых файлов. Классификация задач области речевых технологий. Основные характеристики речевого сигнала. Базовые алгоритмы обработки речевых сигналов. Рекуррентные нейронные сети для распознавания речи. Выделение признаков из звука. Выравнивание. Нормализация текстов для синтеза речи. Проблемы синтеза звука. Современные архитектуры: WaveNet, DeepVoice и их улучшения. Подходы к распознаванию и синтезу речи. Имеющиеся ограничения использования подходов. Обзор вариантов практического использования систем.

4. Тематический план

| № п/п | Разделы и темы | Всего часов | Формы занятий | | | |
|--------------|--|-------------|---------------|----------------------|----------------------|------------------------|
| | | | лекции | практические занятия | лабораторные занятия | самостоятельная работа |
| 1 | Машинное обучение (МО) и области его применения. | 18 | 2 | – | 8 | 8 |
| 2 | Средства Python для работы с алгоритмами МО. | 18 | 2 | – | 8 | 8 |
| 3 | Алгоритмы регрессии. | 18 | 2 | – | 8 | 8 |
| 4 | Алгоритмы кластеризации. | 18 | 2 | – | 8 | 8 |
| 5 | Поиск ассоциативных правил в анализе данных. | 18 | 2 | – | 8 | 8 |
| 6 | Искусственные нейронные сети (ИНС). | 18 | 2 | – | 8 | 8 |
| 7 | Компьютерное зрение. | 18 | 2 | – | 8 | 8 |
| 8 | Распознавание и синтез речи. | 18 | 2 | – | 8 | 8 |
| ИТОГО | | 144 | 16 | – | 64 | 64 |

5. Виды образовательной деятельности

Занятия лекционного типа

Тема 1. Машинное обучение (МО) и области его применения. Примеры задач МО: поиск информации в интернете, распознавание изображений, лиц, эмоций, пола, возраста, распознавание речи, языка, эмоциональной окраски текстов, прогнозирование продаж, прогнозирование оттока клиентов, кредитный скоринг, рекомендательные системы. Сервисы, основанные на обработке данных, машинном обучении и их место в финансовых услугах. Большие данные и МО. Критерии аналитических задач, решение которых предпочтительно с использованием технологий BigData. Процесс решения задач методами МО. Подходы и методы МО: обучение с учителем и обучение без учителя. Типология задач МО, выбор алгоритма в зависимости от типа решаемой задачи.

Тема 2. Средства Python для работы с алгоритмами МО. Библиотека Scikit-learn. NumPy как средство эффективной и SciPy как средство интеллектуальной обработки данных. Предварительная обработка и очистка данных. Алгоритмы классификации: логистическая регрессия (Logistic Regression), метод ближайших соседей (kNearest Neighbor, kNN), наивный байесовский классификатор (Naive Bayes Classifier, NBC), дерево принятия решений (Decision Tree). Фреймворк Django для создания веб – сервисов. Разработка веб – сервиса классификации, прогнозирующего отток клиентов банка.

Тема 3. Алгоритмы регрессии. Постановка задачи. Алгоритмы регрессии: линейная и нелинейная регрессии (Linear Regression, Nonlinear Regression), метод опорных векторов (Support Vector Machine, SVM-Regression), обобщенная линейная модель (Generalized Linear Model), дерево принятия решений (Regression Tree). Множественная линейная регрессия, ее преимущества и недостатки. Проверка регрессионных моделей. Ошибка обучения и ошибка прогноза. Оценка качества регрессионных моделей. Разработка веб – сервиса регрессии, прогнозирующего стоимость квартир.

Тема 4. Алгоритмы кластеризации. Обучение без учителя. Постановка задачи. Типы кластерных структур. Меры сходства в задачах кластеризации. Типы метрик. Жесткая и мягкая кластеризации. Алгоритмы МО для задач кластеризации. Нейронные сети Кохонена, архитектура и обучение. Слой Кохонена. Самоорганизующаяся карта Кохонена, архитектура и правила обучения.

Тема 5. Поиск ассоциативных правил в анализе данных. Понятия и методы выявления закономерностей алгоритмами ассоциаций. Использование выявленных закономерностей в для предсказания неизвестных значений. Класс «Задача маркетинговой корзины». Поддержка,

достоверность, лифт. Алгоритмы Apriori, FP-Growth, Eclat. Анализ исключений, предназначенный для выявления и толкования аномалий в найденных закономерностях. Функция ошибок. Функционал качества.

Тема 6. Искусственные нейронные сети (ИНС). Основные понятия и определения теории ИНС. Однослойные и многослойные ИНС. Неглубокие (Shallow Networks) и глубокие (Deep Networks) нейронные сети. Области применения ИНС. Обучение глубокой нейронной сети. Сверточные нейронные сети (СНС). Функции активации. Рекуррентные нейронные сети (РНС). Назначение, структура РНС. Проблема долгосрочных зависимостей. Сети LSTM. Архитектура LSTM-сетей в зависимости от типа решаемых задач (классификации, регрессии, видео-классификации).

Тема 7. Компьютерное зрение. Введение в компьютерное зрение и устройство зрительной системы человека. Обработка изображений. Простые методы анализа изображений. Представление изображений. Оценка параметров моделей. Машинное обучение и классификация изображений. Сегментация изображений. Детектирование объектов на изображениях. Отслеживание объектов на видео. Атаки на обученные модели нейронных сетей. Поиск изображений по содержанию. Распознавание событий в видео. Компьютерное зрение в реальном времени. Разработка веб – сервиса распознавания изображений.

Тема 8. Распознавание и синтез речи. Понятия звука, речи. Виды и структура звуковых файлов. Классификация задач области речевых технологий. Основные характеристики речевого сигнала. Базовые алгоритмы обработки речевых сигналов. Рекуррентные нейронные сети для распознавания речи. Выделение признаков из звука. Выравнивание. Нормализация текстов для синтеза речи. Проблемы синтеза звука. Современные архитектуры: WaveNet, DeepVoice и их улучшения. Подходы к распознаванию и синтезу речи. Имеющиеся ограничения использования подходов. Обзор вариантов практического использования систем.

Лабораторные работы

№ 1. *Инструменты анализа данных в Python. Библиотека Pandas.*

№ 2-3. *Инструменты анализа данных в Python. Визуализация средствами Matplotlib.*

№ 4-5. *Классификация объектов методом k - ближайших соседей.*

№ 6-7. *Алгоритмы деревьев решений DecisionTreeClassifier.*

№ 8-9. *Наивный байесовский алгоритм (Naive Bayesian).*

№ 10-11. *Алгоритм метода опорных векторов.*

№ 12. *Разработка веб-сервиса Django для реализации алгоритмов классификации.*

№ 13-14. *Алгоритмы кластеризации в машинном обучении.*

№ 15-16. *Алгоритмы регрессии в машинном обучении.*

№ 17. *Разработка веб-сервиса Django для реализации множественной регрессии.*

№ 18- 19. *Ансамблевые методы в машинном обучении.*

№ 20-21. *Искусственные нейронные сети.*

№ 22. *Фреймворк глубокого обучения TensorFlow.*

№ 23-24. *Библиотека глубокого обучения Keras для работы с ИНС.*

№ 25-26. *Библиотека глубокого обучения Keras для распознавания изображений.*

№ 27. *Обучение рекуррентных нейронных сетей.*

№ 28. *Обнаружение объектов на изображении.*

№ 29. *Обнаружение и верификация лиц.*

№ 30. *Разработка веб-сервиса Django для реализации алгоритмов обнаружения объектов на изображении средствами модели сверточной нейронной сети VGG16.*

№ 31. *Веб-сервисы машинного обучения для распознавания речи.*

№ 32. *Веб-сервисы машинного обучения для синтеза речи.*

Задания для лабораторных работ, размещены в системе дистанционного обучения СмолГУ (www.moodle.smolgu.ru). На занятиях для каждой работы задание предоставляется студентам в электронном виде.

Самостоятельная работа

Текущая самостоятельная работа студента направлена на углубление и закрепление знаний студентов, развитие 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 включает:

- проработку лекционного материала, составление конспекта лекций по темам, вынесенным на самостоятельное изучение;
- выполнение домашних заданий;
- подготовку к защите лабораторных работ.

Темы для самостоятельного изучения

1. Алгоритмы ID3, C4.5, CART.
2. Гребневая регрессия.
3. Ансамбли моделей. Стекинг. Бустинг. Бэггинг.
4. Случайные леса. AdaBoost.
5. Ансамблевые методы регрессии.
6. Нейронные сети. Перцептрон Розенблатта.
7. Функции активации.
8. Нейрон с сигмоидальной функцией активации.
9. Перцептрон и его обучение.
10. Структура многослойного перцептрона

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.

Учебно-методическое обеспечение для самостоятельной работы

1. Интернет-Университет Информационных Технологий, <http://www.intuit.ru>
2. Алгоритмы машинного обучения : учебник и практикум для академического бакалавриата [электронный ресурс]/ Д. В. Чистов, П. П. Мельников, А. В. Золотарюк, Н. Б. Ничепорук ; под общ. ред. Д. В. Чистова. — М. : Издательство Юрайт, 2021. — 258 с. — (Серия : Бакалавр. Академический курс). — ISBN 978-5-534-00492-2.
3. Федоров, Д. Ю. Программирование на языке высокого уровня python : учебное пособие для прикладного бакалавриата / Д. Ю. Федоров. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2020. — 161 с. — (Бакалавр. Прикладной курс). — ISBN 978- 5-534-10971-9. — Текст: электронный // ЭБС Юрайт [сайт]. — URL: <https://www.biblioonline.ru/bcode/437489>

6. Критерии оценивания результатов освоения дисциплины (модуля)

6.1. Оценочные средства и критерии оценивания для текущей аттестации

Теоретические вопросы

1. Машинное обучение (МО) и области его применения.
2. Средства Python для работы с алгоритмами МО. Алгоритмы классификации. Разработка веб – сервиса классификации, прогнозирующего отток клиентов банка.
3. Алгоритмы регрессии. Разработка веб – сервиса регрессии прогнозирующего стоимость квартир.
4. Алгоритмы кластеризации.
5. Поиск ассоциативных правил в анализе данных.

6. Искусственные нейронные сети (ИНС).Обучение глубокой нейронной сети. Сверточные и рекуррентные нейронные сети.
7. Компьютерное зрение. Разработка веб – сервиса распознавания изображений.
8. Распознавание и синтез речи.

Критерии оценивания теоретических вопросов

1. Нормы оценивания ответов на теоретические вопросы

| № п/п | Теоретический вопрос | Количество баллов (*) |
|-------|--|-----------------------|
| 1 | Дан краткий ответ на поставленный вопрос | 1 балл |
| 2 | Дан развернутый ответ на вопрос с анализом результатов | 2 балла |

(*) Возможна градация в 0,25 балла.

2. Шкала оценивания. Оценка «зачтено» за ответы на теоретические вопросы выставляется, если набрано не менее 3 баллов при ответе на три вопроса, в противном случае выставляется «не зачтено».

Задания для лабораторных занятий

Лабораторное занятие № 1. *Инструменты анализа данных в Python. Библиотека Pandas.*

Цель занятия: Освоение инструментов анализа данных в Python

Задание 1. Используя Pandas, выполнить импорт набора данных *.CSV, содержащий пропущенные данные. Выполнить:

- замену отсутствующих значений нулями
- замену отсутствующих средними по столбцу
- удаление строк с отсутствующими элементами

Задание 2. Используя Pandas, выполнить импорт набора данных *.CSV, содержащих строковые и числовые поля. Выполнить (с выводом таблиц):

- Отбор колонок (2-4)
- Отбор строк в некотором диапазоне
- Фильтрацию данных по числовому значению (>,<) и по заданному значению строкового поля.

Подсчет количеств строк с каждым значением строкового поля

Задание 3. Используя Pandas, выполнить импорт набора данных *.CSV, содержащий категориальные (текстовые) данные. Выполнить замену категориальных данных числовыми.

Лабораторное занятие № 2. *Инструменты анализа данных в Python. Визуализация средствами Matplotlib. Описательная статистика.*

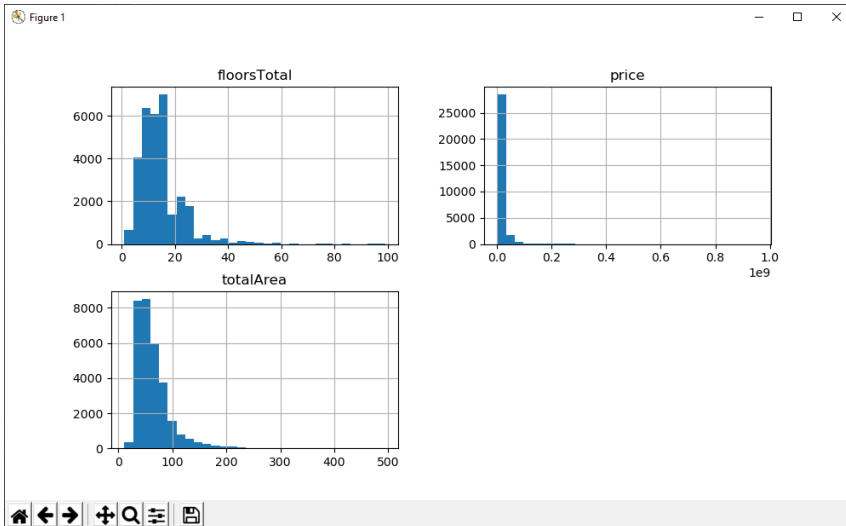
Цель занятия: Освоение инструментов анализа данных в Python

Задание 1. Исследовать объявления о продаже квартир, представленных CSV-файлом. Построить распределение квартирных цен. Построить графики описательной статистики для целевые характеристики квартиры - цены, этажности и общей площади. Проверить, насколько удаление дорогостоящих квартир скажется на распределении цены.

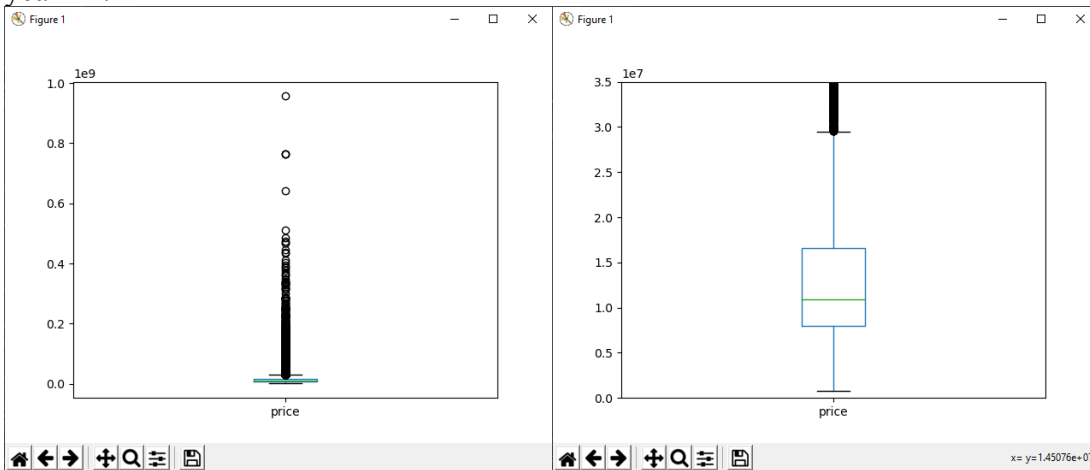
Выборка исследования: архив объявлений о продаже квартир в Москве за несколько лет - данные сервиса Яндекс.Недвижимость (31191 записей). Данные в csv-файле содержат: материал стен, этажность, этажность общая, общая площадь, площадь кухни, широта, долгота, цена. Соответствующиеименаполейдля DataFrame: wallsMaterial, floorNumber, floorsTotal, totalArea, kitchenArea, latitude, longitude, price. Аналогичное исследование, проведенное на данных Санкт-Петербурга, описано на сайте:

<https://code.s3.yandex.net/marketing/%D0%9D%D0%B5%D0%B4%D0%B2%D0%B8%D0%B6%D0%B8%D0%BC%D0%BE%D1%81%D1%82%D1%8C.html>

Задание 2. Сформируйте описательную статистику для целевых характеристик квартиры - цены, этажности и общей площади. Пример результата:

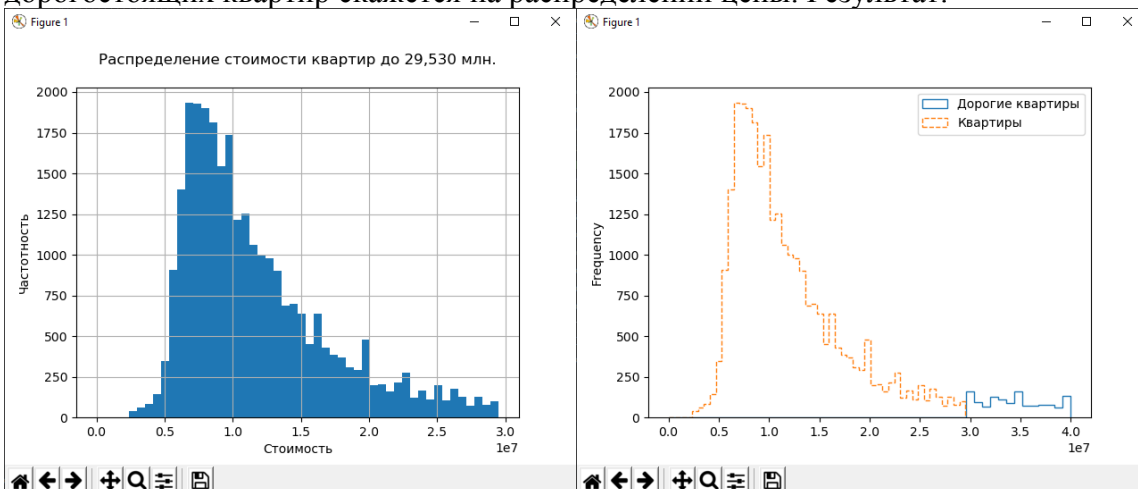


Исследуйте цену квартир. Для этого строим диаграмму размаха и «ящик с усами». Создаем для цены квартиры базовые статистические параметры: медиану (среднее значение), границы (25-й и 75-й процентиля), минимальное и максимальное значения. Лучше всего это видно в «ящике с усами»:



На диаграмме размаха видны выбросы (точки). Одним из способов борьбы с ними - удаление всех экстремальных значений, которые превышают сумму третьего квартиля и полтора интерквартильного размаха. Вычислите границы до выбросов.

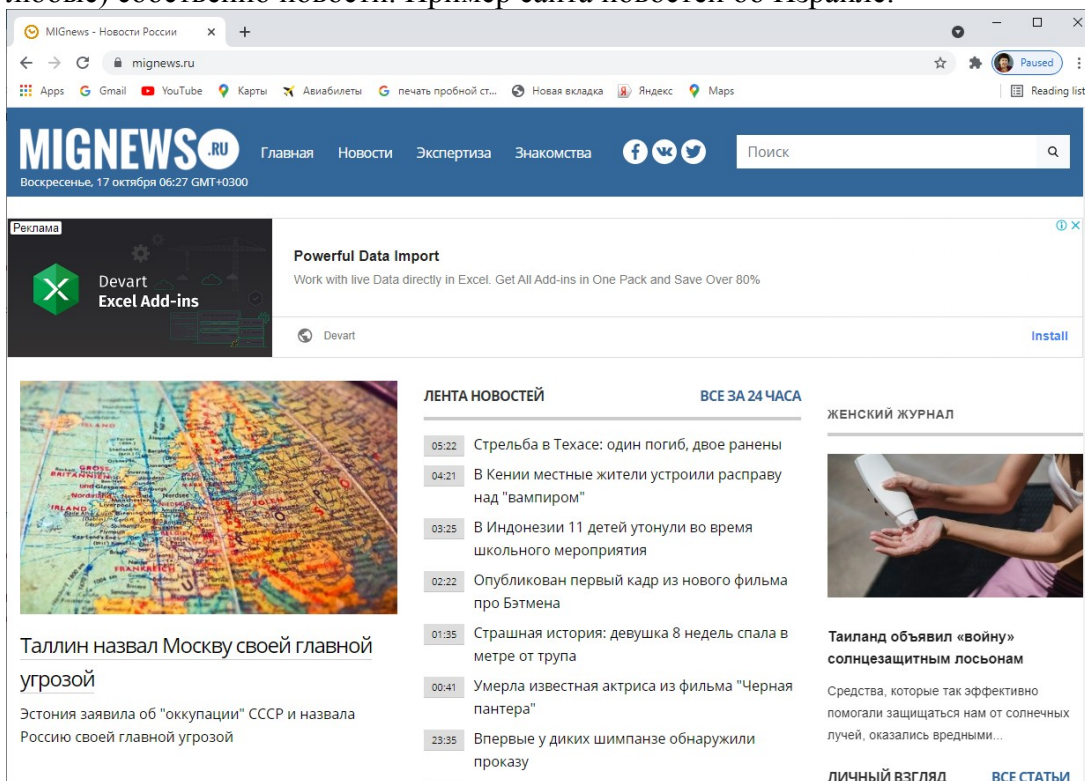
Строим распределение стоимости квартиры до 29,530 млн. Проверим, насколько удаление дорогостоящих квартир скажется на распределении цены. Результат:



Лабораторное занятие № 3. Инструменты анализа данных из интернета в Python. Визуализация средствами Matplotlib. Описательная статистика.

Цель занятия: Освоение инструментов анализа данных веб-страниц средствами Python-библиотеки BeautifulSoup.

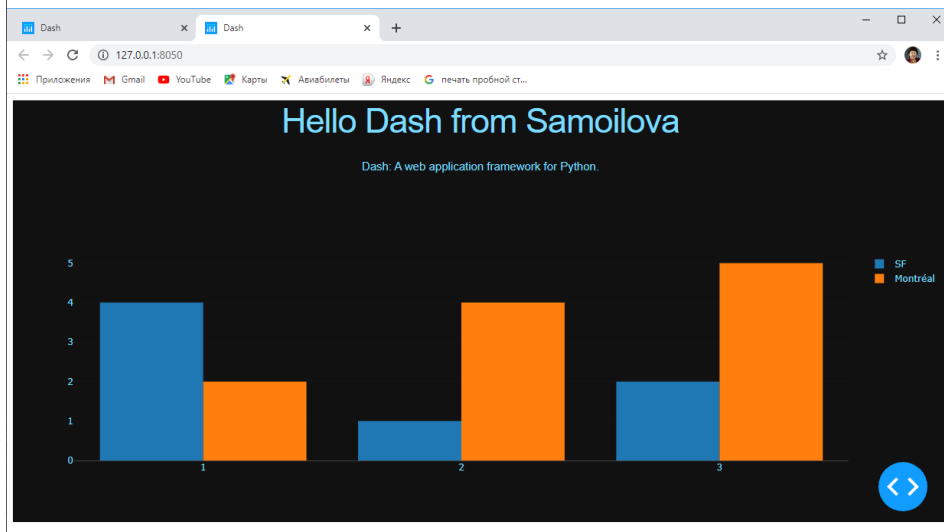
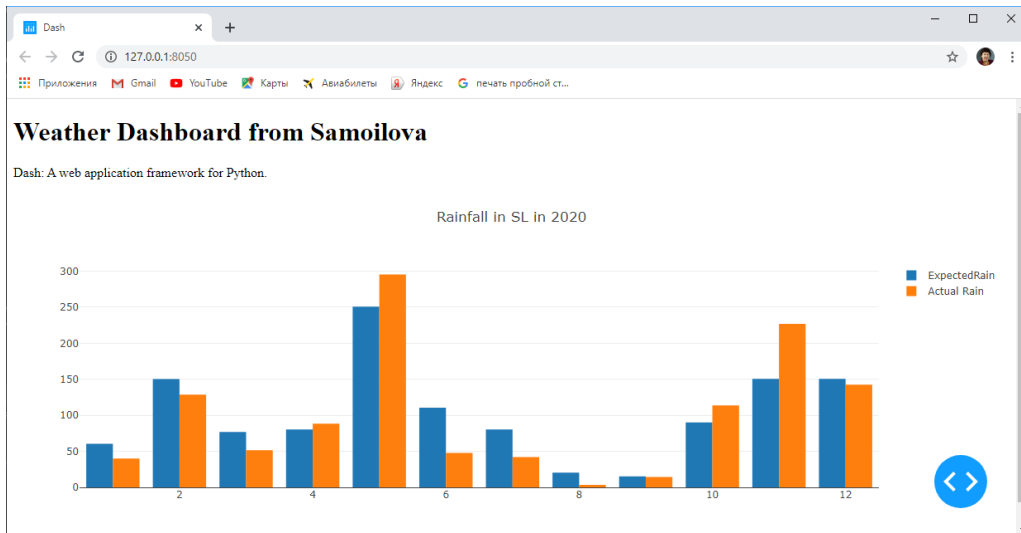
Задание 1. Разработайте парсер сайта на python, который вырезает из сайта новостей (найдите любые) собственно новости. Пример сайта новостей об Израиле:



Результат выбора классов с новостями и сохранения их в файле:

10:24 В Израиле от коронавируса скончался 33-летний мужчина
10:23 Израиль: утверждены поправки к правилам о режиме ЧС
07:01 Аббас: выходим из всех договоров с Израилем и США
21:03 В Израиле – 278 жертв COVID-19, 16659 зараженных
20:25 Впервые в истории: самолет ОАЭ приземлился в Израиле
20:00 Нетаниягу провел ряд встреч с министрами: все подробности
18:48 ЦАХАЛ нанес авиаудары на юге Сирии
17:45 В Израиль прибыли 111 репатриантов из Украины
16:13 Главный раввин хочет, чтобы Нетаниягу срочно открыл синагоги
14:34 Открытие банкетных комплексов с середины июня
14:05 Транспорт в час-пик: конец ограничениям
13:31 Израиль готовится к открытию 136 пляжей: какие правила
13:16 Рестораны и ночные клубы откроют через неделю
13:07 Разрывшая могилу сержанта Голани отправлена к психиатру
11:00 Число жертв COVID-19 в Израиле достигло 277
09:50 Посольство США в Ираке подверглось новому ракетному обстрелу
09:46 Медсестра-репатриантка жестоко избита пациентом
09:01 Более ста авиакомпаний возвращаются в Израиль
08:36 Масштабный пожар возле Эльада взят под контроль
07:45 Возле Эльада вспыхнул масштабный пожар
07:30 Эдельштейн временно отменил маски в школах
06:52 Хаос в иранском порту после израильской кибератаки

Задание 2. Используя Python-фреймворк Dash, создайте веб-приложение вывода произвольной таблицы данных в графическом виде. Примеры:



Лабораторное занятие № 4-5. Классификация объектов методом k - ближайших соседей.

Цель занятия: Реализация алгоритмов классификация объектов методом машинного обучения k - ближайших соседей.

Задание 1. 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. Данные для экспериментов (для каждого студента строго индивидуальные): UCI Machine Learning Repository <http://archive.ics.uci.edu/ml/>

Задание 2. В модели, созданной по заданию 1, замените тестовую выборку обучающей и сравните результаты выполненного моделью прогнозирования со значениями классов из обучающей выборки. По результатам сравнения оцените качество прогноза созданной модели (аккуратность, точности, полноту, F-меру). Измените параметры модели, обеспечивающие лучшую (худшую) точность прогноза.

Задание 3. Измените способ задания тестовой выборки для задания 1. Данные из файла случайным образом делятся на две выборки в заданной пропорции. Оцените точность классификации для каждого из способов.

Задание 4. Выполните сохранение обученной классификационной модели прогнозирования в файл. Используя файл обученной модели, выполните прогноз, не используя этап обучения.

Лабораторное занятие № 6-7. Алгоритмы деревьев решений *DecisionTreeClassifier*.

Цель занятия: Реализация алгоритмов классификация объектов методами деревьев решений

Задание 1. Обучите модель методом деревьев решений по демонстрационным данным "Ирисы". Выборка включает 100 цветочков. Бинарная классификация.

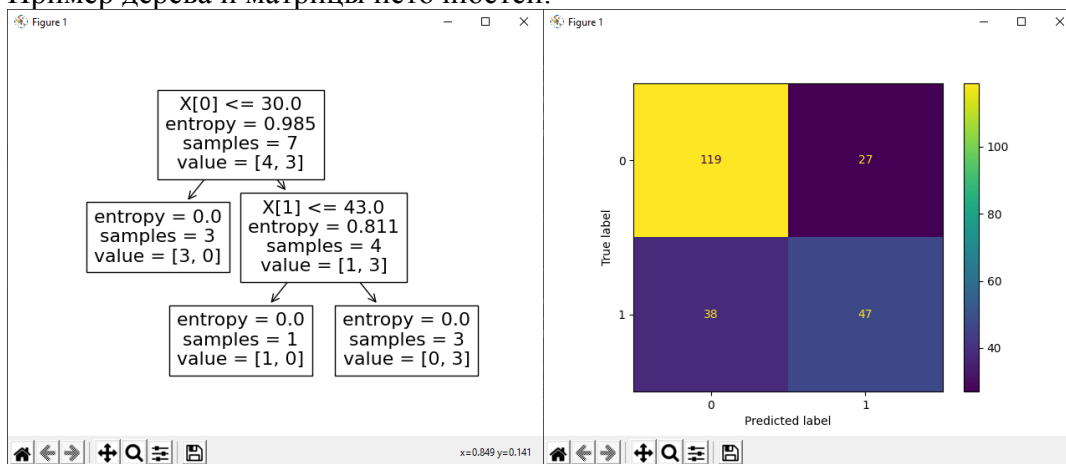
Задание 2. Обучите модель методом деревьев решений по демонстрационным данным "Ирисы". Выборка из трех классов. Тестовая выборка формируется методом кросс - валидации. Результат классификации (обучающая выборка 130 цветочков, тестовая выборка: 0.33 от обучающей) - Оценки качества выполните посредством кросс – валидации. Постройте графики зависимостей класса от признаков.

Задание 3. Методом деревьев решений выполните обучение модели классификации на конкретных индивидуальных 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). До обучения модели, постройте корреляционную матрицу параметров, которая показывает, какие параметры будут влиять на результат. Оставьте для обучения 3-4 наиболее значимых параметра. Замените категориальные параметры числовыми. Тестовую выборку сформируйте методом кросс - валидации. Подберите параметры метода, обеспечивающие лучшее качество.

Задание 4. Обучите модель методом деревьев решений по вашим данным, используя алгоритмы C4.5 и CART. Сравните результаты

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 алгоритмом **RandomForest** по индивидуальным данным . Тестовая выборка формируется методом кросс - валидации. Постройте матрицу неточности, Оцените качество прогнозирования при разных размерах тестовой выборки = 0,2; 0,3; 0,4; 0,5 от обучающей.

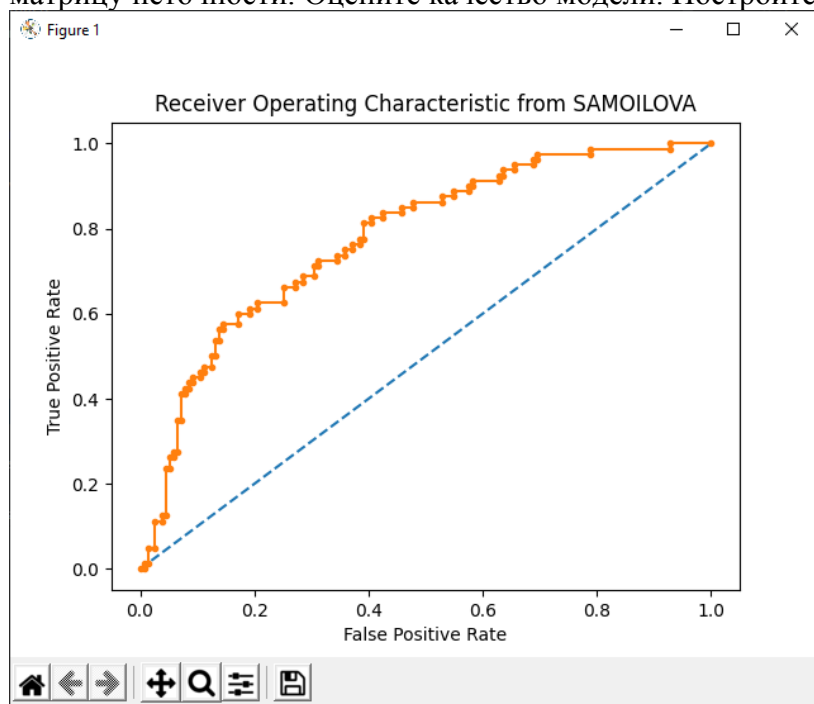
Лабораторное занятие № 8-9. Наивный байесовский алгоритм (НБА, *Naive Bayesian*).

Цель занятия: Реализация алгоритмов классификация объектов методами наивного Байеса.

Задание 1. Методом наивного Байеса выполните обучение модели классификации на конкретных индивидуальных данных. Оцените качество модели на тестовых данных, эквивалентных обучающим и делении данных методом кросс - валидации. Результаты: оценки качества - точность, полнота, f-мера, ROC – кривая. Постройте графики зависимости класса от двух входных параметров (например, возраст, доход). Подберите параметры метода, обеспечивающие лучшее качество.

Задание 2. Обучите модель методом наивного Байеса по данным зависимости победы в спортивной игре (0,1) от погоды (разная). Данные задайте массивами. Оцените точность F1-score. Два класса.

Задание 3. Обучите модель наивного Байеса по демонстрационным данным "Диабет". Выборка из двух классов. Тестовая выборка формируется методом кросс - валидации. Постройте матрицу неточности. Оцените качество модели. Постройте ROC-кривую. Пример ROC-кривой:



Задание 4. Оцените качество прогнозирования методом кросс-валидации (задание 3) при разных размерах тестовой выборки = 0,2; 0,3; 0,4; 0,5 от обучающей.

Задание 5. Обучите модель методом наивного Байеса. Выборка ваша из 7-12 классов. До обучения модели, постройте корреляционную матрицу параметров, которая показывает, как параметры будут влиять на результат. Тестовую выборку сформируйте методом кросс - валидации. Постройте матрицу неточности.

Лабораторное занятие № 10-11. Алгоритм метода опорных векторов(классификация).

Цель занятия: Реализация алгоритмов классификация объектов методами *опорных векторов*.

Задание 1. Разработайте программу классификации методом опорных векторов (ваша обучающая выборка не менее 3 классов) и построить матрицы неточности (Confusion Matrix) для двух тестовых наборов: а) тестовый набор равен обучающей выборке; б) тестовый набор задан в файле или программе.

Задание 2. Разработайте программу классификации методом опорных векторов для конкретных индивидуальных данных. Оцените качество модели на тестовых данных, эквивалентных обучающим и делении данных методом кросс - валидации. Результаты: оценки качества - точность, полнота, f-мера, ROC – кривая. Подберите параметры метода, обеспечивающие лучшее качество.

Задание 3. Обучите модель опорных векторов по демонстрационным данным "Недвижимость". Выборка из двух классов. Тестовая выборка формируется методом кросс - валидации.

Постройте матрицу неточности. Оцените качество модели. Постройте ROC-кривую.

Задание 4. Оцените качество прогнозирования методом кросс-валидации (задание 2) при разных размерах тестовой выборки = 0,2; 0,3; 0,4; 0,5 от обучающей.

Задание 5. Обучите модель методом опорных векторов по вашим данным . Выборка ваша из 3-7 классов. До обучения модели, постройте корреляционную матрицу параметров, которая показывает, как параметры будут влиять на результат. Тестовую выборку сформируйте методом кросс - валидации. Постройте матрицу неточности. Выполните ROC-анализ. Вычислите AUC модели.

Лабораторное занятие № 12. Разработка веб-сервиса Django для реализации алгоритмов классификации.

Цель занятия: Реализация веб - сервисов машинного обучения методом классификации средствами фреймворка Django.

Задание1. Средствами Python создайте модель классификации, реализующую алгоритм наивного Байеса. Обучающая выборка – сведения о клиентах банка. Классы: 0 – клиент ушел из банка, 1 – клиент остался в банке. Фрагмент обучающей выборки файл Churn_Modelling.csv (10000 строк):

```
RowNumber,CustomerId,Surname,CreditScore,Geography,Gender,Age,Tenure,Balance,NumOfProducts,HasCrCard,IsActiveMember,EstimatedSalary,Exited
1,15634602,Hargrave,619,France,Female,42,2,0,1,1,1,101348.88,1
2,15647311,Hill,608,Spain,Female,41,1,83807.86,1,0,1,112542.58,0
3,15619304,Onio,502,France,Female,42,8,159660.8,3,1,0,113931.57,1
4,15701354,Boni,699,France,Female,39,1,0,2,0,0,93826.63,0
```

Exited = 0,1 – класс клиента в обучающей выборке.

Задание 2. Создайте веб-сервис, вычисляющий класс тестового набора данных по готовой, обученной модели **finalized_modelNB.sav** (задание 1). Обращение к сервису выполните из веб-страницы приложения. Пример обращения к сервису машинного обучения из веб-страницы:

The image shows two side-by-side screenshots of a web browser displaying a prediction interface. Both pages have the URL `127.0.0.1:8000/app/ml3` and the title "Здесь будет предсказан уход клиента из банка".

Left Screenshot: The form contains the following input fields: "Код города" with value "2", "Возраст" with value "33", and "Сумма на счете(\$)" with value "199200,00". Below the form is a button "Выполнить прогноз". The result at the bottom is "Прогноз ухода клиента из банка = [0]".

Right Screenshot: The form contains the following input fields: "Код города" with value "3", "Возраст" with value "59", and "Сумма на счете(\$)" with value "115046,74". Below the form is a button "Выполнить прогноз". The result at the bottom is "Прогноз ухода клиента из банка = [1]".

Лабораторное занятие № 13-14. Алгоритмы кластеризации в машинном обучении.

Цель занятия: Реализация алгоритмов кластеризации объектов средствами библиотек Python.

Задание 1. Разработайте программу кластеризации данных алгоритмом K-Means (ваша выборка для кластеризации, набор 1000 – 10000 записей).

Задание 2. Разработайте программу кластеризации данных алгоритмом K-Means (ваша выборка для кластеризации) **Оценки качества.ROC-анализ**

Задание 3. Разработайте программу кластеризации данных алгоритмом K-Means (ваша выборка для кластеризации) . Тестовая выборка формируется методом кросс - валидации.

Задание 4. Для алгоритма задания 3 оцените качество кластеризации с размером тестовой выборки = 0,5 и 0,7 соответственно от обучающей (метод кросс-валидации).

Задание 5. Разработайте программу кластеризации данных алгоритмом DBSCAN (ваша выборка для кластеризации). Выполните вычисление оценок качества и постройте графики зависимостей центроид от признаков выборки.

Задание 6. Выполните оптимизацию числа кластеров для kMeans (задание 1). Для этого необходимо представить задачу кластеризации как задачу оптимизации. Критерий – сумма квадратов расстояний от точек до центроидов кластеров. Постройте зависимость критерия от числа кластеров. Убедитесь, что $J(C_k)$ падает при увеличении числа кластеров k .

Задание 7. Используя пакеи scikit-fuzzy выполните нечеткую кластеризацию кластеризации данных алгоритмом Fuzzy C-Means (данные – демонстрационный пример и ваша выборка). Выполните вычисление коэффициента нечеткого распределения Fuzzy partition coefficient (FPC). Демонстрационный пример включает 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 8. Разработайте программу кластеризации данных алгоритмом метода главных компонент (Principal Component Analysis, PCA) для набора рукописных цифр. Оцените качество кластеризации. Метрики качества кластеризации (пакет sklearn.metrics): Adjusted Rand Index (ARI), Adjusted Mutual Information (AMI), Silhouette(S).

Задание 9. Разработайте программу вычисления метрик качества кластеризации (задание 8) для алгоритмов 'K-means', 'Affinity', 'Spectral', 'Agglomerative', запущенных на данных рукописных цифр.

Лабораторное занятие № 15-16. Алгоритмы регрессии в машинном обучении.

Цель занятия: Реализация алгоритмов регрессии объектов средствами библиотек Python.

Задание 1. 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.

Задание 2. Разработайте программы регрессии методом опорных векторов с использованием линейных, полиномиальных и RBF-ядер. Используйте данные для прогноза цены квартиры. Выборка исследования: архив объявлений о продаже квартир в Москве за несколько лет -

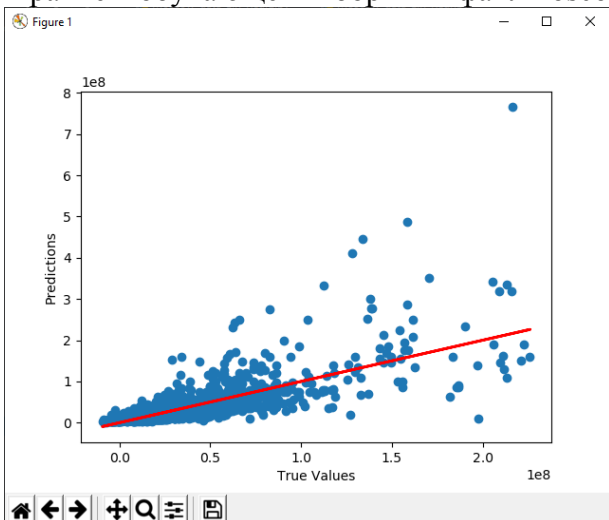
данные (например, moscow_dataset_2020.csv - 31191 записей) сервиса Яндекс. Недвижимость. Они содержат: материал стен, этажность, этажность общая, общая площадь, площадь кухни, широта, долгота, цена. Соответствующие имена полей для DataFrame: wallsMaterial, floorNumber, floorsTotal, totalArea, kitchenArea, latitude, longitude, price. Результаты: Целевая переменная (непрерывная): price – цена квартиры. Вычислите оценки качества для тестовой и обучающей выборки. Постройте графики цены квартиры как функции от этажности, этажности общей, общей площади квартиры, площади кухни.

Задание 3. 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 r^2 _score. Сравните результаты для разных типов ядер. Постройте графики зависимости целевой функции от признаков выборки.

Лабораторное занятие № 17. Разработка веб-сервиса Django для реализации множественной регрессии.

Цель занятия: Реализация алгоритмов регрессии объектов средствами библиотек Python.

Задание 1. Средствами Python создайте модель множественной линейной регрессии. Сохраните обученную модель в файле. Обучающая выборка – сведения о продажах квартир в Москве. Фрагмент обучающей выборки - файл moscow_dataset_2020.csv (31190 строк):



Задание 2. Создайте веб-сервис, вычисляющий прогноз цены квартиры по готовой, обученной модели finalized_model.sav (задание 1). Обращение к сервису выполните из веб-страницы приложения. Пример обращения к сервису машинного обучения из веб-страницы:

| | |
|--|---|
| <p>От Самойловой x +</p> <p>127.0.0.1:8000/app/ml2</p> <p>Сервисы Gmail YouTube Карты Авиабилеты</p> <p>Здесь будет предсказана цена квартиры</p> <p>Заполните поля</p> <p>Этаж квартиры: <input type="text" value="2"/></p> <p>Этажность дома: <input type="text" value="5,0"/></p> <p>Площадь квартиры: <input type="text" value="50,0"/></p> <p>Площадь кухни: <input type="text" value="6,0"/></p> <p>Латитуда: <input type="text" value="55,486698"/></p> <p>Лонгитуда: <input type="text" value="37,59532100"/></p> <p><input type="button" value="Выполнить прогноз"/></p> <p>Цена квартиры Самойловой =</p> | <p>От Самойловой x +</p> <p>127.0.0.1:8000/app/ml2</p> <p>Сервисы Gmail YouTube Карты Авиабилеты</p> <p>Здесь будет предсказана цена квартиры</p> <p>Предсказала отлично!</p> <p>Этаж квартиры: <input type="text" value="2"/></p> <p>Этажность дома: <input type="text" value="5,0"/></p> <p>Площадь квартиры: <input type="text" value="50,0"/></p> <p>Площадь кухни: <input type="text" value="6,0"/></p> <p>Латитуда: <input type="text" value="55,486698"/></p> <p>Лонгитуда: <input type="text" value="37,59532100"/></p> <p><input type="button" value="Выполнить прогноз"/></p> <p>Цена квартиры Самойловой = [7625691.12806916]</p> |
|--|---|

Лабораторное занятие № 18- 19. Ансамблевые методы в машинном обучении.

Цель занятия: Освоение приемов работы по организации ансамблевых методов машинного обучения с использованием алгоритмов, реализованных библиотеками Python.

Задание 1. Разработайте Python - приложение для выполнения стекинг-регрессии средствами класса `sklearn.ensemble.StackingRegressor`. Данные – индивидуальные. Вычислите метрики качества регрессии. Сравните их с методом опорных векторов для регрессии (данные ваши).

Задание 2. Создайте программу для выполнения бэггинг -классификации средствами класса `sklearn.ensemble.RandomForestClassifier()`. Вычислите метрики качества классификации. Сравните их с оценками качества классического дерева решений (`DecisionTreeClassifier`, данные ваши). Сравните с оценками качества метода k-ближайших соседей (KNN, данные ваши).

Задание 3. Создайте программу для выполнения бустинг -классификации средствами класса `sklearn.ensemble.AdaBoostClassifier()`. Вычислите метрики качества классификации. Сравните их с оценками качества классического дерева решений (`DecisionTreeClassifier`).

Задание 4. Создайте программу для выполнения бустинг -классификации средствами класса `sklearn.ensemble.GradientBoostingClassifier()`. Вычислите метрики качества классификации. Сравните их с оценками качества классического дерева решений (`DecisionTreeClassifier`).

Задание 5. Создайте программу для выполнения бустинг -классификации средствами класса `xgboost.XGBClassifier()`. Вычислите метрики качества классификации. Сравните их с оценками качества классического дерева решений (`DecisionTreeClassifier`).

Лабораторное занятие № 20-21. Искусственные нейронные сети.

Цель занятия: Реализация алгоритмов простых искусственных нейронных сетей (ANN) средствами библиотек Python.

Задание 1. Разработайте модель ИНС алгоритмом `Perceptron` для реализации бинарной классификации. Исходные данные – ИРИСЫ

Задание 2. Реализовать модель однослойного персептрона для ваших обучающих данных. Оценить качество для разных значений параметров модели.

Задание 3. Реализовать многослойный персептрон алгоритмом `MLPClassifier` для реализации обучения операции XOR. Создать два скрытых слоя, число нейронов для которых подобрать, сравнивая точности. Пример: 1-й скрытый слой имеет 4 нейрона, 2-й скрытый слой имеет 2 нейрона.

Задание 4. Обучить модель задания 3 вычислению OR и AND.

Задание 5. Разработайте модель ИНС алгоритмом `MLPClassifier` для реализации бинарной классификации. Исходные обучающие данные – ваши. Выведите значения весовых матриц синапсов.

Задание 6. Разработайте модель ИНС алгоритмом `MLPClassifier` для реализации многоклассовой классификации. Исходные данные - ИРИСЫ (три класса).

Задание 7. Разработайте модель ИНС алгоритмом `MLPClassifier` для реализации многоклассовой классификации. Исходные обучающие данные - изображения цифр библиотеки MNIST. Выполнить распознавание 1-2 цифр по обученной модели (нарисуйте цифры).

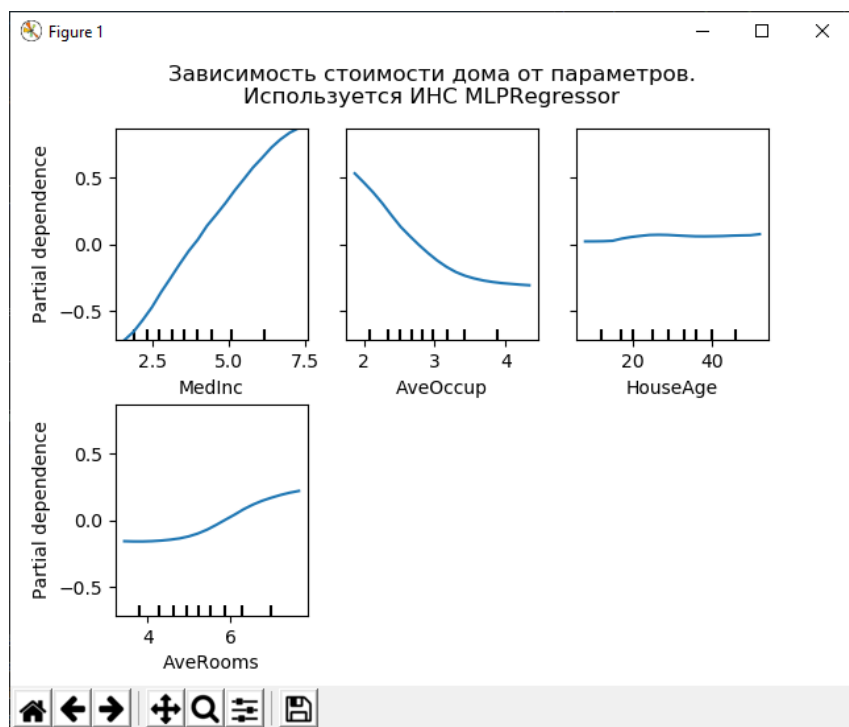
Задание 8. Разработайте модель ИНС алгоритмом `MLPClassifier` для реализации регрессии. Обучающие данные – файл цен на жилье в Калифорнии

https://scikit-learn.org/stable/auto_examples/inspection/plot_partial_dependence.html#california-housing-data-preprocessing

Файл содержит переменные (параметры регрессии) 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`.

Целевые (Target) переменные содержат цены на дома в `ln` форме (медианной). Используйте функцию `exp`, чтобы инвертировать `ln` и получить цену дома. Оцените значение влияющих на цену параметров. Для этого

выполните вывод и анализ графиков частичной зависимости, которые показывают зависимость между целевой функцией (стоимость дома) и набором «целевых» признаков. Пример вывода графиков:



Лабораторное занятие № 22. Фреймворк глубокого обучения TensorFlow

Цель занятия: разработка программ для обучения нейронной сети распознаванию изображений.

Задание 1. Возьмите функцию от 1 или 2 переменных, для которой вы умеете находить производные и их значения в 2-3 точках. Вычислите значения вручную и используя TensorFlow. Сравните.

Задание 2. Возьмите функцию от 1 переменной, для которой вы умеете находить минимум. Вычислите значения вручную и используя TensorFlow. Сравните.

Задание 3. Разработайте python-программу вычисления умножения матрицы на вектор, используя библиотеку TensorFlow. Испытайте на разных размерностях исходных данных.

Задание 4. Разработайте python-программу вычисления умножения матрицы на матрицу, используя библиотеку TensorFlow. Испытайте на разных размерностях исходных матриц.

Задание 5. Разработайте python-программу линейной регрессии, используя нейронную сеть средствами библиотеки TensorFlow. Входные данные – сведения о выпускниках колледжа. Он содержит 6 непрерывных и 1 двоичную функцию, составляющую в общей сложности 7 функций. Ожидаемый результат - это шанс поступления в вуз для выпускника. Это ваша целевая переменная.

Лабораторное занятие № 23-24. Библиотека глубокого обучения Keras для работы с ИНС.

Задание 1. Разработать python-программу вычисления температуры по Фаренгейту, зная температуру по Цельсию, используя ИНС, созданную библиотекой keras из TensorFlow (Не использовать известную формулу зависимости $F = C \times 1,8 + 32$)

Задание 2. Поменяйте в нейронной сети задания 1 функцию оптимизации (<https://keras.io/optimizers/>). Испытайте сеть и сравните результаты.

Задание 3. Обучите сеть, созданную в задании 1, другой функции (выберите самостоятельно).

Задание 4. 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) ^ 2)
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 5. 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 5. Разработать python-программу обучения регрессии, используя ИНС, созданную библиотекой keras из TensorFlow. Выполните прогноз цены квартиры. Выборка исследования: архив объявлений о продаже квартир в Москве за несколько лет - данные (например, moscow_dataset_2020.csv - 31191 записей) сервиса Яндекс. Недвижимость(). Они 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. Результаты: Оценки качества для тестовой выборки + Цена квартиры как функция от этажности, этажности общей, общей площади, площади кухни. Целевая переменная (непрерывная): price – цена квартиры.

Построение регрессионной модели выполняется функцией Sequential с двумя плотно соединенных скрытых слоев и выходной слой, который возвращает единственное, непрерывное значение.

Тренировка модели: обучается модель на 1000 эпохах.

Для оценки точности модели использовать простой метод перекрестной проверки с удержанием: 80% данных являются данными обучения, а 20% - данными испытаний.

Лабораторное занятие № 25-26. Библиотека глубокого обучения Keras для распознавания изображений.

Цель занятия: разработка программ для обучения нейронной сети распознаванию изображений.

Задание 1. Разработать программу для обучения нейронной сети распознаванию рукописных

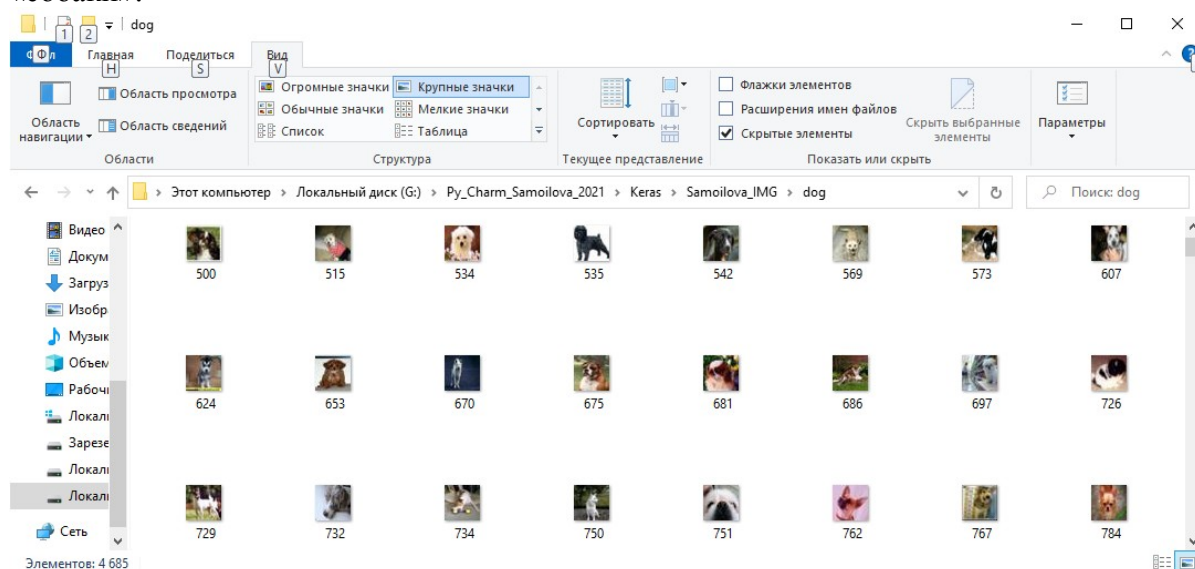
цифр из библиотеки MNIST. Сеть имеет 3 слоя – входной, скрытый и выходной. Сохраните обученную модель на диск.

Задание 2. Разработать программу, которая по обученной и сохраненной модели (задание 1) распознает рукописные цифры. Распознайте 1) 3-4 изображения из библиотеки MNIST; 2) нарисованное вами изображение рукописной цифры размера 28*28.

Задание 3. Разработать программу для обучения нейронной сети распознаванию модных товаров из библиотеки Fashion-MNIST. Сеть имеет 3 слоя – входной, скрытый и выходной. Сохраните обученную модель на диск.

Задание 4. Разработать программу, которая по обученной и сохраненной модели (задание 3) распознает модную одежду. Распознайте 1) 3-4 изображения из библиотеки Fashion-MNIST; 2) нарисованное вами изображение модной одежды размера 56*56.

Задание 5. Разработать программу для обучения нейронной сети распознаванию изображений 10 классов. 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.

Задание 6. Разработать программу, которая по обученной и сохраненной модели (задание 5) распознает изображения. Распознайте ваше изображение размера 32*32.

Лабораторное занятие № 27. Обучение рекуррентных нейронных сетей.

Цель занятия: разработка программ для обучения рекуррентной нейронной сети.

Задание 1. Разработать программу для обучения рекуррентной нейронной сети прогнозированию временного ряда, соответствующего продажам акций компании. Сеть обязательно имеет LSTM, выпадающий и выходной слои. Сравните прогнозируемые и реальные тестовые данные.

Пример данных временного ряда возьмите из сети интернет, например, сведения о продажах акций компании TSLA(сохраните его как TSLA.csv).

Задание 2. Разработать программу для обучения рекуррентной нейронной сети (библиотека Keras) прогнозированию временного ряда, соответствующего продажам конкретного товара (например, шампуня). Сеть обязательно имеет LSTM и выходной слои. Сравните прогнозируемые и реальные тестовые данные.

Задание 3. Разработать программу для обучения RNN (библиотека ARIMA) прогнозированию временного ряда, соответствующего продажам шампуня (файл с данными из задания 2). Сравните прогнозируемые и реальные тестовые данные.

Задание 4. Разработать программу для обучения рекуррентной нейронной сети распознаванию рукописных цифр из библиотеки MNIST. Сеть обязательно имеет LSTM, выпадающий и

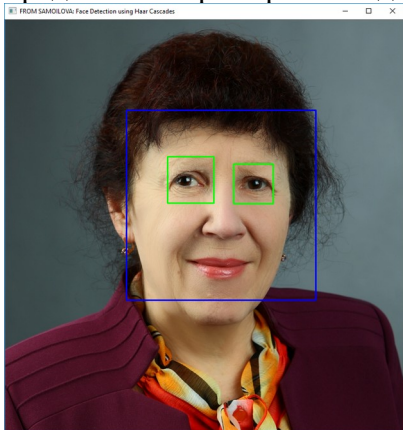
выходной слой. Оцените точность модели и сравните с точностью ИНС, обученных ранее. Сохраните обученную модель на диск.

Задание 5. Разработать программу, которая по обученной и сохраненной модели (задание 4) распознает рукописные цифры. Распознайте 1) 3-4 изображения из библиотеки MNIST; 2) нарисованное вами изображение рукописной цифры размера 28*28.

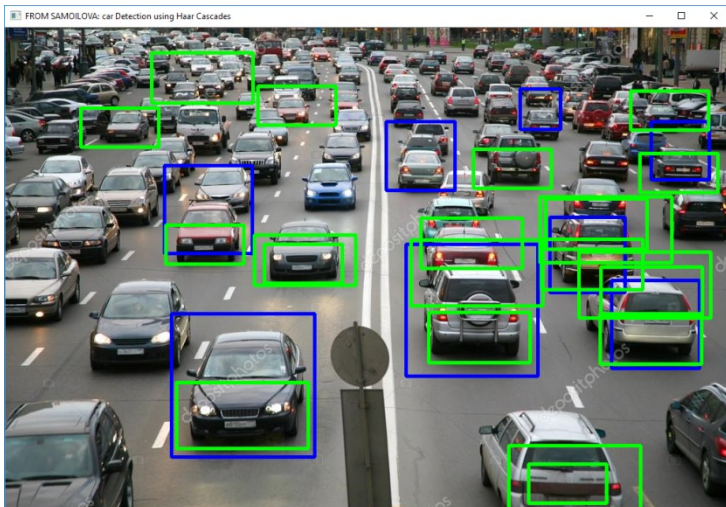
Лабораторное занятие № 28. Обнаружение объектов на изображении.

Цель занятия: разработка программ для обнаружения объектов на изображении.

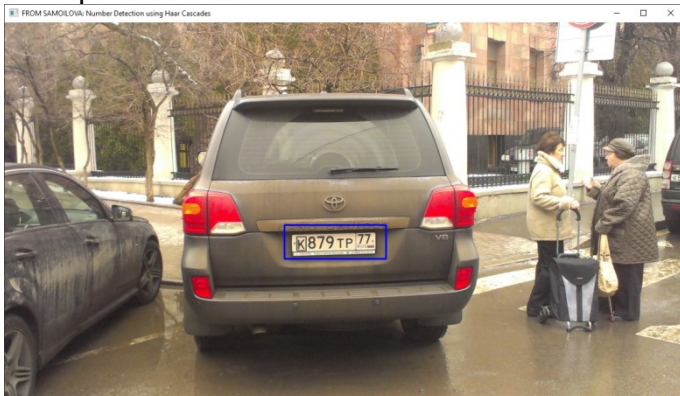
Задание 1. Используя Python-библиотеки распознавания образов (OpenCV), разработать модель распознавания элементов изображений алгоритмом Виолы-Джонса. XML - файл диаграммы Хаара должен соответствовать вашему варианту распознаваемого элемента (автомобиль, пешеход, номер автомобиля, лицо, нос, уши, голова, глаза, торс, полный рост и т.п.). Далее представлен примерный вид формы разработанного приложения (лицо - глаза):



Автомобили:

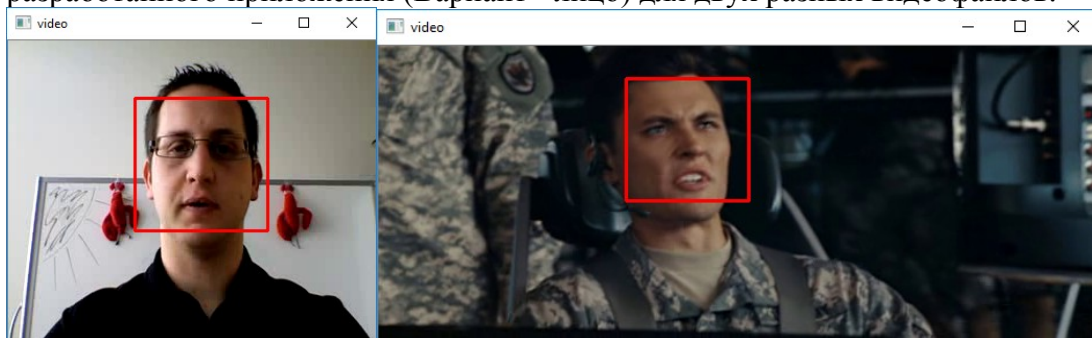


Номера автомобилей:

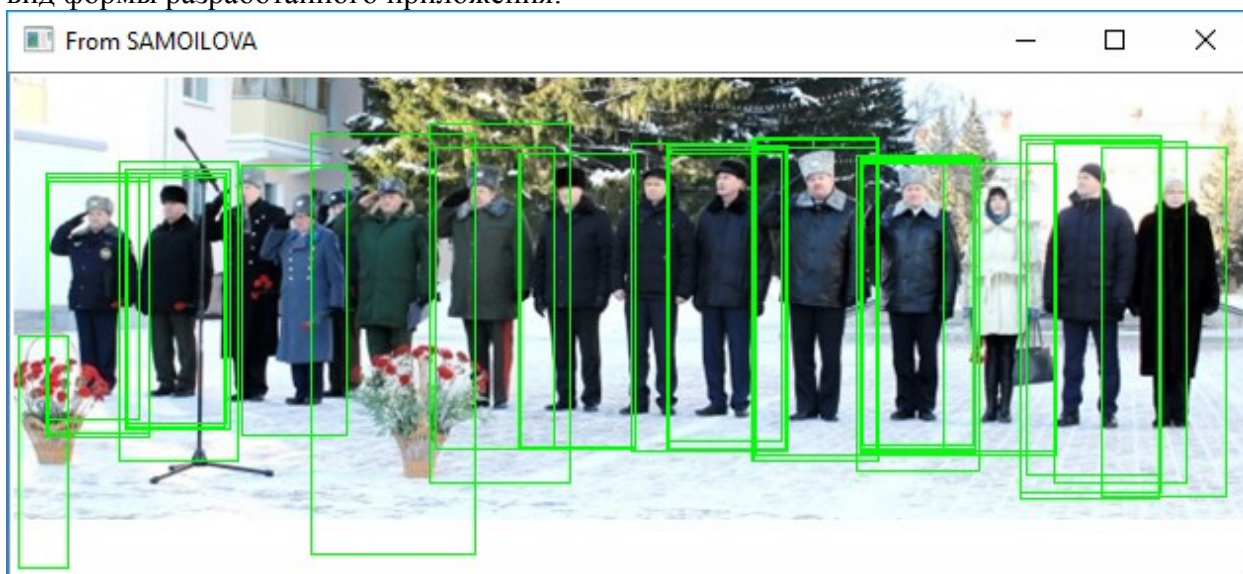


Задание 2. Используя Python-библиотеки распознавания образов, разработать модель распознавания кадров видеопотока алгоритмом Виолы-Джонса. XML - файл диаграммы Хаара

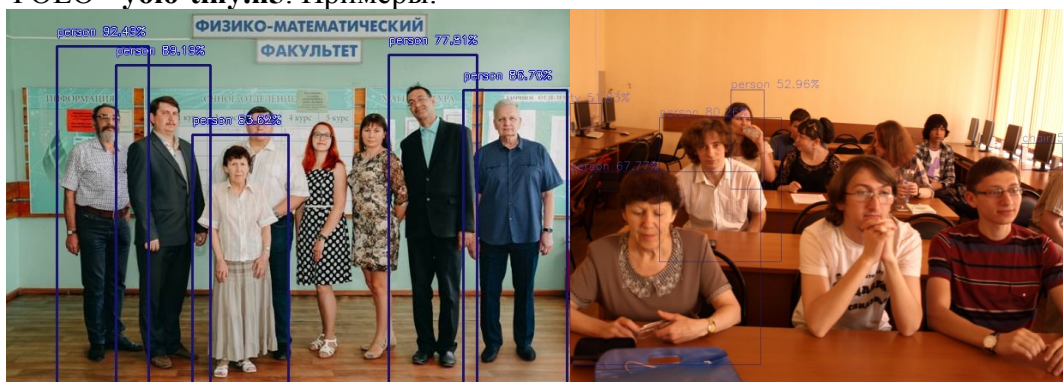
должен соответствовать вашему варианту распознаваемого элемента (нос, уши, голова, глаза, торс, полный рост и т.п.). Далее представлен примерный вид кадров распознавания разработанного приложения (Вариант - лицо) для двух разных видеофайлов.



Задание 3. Используя python-библиотеки opencv распознавания образов и алгоритм HOG, разработать модель распознавания пешеходов на изображении. Далее представлен примерный вид формы разработанного приложения:



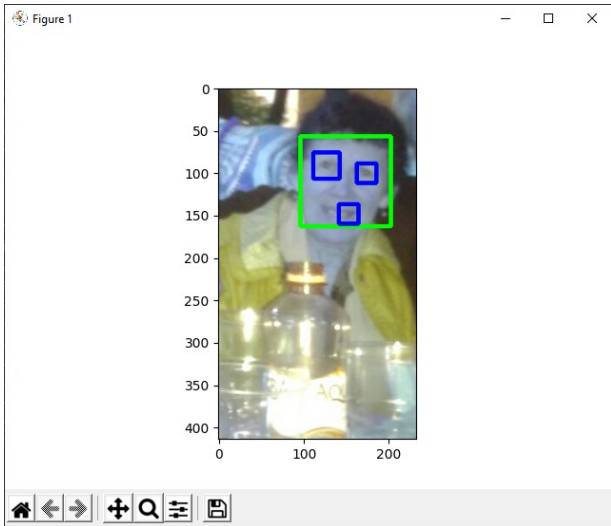
Задание 1. Используя Python-библиотеку ImageAI, разработать программу обращения к обученной модели обнаружения элементов изображения алгоритмом глубокого обучения YOLO - yolo-tiny.h5. Примеры:



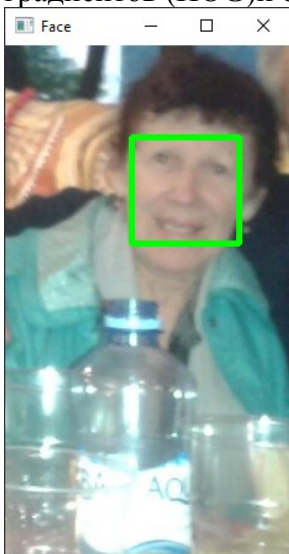
Лабораторное занятие № 29. Обнаружение и верификация лиц.

Цель занятия: разработка программ для обнаружения и распознавания лиц с использованием современных алгоритмов машинного обучения.

Задание 1. Используя XML-файлы каскадов Хаара для распознавания лиц, глаз, ушей и пр., разработать модель обнаружения лица по фото (взять свое). Пример:



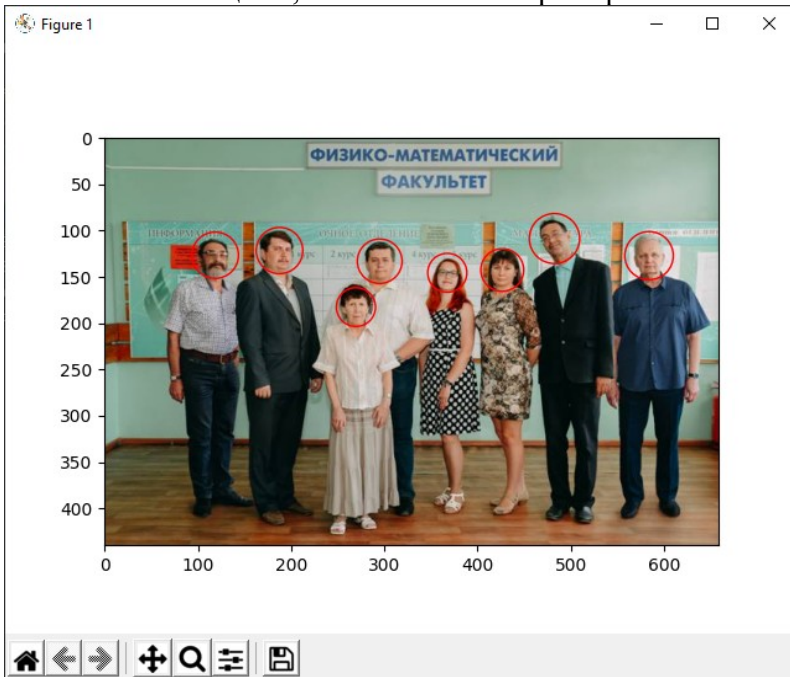
Задание 2. Используя OpenCV и библиотеку **dlib** с фронтальным детектором лица, разработать модель обнаружения лица. Детектор основан на гистограмме ориентированных градиентов (HOG) и обученной линейной SVM. Фото взять свое. Пример:



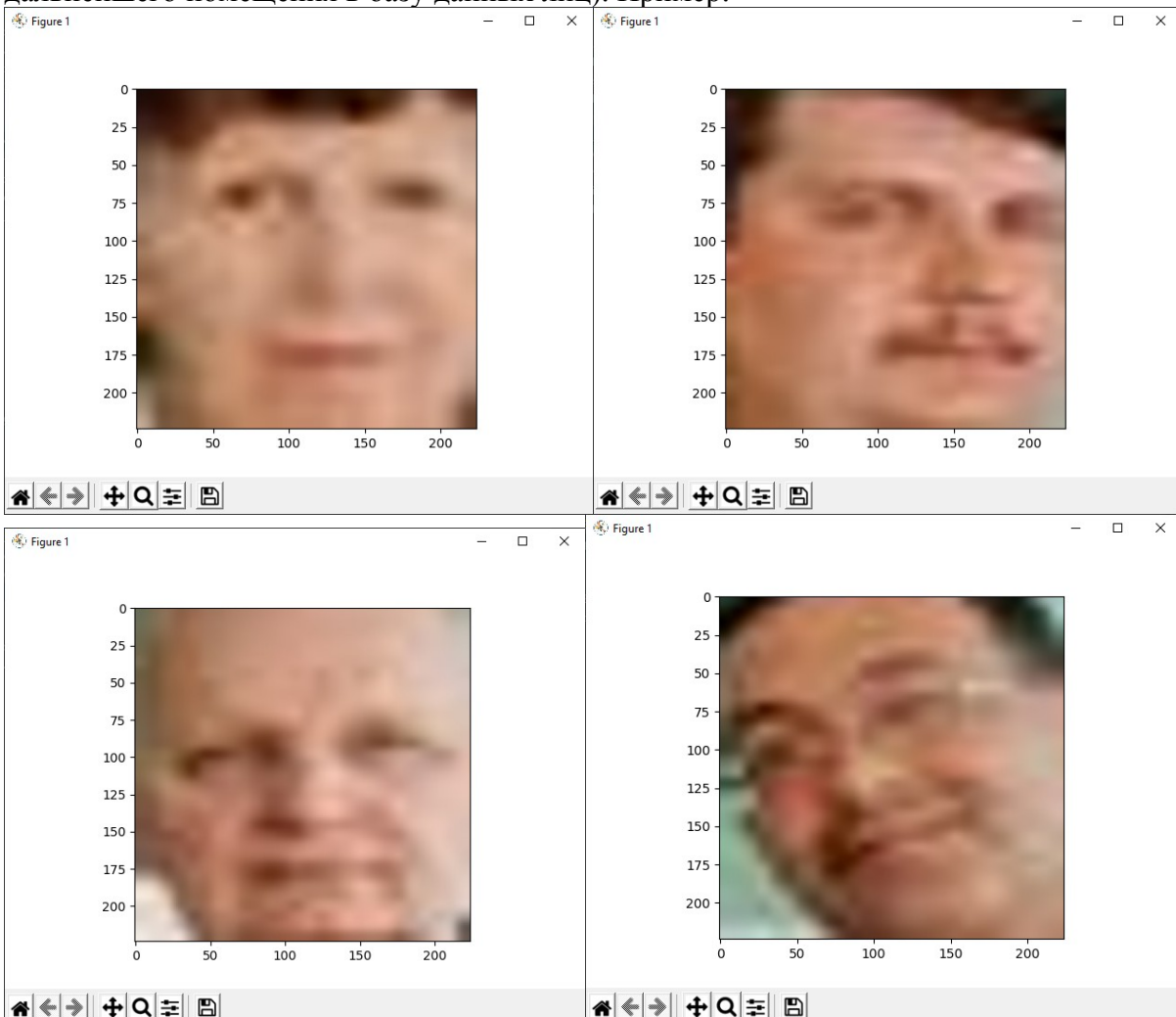
Задание 3. Используя OpenCV и библиотеку **dlib** с CNN детектором лица, разработать модель обнаружения лица в видеопотоке. Детектор основан на обученной сверточной нейронной сети. Файл *.avi найти в интернете. Пример:

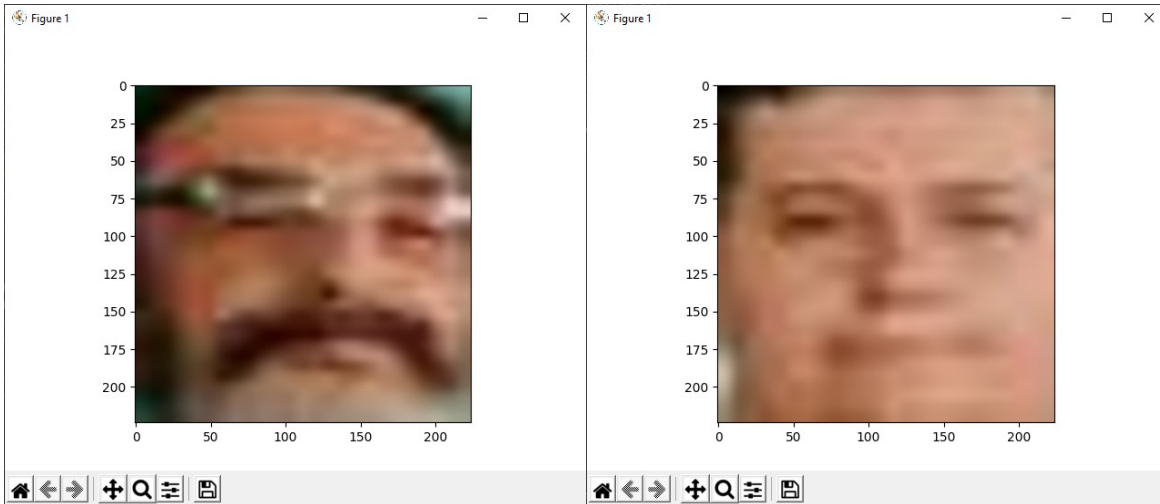


Задание 4. Используя многозадачную каскадную сверточную нейронную сеть и библиотеку MTCNN с CNN детектором лица, разработать модель обнаружения лица. Взять фото с несколькими лицами, включая свое. Пример:

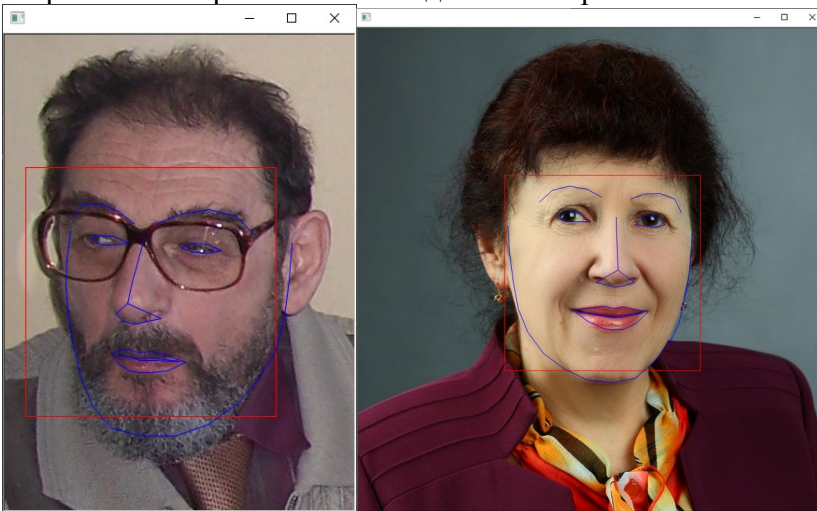


Задание 5. Добавить к модели обнаружения лиц фрагмент кода выделения каждого лица (для дальнейшего помещения в базу данных лиц). Пример:





Задание 6. Используя библиотеку `dlib` с выделением 68 характерных точек лица, сравнить фотографии двух лиц. Вычислить меру схожести по Евклиду. Метод основан на обученной сверточной нейронной сети. Одно из изображений – ваше лицо. Примеры сравниваемых лиц:



Задание 7. Используя Python- библиотеку `deephface`, сравнить фотографии двух лиц. Определить пол, возраст, настроение и расу каждого из лиц. В основе используемых алгоритмов - обучение сверточной нейронной сети. Одно из изображений – ваше лицо.

Лабораторное занятие № 30. Разработка веб-сервиса Django для реализации алгоритмов обнаружения объектов на изображении средствами модели сверточной нейронной сети VGG16.

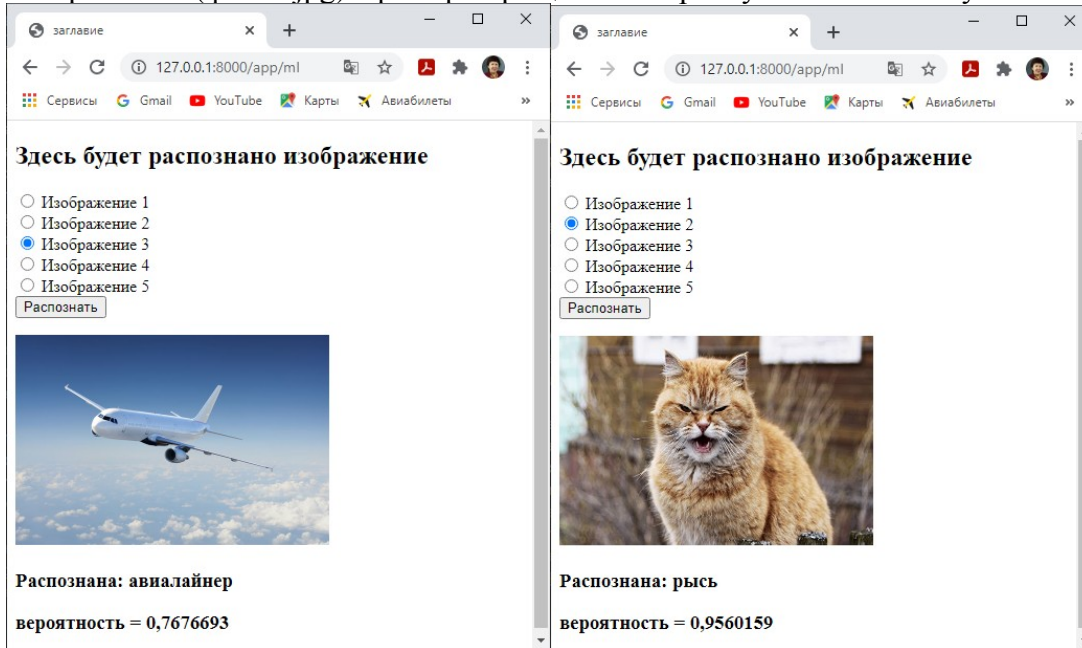
Цель занятия: Реализация веб - сервисов машинного обучения для обнаружения объектов на изображении средствами фреймворка Django.

Задание1. Средствами Python используйте обученную модель VGG16 для распознавания изображений (файлы.jpg). Примеры файлов изображений:





Задание 2. Создайте веб-сервис, использующий обученную модель VGG16 для распознавания изображений (файлы jpg) Пример обращения к сервису машинного обучения из веб-страницы:



Лабораторное занятие № 31. Веб-сервисы машинного обучения для распознавания речи.

Цель занятия: Освоение приемов работы по организации машинного обучения для распознавания речи с использованием библиотек Python.

Задание 1. Разработайте веб-сервис для распознавания речи, используя функцию `recognize_google()`. Источник речи – микрофон. Сохраните различные аудио в файлах wav. Испытайте приложения на записях разной длины, вычисляя процент ошибок в распознавании (WER).

Задание 2. Создайте веб-сервис для распознавания речи, используя функцию `recognize_google()`. Источник речи – файл wav. Испытайте приложения на разных файлах, созданных в задании 1.

Лабораторное занятие № 32. Веб-сервисы машинного обучения для синтеза речи.

Цель занятия: Освоение приемов работы по организации машинного обучения для синтеза речи с использованием веб-сервиса для синтеза речи, используя функцию `gTTS()`. Сохраните аудио в файле MP3.

Задание 2. Создайте веб-сервис для синтеза речи, используя функцию `pyttsx3`. Сохраните аудио в файле MP3 или WAV.

Задание 3. Создайте веб-сервис для синтеза речи, используя функцию `pyttsx3`. Измените параметры голоса (скорость, мужской-женский, громкость). Сохраните аудио в файле MP3 или WAV.

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 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

Зачетная контрольная работа (пример задания)

1. Создайте веб-сервис, использующий обученную модель VGG16 для распознавания конкретного изображения (файл.jpg).
2. Методом деревьев решений выполните обучение модели классификации на конкретных индивидуальных данных. Оцените качество модели.

Критерии оценивания зачетной контрольной работы

1. Нормы оценивания работы

| № п/п | Структурная часть контрольной работы | Количество баллов (*) |
|-------|---|-----------------------|
| 1 | Правильно реализован каждый метод решения | 1 балл |
| 2 | Анализ результатов | 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 |

Критерий получения зачета

Зачет выставляется по результатам работы студента в течение семестра согласно Положению о текущем контроле успеваемости и промежуточной аттестации обучающихся в федеральном государственном бюджетном образовательном учреждении высшего профессионального образования «Смоленский государственный университет» (утверждено приказом ректора № 01-113 от 26.09.2019 г.; внесены дополнения приказом ректора № 01-48 от 30.04.2020).

Для получения зачета студент должен:

- уметь отвечать на теоретические вопросы, рассмотренные на лекциях;
- уметь решать задачи, предложенные на лабораторных занятиях;
- уметь решать задачи, предложенные на зачетной контрольной работе.

7. Перечень основной и дополнительной учебной литературы

7.1. Основная литература

1. Анализ данных : учебник для академического бакалавриата / В. С. Мхитарян [и др.] ; под ред. В. С. Мхитаряна. — М. : Издательство Юрайт, 2021. — 490 с. — (Серия : Бакалавр. Академический курс). — ISBN 978-5-534-00616-2. — Режим доступа: www.biblio-online.ru/book/CC38E97A-CCE5-4470-90F1-3B6D35ACC0B4.
2. Грекул, В. И. Алгоритмы машинного обучения : учебник и практикум для академического бакалавриата [электронный ресурс] / В. И. Грекул, Н. Л. Коровкина, Г. А. Левочкина. — М. :

Издательство Юрайт, 2019. — 385 с. — (Серия : Бакалавр. Академический курс). — ISBN 978-5-9916-8764-5. URL:<https://biblio-online.ru/book/B56731F0-5408-4182-8607-92ACE5A8D7BE>.

7.2. Дополнительная литература

1. Флах, П. Машинное обучение. Наука и искусство построения алгоритмов, которые извлекают знания из данных [Электронный ресурс] / П. Флах. — Электрон. дан. — Москва: ДМК Пресс, 2019. — 400 с. — Режим доступа: <https://e.lanbook.com/book/69955>.
2. Миркин, Б. Г. Введение в анализ данных : учебник и практикум / Б. Г. Миркин. — М. : Издательство Юрайт, 2020. — 174 с. — (Серия : Авторский учебник). — ISBN 978-5-9916-5009-0. — Режим доступа: www.biblio-online.ru/book/46A41F93-BC46-401C-A30E-27C0FB60B9DE

7.3. Перечень ресурсов информационно-телекоммуникационной сети «Интернет»

1. <http://www.machinelearning.ru/> - Профессиональный информационно-аналитический ресурс, посвященный машинному обучению, распознаванию образов и интеллектуальному анализу данных.
2. Воронцов К.В. Машинное обучение Школа Анализа данных Яндекс. МФТИ. национальный открытый университет Интуит, 2019 – Режим доступа: свободный – URL: <https://www.intuit.ru/studies/courses/13844/1241/info>
3. <http://www.niisi.ru/iont/ni> - Российская ассоциация нейроинформат
4. Система дистанционного обучения СмолГУ (moodle.smolgu.ru).
5. Интернет-Университет Информационных Технологий, <http://www.intuit.ru>

8. Материально-техническое обеспечение

Для занятий необходимы:

1. проектор;
2. интерактивная доска;
3. персональные компьютеры.

Для самостоятельной работы подготовлены аудитории № 224, 226, 230, 234 с выходом в Интернет, оснащенные компьютерами IBMPCс процессорами IntelCore 7 и оперативной памятью не менее 16 ГБ.

9. Программное обеспечение

1. Python 3.9.5; среда разработки приложений PyCharm

ДОКУМЕНТ ПОДПИСАН
ЭЛЕКТРОННОЙ ПОДПИСЬЮ

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