



САМАРСКИЙ
ПОЛИТЕХ
Опорный университет

МИНОБРНАУКИ РОССИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Самарский государственный технический университет»
(ФГБОУ ВО «СамГТУ»)

Центр развития современных компетенций
«Дом научной коллаборации им. Н.Н. Семёнова»

Принято на заседании
Методического совета ЦРСК
от «03» апреля 2023 г.

Протокол № 8

 М.А. Климанова

УТВЕРЖДАЮ

Проректор по образовательным
проектам и информационной
политике

 Д.Е. Овчинников
«04» апреля 2023 г.



ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ
ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА
«ОСНОВЫ АНАЛИЗА ДАННЫХ НА PYTHON»
(базовый уровень)

Направленность программы: техническая

Возраст обучающихся: 13-16 лет (7-9 классы)

Срок реализации: 1 год

Язык обучения: русский

Самара 2023

Составитель программы:

Латышев Никита Андреевич
(Фамилия, Имя, Отчество)

старший педагог дополнительного образования ЦРСК «Дом научной
коллаборации им. Н.Н. Семёнова»
(должность составителя программы)

« 10 » 03 2023 г.

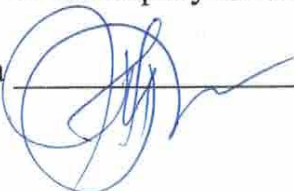
(подпись)

ДООП «Основы анализа данных на Python»
(наименование программы)

Утверждена на заседании Методического совета Центра развития современных компетенций «Дом научной коллаборации им. Н.Н. Семенова»

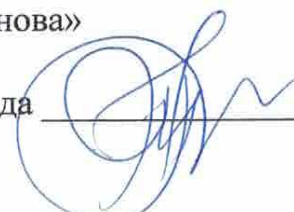
Протокол № 8 от « 3 » апрель 2023 года

Председатель Методического Совета факультета ЦРСК

« 3 » апрель 2023 года  М.А. Климанова

СОГЛАСОВАНО:

Директор Центра развития современных компетенций «Дом научной коллаборации им. Н.Н. Семенова»

« 3 » апрель 2023 года  М.А. Климанова

Настоящая программа является собственностью Самарского государственного технического университета.

Настоящая программа не может быть полностью или частично воспроизведена, тиражирована и распространена в качестве официального издания без разрешения Самарского государственного технического университета.

СОДЕРЖАНИЕ

Пояснительная записка	4
1. Содержание программы	8
2. Материально-технические условия реализации программы	188
3. Методическое обеспечение программы	18
4. Список литературы	23
5. Календарный учебный график	24

Пояснительная записка

(Организационно-педагогические условия реализации программы)

Актуальность программы заключается в необходимости знаний подобного свойства для человека современного компьютеризированного мира и времени цифровых технологий. Сфера машинного обучения в современном мире развивается экспоненциально, анализ данных – это первая ступень в освоении этой крупной области знаний. Обучающиеся по данной программе приобретают необходимые знания и навыки для анализа данных с помощью языка программирования Python, которые необходимы для понимания работы нейронных сетей и самообучающихся систем.

Новизна программы состоит в том, что содержательно она направлена на изучение методов комбинации различных модулей Python для решения поставленной задачи и предполагает глубокое погружение в тему программирования.

Отличительной особенностью программы является изучение структуры, логики использования конкретно каждого рассматриваемого модуля Python.

Педагогическая целесообразность заключается в возможности изучения методов обработки наборов данных, используемых в сфере машинного обучения, тем самым создавая необходимые условия для личностного развития, профессионального самоопределения и творческого труда современных школьников.

Программа предусматривает базовый уровень освоения содержания программы, предполагающий получение обучающимися базовых навыков анализа данных.

Цель программы – погружение в технологию анализа данных при помощи библиотек языка программирования Python.

Задачи программы:

обучающие:

- обучить работе с модулями языка Python: Pandas, Numpy, Scipy, Matplotlib и др.;

- обучить работе в IDE PyCharm или подобных ему;

- дать начальное представление о методах обработки данных и как они используются в машинном обучении;

развивающие:

- поддерживать самостоятельность в учебно-познавательной деятельности;

- способствовать самореализации и целеустремлённости;

- сформировать техническое мышление и творческий подход к работе;

- расширить ассоциативные возможности мышления;

- способствовать развитию внимания и творческих способностей обучающихся;

воспитательные:

- воспитывать коммуникативную культуру, внимание, уважение к людям;

- воспитывать трудолюбие, развить трудовые умения и навыки, расширить политехнический кругозор и умение планировать работу по реализации замысла, предвидение результата и его достижение;

- воспитывать способность к продуктивному общению и сотрудничеству со сверстниками и взрослыми в процессе творческой деятельности;

- способствовать воспитанию аккуратности, терпения, самостоятельности при выполнении работ.

Возраст детей, участвующих в реализации программы: 13-16 лет (обучающиеся с 7-9 классов общеобразовательных организаций).

Сроки реализации: программа рассчитана на 1 учебный год, объем составляет 72 часа.

Формы обучения: очная форма.

Программа предполагает ее частичную реализацию в дистанционном формате.

Формы организации деятельности:

Форма организации занятий: групповая, индивидуально-групповая. Образовательная деятельность обучающихся предусматривает следующие виды учебных занятий: лекции, практические или творческие занятия, мастер-классы, и другие виды учебных занятий.

Режим занятий: одно занятие в неделю продолжительностью 2 академических часа с перерывом.

Наполняемость учебной группы: составляет до 14 человек.

Планируемые результаты:

Предметные:

- использование основных методов обработки наборов данных;
- формирование собственных методов обработки наборов данных;
- умение разрабатывать простые программы на языке Python самостоятельно.

Метапредметные:

- самостоятельность в учебно-познавательной деятельности;
- способность ассоциативного мышления.

Личностные:

- формирование коммуникативной культуры, внимания, уважения к людям;
- проявление трудолюбия, трудовых навыков и политехнического кругозора;
- умение планировать работу, реализовывать план, предвидеть результат, при необходимости вносить коррективы.
- способность к продуктивному общению и сотрудничеству со сверстниками и взрослыми в процессе творческой деятельности.

Критерии оценки знаний, умений и навыков при освоении программы

Для того чтобы оценить усвоение программы, используются следующие методы диагностики: собеседование, наблюдение, выполнение групповых заданий, тестирование, участие в конкурсах, и др.

Применяются три уровня оценки знаний, умений и навыков обучающихся (ниже среднего, средний, выше среднего). Итоговая оценка результативности освоения программы проводится путём вычисления среднего показателя, основываясь на суммарной составляющей по итогам освоения всех модулей.

Уровень освоения программы ниже среднего – обучающийся овладел менее чем 50% предусмотренных знаний, умений и навыков, испытывает серьёзные затруднения при работе с учебным материалом; в состоянии выполнять лишь простейшие практические задания педагога.

Средний уровень освоения программы – объём усвоенных знаний, приобретённых умений и навыков составляет 50-70%; работает с учебным материалом с помощью педагога; в основном, выполняет задания на основе образца; удовлетворительно владеет теоретической информацией по темам курса, умеет пользоваться литературой.

Уровень освоения программы выше среднего – обучающийся овладел на 70-100% предусмотренным программой учебным планом; работает с учебными материалами самостоятельно, не испытывает особых трудностей; выполняет практические задания с элементами творчества; свободно владеет теоретической информацией по курсу, умеет анализировать литературные источники, применять полученную информацию на практике.

Формы контроля качества образовательного процесса:

- выполнение практических заданий;
- соревнования.

Модульный принцип построения программы предполагает описание предметных результатов в каждом конкретном модуле.

1. Содержание программы

Учебный план ДООП «Основы анализа данных на Python»

№ п/п	Название модуля	Количество часов		
		всего	теория	практика
1	Модуль 1 «Основы языка Python»	22	6,5	15,5
2	Модуль 2 «Основы анализа наборов данных»	26	8	18
3	Модуль 3 «Решающие деревья и классификаторы»	24	5,5	18,5
	Итого:	72	20	52

Рабочая программа модуля 1 «Основы языка Python»

Реализация этого модуля направлена на формирование знаний в программировании на языке Python.

Цель модуля: введение в программирование на Python.

Задачи модуля:

- сформировать общее представление о языке программирования Python;
- познакомить с терминологией направления;
- обучить пользоваться Python для решения различных задач программирования;
- сформировать базовый опыт в программировании на языке Python.

Ожидаемые результаты обучения: по итогу освоения модуля

обучающийся будет знать: понятия: «тип данных», «модуль Python», «операторы языка» и т.д.;

обучающийся будет уметь: применять полученные знания для создания простых программ на языке Python;

обучающийся сможет приобрести навык: чтения и понимания кода программы на языке программирования Python.

Учебно-тематический план модуля «Основы языка Python»

№ п/п	Наименование темы	Количество часов			Формы аттестации/контроля
		всего	теория	практика	
1	Инструктаж по технике безопасности. Введение в программирование и Python	2	1	1	Собеседование
2	Типы данных в Python	4	1	3	
3	Операторы в Python	4	1	3	
4	Операторы управления в Python	4	1	3	Выполнение задания в малых группах
5	Функции - определение и вызов функций	4	1	3	
6	Модули - Импорт и использование модулей	2	1	1	
7	Соревнование 1 «Программирование на Python»	2	0,5	1,5	Презентация работ
Итого:		22	6,5	15,5	

Освоение данного модуля позволит обучающимся сформировать компетенцию: способность использовать язык программирования Python для создания простых программ.

Содержание учебного плана модуля «Основы языка Python»

Тема 1: Инструктаж по технике безопасности. Введение в программирование и Python

Теория (1 час): вводное ознакомление о распорядке занятий, требованию к поведению в аудитории и работы с оборудованием и прочими элементами, включенными в образовательную программу согласно нормативам ТБ с последующим внесением результатов в журнал ТБ. Беседа об использовании языка Python в современном мире.

Практика (1 час): знакомство с языком программирования Python, изучение базового синтаксиса и структуры Python, создание программы «Hello world!».

Формы аттестации/контроля: Групповое собеседование в конце занятия по изученной теме.

Тема 2: Типы данных в Python

Теория (1 час): что такое типы данных и чем они отличаются друг от друга.

Практика (3 часа): изучение типов данных в Python - целое число, число с плавающей запятой, строка, логическое значение и методов работы с ними.

Формы аттестации/контроля: Групповое собеседование в конце занятия по изученной теме.

Тема 3: Операторы в Python

Теория (1 час): для чего нужны операторы, особенности использования.

Практика (3 часа): изучение основных операторов - арифметика, сравнение, логика переменные и выражения.

Формы аттестации/контроля: Групповое собеседование в конце занятия по изученной теме.

Тема 4: Операторы управления в Python

Теория (1 час): для чего нужны операторы управления, особенности использования.

Практика (3 часа): изучение условных операторов - if, else, elif; операторов цикла - for, while; создание блок-схем процессов.

Формы аттестации/контроля: Выполнения практического задания в малых группах: создание программы по теме занятия с использованием изученных приемов и методов.

Тема 5: Функции - определение и вызов функций

Теория (1 час): особенности определения и вызова функция в Python.

Практика (3 часа): определение и вызов своих функций для различных задач.

Формы аттестации/контроля: Выполнения практического задания в малых группах: создание программы по теме занятия с использованием изученных приемов и методов.

Тема 6: Модули - Импорт и использование модулей

Теория (1 час): для чего используется `import ... as ...` и особенности подключения модулей

Практика (1 час): работа с модулем `matplotlib` для изображения графиков.

Формы аттестации/контроля: Выполнение практического задания в малых группах: создание программы по теме занятия с использованием изученных приемов и методов.

Соревнование 1 «Программирование на Python»

Теория (0.5 часа): Прием работ

Практика (1.5 часа): Оценка работ и подведение итогов соревнования.

Рабочая программа модуля 2 «Основы анализа наборов данных»

Реализация этого модуля направлена на формирование начальных знаний в сфере анализа данных.

Цель модуля: знакомство с библиотеками для анализа различных объемов входных данных.

Задачи модуля:

- сформировать общее понимание анализа наборов данных;
- познакомить с терминологией направления;
- обучить использовать модули `Pandas`, `Matplotlib`, `numpy` и `Scipy` для анализа наборов данных;
- сформировать базовый опыт анализа наборов данных на языке Python.

Ожидаемые результаты обучения: по итогу освоения модуля

Обучающийся будет знать: как импортировать и экспортировать данные в Python, как очищать и предварительно обрабатывать данные.

Обучающийся будет уметь: визуализировать данные с помощью `Matplotlib`; манипулировать данными с помощью `Pandas`.

Обучающийся сможет приобрести навык: исследовать, манипулировать и визуализировать наборы данных с помощью библиотек Python.

Учебно-тематический план модуля «Основы анализа наборов данных»

№ п/п	Наименование темы	Количество часов			Формы аттестации/контроля
		всего	теория	практика	
1	Введение в анализ наборов данных с помощью Python	2	1	1	Собеседование
2	Импорт и экспорт данных в Python.	4	1	3	
3	Знакомство с Pandas.	4	1	3	
4	Работа с данными с помощью Pandas	4	1	3	Выполнение задания в малых группах
5	Анализ данных с помощью Numpy	4	1	3	
6	Статистический анализ с помощью Numpy и Scipy	4	1	3	
7	Визуализация данных с помощью Matplotlib	2	1	1	
8	Соревнование 2 «Создание набора данных»	2	1	1	Презентация работ
Итого:		26	8	18	

Освоение данного модуля позволит обучающимся сформировать компетенции:

- способность анализировать наборы данных с помощью языка Python и модулей Pandas, Numpy, Scipy;
- способность визуализировать данные с помощью Matplotlib.

Содержание учебного плана модуля «Основы анализа наборов данных»

Тема 1: Введение в анализ наборов данных с помощью Python

Теория (1 час): что такое набор данных и что входит в его анализ.

Практика (1 час): обзор библиотек Python для анализа наборов данных, установка необходимых модулей.

Формы аттестации/контроля: Групповое собеседование в конце занятия по изученной теме.

Тема 2: Импорт и экспорт данных в Python

Теория (1 час): типы файлов, и особенности экспорта различных типов файлов.

Практика (3 часа): чтение и запись файлов данных; обработка недостающих данных.

Формы аттестации/контроля: Групповое собеседование в конце занятия по изученной теме.

Тема 3: Знакомство с Pandas

Теория (1 час): введение в Pandas и его особенности.

Практика (3 часа): Pandas DataFrames и Series, индексирование и выбор данных в Pandas.

Формы аттестации/контроля: Групповое собеседование в конце занятия по изученной теме.

Тема 4: Работа с данными с помощью Pandas

Теория (1 час): особенности работы с данными в Pandas.

Практика (3 часа): фильтрация, сортировка и группировка данных, объединение наборов данных

Формы аттестации/контроля: Выполнения практического задания в малых группах: создание программы по теме занятия с использованием изученных приемов и методов.

Тема 5: Повторение инструктажа по ТБ. Анализ данных с помощью Numpy

Теория (1 час): повторение требований к поведению в аудитории и работы с оборудованием и прочими элементами, включенными в образовательную программу согласно нормативам ТБ с последующим внесением результатов в журнал ТБ. Введение в Numpy и его особенности.

Практика (3 часа): массивы и типы данных в Numpy, индексация и разделение массива.

Формы аттестации/контроля: Выполнение практического задания в малых группах: создание программы по теме занятия с использованием изученных приемов и методов.

Тема 6: статистический анализ с помощью Numpy и Scipy

Теория (1 час): статистический анализ – понятие и способы реализации

Практика (3 часа): распределения вероятностей, проверка гипотез с помощью Scipy

Формы аттестации/контроля: Выполнение практического задания в малых группах: создание программы по теме занятия с использованием изученных приемов и методов.

Тема 7: Визуализация данных с помощью Matplotlib

Теория (1 час): введение в Matplotlib и его возможности.

Практика (1 час): основные функции в Matplotlib, отображение и настройка графиков в Matplotlib.

Формы аттестации/контроля: Выполнение практического задания в малых группах: создание программы по теме занятия с использованием изученных приемов и методов.

Соревнование 2 «Создание набора данных»

Теория (0.5 часа): Прием работ

Практика (1.5 часа): Оценка работ и подведение итогов соревнования.

Рабочая программа модуля 3 «Решающие деревья и классификаторы»

Реализация этого модуля направлена на формирование начальных знаний в теме решающих деревьев и классификаторов.

Цель модуля: рассмотрение технологии работы с библиотекой Scikit-learn для создания деревьев решений и классификаторов.

Задачи модуля:

- сформировать общее понимание основ решающих деревьев и классификаторов;
- познакомить с терминологией направления;

– обучить использовать модуль `scikit-learn` для построения различных моделей машинного обучения;

– сформировать базовый опыт в создании решающих деревьев и классификаторов на языке Python.

Ожидаемые результаты обучения: по итогу освоения модуля

Обучающийся будет знать: как импортировать и экспортировать данные в Python; как очищать и предварительно обрабатывать данные.

Обучающийся будет уметь: ориентироваться в модуле `scikit-learn`.

Обучающийся сможет приобрести навык: исследовать, создавать и эксплуатировать модели машинного обучения, используя модуль `scikit-learn`.

Учебно-тематический план модуля «Решающие деревья и классификаторы»

№ п/п	Наименование темы	Количество часов			Формы аттестации/контроля
		всего	теория	практика	
1	Введение в деревья решений	2	1	1	Собеседование, проверочная работа
2	Предварительная обработка данных	2	0,5	1,5	Выполнение задания в малых группах
3	Построение деревьев решений	2	0,5	1,5	Выполнение задания в малых группах
4	Оценка деревьев решений	2	0,5	1,5	Выполнение задания в малых группах
5	Введение в классификаторы	2	1	1	Собеседование, проверочная работа
6	Построение классификаторов	2	0,5	1,5	Выполнение задания в малых группах
7	Кейс «Классификация объектов»	12	1,5	10,5	Презентация решения
	Итого:	24	5,5	18,5	

Содержанием модуля предусмотрено решение кейса «Классификация объектов», пояснение которого размещено в разделе 3 программы.

Освоение данного модуля позволит обучающимся сформировать компетенцию: способность разбираться в теме машинного обучения на начальном уровне, и создавать простые решающие деревья и классификаторы, используя модуль Python `scikit-learn`.

Содержание учебного плана модуля «Решающие деревья и классификаторы»

Тема 1: Введение в деревья решений

Теория (1 час): введение в деревья решений: что такое деревья решений и почему они полезны, концепция деревьев решений: как деревья решений представляют решения и результаты, основные термины, используемые в деревьях решений: определения ключевых терминов, таких как корневой узел, конечный узел, разделение и энтропия

Практика (1 час): как работают деревья решений: алгоритм деревьев решений и как они принимают решения, введение в модуль `scikit-learn` в Python: обзор библиотеки, используемой для построения деревьев решений и классификаторов.

Формы аттестации/контроля: Групповое собеседование в конце занятия по изученной теме.

Тема 2: Предварительная обработка данных

Теория (0.5 часа): удаление ненужных или повторяющихся данных, стратегии работы с отсутствующими значениями в наборе данных

Практика (1.5 часа): кодирование категориальных данных: преобразование нечисловых данных в числовые значения для анализа, разделение набора данных на наборы для обучения и тестирования: разделение данных на подмножества для обучения и тестирования модели.

Формы аттестации/контроля: Выполнения практического задания в малых группах: создание программы по теме занятия с использованием изученных приемов и методов.

Тема 3: Построение деревьев решений

Теория (0.5 часа): использование библиотеки `scikit-learn` для построения модели дерева решений.

Практика (1.5 часа): создание деревьев решений на Python с помощью модуля `scikit-learn`, варианты дерева решений: различные параметры и

варианты настройки модели, визуализация деревьев решений: как визуализировать модель дерева решений и интерпретировать ее результаты

Формы аттестации/контроля: Выполнения практического задания в малых группах: создание программы по теме занятия с использованием изученных приемов и методов.

Тема 4: Оценка деревьев решений

Теория (0.5 часа): метрики для оценки точности и эффективности модели дерева решений.

Практика (1.5 часа): инструменты для анализа производительности модели, методы повышения производительности модели.

Формы аттестации/контроля: Выполнения практического задания в малых группах: создание программы по теме занятия с использованием изученных приемов и методов.

Тема 5: Введение в классификаторы

Теория (1 час): основные понятия классификаторов: что такое классификаторы и как они работают, как деревья решений вписываются в более широкую категорию классификаторов

Практика (1 час): обзор различных типов классификаторов.

Формы аттестации/контроля: Групповое собеседование в конце занятия по изученной теме.

Тема 6: Построение классификаторов

Теория (0.5 часа): сравнение сильных и слабых сторон различных типов классификаторов и когда использовать каждый тип.

Практика (1.5 часа): создание классификаторов на Python с помощью модуля scikit-learn: как использовать библиотеку для создания разных типов классификаторов, оценка классификаторов эффективности разных классификаторов

Формы аттестации/контроля: Выполнение практического задания в малых группах: создание программы по теме занятия с использованием изученных приемов и методов.

Кейс «Классификация объектов»

Теория - 1,5 часа, Практика – 10,5 часов.

2. Материально-технические условия реализации программы

Занятия проводятся в аудитории для проведения теоретических и практических занятий, оснащенной достаточным количеством рабочих мест. Во время занятий обеспечивается доступ к сети Интернет.

Аудитория оснащена мебелью и необходимым мультимедийным оборудованием, компьютерами или ноутбуками применительно к содержанию представленных модулей и др. расходными материалами по реализации программы.

Программное обеспечение: IDE PyCharm; Оболочка Python 3.11 или его аналоги; ПО для работы с текстовыми файлами и .csv файлами (notepad ++ или его аналоги).

3. Методическое обеспечение программы

Занятия по программе строятся с учетом обучающей, воспитательной и развивающей стороны педагогического процесса:

- лекция-диалог с использованием метода «перевернутый класс» – когда обучающимся предлагается к следующему занятию ознакомиться с материалами (в том числе найденными самостоятельно) на определенную тему для обсуждения в формате диалога на предстоящем занятии;

- Workshop и Tutorial (практическое занятие – hard skills), что, по сути, является разновидностями мастер-классов, где обучающимся предлагается выполнить определенную работу, результатом которой является некоторый продукт (физический или виртуальный результат). Близкий аналог – фронтальная форма работы, когда обучающиеся синхронно работают под контролем педагога;

- конференции, на которых обучающиеся делятся опытом друг с другом и рассказывают о собственных достижениях;

- самостоятельная работа, когда обучающиеся выполняют индивидуальные задания в течение части занятия или нескольких занятий.

- метод кейсов (case-study), «мозговой штурм» (Brainstorming), метод задач (Problem-Based Learning) и метод проектов (Project-Based Learning). Пример: кейс – это конкретная задача («случай» – case, англ.), которую требуется решить, для этого в режиме «мозгового штурма» предлагаются варианты решения, после этого варианты обсуждаются, и выбирается один или несколько путей решения, после чего для решения кейса формируются более мелкие задачи, которые объединяются в проект и реализуются с применением метода командообразования.

Система занятий должна вести к формированию следующих характеристик творческих способностей: критическое мышление, гибкость ума, оригинальность, любознательность, умение выдвигать и разрабатывать гипотезы.

В процессе обучения используются и комбинируются следующие методы обучения:

- репродуктивный;
- объяснительно-иллюстративный;
- исследовательский.

Изложение теоретического материала занятий может осуществляться с помощью традиционных словесных и наглядных методов: рассказ, беседа, демонстрация видеоматериалов, наглядного материала, различного оборудования.

На занятиях применяются различные формы работы, такие как групповые, парные, индивидуальные.

На последнем этапе обучения предполагается выполнение кейса классификация объектов, в процессе решения которого должны быть исследованы задачи классификации объектов.

Оценочные материалы по программе:

Кейс «Классификация объектов»

Реализация этого кейса направлена на формирование начальных знаний в способах классификации объектов по нескольким признакам.

Цель кейса: создание с помощью модуля scikit-learn классификатора объектов.

Задачи кейса:

- сформировать общее понимание задачи классификации объектов и в каких сферах используются изученные решения;
- познакомить с терминологией направления;
- обучить использовать модуль scikit-learn для решения конкретной задачи: классификация объектов;
- сформировать базовый опыт в создании и эксплуатации решающих деревьев и классификаторов на языке Python.

Ожидаемые результаты обучения: по итогу решения кейса

Обучающийся будет знать: как создать с нуля необходимый набор данных, как использовать модуль scikit-learn для решения задач классификации объектов.

Обучающийся будет уметь: использовать модуль scikit-learn для самостоятельного решения простых задач.

Обучающийся сможет приобрести навык: исследовать, создавать и использовать классификаторы, используя модуль scikit-learn.

План кейса «Классификация объектов»

№ п/п	Наименование этапа работы	Количество часов			Формы аттестации/контроля
		всего	теория	практика	
1	Постановка задачи, создание или выбор набора данных для классификации, анализ выбранного набора	4	0,5	3,5	Выполнение задания в малых группах или индивидуально
2	Выбор признаков для классификации, создание	4	0,5	3,5	Выполнение задания в малых группах или индивидуально

	классификатора.				
3	Продолжение создания классификатора объектов, тестирование	4	0,5	3,5	Выполнение задания в малых группах или индивидуально
	Итого:	12	1,5	10,5	

Соревнования

Соревнование 1 «Программирование на Python»

Описание соревнования

Участники создают программу на Python на любую тему, какую считают подходящей, без использования каких-либо модулей и расширений Python. Основное требование к программам – читаемость кода, названия переменных и методов не должно быть просто набором букв, а отражать смысл выполнения.

Участники соревнуются между собой в режиме: каждый сам за себя.

Срок реализации: соревнование рассчитано на дистанционный формат реализации, время индивидуальной работы над программой ограничено лишь сроком от объявления соревнования и заканчивая сроками сдачи работ.

Требования к оборудованию:

ПК, оснащенный ПО для программирования в среде Python.

Тематика: свободная.

Правила соревнования:

Все программы должны быть созданы лично участником.

Поэтапный график соревнования

№ п/п	Этап соревнования	Продолжительность этапа, мин.
1	Анонс соревнования	10
2	Дистанционный этап: создание программы	-
3	Прием работ участников	10
4	Оценка работ	60
5	Подведение итогов	10

Развиваемые компетенции:

- способность создания программ на языке Python;
- способность к творческому подходу в решении проблемы;

- способность к самостоятельной работе;
- готовность к работе в условиях ограниченного времени.

Соревнование 2 «Создание набора данных»

Описание соревнования

Участники создают набор данных из открытых источников, либо сами набирают данные с помощью различных методов, например анкетирование других людей. Задача соревнования: создать набор данных, подходящих под условия программы, созданной педагогом. Если удалось классифицировать объект, используя набор данных – соревнование пройдено, если программа не работает, выдает ошибки и т.д. – соревнование не пройдено.

Участники соревнуются между собой в режиме: каждый сам за себя.

Срок реализации: соревнование рассчитано на дистанционный формат реализации, время индивидуальной работы над набором данных ограничено лишь сроком от объявления соревнования и заканчивая сроками сдачи работ.

Требования к оборудованию:

ПК, оснащенный ПО для программирования в среде Python.

Тематика: свободная.

Правила соревнования:

Набор данных, должен быть создан лично участником.

Поэтапный график соревнования

№ п/п	Этап соревнования	Продолжительность этапа, мин.
1	Анонс соревнования	10
2	Дистанционный этап: создание набора данных	-
3	Прием работ участников	10
4	Оценка работ	60
5	Подведение итогов	10

Развиваемые компетенции:

- способность создавать набор данных под конкретные требования с использованием модуля Pandas;
- способность к творческому подходу в решении проблемы;

- способность к самостоятельной работе;
- готовность к работе в условиях ограниченного времени.

4. Список литературы

Основная литература

1. Лутц М. Изучаем Python, том 1 [Текст] / Лутц М. — 5-е изд.: Пер. с англ. . — СПб.: Диалектика, 2019 — 833 с.
2. Маккинни У. Python и анализ данных: Первичная обработка данных с применением pandas, NumPy и Jupiter [Текст] / пер. с англ. А. Слинкина. 3-е изд. - М.: МК Пресс, 2023. - 536 с.

Дополнительная литература

1. Могилев, А.В., Пак, Н.И. Информатика: учеб. пособие для студ. пед. вузов [Текст] / А.В. Могилев, Н.И. Пак — 3-е изд. — М.: АКАДЕМИЯ, 2004. — 848 с.
2. Николенко, С.И., Тулупьев, А.Л. Самообучающиеся системы. [Текст] / С.И. Николенко, А.Л. Тулупьев — 1-е изд. — М.: МЦНМО, 2009. — 289 с.
3. Фридланд, А.Я., Фридланд, И.А. Информатика и компьютерные технологии: Основные термины: Толков. слов.: Более 1000 базовых понятий и терминов [Текст] / А.Я. Фридланд, И.А. Фридланд — 3-е изд. — М.: АСТ, 2003 — 272 с.
4. Херманс Ф. Ум программиста: как понять и осмыслить любой код [Текст] / Херманс Ф. — 1-е изд. — СПб.: БХВ-Петербург, 2023 — 274 с.

Интернет-ресурсы

1. Welcome to Python / [Электронный ресурс] // Python: [сайт]. — URL: <https://www.python.org> (дата обращения: 30.03.2023).
2. Анализ данных с использованием Python/Хабр / [Электронный ресурс] // Хабр: [сайт]. — URL: <https://habr.com/ru/post/353050/> (дата

обращения: 30.03.2023).

5. Календарный учебный график

Год обучения	Дата начала обучения по программе	Дата окончания обучения по программе	Всего учебных недель	Кол-во часов	Режим занятий
2023-2024	01.09.2023	31.05.2024	36	72	1 раз в неделю по 2 акад. часа