

Ассистент ученика

Команда: AI больно

Хакатон 1.1

ИИ

Команда



Кирилл Черников
Главный программист



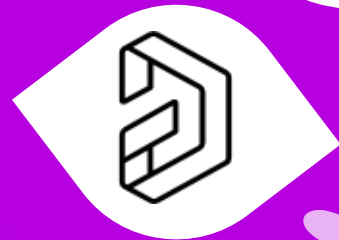
Арина Блохина
Data-аналитик



Николай Вологдин
UI дизайнер



Николай Щежин
Программист-тестировщик



Кейс ИИ

Ассистент ученика

Составить программу, основанную на ИИ-модели для прогнозирования успешности обучения ученика. Программа должна указывать на вероятность получения итоговой оценки учеником, давать рекомендации по необходимости приложения усилий по тому или иному предмету для достижения максимального результата.

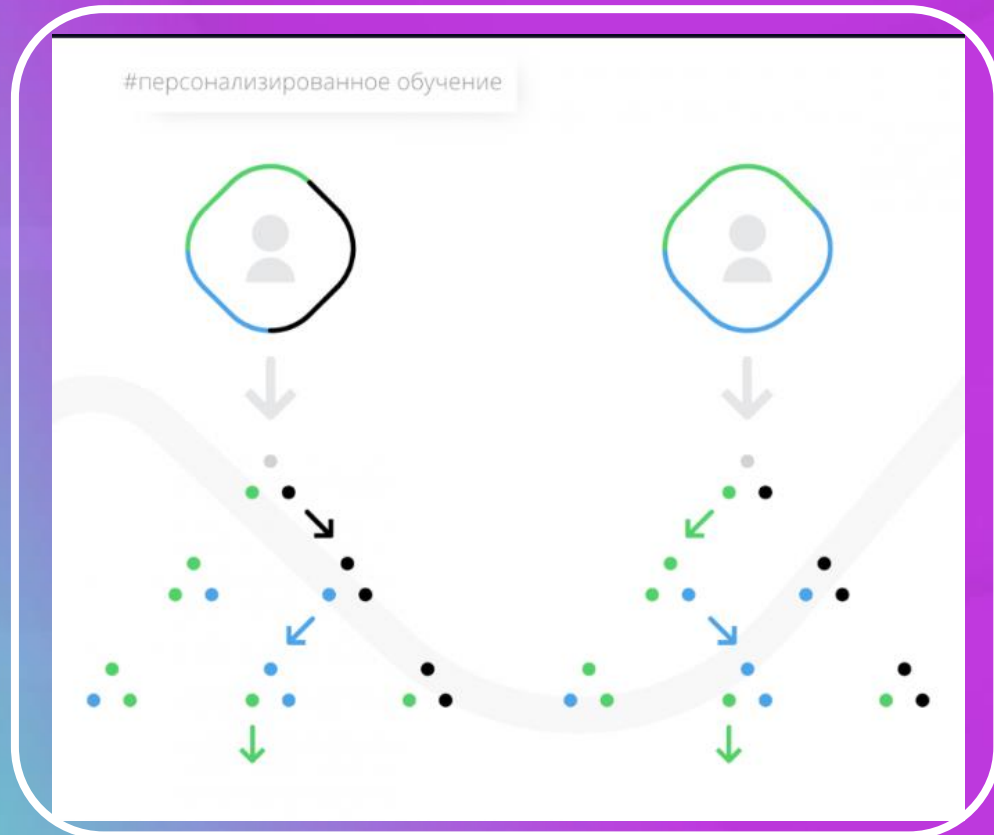
Задачи

- 01** • Проанализировать существующие решения.
- 02** • Разработать базу данных
- 03** • Сгенерировать синтетическую базу данных
- 04** • Выявить наиболее важные параметры
- 05** • Обучить ML модели и нейронную сеть

Что есть сейчас ?

Stepik

Рекомендательные
СИСТЕМЫ



Сириус.Курсы

Обучающие
СИСТЕМЫ

Фактор	Описание фактора	Возможные значения
Школа	Название школы, в которой учится ученик	«GP» (сокращение) «MS» (сокращение)
Состояние здоровья	Состояние здоровья ученика	Целые числа из диапазона от 1 до 5
Желание получить высшее образование	Хочет ли ученик после окончания школы пойти учиться в университет	«нет» «да»
Время в пути до школы	Время в минутах, которое ученик тратит на то, чтобы добраться до школы	«< 15» «15–30» «30–60» «> 60»
Возраст	Возраст ученика	Целые числа из диапазона от 15 до 22
Работа матери	Сфера профессиональной деятельности матери ученика	«образование» «медицина» «госслужба» «нет работы» «другое»

Наше решение



- **Обширная база данных**
- ↓
- **Ансамблевый подход**
- ↓
- **Наиболее точные предсказания касательно дальнейших оценок и действий**

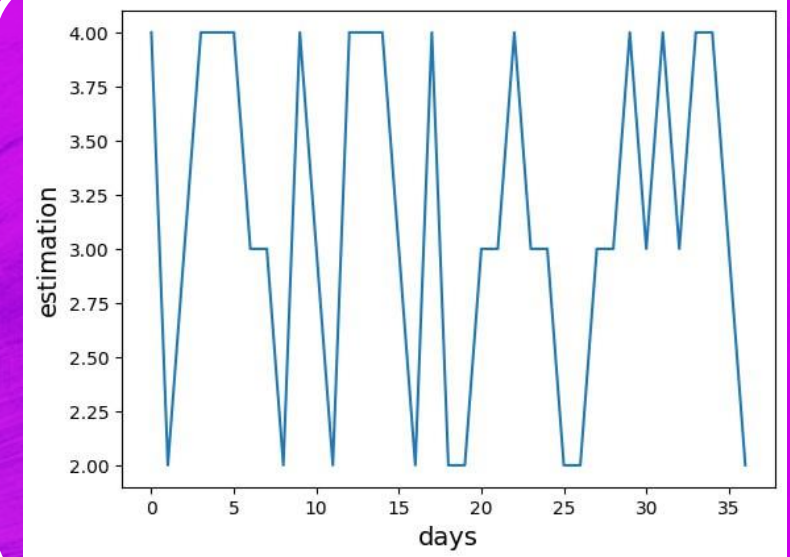
Входящие данные

Предметы	Средняя оценка	1 четверть	2 четверть	3 четверть	Средняя оценка+средняя оценка за 3 четверти
Русский язык	3,6	4	4	4	4
Английский язык	4	4	4	4	4
Литература	4	5	5	4	4,75
Математика	3,2	4	4	4	3,75
Химия	5	5	5	5	5
Физика	3,6	5	4	4	4,25
Физкультура	5	5	5	5	5
Информатика	5	5	5	5	5
История	4,5	5	4	5	5
Обществознание	5	5	5	5	5
Биология	4,5	4	5	4	4,5
География	4	5	4	5	4,5

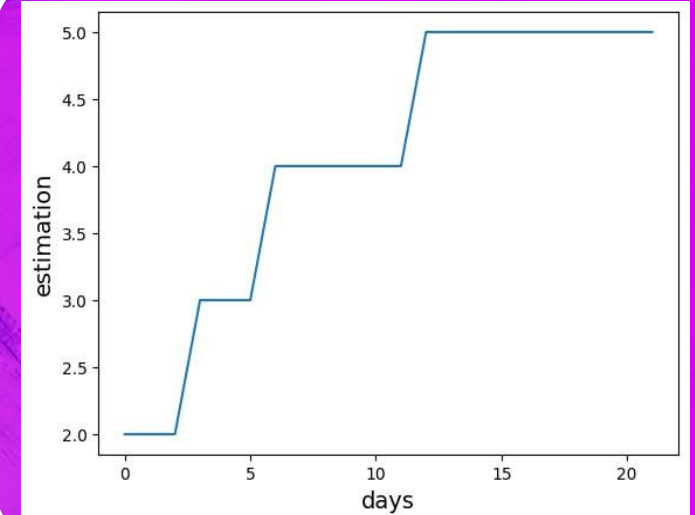
Таблица успеваемости

	10.04.2023	11.04.2023	12.04.2023
Русский язык	1, 1, 2	2, 1, 2	3, 1, 2
Английский язык	1, 1, 3	2, 1, 3	3, 1, 3
Литература	1, 1, 4	2, 1, 4	3, 1, 4
Математика	1, 1, 5	2, 1, 5	3, 1, 5
Химия	1, 1	2, 1	3, 1
Физика	1, 0	2, 0	3, 0
Физкультура	1, 1, 2	2, 1, 2	3, 1, 2
Информатика	1, 1, 3	2, 1, 3	3, 1, 3
История	1, 1, 4	2, 1, 4	3, 1, 4
Обществознание	1, 1, 5	2, 1, 5	3, 1, 5
Биология	1, 1	2, 1	3, 1
География	1, 0	2, 0	3, 0

Матрица учащегося



Общая динамика успеваемости



Динамика успеваемости по теме



Промежуточные и выходящие данные

Средние параметры (с которыми мы скорее всего и будем работать)

Текущая средняя оценка	Число
Максимально возможная оценка	Число
Динамика успеваемости (общая) (предмет)	Логистическая регрессия
Динамика успеваемости по теме (предмет)	Логистическая регрессия
Посещаемость (динамика)	Логистическая регрессия
Сколько уроков осталось до конца четверти	Число
Максимальная возможная оценка за четверть	Число по лучшим предсказаниям
Максимально возможная оценка за год	Число по лучшим предсказаниям

Выходящие данные

Входящие данные для ИИ

Предмет	Тема	Максимальная оценка(по предмету)	Максимальная оценка(за год)	Лучшая возможность исправить оценку
Русский язык	1	4	4	24.04.2023
Английский язык	2	4	4	05.05.2023
Литература	3	5	5	07.05.2023
Математика	4	4	4	09.05.2023
Химия	5	5	5	13.04.2023
Физика	6	4	5	20.04.2023
Физкультура	7	5	5	25.05.2023
Информатика	8	5	5	14.05.2023
История	9	5	5	03.05.2023
Обществознание	10	5	5	03.05.2023
Биология	11	4	5	22.04.2023
География	12	5	5	17.04.2023

Инструменты



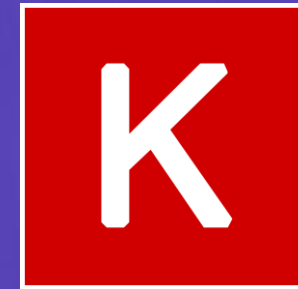
Numpy



Pandas



Sklearn



Keras



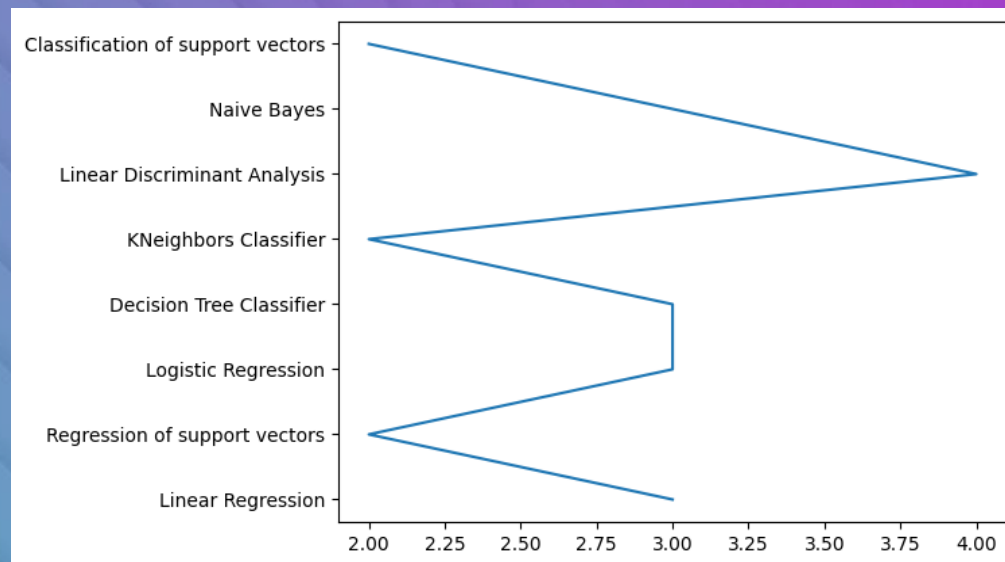
Tensorflow



Метрики Ансамбль Roadmap ИИ

МІ модели

Лучшая точность - 0.4
Лучшая модель -
дискриминантный анализ
Лучшие параметры - все



Обычные нейронные сети

Лучшая точность - 0.672
Архитектура;

	Кол-во нейронов	Активационная функция
Первый слой	100	Softmax
Второй слой	150	Relu
Третий слой	50	Relu

Лучшие параметры -
все
Кол-во эпох - 10

Немного кода

```
def get_sint(x,n,d):#x- масс, n-количество учеников, d- количество дней в четверти
fir=[]
for i in range(n):
    fir.append(np.random.choice([2,3,4,5],3))
kol=np.random.randint(1,d,1)
k=1
ans=[]
a=[]
for j in range(n):
    for i in range(int(kol)):
        ans=[]
        ans.append(k)
        ans.append(int(np.random.randint(0,2,1)))
        if ans[1]==1:
            ans.append(int(np.random.randint(2,5,1)))
        a.append(ans)
    k+=1
    for i in range(d-int(kol)):
        ans=[]
        ans.append(k)
        ans.append(int(np.random.randint(0,2,1)))
        if ans[1]==1:
            ans.append(int(np.random.randint(2,5,1)))
        a.append(ans)
return(fir,a)
```

```
def get_sint(x,n,d):
    fir=[]
    for i in range(n):
        fir.append(np.random.choice([2,3,4,5],3))
    kol=np.random.randint(1,d,1)
    k=1
    ans=[]
    a=[]
    for j in range(n):
        for i in range(int(kol)):
            ans=[]
            ans.append(k)
            ans.append(int(np.random.randint(0,2,1)))
            if ans[1]==1:
                ans.append(int(np.random.randint(2,5,1)))
            a.append(ans)
        k+=1
        for i in range(d-int(kol)):
            ans=[]
            ans.append(k)
            ans.append(int(np.random.randint(0,2,1)))
            if ans[1]==1:
                ans.append(int(np.random.randint(2,5,1)))
            a.append(ans)
    return(fir,a)
```

Выводы



Мы выяснили, что наиболее точной является сеть, у которой будет большая база данных



Мы использовали ансамблевый подход тем самым смогли реализовать нейронную сеть и ML модели что бы сравнить их на разных базах данных



Мы определили закономерность между размерностью баз данных, а так же их сложностью



В процессе мы столкнулись с проблемой переобучения



Thank You

Код в полном
объёме

