

Приложение № 1
к приказу ректора
АНО ВО «Центральный университет»
от «17» августа 2026 г. № 0817.93

**Автономная некоммерческая организация высшего образования
«Центральный университет»**

УТВЕРЖДАЮ

Ректор АНО ВО «Центральный
университет»

Е.В. Ивашкевич

**ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ ПРОГРАММА –
ДОПОЛНИТЕЛЬНАЯ ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА ДЛЯ ДЕТЕЙ
«ВВЕДЕНИЕ В ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ»**

Трудоемкость обучения: 112 ак. часов

Москва

2026

1. ОБЩАЯ ХАРАКТЕРИСТИКА ПРОГРАММЫ

1.1. Общие положения

Настоящая дополнительная общеобразовательная программа – дополнительная общеразвивающая программа для детей «Введение в искусственный интеллект» (далее – дополнительная общеразвивающая программа для детей, программа) разработана на основании Федерального закона от 29 декабря 2012 г. № 273-ФЗ «Об образовании в Российской Федерации», приказа Минпросвещения России от 27 июля 2022 г. № 629 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам» и приказа Минобрнауки России от 11 октября 2023 г. № 1678 «Об утверждении Правил применения организациями, осуществляющими образовательную деятельность, электронного обучения, дистанционных образовательных технологий при реализации образовательных программ».

Дополнительная общеразвивающая программа для детей реализуется на базе образовательных организаций – участников проекта по внедрению дополнительной образовательной программы «Введение в искусственный интеллект» (далее — проект) в школы при организационном и учебно-методическом сопровождении Автономной некоммерческой организации высшего образования «Центральный университет» (далее – АНО ВО «Центральный университет», Центральный университет).

Разработчики программы:

Ю.А. Кириленко, преподаватель АНО ВО «Центральный университет»;

В.М. Гуровиц, учитель профильной информатики, преподаватель курсов по машинному обучению и квантовым вычислениям АОНО «Летово»;

А.Е. Гущин, академический руководитель олимпиадного направления искусственного интеллекта АНО ВО «Центральный университет».

Дополнительная общеразвивающая программа для детей разработана в инициативном порядке.

Электронный документ

Программа реализуется на русском языке.

1.2. Цель реализации программы

Дополнительная общеразвивающая программа для детей нацелена на:

- удовлетворение индивидуальных потребностей обучающихся в интеллектуальном, нравственном, художественно-эстетическом развитии и физическом совершенствовании;
- профессиональную ориентацию обучающихся;
- выявление, развитие и поддержку обучающихся, проявивших выдающиеся способности;
- удовлетворение иных образовательных потребностей и интересов обучающихся, не противоречащих законодательству Российской Федерации, осуществляемых за пределами федеральных государственных образовательных стандартов и федеральных государственных требований.

Направленность программы: техническая.

Цель программы – формирование у обучающихся базовых знаний и практических навыков анализа данных, построения моделей машинного обучения и разработки решений компьютерного зрения на языке программирования Python.

Для достижения цели решаются следующие задачи:

- сформировать представление о жизненном цикле проекта в области анализа данных и искусственного интеллекта;
- научить загружать, очищать, преобразовывать, агрегировать и визуализировать данные;
- освоить основы описательной статистики, проверки гипотез и кластеризации;
- научить строить и оценивать модели регрессии и классификации, предупреждать переобучение и утечку данных;
- познакомить с классическими методами обработки изображений и основами нейронных сетей;

- сформировать навыки документирования кода, интерпретации результатов и защиты проекта;

- развить ответственное отношение к данным, воспроизводимости экспериментов и корректной оценке качества моделей.

Актуальность программы обусловлена ростом объёма данных и распространением технологий искусственного интеллекта в науке, промышленности и повседневной жизни. Освоение методов анализа данных позволяет обучающимся перейти от использования готовых цифровых сервисов к осознанному проектированию собственных решений, основанных на данных.

Педагогическая целесообразность программы определяется последовательным переходом от работы с табличными данными и статистикой к классическому машинному обучению и компьютерному зрению. Каждый содержательный раздел завершается промежуточной аттестацией, что обеспечивает связь теории с практикой, развивает самостоятельность, критическое мышление и навыки представления результатов.

Отличительные особенности программы:

- сквозная логика обучения: анализ данных — машинное обучение — компьютерное зрение;

- сочетание теоретических и практических занятий, автоматизированных проверок и проектной работы;

- использование профессионального программного стека: Python, NumPy, Pandas, Scikit-Learn, XGBoost, LightGBM, CatBoost, OpenCV, MediaPipe, PyTorch и torchvision;

- работа с учебными наборами данных, а в разделе 2 — со скрытыми тестовыми выборками и таблицей результатов, организованной по принципу открытых соревнований по анализу данных;

- три промежуточные аттестации в форме проектов и соревнования, формирующих портфолио обучающегося.

1.3. Категории обучающихся

Основными категориями обучающихся, на которых рассчитана дополнительная общеразвивающая программа для детей, являются обучающиеся общеобразовательных организаций 8–11-х классов, проявляющие интерес к программированию, математике и технологиям искусственного интеллекта.

1.4. Требования к уровню подготовки поступающего на обучение, необходимому для освоения программы

Для зачисления на образовательную программу необходимо владение базовыми конструкциями языка программирования Python: переменными, условными операторами, циклами, функциями и коллекциями.

1.5. Критерии отбора на программу

До начала обучения проводится входное диагностическое тестирование в системе управления обучением (LMS) Центрального университета по базовым конструкциям языка программирования Python.

К прохождению следующего этапа отбора – индивидуального собеседования – рекомендуется допускать обучающихся, выполнивших не менее 70 % заданий входного тестирования.

Индивидуальное собеседование проводится с целью определения мотивации обучающегося к освоению программы, уточнения уровня его предметной подготовки, а также оценки готовности к учебной нагрузке и регулярному участию в занятиях.

Решение о зачислении обучающегося на программу принимает педагог, ведущий образовательную программу, на основании совокупности результатов входного тестирования и индивидуального собеседования.

Результаты входной диагностики также используются для определения стартового уровня подготовки обучающихся и формирования индивидуальных

рекомендаций по повторению и актуализации базовых знаний и навыков программирования.

1.6. Идентификаторы достижения планируемых результатов обучения

Обучающийся, освоивший дополнительную общеразвивающую программу для детей, должен:

знать:

- назначение и устройство массивов библиотеки NumPy, структур Series и DataFrame библиотеки Pandas;
- основные этапы очистки, преобразования, агрегации и визуализации данных;
- базовые понятия описательной статистики, проверки статистических гипотез и кластеризации;
- типы задач машинного обучения, принципы регрессии, классификации и ансамблевых методов;
- назначение обучающей, валидационной и тестовой выборок, признаки переобучения и утечки данных;
- основы представления и обработки изображений, принципы работы свёрток и аугментаций;
- базовую архитектуру нейронной сети, назначение функции потерь, оптимизатора и алгоритма обратного распространения ошибки.

уметь:

- загружать табличные данные, выполнять первичный анализ их качества и корректировать типы данных;
- фильтровать, объединять, группировать и преобразовывать данные средствами библиотек Pandas и NumPy;
- строить графики, интерпретировать распределения и формулировать выводы по данным;

- выбирать статистический критерий и интерпретировать результат проверки гипотезы;
- строить воспроизводимый конвейер машинного обучения и выбирать метрику качества;
- обучать и сравнивать линейные модели, решающие деревья, случайный лес и градиентный бустинг;
- обрабатывать изображения средствами NumPy и библиотеки компьютерного зрения OpenCV, создавать и обучать базовую нейронную сеть средствами библиотеки PyTorch;
- оформлять воспроизводимый ноутбук, отчёт и презентацию, аргументированно защищать проект.

владеть:

- приёмами работы в среде Jupyter Notebook и на платформе StackMoreLayers;
- инструментами NumPy, Pandas, Matplotlib, Seaborn и SciPy;
- базовыми средствами библиотек Scikit-Learn, XGBoost, LightGBM и CatBoost для построения моделей;
- инструментами Pillow, OpenCV, MediaPipe, PyTorch и torchvision, а также средствами разработки простого веб-приложения;
- приёмами воспроизводимого эксперимента, контроля качества данных и анализа ошибок модели;
- навыками индивидуальной проектной работы, документирования и публичного представления результатов.

1.7. Трудоёмкость программы

Нормативная трудоемкость обучения по данной программе – 112 академических часов, включая все виды контактной работы слушателя.

1.8. Форма и сроки обучения

Обучение по программе осуществляется в очной форме с применением электронного обучения и дистанционных образовательных технологий.

Минимальный срок обучения на программе составляет 28 недель.

Электронное обучение и применение дистанционных образовательных технологий обеспечиваются с использованием системы управления обучением (LMS) Центрального университета, платформы StackMoreLayers, среды Jupyter Notebook или совместимой облачной среды выполнения кода, автоматизированных систем тестирования, закрытых наборов тестовых данных и электронных учебно-методических материалов программы.

Основными формами организации занятий являются: интерактивная лекция, практическое занятие, работа на образовательной онлайн-платформе для анализа данных и машинного обучения, консультация, проектная работа и защита проекта.

В процессе реализации программы применяются технологии проблемного и проектного обучения, разбор практических ситуаций, взаимная проверка и формирующее оценивание.

Педагог самостоятельно определяет формы, методы и приёмы организации образовательной деятельности, последовательность изложения материала в рамках отдельного занятия, а также целесообразность использования предлагаемых учебных и методических материалов с учётом уровня подготовки и образовательных потребностей обучающихся. При этом содержание программы, предусмотренное учебным планом и календарно-тематическим планированием, подлежит реализации в полном объеме.

1.9. Режим занятий

Для всех занятий академический час устанавливается продолжительностью 40 минут.

Учебная нагрузка устанавливается не более 4 академических часов в неделю, включая все виды контактной работы слушателя.

2. ПРОГРАММА УЧЕБНОГО КУРСА

2.1. Учебный план дополнительной общеразвивающей программы для детей «Введение в искусственный интеллект»

Продолжительность обучения – 112 ак. часов.

Форма обучения – очная.

№ п/п	Наименование разделов учебного курса	Всего, академ. ч.	Контактная работа (академ. ч.)		Формы аттестации и контроля знаний
			лекции	практич. и лаборат. занятия	
1	Pandas и статистика	40	16	24	Тесты, Практические задания, Проект
2	Введение в машинное обучение	40	16	24	Тесты, практические задания, Соревнование
3	Введение в компьютерное зрение	32	12	20	Тесты, Практические задания, Проект
	ИТОГО:	112	44	68	

2.2. Учебно-тематический дополнительной общеразвивающей программы для детей «Введение в искусственный интеллект»

№ п/п	Наименование разделов блоков и тем	Всего, академ. ч.	Контактная работа (академ. ч.)		Формы аттестации и контроля знаний
			лекции	практич. и лаборат. занятия	
1.	Pandas и статистика	40	16	24	Тесты, Практические задания, Проект
1.1.	Знакомство с инструментами NumPy и Pandas	4	2	2	Тесты, Практические задания
1.2.	Фильтрация, срезы и условный выбор данных	4	2	2	Тесты, Практические задания
1.3.	Пользовательские преобразования и объединение данных	4	2	2	Тесты, Практические задания
1.4.	Работа с датами, агрегации и сводные таблицы	4	2	2	Тесты, Практические задания
1.5.	Визуализация данных средствами Pandas, Matplotlib и Seaborn	4	2	2	Тесты, Практические задания
1.6.	Описательная статистика и визуализация распределений	4	2	2	Тесты, Практические задания
1.7.	Кластеризация данных: статистический взгляд на группы	4	2	2	Тесты, Практические задания
1.8.	Разброс, аномалии и проверка статистических гипотез	4	2	2	Тесты, Практические задания
1.9.	Проект. Анализ данных: от постановки задачи до выводов	8		8	Проект
2.	Введение в машинное обучение	40	16	24	Тесты, Практические задания, Соревнование
2.1.	Жизненный цикл проекта машинного обучения и постановка задачи	4	2	2	Тесты, Практические задания
2.2.	Предобработка данных	4	2	2	Тесты, Практические задания

2.3.	Регрессия: математические основы и метрики качества	4	2	2	Тесты, Практические задания
2.4.	Классификация: вероятностная логика и метрики качества	4	2	2	Тесты, Практические задания
2.5.	Переобучение, валидация и воспроизводимость	4	2	2	Тесты, Практические задания
2.6.	Решающие деревья и управление переобучением	4	2	2	Тесты, Практические задания
2.7.	Ансамбли моделей: бэггинг и случайный лес	4	2	2	Тесты, Практические задания
2.8.	Градиентный бустинг для задач классификации	4	2	2	Тесты, Практические задания
2.9.	Соревнование и защита решения «Модель машинного обучения для прикладной задачи»	8		8	Соревнование
3.	Введение в компьютерное зрение	32	12	20	Тесты, Практические задания, Проект
3.1.	Изображение как массив чисел	4	2	2	Тесты, Практические задания
3.2.	Базовые операции с изображениями, аугментация и свёртки	4	2	2	Тесты, Практические задания
3.3.	Прикладное компьютерное зрение: OpenCV и MediaPipe	4	2	2	Тесты, Практические задания
3.4.	Экстремумы и градиентный спуск	4	2	2	Тесты, Практические задания
3.5.	Введение в нейронные сети и PyTorch	4	2	2	Тесты, Практические задания
3.6.	Регуляризация и свёрточные нейронные сети	4	2	2	Тесты, Практические задания
3.7.	Проект. «CV-модель и веб-приложение»	8		8	Проект
	ИТОГО:	112	44	68	

2.3. Календарный график

№ п/ п	Наименование раздела	Наименование темы раздела	Академ. часов	Учебные недели		
				1 – 10	11 - 20	21 - 28
1.	Pandas и статистика	Знакомство с инструментами NumPy и Pandas	4	4		
		Фильтрация, срезы и условный выбор данных	4	4		
		Пользовательские преобразования и объединение данных	4	4		
		Работа с датами, агрегации и сводные таблицы	4	4		
		Визуализация данных средствами Pandas, Matplotlib и Seaborn	4	4		
		Описательная статистика и визуализация распределений	4	4		
		Кластеризация данных: статистический взгляд на группы	4	4		
		Разброс, аномалии и проверка статистических гипотез	4	4		
		Проект 1. Анализ данных: от постановки задачи до выводов	8	8		
2.	Введение в машинное обучение	Жизненный цикл проекта машинного обучения и постановка задачи	4		4	
		Предобработка данных	4		4	
		Регрессия: математические основы и метрики качества	4		4	
		Классификация: вероятностная логика и метрики качества	4		4	
		Переобучение, валидация и воспроизводимость	4		4	
		Решающие деревья и управление переобучением	4		4	
		Ансамбли моделей: бэггинг и случайный лес	4		4	
		Градиентный бустинг для задач классификации	4		4	

		Соревнование и защита решения «Модель машинного обучения для прикладной задачи»	8		8	
3.	Введение в компьютерное зрение	Изображение как массив чисел	4			4
		Базовые операции с изображениями, аугментация и свёртки	4			4
		Прикладное компьютерное зрение: OpenCV и MediaPipe	4			4
		Экстремумы и градиентный спуск	4			4
		Введение в нейронные сети и PyTorch	4			4
		Регуляризация и свёрточные нейронные сети	4			4
		Проект 3. «CV-модель и веб-приложение»	8			8

Конкретные календарные сроки проведения занятий определяются расписанием образовательной площадки, реализующей программу. В случае изменения сроков проведения занятий сохраняются предусмотренные программой последовательность освоения тем, содержание и общий объем учебной нагрузки.

3. РАБОЧАЯ ПРОГРАММА КУРСА

3.1.Содержание курса

№ п/п	Наименование раздела	Наименование темы раздела	Содержание темы (тезисно)
1.	Pandas и статистика	Знакомство с инструментами NumPy и Pandas	Занятие 1. Интерактивная лекция. Жизненный цикл проекта по анализу данных. Рабочая среда: ноутбук, ячейки, диалоговый режим, разметка Markdown. NumPy: ndarray против списка, векторизация без циклов, dtype и shape, агрегаты. Pandas поверх NumPy: Series, DataFrame, индексы, типы данных. <i>Инструменты: StackMoreLayers, NumPy, Pandas.</i> <i>Форма контроля:</i> Тестирование по теории. Занятие 2. Практическое занятие. Скорость NumPy против списков (%timeit). Агрегаты без циклов: сумма, среднее, минимум, максимум. Формат CSV и отличия от электронных таблиц. Чтение данных методами read_.... Первичный осмотр таблицы. Индексация: столбцы, .loc и .iloc, строки, клетки, срезы. <i>Инструменты: StackMoreLayers, NumPy, Pandas.</i> <i>Форма контроля:</i> Практическое задание.
		Фильтрация, срезы и условный выбор данных	Занятие 1. Интерактивная лекция. Булево маскирование и операторы сравнения. Сложные условия: «И» (&), «ИЛИ» (), «НЕ» (~). Метод .query() как читаемая альтернатива маскам. Копирование через .copy(). Пропуски: .isna(), .dropna(), .fillna(), выбор стратегии. <i>Инструменты: StackMoreLayers, Pandas, NumPy.</i> <i>Форма контроля:</i> Тестирование по теории. Занятие 2. Практическое занятие. Маски, .query(), .isin(). Выделение и сравнение целевых групп. Поиск аномалий по жёстким условиям. Обработка пропусков. Первая модель: линейная регрессия в несколько строк, прогноз по одному и по нескольким признакам (без метрик и конвейера). <i>Инструменты: StackMoreLayers, Pandas, NumPy, Scikit-Learn.</i> <i>Форма контроля:</i> Практическое задание.
		Пользовательские преобразования и объединение данных	Занятие 1. Интерактивная лекция. Изменение структуры таблицы: добавление столбца, .drop(), .rename(), добавление и удаление строк. Пользовательские преобразования: .apply() по столбцам и по строкам, lambda-функции. Текстовые столбцы через аксессор .str. Объединение данных: concat по строкам и столбцам, merge по ключу, типы

			соединений (внутреннее, левое, правое, полное), соединение по нескольким и разноимённым ключам. <i>Инструменты:</i> <i>StackMoreLayers, Pandas.</i> <i>Форма контроля:</i> Тестирование по теории. Занятие 2. Практическое занятие. Разбор текстовых признаков собственными функциями: домены электронной почты, нормализация телефонов, <code>.str.split()</code>. Генерация признаков через <code>.apply()</code> по строкам. Сборка единого набора: <code>concat</code> основной и дополнительной таблиц, обогащение справочниками через <code>merge</code>, сравнение типов соединений. <i>Инструменты:</i> <i>StackMoreLayers, Pandas.</i> <i>Форма контроля:</i> Практическое задание.
		Работа с датами, агрегации и сводные таблицы	Занятие 1. Интерактивная лекция. Типы <code>Timestamp</code> и <code>Timedelta</code>. Преобразование текста в даты: <code>pd.to_datetime()</code>. Компоненты времени через <code>.dt</code>: год, месяц, день недели, час. Интервалы между событиями и фильтрация по ним. <code>Split-apply-combine</code> и <code>groupby</code>. Несколько агрегаций через <code>.agg()</code>, свои функции по столбцам, группировка по нескольким колонкам. Сводные таблицы: <code>pivot_table</code> для чисел, <code>crosstab</code> для частот и долей. <i>Инструменты:</i> <i>StackMoreLayers, Pandas, NumPy.</i> <i>Форма контроля:</i> Тестирование по теории. Занятие 2. Практическое занятие. Разбор дат в заказах, извлечение часа и дня недели. Интервалы между событиями и скорость доставки. Аналитическая витрина через <code>groupby().agg()</code>: средний чек, число заказов. Двумерный отчёт через <code>pivot_table</code>: категории и дни недели. <i>Инструменты:</i> <i>StackMoreLayers, Pandas, NumPy.</i> <i>Форма контроля:</i> Практическое задание.
		Визуализация данных средствами Pandas, Matplotlib и Seaborn	Занятие 1. Интерактивная лекция. Зачем нужна визуализация. Графики из <code>Pandas</code>: <code>.plot()</code> и параметр <code>kind</code> — линия, столбцы, гистограмма, круговая, точечный график, составные столбцы; частоты через <code>value_counts()</code>. <code>Matplotlib</code>: <code>Figure</code> и <code>Axes</code>, заголовок, подписи осей, легенда, несколько линий и несколько графиков на полотне. <code>Seaborn</code>: <code>countplot</code>, <code>barplot</code>, разбивка через <code>hue</code>, <code>lineplot</code>, <code>histplot</code>, <code>scatterplot</code>, <code>heatmap</code>. Выбор типа графика под задачу. <i>Инструменты:</i> <i>StackMoreLayers, Pandas, Matplotlib, Seaborn.</i> <i>Форма контроля:</i> Тестирование по теории.

			Занятие 2. Практическое занятие. Графики из Pandas: частоты категорий столбцами, динамика линией, распределение гистограммой. Seaborn: barplot с разбивкой через hue, lineplot по месяцам, тепловая карта по pivot table. Оформление графика и вывод по нему. Сохранение графика в файл. <i>Инструменты:</i> StackMoreLayers, Pandas, Matplotlib, Seaborn. <i>Форма контроля:</i> Практическое задание.
		Описательная статистика и визуализация распределений	Занятие 1. Интерактивная лекция. Случайные величины. Типы переменных: количественные и качественные. Меры центральной тенденции: среднее, медиана, мода; расхождение среднего и медианы. Меры разброса: дисперсия, стандартное отклонение, квантили, перцентили, межквартильный размах; сводка через .describe(). Визуализация распределений: гистограмма, ядерная оценка плотности (KDE). Форма распределения: асимметрия и эксцесс. Виды распределений: нормальное, логнормальное, экспоненциальное. Генеральная совокупность и выборка. <i>Инструменты:</i> StackMoreLayers, Pandas, NumPy, SciPy, Matplotlib, Seaborn. <i>Форма контроля:</i> Тестирование по теории. Занятие 2. Практическое занятие. Гистограммы и графики плотности, в том числе с разбивкой по группам. Оценка асимметрии через .skew() и чтение её величины. Логарифмирование скошенных распределений. Возврат из логарифма к исходной шкале. <i>Инструменты:</i> StackMoreLayers, Pandas, NumPy, SciPy, Matplotlib, Seaborn. <i>Форма контроля:</i> Практическое задание.
		Кластеризация данных: статистический взгляд на группы	Занятие 1. Интерактивная лекция. Зачем группировать объекты. Похожесть и расстояние между точками. Центр группы (центроид). Алгоритм k-средних (K-Means). Иерархическая кластеризация и дендрограмма. Выбор числа групп: метод локтя. Стандартизация признаков. Оценка качества разбиения: коэффициент силуэта. Ограничения K-Means: кластер как гипотеза. <i>Инструменты:</i> StackMoreLayers, Pandas, NumPy, Scikit-Learn, SciPy, Matplotlib, Seaborn. <i>Форма контроля:</i> Тестирование по теории. Занятие 2. Практическое занятие. Иерархическая кластеризация: дендрограмма, каменистая осыпь, разрез дерева через fcluster, профили групп. Кластеризация методом k-средних и сравнение методов. Два набора данных: навыки студентов и

			показатели здоровья. Группы на диаграмме рассеяния и их описание простыми словами. <i>Инструменты:</i> StackMoreLayers, Pandas, NumPy, Scikit-Learn, SciPy, Matplotlib, Seaborn. <i>Форма контроля:</i> Практическое задание.
	Разброс, аномалии и проверка статистических гипотез		Занятие 1. Интерактивная лекция. Природа выбросов. Правила поиска: межквартильный размах (IQR) и Z-оценка, причины их расхождения. Геометрия диаграммы размаха («ящик с усами»): медиана, квартили, усы, выбросы; слепое пятно и скрипичный график. Проверка гипотез: случайные различия, H_0 и H_1, достигаемый уровень значимости (p-value), типичные ошибки трактовки. Критерии: t-критерий Стьюдента, критерий Манна — Уитни, критерий хи-квадрат для долей. Схема A/B-теста как прикладная рамка проверки гипотез: контрольная и тестовая группы, случайное распределение участников по группам, сравнение целевой метрики между группами и условия корректности вывода. <i>Инструменты:</i> StackMoreLayers, Pandas, NumPy, SciPy, Matplotlib, Seaborn. <i>Форма контроля:</i> Тестирование по теории. Занятие 2. Практическое занятие. Диаграммы размаха и скрипичные графики для сравнения групп. Поиск выбросов по межквартильному размаху и по Z-оценке. Формулировка гипотезы и выбор критерия. Разбор учебного A/B-теста: сравнение контрольной и тестовой групп по целевой метрике. Сравнение групп t-критерием или критерием Манна — Уитни, сравнение долей критерием хи-квадрат. Вывод для заказчика о значимости различий. <i>Инструменты:</i> StackMoreLayers, Pandas, NumPy, SciPy, Matplotlib, Seaborn. <i>Форма контроля:</i> Практическое задание.
	Проект 1. Анализ данных: от постановки задачи до выводов		Занятие 1. Проект. Постановка задачи и данные. Пять учебных наборов данных на основе реальных бизнес-кейсов на выбор: сервисный сбор в сервисе «Топливо», поездки в сервисе «Самокаты», клиенты «Города», инвестиционный сервис, отзывы клиентов; выбирается один набор для сквозного анализа. Бизнес-контекст и вопрос заказчика, описание колонок и кодировок. Формулировка гипотез и вопросов к данным. Загрузка и первичный осмотр: размер, типы, .info(), .describe(), пропуски, дубликаты, диапазоны значений, перечень найденных проблем. <i>Инструменты:</i> StackMoreLayers, Pandas, NumPy. <i>Форма контроля:</i> Выбор набора данных и защита постановки задачи: обучающийся фиксирует вопрос заказчика, формулирует проверяемые гипотезы и

			<i>предъявляет результаты первичного осмотра данных; оценивается по подкритериям блока «Качество EDA (разведочного анализа данных)» (первичный осмотр и оценка качества данных).</i> Занятие 2. Проект. Очистка и подготовка данных. Обработка пропусков и выбросов (IQR или Z-оценка) с обоснованием решений. Приведение типов, нормализация категорий. Генерация признаков: компоненты дат, интервалы между событиями, агрегаты через <code>groupby</code> и <code>pivot_table</code>, объединение источников через <code>merge</code>. Аналитическая витрина для дальнейшего анализа. <i>Инструменты: StackMoreLayers, Pandas, NumPy.</i> <i>Форма контроля:</i> Промежуточный контроль хода проекта: обучающийся предъявляет результаты разведочного анализа данных, сформулированные гипотезы и результаты их статистической проверки. Занятие 3. Проект. Разведочный анализ данных. Описательная статистика, гистограммы и графики плотности, асимметрия и логарифмические преобразования, корреляции и их интерпретация, вывод текстом к каждому графику. Не менее трёх бизнес-гипотез с явными H_0 и H_1. Выбор критерия под данные: t-критерий Стьюдента, критерий Манна — Уитни, критерий хи-квадрат. Интерпретация p-value и оценка размера эффекта. Сегментация кластеризацией: обоснование числа кластеров, описание сегментов. <i>Инструменты: StackMoreLayers, Pandas, NumPy, SciPy, Scikit-Learn, Matplotlib, Seaborn.</i> <i>Форма контроля:</i> Промежуточный контроль хода проекта: обучающийся предъявляет результаты разведочного анализа данных, сформулированные гипотезы и результаты их статистической проверки. Занятие 4. Проект. Выводы и защита. Итоговый отчёт: ответы на вопросы бизнеса, ключевые находки и визуализации. Не менее трёх рекомендаций в формате «сделать X — ожидаемый эффект Y» с оценкой по деньгам или метрике. Ограничения анализа, риски выводов, следующие шаги. Презентация на 8–12 слайдов: задача — данные — находки — рекомендации — следующие шаги. Чистый воспроизводимый ноутбук. Защита проекта. <i>Инструменты: StackMoreLayers, Pandas, Matplotlib, Seaborn, Markdown.</i> <i>Форма контроля:</i> Промежуточная аттестация: выполнение и защита учебного проекта.
--	--	--	--

2.	Введение в машинное обучение	Жизненный цикл проекта машинного обучения и постановка задачи	Занятие 1. Интерактивная лекция. Концепция жизненного цикла машинного обучения и классификация задач. Сквозной цикл построения моделей: данные — предобработка — конструирование признаков — обучение модели — оценка метриками — валидация — анализ ошибок. Постановка задач машинного обучения: обучение с учителем и обучение без учителя. Перевод прикладной задачи в постановку машинного обучения: объект, целевая переменная, момент прогноза. <i>Инструменты:</i> StackMoreLayers, Pandas, NumPy, Scikit-Learn. <i>Форма контроля:</i> Тестирование по теории. Занятие 2. Практическое занятие. Проектирование базового конвейера машинного обучения. Написание первого сквозного конвейера «с нуля» на табличном наборе данных. Реализация константного эталона качества (тривиальный классификатор или регрессор). Оценка качества эталона, фиксация значения функции потерь и расчёт простых метрик. <i>Инструменты:</i> StackMoreLayers, Pandas, NumPy, Scikit-Learn. <i>Форма контроля:</i> Практическое задание.
		Предобработка данных	Занятие 1. Интерактивная лекция. Подготовка табличных признаков к обучению модели. Масштабирование признаков: стандартизация и приведение к диапазону, назначение каждого способа. Кодирование категориальных признаков: прямое кодирование (One-Hot), порядковое кодирование, кодирование по целевой переменной; редкие и неизвестные категории. Обработка пропусков и индикаторы пропусков. Риск утечки данных при некорректной предобработке: почему параметры преобразований вычисляются только по обучающей выборке. <i>Инструменты:</i> StackMoreLayers, Pandas, NumPy, Scikit-Learn. <i>Форма контроля:</i> Тестирование по теории. Занятие 2. Практическое занятие. Сборка изолированных предобработчиков и проектирование автоматического конвейера предобработки: последовательное применение преобразований средствами библиотеки Scikit-Learn, разная логика обработки числовых и категориальных столбцов, объединение шагов в единый конвейер. <i>Инструменты:</i> StackMoreLayers, Pandas, NumPy, Scikit-Learn. <i>Форма контроля:</i> Практическое задание.

		Регрессия: математические основы и метрики качества	Занятие 1. Интерактивная лекция. Линейные модели для задач прогноза непрерывной величины: одномерная и множественная регрессия. Метод наименьших квадратов и его геометрический смысл. Разбор метрик регрессии: средняя абсолютная ошибка (MAE) и её устойчивость к выбросам, средняя квадратичная ошибка (MSE) и штраф за крупные ошибки, корень из средней квадратичной ошибки (RMSE) в исходной размерности, средняя абсолютная процентная ошибка (MAPE), коэффициент детерминации. Константный эталон качества; разбиение на обучающую, валидационную и тестовую выборки для задачи регрессии. <i>Инструменты:</i> StackMoreLayers, Pandas, NumPy, Scikit-Learn, Matplotlib. <i>Форма контроля:</i> Тестирование по теории. Занятие 2. Практическое занятие. Построение модели линейной регрессии на табличном наборе данных, прогноз непрерывной величины, сравнение моделей по MAE и RMSE и выбор лучшей. Нелинейные признаки и регуляризованные линейные модели: гребневая регрессия и лассо; влияние масштабирования и выбросов на качество прогноза. <i>Инструменты:</i> StackMoreLayers, Pandas, NumPy, Scikit-Learn, Matplotlib. <i>Форма контроля:</i> Практическое задание.
		Классификация: вероятностная логика и метрики качества	Занятие 1. Интерактивная лекция. Линейные модели для задач классификации. Логистическая регрессия и вероятностный смысл предсказаний. Матрица ошибок. Метрики классификации: доля правильных ответов (accuracy), точность (precision), полнота (recall), F1-мера; площадь под ROC-кривой. Выбор метрики и порога принятия решения под задачу, поведение метрик при дисбалансе классов. <i>Инструменты:</i> StackMoreLayers, Pandas, NumPy, Scikit-Learn, Matplotlib. <i>Форма контроля:</i> Тестирование по теории. Занятие 2. Практическое занятие. Обучение логистической регрессии для бинарной классификации (например, прогноз оттока клиентов). Получение вероятностей принадлежности к классу, построение матрицы ошибок, расчёт точности, полноты и F1-меры, подбор порога принятия решения под условия задачи. <i>Инструменты:</i> StackMoreLayers, Pandas, NumPy, Scikit-Learn, Matplotlib. <i>Форма контроля:</i> Практическое задание.

	Переобучение, валидация и воспроизводимость	Занятие 1. Интерактивная лекция. Диагностика переобучения и недообучения, их природа и внешние признаки. Компромисс между смещением и разбросом. Схемы валидации: разбиение на обучающую и тестовую выборки, кросс-валидация по K блокам, стратифицированная кросс-валидация для сохранения баланса классов. Воспроизводимость эксперимента: фиксация начального значения генератора случайных чисел, учёт версий данных и моделей. <i>Инструменты:</i> StackMoreLayers, Pandas, NumPy, Scikit-Learn, Matplotlib. <i>Форма контроля:</i> Тестирование по теории. Занятие 2. Практическое занятие. Сравнение моделей на честной кросс-валидации: настройка обычного и стратифицированного разбиения, расчёт метрик по блокам. Построение кривых обучения, выявление переобучения, фиксация случайности и достижение воспроизводимого результата. <i>Инструменты:</i> StackMoreLayers, Pandas, NumPy, Scikit-Learn, Matplotlib. <i>Форма контроля:</i> Практическое задание.
	Решающие деревья и управление переобучением	Занятие 1. Интерактивная лекция. Устройство решающего дерева и логика последовательных разбиений. Критерии информативности: энтропия, прирост информации, индекс Джини; критерий для задач регрессии (минимизация дисперсии в листьях). Регуляризация дерева: максимальная глубина, минимальное число объектов для разбиения и в листе, число признаков, рассматриваемых при разбиении. <i>Инструменты:</i> StackMoreLayers, Pandas, NumPy, Scikit-Learn, Matplotlib. <i>Форма контроля:</i> Тестирование по теории. Занятие 2. Практическое занятие. Обучение решающего дерева, экспорт и разбор структуры обученного дерева, анализ отдельных разбиений. Построение графиков зависимости качества на обучающей и валидационной выборках от глубины дерева и поиск точки отсечения переобучения. Расчёт важности признаков. <i>Инструменты:</i> StackMoreLayers, Pandas, NumPy, Scikit-Learn, Matplotlib. <i>Форма контроля:</i> Практическое задание.
	Ансамбли моделей: бэггинг и случайный лес	Занятие 1. Интерактивная лекция. Идея ансамблирования: почему несколько моделей вместе работают лучше одной. Бэггинг как усреднение предсказаний моделей, обученных на бутстрап-выборках, и снижение разброса. Метод случайных подпространств. Алгоритм случайного леса и оценка

		качества по объектам вне выборки. Автоматический подбор гиперпараметров перебором по сетке с кросс-валидацией. <i>Инструменты:</i> <i>StackMoreLayers</i>, <i>Pandas</i>, <i>NumPy</i>, <i>Scikit-Learn</i>, <i>Matplotlib</i>. <i>Форма контроля:</i> Тестирование по теории. Занятие 2. Практическое занятие. Обучение случайного леса на табличном наборе данных и сравнение его с одиночным решающим деревом. Подбор гиперпараметров (число деревьев, глубина, число признаков) с кросс-валидацией, выбор лучшей комбинации, оценка качества и анализ важности признаков. <i>Инструменты:</i> <i>StackMoreLayers</i>, <i>Pandas</i>, <i>NumPy</i>, <i>Scikit-Learn</i>, <i>Matplotlib</i>. <i>Форма контроля:</i> Практическое задание.
	Градиентный бустинг для задач классификации	Занятие 1. Интерактивная лекция. Градиентный бустинг как последовательное исправление ошибок: остатки и антиградиент функции потерь, слабые деревья, снижение смещения. Высокий риск переобучения при добавлении деревьев и защита от него — ранняя остановка по валидационной выборке. Регуляризаторы бустинга: темп обучения, доля объектов и признаков. Сравнение бэггинга, случайного леса и бустинга. Обзор реализаций: <i>HistGradientBoosting</i>, <i>XGBoost</i>, <i>LightGBM</i> и <i>CatBoost</i>, особенности обработки категориальных признаков. <i>Инструменты:</i> <i>StackMoreLayers</i>, <i>Pandas</i>, <i>NumPy</i>, <i>Scikit-Learn</i>, <i>XGBoost</i>, <i>LightGBM</i>, <i>CatBoost</i>, <i>Matplotlib</i>. <i>Форма контроля:</i> Тестирование по теории. Занятие 2. Практическое занятие. Подготовка набора данных с категориальными признаками и обучение градиентного бустинга с передачей категорий без ручного кодирования. Настройка ранней остановки на валидационном блоке, подбор темпа обучения и числа деревьев, сравнение реализаций бустинга по качеству и времени обучения. <i>Инструменты:</i> <i>StackMoreLayers</i>, <i>Pandas</i>, <i>NumPy</i>, <i>Scikit-Learn</i>, <i>XGBoost</i>, <i>LightGBM</i>, <i>CatBoost</i>, <i>Matplotlib</i>. <i>Форма контроля:</i> Практическое задание.
	Соревнование и защита решения «Модель машинного обучения для прикладной задачи»	Занятие 1. Проект. Постановка задачи машинного обучения. Разбор прикладной задачи и перевод её в постановку машинного обучения (регрессия или классификация). Анализ предоставленного набора данных, выбор целевой метрики качества, разделение на обучающую и тестовую выборки, построение простого эталона качества.

			<i>Инструменты:</i> StackMoreLayers, Pandas, NumPy, Scikit-Learn. <i>Форма контроля:</i> Открытие учебного соревнования на платформе: обучающиеся изучают условие задачи, данные и метрику, по которой ранжируется таблица результатов. Занятие 2. Проект. Подготовка данных и конвейер. Предобработка данных: масштабирование числовых признаков, кодирование категориальных признаков, обработка пропусков. Аккуратное разделение выборок для исключения утечки данных. Генерация и отбор признаков. Сборка воспроизводимого конвейера обучения. <i>Инструменты:</i> StackMoreLayers, Pandas, NumPy, Scikit-Learn. <i>Форма контроля:</i> Первые отправки решений; платформа автоматически рассчитывает метрику качества и формирует таблицу результатов. Занятие 3. Проект. Обучение и подбор моделей. Обучение и сравнение моделей: линейные модели, решающие деревья, случайный лес, градиентный бустинг. Подбор гиперпараметров по валидационной выборке и кросс-валидации, анализ ошибок и важности признаков. <i>Инструменты:</i> StackMoreLayers, Pandas, NumPy, Scikit-Learn, XGBoost, LightGBM, CatBoost, Matplotlib. <i>Форма контроля:</i> Повторные отправки решений с автоматическим расчётом метрики качества и обновлением таблицы результатов. Занятие 4. Проект. Финальная модель и защита. Выбор финальной модели по метрикам с обоснованием, оценка на отложенной выборке, интерпретация результата и ограничений решения. Подготовка воспроизводимого ноутбука, краткого отчёта и презентации. Защита решения. <i>Инструменты:</i> StackMoreLayers, Pandas, NumPy, Scikit-Learn, XGBoost, LightGBM, CatBoost, Matplotlib. <i>Форма контроля:</i> Промежуточная аттестация: участие в учебном соревновании и защита решения. Финальная отправка фиксирует результат в таблице результатов; на занятии проводится защита решения перед педагогом.
3.	Введение в компьютерное зрение	Изображение как массив чисел	Занятие 1. Интерактивная лекция. Как компьютер «видит» изображение: пиксель, яркость, матрица значений. Высота, ширина и каналы изображения. Изображение как многомерный массив NumPy. Индексация и срезы по пикселям и областям.

			Простейшие преобразования: обрезка, отражение, изменение яркости через арифметику над массивом. <i>Инструменты: NumPy, Pillow, Matplotlib.</i> <i>Форма контроля:</i> Тестирование по теории. Занятие 2. Практическое занятие. Загружаем изображение, превращаем его в массив NumPy, исследуем форму и значения. Делаем срезы (вырезаем фрагмент), переворачиваем изображение, меняем яркость и контраст простыми операциями над массивом. Сохраняем результат. <i>Инструменты: NumPy, Pillow, Matplotlib.</i> <i>Форма контроля:</i> Практическое задание.
		Базовые операции с изображениями, аугментация и свёртки	Занятие 1. Интерактивная лекция. Что такое фильтр и ядро свёртки. Как свёртка по соседним пикселям даёт размытие, повышение резкости и выделение границ. Идея аугментации данных: повороты, сдвиги, отражения, шум — зачем это нужно для обучения моделей. Каналы изображения и цветовые модели: «красный — зелёный — синий» (RGB), оттенки серого, «цветовой тон — насыщенность — яркость» (HSV); когда удобно переходить в оттенки серого, а когда — в HSV. Пороговая обработка и выделение объектов по цвету. <i>Инструменты: NumPy, OpenCV, Pillow, Matplotlib.</i> <i>Форма контроля:</i> Тестирование по теории. Занятие 2. Практическое занятие. Применяем классические свёртки: размытие, повышение резкости, детектор границ. Собираем простой конвейер аугментаций над набором изображений. Конвертируем изображение между цветовыми моделями и выделяем объект заданного цвета через маску в модели HSV с пороговой обработкой. <i>Инструменты: NumPy, OpenCV, Pillow, Matplotlib.</i> <i>Форма контроля:</i> Практическое задание.
		Прикладное компьютерное зрение: OpenCV и MediaPipe	Занятие 1. Интерактивная лекция. Обзор готовых инструментов компьютерного зрения. Возможности библиотеки OpenCV для обработки и анализа изображений. Идея готовых решений в библиотеке MediaPipe: обнаружение лиц, рук и позы человека без обучения собственной модели. Где это применяется в реальных продуктах. <i>Инструменты: OpenCV, MediaPipe, NumPy, Matplotlib.</i> <i>Форма контроля:</i> Тестирование по теории. Занятие 2. Практическое занятие. Собираем мини-проект: обнаружение лица или руки в реальном времени с веб-камеры (либо на подготовленных фотографиях и видеозаписях) средствами MediaPipe и OpenCV. Рисуем разметку поверх изображения, опираясь на готовые примеры.

			<i>Инструменты: OpenCV, MediaPipe, NumPy, Matplotlib.</i> <i>Форма контроля: Практическое задание.</i>
	Экстремумы и градиентный спуск		Занятие 1. Интерактивная лекция. Минимумы, максимумы и седловые точки функции. Связь градиента с экстремумами (в точке минимума градиент равен нулю). Идея градиентного спуска: движение против градиента для спуска к минимуму. Скорость обучения (величина шага) и почему она важна: слишком большой шаг — проскакиваем минимум, слишком маленький — сходимся медленно. <i>Инструменты: NumPy, Matplotlib.</i> <i>Форма контроля: Тестирование по теории.</i> Занятие 2. Практическое занятие. Програмируем градиентный спуск с нуля: ищем минимум функции одной и двух переменных. Рисуем траекторию спуска по поверхности. Экспериментируем со скоростью обучения и наблюдаем, как меняется сходимость. <i>Инструменты: NumPy, Matplotlib.</i> <i>Форма контроля: Практическое задание.</i>
	Введение в нейронные сети и PyTorch		Занятие 1. Интерактивная лекция. Что такое нейронная сеть простыми словами: нейрон, веса, слои. Тензоры в PyTorch и автоматическое дифференцирование. Полносвязные сети. Идея обучения: ошибка, обратное распространение ошибки, шаг градиентного спуска. <i>Инструменты: NumPy, PyTorch, Matplotlib.</i> <i>Форма контроля: Тестирование по теории.</i> Занятие 2. Практическое занятие. Создаём тензоры, считаем градиенты средствами автоматического дифференцирования. Собираем первую полносвязную сеть на PyTorch и обучаем её на простой задаче классификации, наблюдая, как убывает ошибка. <i>Инструменты: NumPy, PyTorch, Matplotlib.</i> <i>Форма контроля: Практическое задание.</i>
	Регуляризация и свёрточные нейронные сети		Занятие 1. Интерактивная лекция. Переобучение и борьба с ним: прореживание (Dropout), L2-регуляризация, аугментация. Почему полносвязные сети плохо подходят для изображений. Свёрточные нейронные сети: свёрточные слои и слои подвыборки, как сеть самостоятельно выделяет признаки изображения. <i>Инструменты: PyTorch, torchvision, NumPy, Matplotlib.</i> <i>Форма контроля: Тестирование по теории.</i> Занятие 2. Практическое занятие. Обучаем свёрточную сеть классифицировать изображения (например, рукописные цифры или предметы одежды). Добавляем регуляризацию и аугментацию, сравниваем качество до и после, анализируем переобучение. <i>Инструменты: PyTorch, torchvision, NumPy, Matplotlib.</i> <i>Форма контроля: Практическое задание.</i>

		Проект 3. «CV-модель и веб-приложение»	Занятие 1. Проект. Постановка задачи компьютерного зрения. Выбор задачи компьютерного зрения (классификация изображений и аналогичные). Сбор и осмотр набора изображений, разметка, формирование обучающей и тестовой выборки, определение метрики качества. <i>Инструменты: NumPy, Pillow, OpenCV, Matplotlib.</i> <i>Форма контроля:</i> Представление постановки прикладной задачи, описания набора изображений и выбранной метрики качества. Занятие 2. Проект. Подготовка данных и аугментация. Предобработка изображений: изменение размера, нормализация, работа с каналами. Аугментация данных для расширения выборки. Сборка загрузчика данных для обучения. <i>Инструменты: OpenCV, PyTorch, torchvision, NumPy, Matplotlib.</i> <i>Форма контроля:</i> Представление подготовленного набора данных, реализованного конвейера предобработки и аугментации изображений. Занятие 3. Проект. Обучение модели. Построение и обучение свёрточной нейронной сети. Контроль переобучения (прореживание, аугментация), мониторинг функции потерь и метрик, подбор гиперпараметров обучения. <i>Инструменты: PyTorch, torchvision, NumPy, Matplotlib.</i> <i>Форма контроля:</i> Промежуточный контроль хода проекта: обучающийся предъявляет обученную модель, графики функции потерь и метрик, обоснование выбранных параметров обучения. Занятие 4. Проект. Демонстрация и защита. Оценка качества финальной модели, анализ ошибок. Разработка веб-приложения, обеспечивающего загрузку изображения или получение данных с веб-камеры, применение обученной модели и отображение результата распознавания. Проверка пользовательского сценария и обработка типовых ошибок. Оформление отчёта и презентации. Защита проекта. <i>Инструменты: PyTorch, OpenCV, MediaPipe, Matplotlib.</i> <i>Форма контроля:</i> Промежуточная аттестация: выполнение и защита прикладного проекта. На занятии проводится демонстрация работы веб-приложения и защита проекта перед педагогом.
--	--	---	--

3.2. Методические указания для обучающихся по освоению курса

Интерактивная лекция — последовательное изложение педагогом учебного материала теоретического характера, сопровождаемое разбором примеров, вопросами к аудитории и демонстрацией кода в учебном ноутбуке.

В процессе лекции рекомендуется вести конспект: кратко и схематично фиксировать основные идеи, выводы и обобщения; выделять ключевые термины и обозначения. Вопросы и материалы, вызвавшие затруднение, следует отметить и найти ответы в рекомендованных источниках. Если разобраться в материале самостоятельно не удаётся, следует сформулировать вопрос и задать его педагогу на консультации или на практическом занятии.

Практическое занятие — форма учебной деятельности, проводимая под руководством педагога, в ходе которой обучающиеся самостоятельно решают задачи на учебных наборах данных, разбирают типовые ошибки и обсуждают полученные результаты.

Для успешной работы на практическом занятии рекомендуется заранее ознакомиться с темой и материалами предшествующей лекции, проверить работоспособность учебной среды и подготовить вопросы для обсуждения. Код следует оформлять аккуратно и снабжать краткими комментариями: воспроизводимость решения является отдельным критерием оценивания.

Тестирование — проводится после освоения теоретической части темы и позволяет оценить степень понимания обучающимися основных понятий изученного материала. Тестовые задания составляются по ключевым понятиям и терминам соответствующей темы.

Для подготовки к тестированию необходимо знать основные термины темы, понимать содержание изученных понятий и уметь применять их при решении учебных задач. Результаты тестирования позволяют педагогу определить степень готовности обучающихся к выполнению практических заданий.

Соревнование — выполнение задачи на закрытых данных с автоматическим расчётом метрики качества и ранжированием решений в таблице результатов. Оценивается качество решения и его воспроизводимость, а не только позиция в таблице; на защите обучающийся должен объяснить, за счёт каких решений получен результат.

Проект — самостоятельная индивидуальная исследовательская и практическая работа с презентацией результатов.

Для успешного выполнения проекта рекомендуется: чётко определить цель и задачи проекта; составить план работы, разбив его на этапы с указанием сроков; фиксировать все проведённые эксперименты и их результаты; регулярно проверять прогресс и при необходимости корректировать план; заблаговременно подготовить воспроизводимый ноутбук, отчёт и презентацию.

Практическое задание — контрольная точка текущего контроля успеваемости, выполняемая обучающимся на учебном наборе данных по теме соответствующего практического занятия. Задание предполагает самостоятельное написание программного кода в учебном ноутбуке, который отправляется обучающимся на образовательной платформе. Платформа проверяет результат автоматически по заданным критериям и формирует рекомендацию к оценке для педагога. Оценивается корректность полученного результата, обоснованность выбранного способа решения и воспроизводимость кода.

Для успешного выполнения практического задания рекомендуется: повторить материал соответствующей лекции и разобранные на занятии примеры; выполнять задание последовательно, проверяя промежуточные результаты; сохранять код в ноутбуке в том порядке, в котором он должен выполняться, чтобы решение воспроизводилось при повторном запуске сверху вниз.

4. ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИЕ УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ

4.1. Требования к кадровым условиям реализации программы

Реализация программы на образовательных площадках осуществляется педагогическими работниками организаций, на базе которых проводится обучение.

К реализации программы могут привлекаться учителя информатики и педагоги дополнительного образования, имеющие образование и (или) профессиональную подготовку, соответствующие направленности программы.

Педагог, реализующий программу, должен владеть основами программирования на языке Python, обладать знаниями и практическими навыками в области анализа данных и машинного обучения, а также иметь опыт использования современных цифровых инструментов и сред разработки, применяемых при реализации программы.

Педагог должен быть готов к организации практической и проектной деятельности обучающихся, сопровождению выполнения ими программных и исследовательских заданий, а также применению электронного обучения и дистанционных образовательных технологий, предусмотренных программой.

4.2. Требования к материально-техническим условиям реализации программы

Для реализации программы необходимо обеспечить следующие материально-технические условия:

- наличие учебных аудиторий, оснащённых мультимедийным оборудованием, в том числе проектором и (или) интерактивной панелью;
- наличие персонального компьютера для каждого обучающегося с устойчивым доступом к информационно-телекоммуникационной сети «Интернет» и установленным современным веб-браузером;

— наличие установленной среды Jupyter Notebook, либо доступ к совместимой облачной среде, обеспечивающей возможность выполнения программного кода на языке Python;

— доступ обучающихся и педагогов к системе управления обучением (LMS) и образовательной платформе StackMoreLayers (доступы к личным кабинетам предоставляются АНО ВО «Центральный университет»);

— наличие технической возможности установки и использования в локальной среде обязательных библиотек и программных средств, необходимых для выполнения практических заданий программы: NumPy, Pandas, Matplotlib, Seaborn, SciPy, Scikit-Learn, XGBoost, LightGBM, CatBoost, Pillow, OpenCV, MediaPipe, PyTorch и torchvision, а также средства разработки простого веб-приложения;

— наличие веб-камер для выполнения практических заданий, предусмотренных разделом «Введение в компьютерное зрение»;

— наличие учебных наборов табличных данных и изображений, используемых при выполнении практических заданий и не содержащих персональных данных либо прошедших процедуру обезличивания; соответствующие учебные материалы предоставляются АНО ВО «Центральный университет».

4.3. Учебно-методическое обеспечение программы

Для реализации программы образовательная площадка обеспечивает наличие программного обеспечения, необходимого для освоения содержания программы и выполнения практических заданий, включая свободно распространяемое и (или) лицензионное программное обеспечение.

На период освоения программы обучающимся и педагогам предоставляется доступ к системе управления обучением (LMS), платформе StackMoreLayers и электронным учебно-методическим материалам программы. Доступ к указанным цифровым ресурсам и материалам предоставляется АНО ВО «Центральный университет».

Электронный документ

Учебно-методическое сопровождение каждого занятия обеспечивается комплектом материалов, включающим презентацию и (или) конспект занятия, демонстрационные материалы в формате Jupyter Notebook или совместимом формате, практические задания, учебные наборы данных, а также критерии оценивания результатов выполнения заданий.

Для организации проектной деятельности обучающихся предоставляется комплект учебно-методических материалов, включающий постановку проектной задачи, шаблон воспроизводимого ноутбука, описание используемых метрик, требования к содержанию и оформлению результатов проектной работы, а также критерии и форму её оценивания.

Учебно-методические материалы, необходимые для проведения занятий и организации проектной деятельности обучающихся, разрабатываются и предоставляются АНО ВО «Центральный университет».

Основная литература:

1. Онлайн учебник по машинному обучению от ШАД <https://education.yandex.ru/handbook/ml>.
2. Python Software Foundation. The Python Tutorial : официальная документация : [сайт]. — URL: <https://docs.python.org/3/tutorial>.
9. NumPy Developers. NumPy user guide : официальная документация : [сайт]. — URL: <https://numpy.org/doc/stable/user>.
4. The pandas development team. Getting started tutorials : официальная документация : [сайт]. — URL: https://pandas.pydata.org/docs/getting_started/intro_tutorials.

Дополнительная литература:

1. Matplotlib Development Team. Tutorials : официальная документация : [сайт]. — URL: <https://matplotlib.org/stable/tutorials>.
2. Scikit-learn developers. User Guide : официальная документация : [сайт]. — URL: https://scikit-learn.org/stable/user_guide.html.

3. OpenCV team. OpenCV-Python Tutorials : официальная документация : [сайт]. — URL: https://docs.opencv.org/4.x/d6/d00/tutorial_py_root.html.

4. PyTorch contributors. Learn the Basics : официальная документация : [сайт]. — URL: <https://docs.pytorch.org/tutorials/beginner/basics/intro.html>.

5. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ

5.1. Формы контроля

Текущий контроль успеваемости осуществляется в форме автоматизированного тестирования, выполнения практических заданий на учебных наборах данных, автоматизированной проверки программного кода на закрытых тестах и устного комментирования обучающимся выполненных решений.

При проведении текущего контроля оцениваются степень освоения теоретического материала, правильность и полнота выполнения практических заданий, работоспособность и воспроизводимость программного решения, а также способность обучающегося объяснить применённый подход.

Промежуточная аттестация проводится по завершении каждого из трёх разделов программы и предусматривает оценку результатов освоения соответствующего раздела:

- по завершении раздела 1 — выполнение и защита учебного проекта;
- по завершении раздела 2 — участие в учебном соревновании по решению практической задачи с последующей защитой полученного решения;
- по завершении раздела 3 — выполнение и защита прикладного проекта.

Итоговый результат освоения программы определяется на основании совокупности результатов трёх промежуточных аттестаций.

5.2. Система оценивания результатов обучения по курсу

Балльная структура оценивания:

Оценивание по программе является накопительным. Баллы начисляются за каждую контрольную точку по следующим правилам:

- тест к лекции — 10 баллов;
- практическое задание — 30 баллов;

— проект/соревнование по итогам раздела — 100 баллов.

Распределение баллов по разделам программы:

Раздел программы	Тесты к лекциям (10 баллов за тест)	Практические задания (30 баллов за задание)	Проект/соревнование по итогам раздела	Всего за раздел, баллов	Доля от общей суммы
Раздел 1. Pandas и статистика	$8 \times 10 = 80$	$8 \times 30 = 240$	100	420	35,6 %
Раздел 2. Введение в машинное обучение	$8 \times 10 = 80$	$8 \times 30 = 240$	100	420	35,6 %
Раздел 3. Введение в компьютерное зрение	$6 \times 10 = 60$	$6 \times 30 = 180$	100	340	28,8 %
ИТОГО:	$22 \times 10 = 220$	$22 \times 30 = 660$	$3 \times 100 = 300$	1180	100 %

Максимальная сумма баллов, которую обучающийся может набрать за весь период освоения программы, составляет 1180 баллов.

Тест к лекции считается выполненным при наборе не менее 50 % баллов теста. Практическое задание засчитывается, если программа выполняется без критических ошибок, результат соответствует условию задания, а обучающийся способен объяснить применённый подход; частично выполненное задание оценивается пропорционально доле выполненных требований.

5.3. Контроль посещаемости и минимальной успеваемости

Контроль посещаемости занятий и минимальной успеваемости проводится один раз — по завершении каждого раздела программы. Распределение контрольных точек по разделам приведено в таблице.

Раздел	Учебные недели	Контрольные точки	Максимум баллов за раздел
Pandas и статистика	1–10	Темы 1.1–1.8 (тесты и практические задания); проект 1 — выполнение и защита учебного проекта	420

Раздел	Учебные недели	Контрольные точки	Максимум баллов за раздел
		(промежуточная аттестация)	
Введение в машинное обучение	11–20	Темы 2.1–2.8 (тесты и практические задания); участие в учебном соревновании и защита решения (промежуточная аттестация)	420
Введение в компьютерное зрение	21–28	Темы 3.1–3.6 (тесты и практические задания); проект 3 — выполнение и защита прикладного проекта (промежуточная аттестация)	340
ИТОГО:	1–28		1180

Устанавливаются следующие минимальные требования по итогам разделов:

— посещаемость — не менее 50% занятий, проведённых в соответствующем разделе;

— успеваемость — не менее 30% от максимальной суммы баллов за контрольные точки соответствующих разделов (не менее 126 баллов из 420 по итогам первого раздела, не менее 126 баллов из 420 по итогам второго раздела и не менее 102 баллов из 340 по итогам третьего раздела).

Невыполнение любого из указанных требований является основанием для рассмотрения вопроса об отчислении обучающегося с программы.

5.4. Промежуточная аттестация: формат и критерии оценивания

В ходе программы проводятся три промежуточные аттестации: по завершении раздела 1 – проект «Анализ данных: от постановки задачи до выводов»; по завершении раздела 2 – соревнование и защита решения «Модель

машинного обучения для прикладной задачи»; по завершении раздела 3 – проект «CV-модель и веб-приложение».

Обучающийся представляет воспроизводимый ноутбук с решением, краткий отчёт с описанием данных, применённых методов, полученных метрик и выводов, а также презентацию результатов. Защита проводится в форме публичного выступления с ответами на вопросы педагога.

Критерии оценивания проекта 1

«Анализ данных: от постановки задачи до выводов»

Максимальный балл – 100.

Критерии оценивания	Максимальный балл
1. Качество EDA (разведочного анализа данных)	25
Первичный осмотр и оценка качества данных: типы, пропуски, дубликаты, диапазоны значений	5
Очистка данных: обработка пропусков и выбросов с обоснованием решений, приведение типов	5
Генерация признаков и сборка аналитической витрины	5
Анализ распределений и связей: описательная статистика, форма распределений, корреляции	5
Визуализация: читаемые графики нужного типа, к каждому — вывод текстом	5
2. Качество кода и ноутбука	25
Воспроизводимость: ноутбук запускается сверху вниз без ошибок, <code>random_state</code> (начальное значение генератора случайных чисел) зафиксирован	6
Структура ноутбука: логичные разделы и сквозное повествование от задачи к выводам	6
Чистота кода: осмысленные имена, повторяющаяся логика в функциях, векторизация вместо циклов	7
Комментарии и markdown: перед блоком — зачем он нужен, после расчёта — что получилось	6
3. Гипотезы и корректность выводов	25
Сформулировано не менее трёх проверяемых бизнес-гипотез с явными H_0 и H_1	6
Статистический критерий подобран под данные, выбор обоснован	7
Результат интерпретирован корректно: уровень значимости, размер эффекта, без подмены корреляции причинностью	7

Критерии оценивания	Максимальный балл
Проведена сегментация: число кластеров обосновано, каждый сегмент описан словами	5
4. Презентация и рекомендации бизнесу	25
Структура презентации: задача → данные → находки → рекомендации → следующие шаги, 8–12 слайдов	6
Ясность изложения: понятно слушателю без технического бэкграунда	6
Не менее трёх рекомендаций «сделать X → ожидаемый эффект Y» с оценкой по деньгам или метрике	8
Названы ограничения данных и анализа, риски выводов и что проверить дополнительно	5

Критерии оценивания соревнования и защиты решения

«Модель машинного обучения для прикладной задачи»

Максимальный балл – 100

Критерии оценивания	Максимальный балл
1. Подготовка данных, EDA (разведочный анализ) и воспроизводимость (баллы за подпункты суммируются)	25
Корректная постановка задачи: объект, целевая переменная, смысл прогноза, метрика ROC-AUC	5
Первичный анализ данных: размер, типы признаков, пропуски, баланс классов, базовые наблюдения	5
Честное разбиение / кросс-валидация без утечки	5
Корректная предобработка: пропуски, категории, масштабирование там, где нужно	5
Воспроизводимый pipeline (конвейер): ноутбук запускается сверху вниз	5
2. ML-подход и эксперименты (баллы за подпункты суммируются)	25
Есть baseline (эталон качества) и сравнение с ним	4
Проверено несколько осмысленных моделей	5
Подбор гиперпараметров выполнен по validation/CV (валидационной выборке и кросс-валидации), а не по leaderboard (таблице результатов)	5
Использована подходящая метрика и результаты моделей сравниваются на одной схеме проверки	4
Есть анализ важности признаков, ошибок или причин улучшения	4
Финальная модель выбрана аргументированно, а не только «потому что выше на leaderboard (в таблице результатов)»	3

Критерии оценивания	Максимальный балл
3. Leaderboard (таблица результатов); баллы выставляются по приведённой шкале	25
не ниже G — балла хорошего авторского решения на private-датасете (закрытой тестовой выборке)	25
не выше B — балла простого авторского baseline на private-датасете (закрытой тестовой выборке)	0
между B и G, верхняя четверть диапазона	20
между B и G, вторая сверху четверть диапазона	15
между B и G, вторая снизу четверть диапазона	10
между B и G, нижняя четверть диапазона	5
4. Устная защита собственного решения	25
Обучающийся объяснил архитектуру своего решения: какие шаги есть в pipeline (конвейере), в каком порядке они выполняются и зачем	5
Обучающийся объяснил свои решения по признакам и предобработке: что добавил, удалил, закодировал, масштабировал и почему	5
Обучающийся объяснил выбранные модели и настройки: почему использовал именно их, какие гиперпараметры менял, как и зачем применял регуляризацию, ограничения и early stopping (раннюю остановку)	6
Обучающийся связал эксперименты с результатом: какие изменения реально улучшили качество, какие не сработали, как он это понял	5
Обучающийся уверенно отвечает на уточняющие вопросы по своему коду и может объяснить отдельные фрагменты решения без чтения «по бумажке»	4

Критерии оценивания проекта 3 «CV-модель и веб-приложение»

Максимальный балл – 100

Критерии оценивания	Максимальный балл
1. Постановка задачи, данные и воспроизводимость	25
Ясно сформулирована прикладная задача	5
Описаны входные данные: источник, формат, классы/разметка, ограничения	5
Данные корректно подготовлены: разбиение, проверка разметки, отсутствие очевидных утечек	6
Код обучения воспроизводим: зависимости, порядок запуска, сохранение модели	5

Критерии оценивания	Максимальный балл
Есть простой baseline (эталон качества) или понятная стартовая точка для сравнения	4
2. ML/CV-решение	25
Выбран подход, соответствующий задаче	4
Корректно реализована предобработка изображений или видеопотока	4
Использована адекватная модель или библиотека для CV-задачи	5
Проведены эксперименты с настройками, моделями, аугментациями или порогами	5
Качество оценено подходящими метриками на честной проверочной выборке	4
Есть анализ ошибок модели	3
3. Веб-приложение и прикладной сценарий	25
Приложение запускается и позволяет использовать камеру или загрузку изображения/видео	5
Модель подключена к приложению и выдаёт результат в пользовательском сценарии	5
Реализовано прикладное действие на основе распознавания	5
Интерфейс понятен пользователю: видно вход, результат, статус обработки и ошибки	4
Приложение устойчиво к типовым проблемам: нет камеры, плохой кадр, неуверенное предсказание	4
Код приложения отделён от кода обучения модели	2
4. Отчёт, демонстрация и индивидуальная защита	25
Обучающийся объяснил задачу, пользовательский сценарий и выбор CV-подхода	5
Обучающийся объяснил данные, предобработку, разбиение и ограничения датасета	5
Обучающийся объяснил модель, настройки, метрики и выбор финального варианта	6
Обучающийся показал работу приложения и связь между моделью и действием в интерфейсе	5
Обучающийся уверенно отвечает на вопросы по коду, ошибкам модели и улучшениям	4

5.5. Порядок определения итогового результата освоения программы

Отдельная итоговая аттестация по программе не проводится. Итоговый результат освоения программы определяется по двум показателям: общей сумме

Электронный документ

баллов, полученных за текущий контроль и три промежуточные аттестации, и результатам каждой из трёх промежуточных аттестаций. Максимальная общая сумма составляет 1180 баллов; максимальная сумма за три промежуточные аттестации — 300 баллов.

5.6. Примерные задания текущего контроля успеваемости

Примерные задания к тестам по разделу 1 «Pandas и статистика»

Вопрос 1. Чем массив библиотеки NumPy принципиально отличается от списка языка Python?

- A. Может хранить значения разных типов в одной структуре.
- B. Хранит элементы одного типа подряд в памяти, что позволяет выполнять векторизованные операции без циклов.
- C. Не поддерживает индексацию по номеру элемента.
- D. Всегда является двумерным.

Правильный ответ: B

Вопрос 2. Что вернёт применение булевой маски к объекту DataFrame?

- A. Отсортированную таблицу.
- B. Только имена столбцов, удовлетворяющих условию.
- C. Подмножество строк, для которых условие истинно.
- D. Копию таблицы без изменений.

Правильный ответ: C

Вопрос 3. Какой тип соединения таблиц сохраняет все строки левой таблицы независимо от наличия совпадения по ключу?

- A. Внутреннее (inner).
- B. Левое (left).
- C. Правое (right).
- D. Соединения такого типа не существует.

Правильный ответ: B

Вопрос 4. Какая мера центральной тенденции наиболее устойчива к выбросам?

- A. Среднее арифметическое.
- B. Медиана.
- C. Дисперсия.
- D. Размах.

Правильный ответ: B

Вопрос 5. Что означает достигаемый уровень значимости (p-value) меньше принятого порога?

- A. Нулевая гипотеза доказана.
- B. Различие между группами заведомо велико по величине.
- C. Есть основания отвергнуть нулевую гипотезу в пользу альтернативной.
- D. Выборка сформирована некорректно.

Правильный ответ: C

Примерные задания к тестам по разделу 2

«Введение в машинное обучение»

Вопрос 1. Какая метрика регрессии сильнее штрафует крупные ошибки?

- A. Средняя абсолютная ошибка.
- B. Средняя квадратичная ошибка.
- C. Медианная ошибка.
- D. Доля правильных ответов.

Правильный ответ: B

Вопрос 2. Что показывает полнота (recall) в задаче бинарной классификации?

- A. Долю правильно предсказанных объектов среди всех объектов.
- B. Долю действительно положительных объектов, найденных моделью.
- C. Долю верных предсказаний среди объектов, отнесённых моделью к положительному классу.
- D. Число признаков, использованных моделью.

Правильный ответ: B

Вопрос 3. К чему приводит вычисление параметров масштабирования по всей выборке до её разделения на обучающую и тестовую части?

- A. К ускорению обучения без побочных эффектов.
- B. К утечке данных и завышенной оценке качества модели.
- C. К снижению числа признаков.
- D. Ни к каким последствиям.

Правильный ответ: B

Вопрос 4. Зачем в решающем дереве ограничивают максимальную глубину?

- A. Чтобы увеличить число листьев.
- B. Чтобы снизить риск переобучения.
- C. Чтобы дерево обучалось дольше.
- D. Чтобы отказаться от разделения выборки.

Правильный ответ: B

Вопрос 5. В чём состоит основная идея градиентного бустинга?

- A. Модели обучаются независимо, а их предсказания усредняются.
- B. Каждая следующая модель обучается исправлять ошибки предыдущих.
- C. Обучается одна модель максимальной глубины.
- D. Признаки отбираются случайным образом без обучения.

Правильный ответ: B

Примерные задания к тестам по разделу 3

«Введение в компьютерное зрение»

Вопрос 1. Как цветное изображение представляется в виде массива NumPy?

- A. Как одномерный массив длиной, равной числу пикселей.
- B. Как трёхмерный массив «высота × ширина × каналы».
- C. Как список строк.
- D. Как словарь координат.

Правильный ответ: В

Вопрос 2. Для чего применяется аугментация данных при обучении моделей на изображениях?

- А. Для уменьшения размера изображений.
- В. Для искусственного расширения выборки и снижения переобучения.
- С. Для перевода изображения в текстовый формат.
- Д. Для ускорения загрузки файлов с диска.

Правильный ответ: В

Вопрос 3. Что произойдёт, если задать слишком большую скорость обучения в градиентном спуске?

- А. Алгоритм гарантированно сойдётся быстрее.
- В. Алгоритм может «проскакивать» минимум и расходиться.
- С. Градиент перестанет вычисляться.
- Д. Функция потерь станет отрицательной.

Правильный ответ: В

Вопрос 4. Какую роль выполняет слой подвыборки (пулинга) в свёрточной нейронной сети?

- А. Увеличивает пространственный размер карты признаков.
- В. Уменьшает пространственный размер карты признаков и повышает устойчивость к сдвигам.
- С. Заменяет функцию потерь.
- Д. Выполняет нормализацию меток классов.

Правильный ответ: В

Вопрос 5. Что делает механизм прореживания (Dropout) при обучении нейронной сети?

- А. Удаляет объекты из обучающей выборки.
- В. Случайно отключает часть нейронов на каждом шаге обучения, снижая переобучение.
- С. Уменьшает число классов в задаче.
- Д. Фиксирует значения весов на всё время обучения.

Правильный ответ: В

Примерное описание практического задания

Тема 1.8. Разброс, аномалии и проверка статистических гипотез. По предоставленной витрине пользователей необходимо: построить диаграмму размаха трат по двум группам пользователей; выделить выбросы по правилу межквартильного размаха и указать их количество; сформулировать нулевую и альтернативную гипотезы о различии средних трат в группах; выбрать и обосновать статистический критерий; вычислить достигаемый уровень значимости и сформулировать вывод. Результат вводится на платформе и автоматически сверяется с эталоном; обоснование выбора критерия комментируется устно.

Примерное описание задания и критерии оценивания проекта:

Цель проекта — самостоятельно пройти полный цикл работы над прикладной задачей в области анализа данных, машинного обучения или компьютерного зрения: от постановки задачи и подготовки данных до обучения модели, оценки её качества и публичной защиты полученного решения.

Задачи и этапы выполнения проекта:

1. Постановка задачи. Изучение условия, предметной области и предоставленного набора данных. Формулировка решаемой задачи и вопросов к данным. Определение метрики качества, по которой будет оцениваться решение.

2. Подготовка данных. Первичный осмотр и оценка качества данных: пропуски, дубликаты, типы, аномалии. Очистка и преобразование данных, генерация признаков. Корректное разделение выборок для исключения утечки данных.

3. Построение решения. Для проекта 1 — разведочный анализ данных, проверка гипотез, сегментация. Для соревнования 2 — обучение и сравнение моделей, подбор гиперпараметров, честная валидация. Для проекта 3 — построение и обучение свёрточной нейронной сети с контролем переобучения, создание веб-приложения.

4. Оценка качества и анализ ошибок. Расчёт метрик, сравнение с эталоном качества, анализ типовых ошибок модели и ограничений полученного решения.

5. Оформление и защита. Подготовка воспроизводимого ноутбука, краткого отчёта и презентации. Публичная защита проекта с ответами на вопросы педагога. Для соревновательной работы финальная отправка решения фиксирует результат в таблице результатов.

Проект/соревнование оценивается по критериям, приведённым в пункте 5.4 настоящей программы. Максимальный балл за проект — 100. Проект считается защищённым при результате не менее 50 баллов.

6. ВЫХОДНЫЕ ДОКУМЕНТЫ

Успешное освоение программы подтверждается сертификатом установленного образца, выдаваемым образовательной организацией, на базе которой реализуется образовательная программа.

Сертификат об обучении выдаётся лицам, успешно освоившим соответствующую дополнительную общеразвивающую программу для детей, что проверяется соответствием следующим критериям:

— набрано не менее 50 % от максимальной суммы баллов по программе, то есть не менее 590 баллов из 1180;

— пройдены три промежуточные аттестации: по результатам каждой промежуточной аттестации набрано не менее 50 баллов из 100.

Обучающимся, не набравшим необходимое количество баллов по результатам промежуточной аттестации, предоставляется возможность доработать соответствующую работу и повторно пройти промежуточную аттестацию в течение двух недель после окончания соответствующего раздела.

Всем слушателям, которые успешно завершили программу, начисляются 100 XP¹ (баллы появятся в личном кабинете в течение двух недель, дополнительных действий со стороны участников или организаторов программы не требуется) и предоставляется возможность участия в XP Game² от Центрального университета, которая откроет доступ обучающимся к мерчу, закрытым выездам и событиям, личному пропуску в кампус университета, а также возможности получить до +10% к дополнительному гранту.

¹ XP («experience points») – очки опыта за участие в мероприятиях университета, выполнение заданий и привлечение друзей.

² XP Game – программа, предоставляющая возможность абитуриентам повысить грант на обучение в Центральном университете в процессе участия в эксклюзивных активностях, приглашения друзей и развития университета через выполнение заданий в течение года.