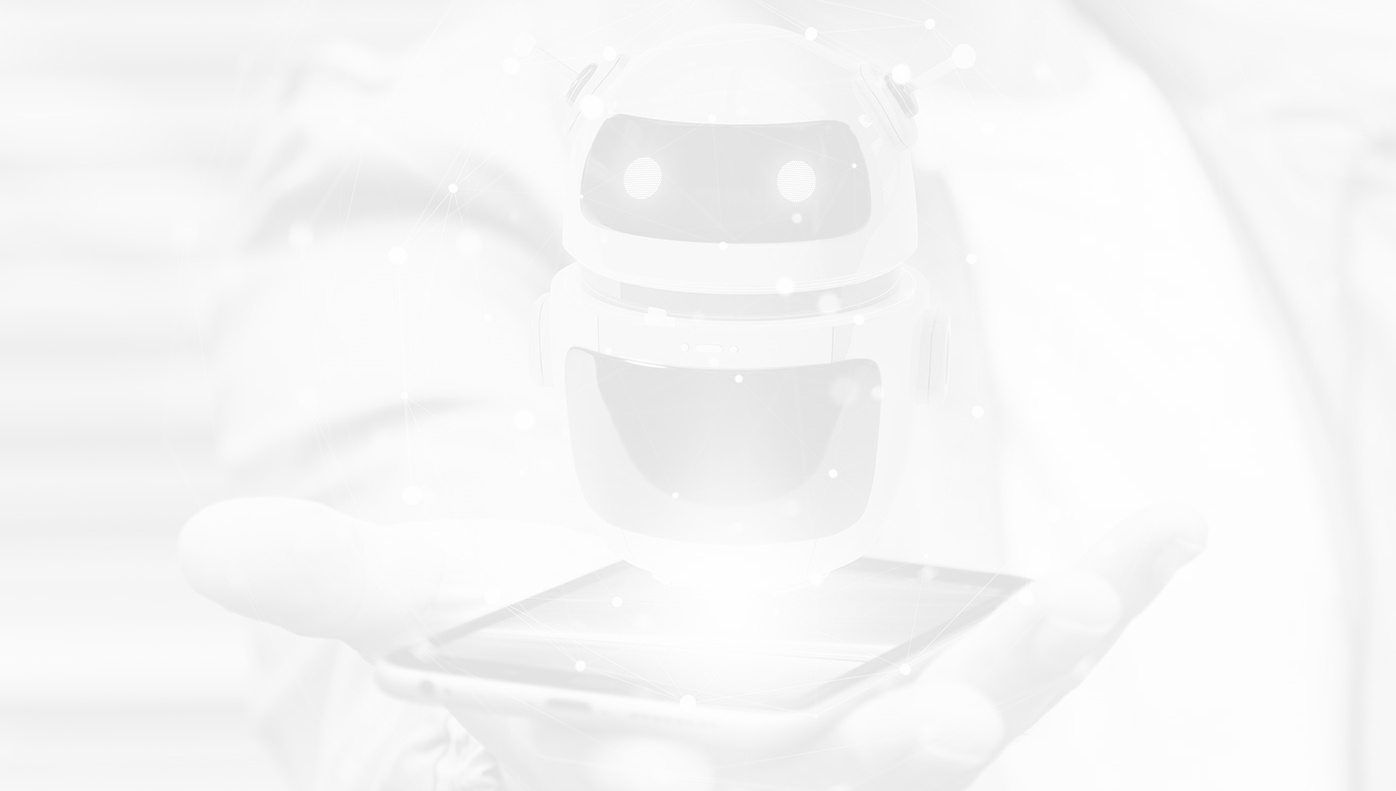
****

***Курс***

***«Машинное обучение для школьников»***

**ЛАБОРАТОРИЯ**

**НАНОСЕМАНТИКА**

^

*Руководитель и основные исполнители:*

Павел Сухачев, куратор, Наносемантика

Дмитрий Пантюхин, старший преп. НИУ ВШЭ, преп. МФТИ

Карина Садова, инженер, Наносемантика

**МАШИННОЕ ОБУЧЕНИЕ**

для школьников

10 класс, 64 часа.

Курс «Машинное обучение для школьников» предназначен для учащихся 10-11 классов в качестве основного или дополнительного модуля информатики.

Изучаются методы работы с данными, методы машинного обучения и нейронных сетей.

Курс практический и представлен в виде «тетрадок» Jupyter Notebook на языке Python, содержащих теоретическую и программную часть.

Для изучения курса необходимо:

- знать базовые конструкции языка Python, уметь устанавливать, запускать «тетрадки» Jupyter Notebook, необходимые библиотеки Python (при работе со своего компьютера). Рекомендуется установить расширения nbextension для Jupyter Notebook.

- знать английский язык для чтения документации, уметь находить необходимую информацию в документации, сети Интернет.

- для работы с видео и звуком нужны установленные микрофон, динамики и web-камера.

**СОДЕРЖАНИЕ КУРСА**

\* ТК – текущий контроль, задания.

\* ДЗ – домашнее задание

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **№ п\п** | **Модуль** | **Наименование раздела** | **Коли-чество часов** | |
| ***1 полугодие 10 класса*** | | | | | |
| 01 | Работа с данными в Python | Математика в Python. Библиотеки *math, numpy, random*. ТК\*. ДЗ\* | 2 | |
| 02 | Работа с табличными данными. Библиотека *pandas*. ТК | 2 | |
| 03 | Работа с текстовыми данными. Библиотеки *spacy, ntlk*. ДЗ | 2 | |
| 04 | Работа с файлами с нетривиальной разметкой, JSON и XML. Библиотека *json*. | 2 | |
| 05 | Работа с изображениями. Библиотека *pil*. ДЗ | 2 | |
| 06 | Работа с видео. Библиотека *opencv*. ТК | 2 | |
| 07 | Работа с аудио. Библиотеки *pydub, simpleaudio, pyaudio, sounddevice*. | 2 | |
| 08 | Визуализация данных в Python | Визуализация данных, графики *Matplotlib*. ТК | 2 | |
| 09 | Визуализация изображений (карт). Библиотека *folium*. ТК. | 2 | |
| 10 | Визуализация табличных данных. Графики в *pandas*. Библиотеки *seaborn*, *plotly*. ТК | 2 | |
| 11 | Визуализация звуковых данных. Библиотека *librosa.* ТК | 2 | |
| 12 | Введение в машинное обучение | Задачи машинного обучения | 2 | |
| 13 | Линейная и логистическая регрессия в библиотеке *sklearn*. ТК | 2 | |
| 14 | Деревья решений и случайный лес в библиотеке *sklearn.* Библиотека *graphviz*. ТК. | 2 | |
| 15 | Реализация итогового проекта. | 2 | |
| 16 | Презентация результатов итогового проекта | 2 | |
| ИТОГО: | | | | 32 | |
| ***2 полугодие 10 класса*** | | | | | |
| 17 | Машин-ное обучение. | Машины опорных векторов в библиотеке *sklearn*. ТК | 2 | |
| 18 | Байесовские модели в библиотеке *sklearn.* | 2 | |
| 19 | Методы ближайших соседей в библиотеке *sklearn*. ТК | 2 | |
| 20 | Ансамбли: беггинг в библиотеке *sklearn*. ТК. | 2 | |
| 21 | Ансамбли: бустинг в библиотеках *sklearn*, *xgboost*, *catboost*. ТК. | 2 | |
| 22 | Ансамбли: стекинг в библиотеке *sklearn*. | 2 | |
| 23 | Матричные разложения. Метод главных компонент, метод T-SNE и анализ независимых компонент в библиотеке *sklearn.* | 2 | |
| 24 | Кластеризация. Алгоритм K-средних, Иерархическая кластеризация, Метод DBSCAN в библиотеках *scipy* и *sklearn*. ТК. | 2 | |
| 25 | Обучение метрических преобразований, метод LMNN в библиотеке *sklearn* | 2 | |
| 26 | Подбор гиперпараметров. ТК | 2 | |
| 27 | Нейронные сети. PuzzleLib | Нейронные сети. Фреймворки глубокого обучения. Библиотека PuzzleLib | 2 | |
| 28 | Многослойный персептрон. Реализация в PuzzleLib и sklearn | 2 | |
| 29 | Играем с персептроном. Playground Tensorflow | 2 | |
| 30 | Сверточные нейронные сети. Реализация в Keras и PuzzleLib. | 2 | |
| 31 | Реализация итогового проекта. | 2 | |
| 32 | Презентация результатов итогового проекта | 2 | |
| ИТОГО: | | | | 32 | |
|  | | | |  | |

**Краткое содержание разделов**

**1 полугодие 10 класса:**

**01. Математика в Python. Библиотеки math, numpy, random.**

Основные математические операции в Python. Скаляры: сложение, вычитание, умножение, деление. Библиотека *math*: округления, модуль числа, остаток от деления, математические константы, возведение в степень, экспонента, логарифм, квадратный корень, тригонометрические функции, факториал. Библиотека *numpy*: создание массивов (скаляры, векторы, матрицы, тензоры (многомерные массивы)); поэлементные операции с массивами (сложение, вычитание, умножение, деление, возведение в степень, остаток от деления); бесконечности (Inf) и «нечисла» (NaN); тригонометрические функции; округления; агрегирующие функции (суммирование элементов массива, минимальный\максимальный элемент); индексы, срезы, итерирование массивов; изменение массивов, объединение, разбиение массивов; копии и представления (view) массивов; случайные числа, библиотеки *random* и *numpy.random*, инициализация генератора случайных чисел. Пример по формированию нужного массива. Контрольные задания и домашнее задание.

**02. Работа с табличными данными. Библиотека pandas.**

Библиотека Pandas. Последовательности (Series) и таблицы (DataFrame), создание и индексация: виды индекса, изменение индекса, названия индекса, столбцов, обращение к и изменение содержимого. Обращение по числовому индексу (iloc). Логический индекс, логические операции. Загрузка из файла. Группировка данных, методы агрегирования. Разбивка значений на интервалы (категории). Контрольные задания.

**03. Работа с текстовыми данными. Библиотеки spacy, ntlk.**

Открытие, закрытие и чтение текстовых файлов. Чтение по строкам. Форматы и режимы работы с файлами. Запись в текстовый файл. Добавление текста к файлу. Перезапись файла. Работа со строками: индексация, длина, поиск подстроки, разбиение строк, склейка строк. Продвинутые методы обработки текста, библиотека spacy: извлечение именованных сущностей. Библиотека nltk: токенизация (сегментация) по предложениям, словам; лемматизация и стемминг текста; обработка стоп-слов. Чистка текста. Домашнее задание.

**04. Работа с файлами с нетривиальной разметкой, JSON и XML. Библиотека json.**

Формат Json сериализация и десериализация. Библиотека json: чтение, парсинг. Формат XML, чтение и парсинг методами работы со строками.

**05. Работа с изображениями. Библиотека pil.**

О формировании изображений. Библиотека PIL: открытие, отображение, разделение цветовых каналов, гистограмма цветов, информация о файле изображения, обрезка, сохранение, поворот, загрузка из URL (библиотека requests), рисование графических примитивов (прямоугольник, эллипс), добавление текста; фильтры изображений (размытие, увеличение резкости, выделение контура и др.), изменение размеров. Домашнее задание.

**06. Работа с видео. Библиотека opencv.**

О представлении видео. Библиотека opencv. Работа с изображениями: чтение, отображение, замечание о несогласованности библиотек, изменение размера, поворот, преобразование цветов, размытие и сглаживание, рисование графических примитивов, добавление текста, сохранение. Работа с видео: захват видеопотока, чтение кадра, проверка чтения, освобождение потока, закрытие окон. Пример отслеживания изменений между кадрами. Контрольное задание.

**07. Работа с аудио. Библиотеки pydub, simpleaudio, pyaudio, sounddevice.**

О звуке, представлении звука в компьютере. Библиотека pydub. Загрузка звукового файла. Пример изменения звука. Запись в файл. Прослушивание (также с помощью IPython.display). Библиотека simpleaudio. Создание звукового объекта, асинхронное воспроизведение. Остановка воспроизведения. Создание простого звука. Библиотека pyaudio. Открытие, воспроизведение. Инициализация, открытие и закрытие звуковой подсистемы. Чтение файла, запись в поток. Библиотека sounddevice, запись звука. Информация о звуковых устройствах. Запись с микрофона. Прослушивание записи. Запись звука в pyaudio.

**08. Визуализация данных, графики Matplotlib.**

Библиотека Matplotlib, описание, полотна, оси, виды графиков, подграфики, элементы графиков, обращение к элементам. Примеры графиков: Stackplot; Scatter и hist на одном полотне (с заданием цветов, маркеров, текста…). Загрузка и отображение данных, библиотеки io, tarfile, и urllib, пример графика с заданием цветовой карты, логарифмическим масштабом, заголовком, цветовой палитры. Отображение матриц, пример тепловой карты. Графики в pandas. Контрольные задания.

**09. Визуализация изображений (карт). Библиотека folium.**

Библиотека folium. Источники карт, маркеры, всплывающие окна, маркеры с графиками. Слои GeoJSON/TopoJSON. Карты Choropleth. Контрольные задания.

**10. Визуализация табличных данных. Графики в pandas. Библиотеки seaborn, plotly.**

Пример таблицы для визуализации, графики pandas. Библиотека seaborn: графики попарных зависимостей значений одного столбца от другого, гистограмма столбца, график «ящик с усами», таблица «тепловая карта». Библиотека plotly для интерактивных графиков: примеры дискретных, столбчатых графиков, «ящик с усами». Контрольные задания.

**11. Визуализация звуковых данных. Библиотека librosa.**

Библиотека librosa: временной график, пример с добавлением шума. Частотное представление и спектр: преобразования в спектр и временное представление, отображение спектров. Контрольные задания.

**12. Задачи машинного обучения**

(Теоретический). Модели физические и математические, параметрические модели. Обучение, машинное обучение. Ошибка модели. Тестирование и обобщающая способность. Параметры и гиперпараметры модели. О приближенности результатов. Данные, требования к данным. Признаки, виды признаков. Виды обучения: с учителем, без учителя, с подкреплением. Способы организации обучения: динамическое (онлайн), отложенное (оффлайн), пакетное, активное. Задачи машинного обучения: классификация, кластеризация, регрессия, прогнозирование, оптимизация и др. О функционалах ошибки и метриках. Некоторые метрики для задач классификации. О моделях.

**13. Линейная и логистическая регрессия в библиотеке sklearn.**

Модель линейной регрессии. Обучение линейной регрессии - итерационный метод наименьших квадратов. Градиентный спуск. Пример (ручной) генерации данных, разделения на обучающее и тестовое множества, инициализации и обучения модели, проверки и отображения результатов. Библиотека sklearn. Пример линейной регрессии в библиотеке sklearn. Модель логистической регрессии: «сигмоида», функция ошибки, пример в sklearn. Понятие разделяющей поверхности, пример. Бинарное (One-hot) кодирование. Контрольные задания.

**14. Деревья решений и случайный лес в библиотеке sklearn. Библиотека graphviz**

Дерево принятия решений, правила если-тогда. Обучение (создание) дерева методом CART, критерии (неопределенность Джини, энтропия, ошибка классификации, среднеквадратичная ошибка). Реализация в sklearn, обучение, тестирование и визуализация. Библиотека graphviz для визуализации деревьев. Пример разделяющей поверхности для дерева. Случайный лес, реализация в sklearn, оценка важности признаков. Пример случайного леса на синтетических данных. Пример на другом наборе данных, матрица ошибок. Контрольные задания.

**15. Реализация итогового проекта.**

Проект с предсказанием стоимости подержанного автомобиля методами регрессии.

Варианты:

а) индивидуальный – каждый обучающийся выполняет проект самостоятельно, заполняя пропуски в прездаполненном файле-задании. Учитель проверяет правильность.

б) групповой – группы обучающихся соревнуются между собой за достижение лучшего результата, изменяя параметры методов, представленные в файле-решении. Оценка по занятому месту.

**16. Презентация результатов итогового проекта**

Обучающиеся презентуют свои решения, сравнивают с предложенным в файле-решении.

**2 полугодие 10 класса:**

**17. Машины опорных векторов в библиотеке sklearn.**

Линейные машины опорных векторов: опорные вектора, функция ошибки, реализация и пример в sklearn. Пример многоклассовой классификация на наборе данных. Нелинейные машины опорных векторов, понятие ядра. Примеры разделяющих поверхностей для разных ядер. Контрольные задания

**18. Байесовские модели в библиотеке sklearn.**

О случайных процессах и вероятности, условная вероятность. Теорема Байеса, вероятности независимых событий. Пример расчета вероятностей. Наивный Байесовский классификатор. Реализация в sklearn: реализация Гауссовского наивного Байесовского классификатора, пример разделяющей поверхности, другие реализации.

**19. Методы ближайших соседей в библиотеке sklearn.**

Метод К-ближайших соседей, содержание, недостатки, реализация в sklearn, пример разделяющей поверхности. Пример с дорисовкой закрытой части лица. Метод ближайших соседей с ограничением по радиусу, разделяющая поверхность. Контрольные задания

**20. Ансамбли: беггинг в библиотеке sklearn.**

Ансамбли моделей. Беггинг. Методе бутстрэпа. Ручной пример реализации бутстрэпа. Реализация беггинга в sklearn, пример беггинга деревьев с изучением влияния на смещение-дисперсию результатов. Пример разделяющей поверхности. Контрольные задания.

**21. Ансамбли: бустинг в библиотеках sklearn, xgboost, catboost**

Бустинг. Алгоритм Adaboost и реализация в sklearn, пример с деревьями и машинами опорных векторов, разделяющие поверхности. Градиентный бустинг Xgboost, алгоритм и реализация в sklearn, библиотека xgboost, пример. Алгоритм Catboost и реализация в библиотеке Catboost, кроссвалидация, отображение процесса обучения. Контрольные задания.

**22. Ансамбли: стекинг в библиотеке sklearn.**

Стекинг, реализация в sklearn, пример стекинга известных классификаторов, разделяющая поверхность.

**23. Матричные разложения. Метод главных компонент, метод T-SNE и анализ независимых компонент в библиотеке sklearn.**

О проекциях. Метод главных компонент, идея, визуализация. Пример ручного расчета главных компонент. Реализация и пример в sklearn. Пример классификации с использованием метода главных компонент. Пример с рукописными цифрами из MNIST. Метод T-SNE, пример.

Анализ независимых компонент, сходство и отличие от метода главных компонент. Реализация и пример в sklearn.

**24. Кластеризация. Алгоритм K-средних, Иерархическая кластеризация, Метод DBSCAN в библиотеках scipy и sklearn.**

Задача кластеризации, отличия от классификации. Алгоритм K-средних, самостоятельная реализация (с использованием функций библиотеки scipy) и реализация в sklearn. Пример кластеризации и метрик кластеризации. Примеры других методов кластеризации. Иерархическая кластеризация, алгоритм и реализация в scipy, пример. Метод кластеризации на основе плотности DBSCAN, алгоритм, реализация и пример в sklearn. Контрольные задания

**25. Обучение метрических преобразований, метод LMNN в библиотеке sklearn**

О «расстояниях». Реализация и пример в sklearn. Метод LMNN, идея, реализация и пример в sklearn.

**26. Подбор гиперпараметров**

О проблеме подбора гиперпараметров. Случайный поиск и поиск по сетке, реализация и пример в sklearn. Пример со сравнением классификаторов на нескольких наборах данных. Контрольные задания.

**27. Нейронные сети. Фреймворки глубокого обучения. Библиотека PuzzleLib**

Нейроны и нейронные сети. О глубоком обучении. О графических ускорителях. О фреймворках глубокого обучения, автодифференцирование, пример графа вычислений в PyTorch. О библиотеке PuzzleLib.

**28. Многослойный персептрон. Реализация в PuzzleLib и sklearn**

Многослойный персептрон. Реализация в sklearn. Реализация средствами PuzzleLib**.** Разделяющая поверхность.Пример задачи регрессии многослойным персептроном. Контрольные задания.

**29. Играем с персептроном. Playground Tensorflow**

Исследование влияния различных гиперпараметров персептрона в <https://playground.tensorflow.org> (рекомендуется совместить с Темой 28)

**30. Сверточные нейронные сети. Реализация в PuzzleLib.**

О сверточных нейронных сетях. Реализация в Keras и PuzzleLib. Контрольные вопросы. Пример с распознаванием рукописных цифр.

**31. Реализация итогового проекта.**

Hello world в нейронных сетях: распознавание типа одежды по изображению.

**32. Презентация результатов итогового проекта**

**Порядок проведения занятий**

Рекомендуется проводить занятия в порядке нумерации уроков. В случае сложностей освоения допускается опускать некоторые темы (например работа со звуком, если таковая не предполагается или отсутствует техническая возможность воспроизводить и записывать звук, сложные темы и т.п.).

Время изучения темы преподаватель регулирует самостоятельно, в зависимости от ожидаемого уровня освоения. Предполагается три уровня освоения:

- Начальный – ученик способен читать материалы, запускать коды, наблюдать решения и объяснять их, не внося модификации в код. Рекомендуемая оценка не выше «хорошо».

- Основной – ученик способен читать материалы, запускать коды, наблюдать решения, вносить необходимые изменения в код по аналогии (по заданиям из материалов, заданиям учителя или самостоятельно).

- Продвинутый – ученик способен читать материалы, запускать коды, наблюдать решения, вносить необходимые изменения по аналогии и предлагать свои решения: изменять постановки задач, изменять данные, изменять методы решения, может объяснить результаты. Предполагается интенсивная самостоятельная работа (чтение документации, эксперименты с кодом).