

Разработка факультатива

на примере курса: «Оракул на питоне»

Цель курса

- Зачем я делаю этот курс?
- Зачем школьник придет на мой курс?

Плохая формулировка 👎

Рассказать школьникам как разрабатывать ИИ

Хорошая формулировка 👍

Научить школьников разрабатывать простые программы, которые смогут прогнозировать различные значения (цены, результаты или рейтинги) на основе данных.

Что нужно для достижения цели?

Что должен уметь и знать школьник, чтобы разрабатывать простые программы для прогнозирования?

- **Что включает в себя итоговый навык?** (ключевые этапы разработки программы):
 - Первичный анализ и предобработка данных
 - Выбор и обучение модели
 - Анализ и оценка качества работы
 - Визуализация результатов
- **Какой инструментарий используется?** Навыки и знания, на прямую не относящиеся к тематике курса.
(какой язык программирования и библиотеки будем использовать)
 - Язык Python
 - Библиотеки: pandas, numpy, sklearn, plotly
- **В какой среде (на каком оборудовании) будет проходить работа?**
 - Jupiter Notebook (Google Colab)

Какие требования к школьникам?

- Какие из умений и навыков мы хотим видеть у школьников на старте, а какие будем формировать на занятиях.
- **Зеленая зона** - навыки и знания, которые планирую прорабатывать на курсе
-> **образовательные результаты**
- **Красная зона** - навыки и знания, которые я не готов прорабатывать на курсе
-> **требования к школьникам**
 - Базовые знания языка Python (синтаксис, массивы, циклы, условные операторы, импорт библиотек и методов из них)
- **Серая зона** - навыки и знания, которые я при необходимости могу проработать, но было бы хорошо, если бы они были на старте

«Что я буду делать, если ко мне придут школьники с разными набором знаний из серой зоны?» 🤔

Отсутствие знаний из зеленой зоны - **требование к школьникам**

Как мы проверим достижение цели курса?

Проверка сформированности навыков через:

- решение школьником набора **учебных задач** - контрольная, активные групповые формы.
- успешное выполнение школьником всех этапов **на комплексной практической задаче** - проекты.

групповые или индивидуальные проекты

- Выставка Итоговых Проектов (ВИП) - возможность школьникам представить результаты своих решений комплексных задач.

Образовательные результаты

Теоретические результаты:

- Учащиеся могут воспроизвести **основные понятия** машинного обучения и предикативной аналитики (какие?).
- Учащиеся могут воспроизвести **принципы работы алгоритмов** машинного обучения, используемых для прогнозирования (каких?) и выбрать подходящий алгоритм в зависимости от контекста задачи.

«Инструментальные» результаты:

- Учащиеся могут **запустить, написать и выполнить код** в среде Jupiter Notebook.
- Учащиеся могут выполнить **базовые действия с датасетами в Pandas** (какие?): загрузка файлов *.csv и *.xlsx, поиск пропущенных значений, формирование выборок по условиям, построение сводных таблиц.
- Учащиеся могут выполнить **базовые действия с числовыми массивами в NumPy** (какие?): ...
- Учащиеся могут **построить простейшие графики с помощью библиотеки Plotly** (какие?): ...

Основные результаты

- Учащиеся могут выполнить очистку и подготовку данных для использования алгоритмов машинного обучения.
- Учащиеся могут выполнить загрузку и обучение необходимой модели из библиотеки sklearn.
- Школьник должен познакомиться и научиться использовать метрики оценки качества работы моделей.
- Учащиеся могут выбрать адекватную метрику оценки модели, импортировать ее из библиотеки sklearn и использовать для оценки обученной модели.
- Учащиеся могут визуализировать результат работы модели.

План занятий

- Практические навыки формируются только в активной деятельности обучающихся.
- Обучающийся в среднем способен сфокусироваться на одной деятельности в течение 12-15 минут.
- Для каждого образовательного результата необходимо предусмотреть в каком месте курса происходит его формирование (для этого важно понимать связи между ними) и в каком месте вы сможете оценить его сформированность.
- «Учащиеся могут **запустить, написать и выполнить код** в среде *Jupyter Notebook*.»
 - *Формирование на 1 занятии*
 - *Оценка сформированности на 2, когда будет решать задачу*

1. Введение в машинное обучение

Теория (20 минут):

- Что такое машинное обучение? Основные понятия: данные, признаки, алгоритмы, модель, обучение с учителем и без учителя.
- Предиктивная аналитика и типы практических задач, которые можно решать с ее помощью: прогнозирование цен, погоды, спроса и т.д.
- Ключевые шаги разработки программы для прогнозирования: сбор данных, обучение модели, проверка точности.

Практика (30 минут):

- Загрузка Jupyter Notebook: создание нового проекта, знакомство с интерфейсом, создание и запуск простейшей программы на python.
- Введение в работу с библиотекой pandas для анализа данных (например, данные о ценах на жилье):
 - импорт библиотеки;
 - загрузка набора данных из файла;
 - отображение и структура датафрейма;
 - получение значения ячейки, запись значения в ячейку;
 - формирование выборок по условию.

2. Работа с данными. Как правильно подготовить данные для прогноза

Практика (5 минут)

- Запуск Jupiter Notebook, открытие проекта, коррекция и исполнение кода

Теория (15 минут):

- Почему правильные данные — это залог точных прогнозов.
- Основные этапы подготовки данных: удаление пропусков, преобразование переменных, нормализация.

Практика (30 минут):

- Поиск и заполнение пропусков с помощью методов pandas.
- Преобразование переменных с помощью методов pandas.
- Объединение двух датафреймов в pandas.
- Преобразование части датафрейма в массив Numpy.
- Использование методов нормализации из библиотеки Sklearn.

3. Первая предсказательная модель

Практика (5 минут):

- открываем проект в Jupiter Notebook, загружаем датасет, выполняем код его обработки, написанный на прошлом занятии;

Теория (15 минут):

- Введение в линейную регрессию — как её используют для прогнозирования числовых значений.
- Основные термины: зависимые и независимые переменные, коэффициенты регрессии.
- Применение линейной регрессии для предсказания.

Практика (30 минут):

- импорт модели линейной регрессии из библиотеки sklearn;
- нормирование данных;
- разделение выборки на обучающую и тестовую;
- обучение модели линейной регрессии на обучающей выборке;
- обработка тестовой выборки обученной моделью.

4. Оценка качества прогноза

Практика (10 минут)

- открываем проект в Jupiter Notebook, загружаем датасет, выполняем код его обработки и обучения модели, написанные на прошлых занятиях;
- обсуждаем и вспоминаем

Теория (15 минут):

- почему модели делают ошибки: понятие «переобучение» и «недообучение»;
- метрики оценки моделей;
- кросс-валидация и её роль в оценке моделей.

Практика (25 минут):

- оценка качества построенной модели с использованием различных метрик;
- проведение кросс-валидации;
- интерпретация результатов и визуализация предсказаний.

5. Методы прогнозирования: деревья решений

Теория (15 минут):

- что такое дерево решений;
- алгоритм прогнозирования значений параметров с использованием дерева решений;
- в каких случаях лучше использовать деревья решений.

Практика (20 минут):

- реализация модели дерева решений для предсказания;
- сравнение точности дерева решений с линейной регрессией / визуализация.

Введение в проекты (15 минут):

- рассказ о том, что такое проект и представление проектных задач;
- объединение учащихся в группы и выбор проектов;
- обсуждение логики проекта в группах.

6. Подготовка проекта к ВИП

на занятии 5 учащиеся выбрали небольшую задачу (например, предсказание стоимости акций)

Установка (5 минут)

- покажем где хранятся исходные данные для проектов, уточним требования, проговорим этапы решения.

Работа над проектом (30 минут):

- Загрузка и предобработка данных.
- Выбор и обучение модели.
- Выбор и оценка метрик качества.
- Визуализация работы модели на графиках.

Обсуждение презентации проекта (15 минут):

- Как демонстрируем работу моделей?
- Что выносим на плакат?