

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Воронежский государственный технический университет»

УТВЕРЖДАЮ

Декан факультета информационных технологий и
компьютерной безопасности

П.Ю. Гусев / _____ /

31.08.2021 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
«Системы искусственного интеллекта»

Специальность 10.05.01 Компьютерная безопасность

Специализация № 4 "Безопасность компьютерных систем и сетей (связь, информационные и коммуникационные технологии)"

Квалификация выпускника специалист по защите информации

Нормативный период обучения 5 лет и 6 м.

Форма обучения очная

Год начала подготовки 2021

Автор программы _____ А.Е. Дешина

Заведующий кафедрой
Систем информационной
безопасности _____ А.Г. Остапенко

Руководитель ОПОП _____ К.А. Разинкин

Воронеж 2021

1. ЦЕЛИ И ЗАДАЧИ ДИСЦИПЛИНЫ

1.1. Цели дисциплины методах, средствах и языках, используемых при разработке систем искусственного интеллекта; ознакомить студентов с основными методами поиска решений, применяемых в системах искусственного интеллекта; сформировать у студента аналитические способности, которые бы позволяли ему делать обоснованный выбор изученных методов, средств и языков при решении задач из проблемной области.

1.2. Задачи освоения дисциплины формирование в рамках компетентностно-ориентированного подхода знаний, умений и навыков направленных на: изучение основных методов машинного обучения для работы с табличными данными, систем глубокого обучения и обучения с подкреплением.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП

Дисциплина «Системы искусственного интеллекта» относится к дисциплинам части, формируемой участниками образовательных отношений блока Б1.

3. ПЕРЕЧЕНЬ ПЛАНИРУЕМЫХ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ

Процесс изучения дисциплины «Системы искусственного интеллекта» направлен на формирование следующих компетенций:

ПК-4.8 - Способен использовать знание основных методов искусственного интеллекта в последующей профессиональной деятельности в качестве научных сотрудников, преподавателей образовательных организаций высшего образования, инженеров, технологов

ПК-4.9 - Способен выявить естественнонаучную сущность проблем, возникающих в ходе профессиональной деятельности в области моделирования и анализа сложных естественных и искусственных систем

Компетенция	Результаты обучения, характеризующие сформированность компетенции
ПК-4.8	знать методы разработки оригинальных алгоритмов и программных решений с использованием современных технологий
	уметь применять методы машинного обучения, подготавливать данные и интерпретировать результаты
	владеть подходами к оцениванию применимости алгоритмов, возможных рисков и последствий ошибок, нахождением оптимальных решений для рабочих задач
ПК-4.9	знать принципы обучения и применения нейронных сетей, архитектуры глубоких нейронных сетей,

	теоретических основ и алгоритмов обучения с подкреплением применяемых в решении практических задач
	уметь настраивать необходимое окружение для работы с нейронными сетями, применять и дообучать предобученные нейронные сети из доступных библиотек, выбирать и реализовывать алгоритмы обучения с подкреплением с учетом специфики задачи, осуществлять адаптацию и настройку алгоритмов обучения с подкреплением под определенную среду
	владеть навыком проведения полного цикла вычислительного эксперимента, отражения хода выполнения проекта и получения результатов в отчетах и документации; владеет навыком использования существующих программных библиотек и моделей, создания программных реализаций глубоких нейронных сетей; владеть навыками декомпозиции, формализации процессов и объектов для использования интеллектуальных программных решений

4. ОБЪЕМ ДИСЦИПЛИНЫ

Общая трудоемкость дисциплины «Системы искусственного интеллекта» составляет 2 з.е.

Распределение трудоемкости дисциплины по видам занятий
очная форма обучения

Виды учебной работы	Всего часов	Семестры
		8
Аудиторные занятия (всего)	54	54
В том числе:		
Лекции	18	18
Практические занятия (ПЗ)	18	18
Лабораторные работы (ЛР)	18	18
Самостоятельная работа	18	18
Виды промежуточной аттестации - зачет	+	+
Общая трудоемкость:		
академические часы	72	72
зач.ед.	2	2

5. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

5.1 Содержание разделов дисциплины и распределение трудоемкости по видам занятий очная форма обучения

№ п/п	Наименование темы	Содержание раздела	Лекц	Прак зан.	Лаб. зан.	СРС	Всего, час
1	Введение в искусственный интеллект и основные методы машинного обучения для работы с табличными данными	Основные задачи систем искусственного интеллекта. Классификация, кластеризация, регрессия. Типы машинного обучения: с учителем, без учителя, с частичным привлечением учителя, обучение с подкреплением. Классификация на примере алгоритма k-ближайших соседей (kNN). Метрики оценки классификации: полнота, точность, F1, ROC, AUC. Валидационная и тестовая выборка. Кросс-валидация. Работа с категориальными признаками. Регрессия. Метрики оценки регрессии: MSE, MAE, R2 - коэффициент детерминации. Линейная регрессия, полиномиальная регрессия. Переобучение и регуляризация, гребневая регрессия, LASSO, Elastic Net. Линейные модели для классификации. Перцептрон, логистическая регрессия, полносвязные нейронные сети, стохастический градиентный спуск и обратное распространение градиента. Регуляризация линейных моделей классификации. Кластеризация. k-means, k-means++, DBSCAN, агломеративная кластеризация. Метрики оценки кластеризации. Алгоритмы, основанные на применении решающих деревьев. Критерии разделения узла: информационный выигрыш, критерий Джини. Ансамбли решающих деревьев: случайный лес, градиентный бустинг. Метод опорных векторов. Прямая и обратная задача. Определение опорных векторов. Ядерный трюк. Наивный байесовский классификатор. Методы оценки распределения признаков. ЕМ-алгоритм на примере смеси гауссиан.	10	10	8	7	35

		Методы безградиентной оптимизации: случайный поиск, hill climb, отжиг, генетический алгоритм.					
2	Системы глубокого обучения	Нейронные сети. Функции ошибки нейронных сетей и обучение с помощью обратного распространения градиента. Понятие батча и эпохи. Работа с изображениями с помощью нейронных сетей. Сверточные нейронные сети. Операции свертки, max-pooling. Архитектуры сверточных нейронных сетей: AlexNet, VGG, Inception (GoogLeNet), ResNet. Трансферное обучение. Обработка текстов. Работа с естественным языком с помощью нейронных сетей. Векторные представления для текста: word2vec, skipgram, CBOW, fasttext. Рекуррентные нейронные сети, LSTM, GRU. Трансформеры, BERT, GPT.	5	4	6	6	21
3	Обучение с подкреплением	Понятия агента, среды, состояния, действий и награды. Функция ценности состояния (Value function) и функция качества действия (Q- function). Оптимизация стратегии с помощью максимизации функций ценности и качества. Q-обучение. Глубокое обучение с подкреплением. Deep Q-Networks, Actor-critic. Для уровня экспертный: REINFORCE, A2C, PPO, DDPG.	3	4	4	5	16
Итого			18	18	18	18	72

5.2 Перечень лабораторных работ

1. Методы работы с таблицами в Python. Агрегация и визуализация данных. Проведение первичного анализа данных.
2. Использование и сравнение алгоритмов классификации: kNN, решающие деревья и их ансамбли, логистическая регрессия.
3. Использование и оценка алгоритмов регрессии. Подбор оптимальных параметров регрессии.
4. Оптимизационные задачи и их решения. Подбор гиперпараметров алгоритма с помощью методов оптимизации.
5. Классификация изображений и трансферное обучение.
6. Работа с текстами и их векторными представлениями текстов.
7. Применение Q-Networks для решения простых окружений.

6. ПРИМЕРНАЯ ТЕМАТИКА КУРСОВЫХ ПРОЕКТОВ (РАБОТ)

И КОНТРОЛЬНЫХ РАБОТ

В соответствии с учебным планом освоение дисциплины не предусматривает выполнение курсового проекта (работы) или контрольной работы.

7. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ

7.1. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания

7.1.1 Этап текущего контроля

Результаты текущего контроля знаний и межсессионной аттестации оцениваются по следующей системе:

«аттестован»;

«не аттестован».

Компетенция	Результаты обучения, характеризующие сформированность компетенции	Критерии оценивания	Аттестован	Не аттестован
ПК-4.8	знать методы разработки оригинальных алгоритмов и программных решений с использованием современных технологий	знание методов разработки оригинальных алгоритмов и программных решений с использованием современных технологий	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
	уметь применять методы машинного обучения, подготавливать данные и интерпретировать результаты	умение применять методы машинного обучения, подготавливать данные и интерпретировать результаты	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
	владеть подходами к оцениванию применимости алгоритмов, возможных рисков и последствий ошибок, нахождением оптимальных решений для рабочих задач	владение подходами к оцениванию применимости алгоритмов, возможных рисков и последствий ошибок, нахождением оптимальных решений для рабочих задач	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
ПК-4.9	знать принципы обучения и применения нейронных сетей, архитектуры глубоких нейронных сетей, теоретических основ и алгоритмов обучения с подкреплением применяемых в решении практических задач	знание принципов обучения и применения нейронных сетей, архитектуры глубоких нейронных сетей, теоретических основ и алгоритмов обучения с подкреплением применяемых в решении практических задач	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
	уметь настраивать необходимое окружение для работы с	умение настраивать необходимое окружение для работы с нейронными сетями,	Выполнение работ в срок, предусмотренный в	Невыполнение работ в срок, предусмотренный в

	нейронными сетями, применять и дообучать предобученные нейронные сети из доступных библиотек, выбирать и реализовывать алгоритмы обучения с подкреплением с учетом специфики задачи, осуществлять адаптацию и настройку алгоритмов обучения с подкреплением под определенную среду	применять и дообучать предобученные нейронные сети из доступных библиотек, выбирать и реализовывать алгоритмы обучения с подкреплением с учетом специфики задачи, осуществлять адаптацию и настройку алгоритмов обучения с подкреплением под определенную среду	рабочих программах	рабочих программах
	владеть навыком проведения полного цикла вычислительного эксперимента, отражения хода выполнения проекта и получения результатов в отчетах и документации; владеет навыком использования существующих программных библиотек и моделей, создания программных реализаций глубоких нейронных сетей; владеть навыками декомпозиции, формализации процессов и объектов для использования интеллектуальных программных решений	владение навыком проведения полного цикла вычислительного эксперимента, отражения хода выполнения проекта и получения результатов в отчетах и документации; владеет навыком использования существующих программных библиотек и моделей, создания программных реализаций глубоких нейронных сетей; владеть навыками декомпозиции, формализации процессов и объектов для использования интеллектуальных программных решений	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах

7.1.2 Этап промежуточного контроля знаний

Результаты промежуточного контроля знаний оцениваются в 8 семестре для очной формы обучения по двухбалльной системе:

«зачтено»

«не зачтено»

Компетенция	Результаты обучения, характеризующие сформированность компетенции	Критерии оценивания	Зачтено	Не зачтено
ПК-4.8	знать методы разработки оригинальных алгоритмов и программных решений с использованием современных технологий	Тест	Выполнение теста на 70-100%	Выполнение менее 70%
	уметь применять методы машинного обучения, подготавливать данные и интерпретировать	Решение стандартных практических задач	Продемонстрировать верный ход решения в большинстве задач	Задачи не решены

	результаты владеть подходами к оцениванию применимости алгоритмов, возможных рисков и последствий ошибок, нахождением оптимальных решений для рабочих задач	Решение прикладных задач в конкретной предметной области	Продемонстрировать верный ход решения в большинстве задач	Задачи не решены
ПК-4.9	знать принципы обучения и применения нейронных сетей, архитектуры глубоких нейронных сетей, теоретических основ и алгоритмов обучения с подкреплением применяемых в решении практических задач	Тест	Выполнение теста на 70-100%	Выполнение менее 70%
	уметь настраивать необходимое окружение для работы с нейронными сетями, применять и дообучать предобученные нейронные сети из доступных библиотек, выбирать и реализовывать алгоритмы обучения с подкреплением с учетом специфики задачи, осуществлять адаптацию и настройку алгоритмов обучения с подкреплением под определенную среду	Решение стандартных практических задач	Продемонстрировать верный ход решения в большинстве задач	Задачи не решены
	владеть навыком проведения полного цикла вычислительного эксперимента, отражения хода выполнения проекта и получения результатов в отчетах и документации; владеет навыком использования существующих программных библиотек и моделей, создания программных реализаций глубоких нейронных сетей; владеть навыками декомпозиции, формализации процессов и объектов для использования интеллектуальных программных решений	Решение прикладных задач в конкретной предметной области	Продемонстрировать верный ход решения в большинстве задач	Задачи не решены

7.2 Примерный перечень оценочных средств (типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности)

7.2.1 Примерный перечень заданий для подготовки к тестированию

1. Что, из ниже перечисленного, относится к обучающей выборке?

классификация данных

объекты с известными ответами +

алгоритм решающий функцию

2. Объекты состоят из признаков?

Да +

Нет

3. Что называют данными в машинном обучении?

матрицы +

объекты +

признаки +

алгоритм

функция

4. Выберите правильный ответ. Задача классификации - это:

множество объектов, разделенных на классы +

исследование влияние одного или нескольких признаков на объект

определение порядка признака согласно рангу

5. Выберите правильный ответ. Задача регрессии - это:

множество объектов, разделенных на классы

исследование влияние одного или нескольких признаков на объект +

определение порядка признака согласно рангу

6. Выберите правильный ответ. Задача ранжирования - это:

множество объектов, разделенных на классы

исследование влияние одного или нескольких признаков на объект

определение порядка признака согласно рангу +

7. Что служит индикатором ошибки для задач классификации?

$$j(a, x) = \frac{1}{2} (a(x) - y^*(x))^2 +$$

$$j(a, x) = |a(x) - y^*(x)|$$

$$j(a, x) = (a(x) - y^*(x))$$

8. Как формула подходит для абсолютного значения ошибки для задач регрессии?

$$j(a, x) = \frac{1}{2} (a(x) - y^*(x))^2 +$$

$$j(a, x) = |a(x) - y^*(x)| +$$

$$j(a, x) = (a(x) - y^*(x))$$

9. Что является квадратичной ошибкой для задачи регрессии?

$$j(a, x) = \frac{1}{2} (a(x) - y^*(x))^2 +$$

$$j(a, x) = |a(x) - y^*(x)|$$

$$j(a, x) = (a(x) - y^*(x))^2 +$$

10. Если происходит средняя потеря на всех объектах, то это есть: переобучение

эмпирический риск +

оценка релевантности

11. Выберите верные утверждения.

класс - это множество всех объектов с определенным значением +

в задачах регрессии допустимым ответом является действительное число или числовой вектор+

в задачах ранжирования ответы получают сразу на множестве объектов+

области минимального объёма с достаточно гладкой границей являются основной составляющей задач ранжирования

12. Какие задачи из ниже перечисленных относятся к задачам классификации?

определение наиболее целесообразного способа лечения+

определение длительности и исхода заболевания+

оценивание кредитоспособности заёмщика+

задачи поискового вывода

7.2.2 Примерный перечень заданий для решения стандартных задач

1. Идея алгоритма ЕМ заключается в следующем:

при слишком узком окне ($h \rightarrow 0$) плотность концентрируется вблизи обучающих объектов, при слишком широком окне плотность чрезмерно сглаживается и в пределе $h \rightarrow \infty$ вырождается в константу.

задаются функции правдоподобия $p_y(x)$ и априорные вероятности p_y . Согласно распределению $p_{y,y}(x)$ генерируются 2 выборки: обучающая X^l и контрольная X^k . По обучающей выборке X^l настраивается тестируемый алгоритм $a(x)$. По контрольной выборке вычисляется эмпирическая оценка среднего риска.

имея некоторый набор компонент, можно выделить объекты x_i , которые хуже всего описываются смесью - это объекты с наименьшими значениями правдоподобия $p(x_i)$. По этим объектам строится еще одна компонента. Затем она добавляется в смесь и запускаются ЕМ - итерации, чтобы новая компонента и старые "притёрлись друг к другу". Так продолжается до тех пор, пока все объекты не окажутся покрыты компонентами.

искусственно вводится вспомогательный вектор скрытых переменных G , обладающий 2-мя свойствами: первое - он может быть вычислен, если известны значения вектора параметров Q ; второе - поиск максимума правдоподобия сильно упрощается, если известны значения скрытых переменных.+

2. Искусственные нейронные сети (ИНС) — модели машинного

обучения, использующие комбинации распределенных простых операций, зависящих от обучаемых параметров, для обработки входных данных. Какого вида ИНС не существует?

Рекуррентные
Противоборствующие
Наивные +
Импульсные

3. У машинного обучения есть ряд задач. Как называется та, что направлена на предсказание значения той или иной непрерывной числовой величины для входных данных?

Кластеризация
Классификация
Переобучение
Регрессия+

4. Конкретный вид алгоритма, позволяющий решать задачу классификации в машинном обучении, — это:

функция классификации
функционал классификации +
модель классификации
уравнение классификации
структура классификации

5. Свертка значений признаков с некоторыми весовыми коэффициентами, определяющими степень важности данного признака, называется:

взвешенная +

6. Какова будет апостериорная вероятность того, что студент готовился к экзамену, если он его сдал, вероятность сдать экзамен - 40%, вероятность сдать при условии подготовки - 60%, а всего к экзаменам готовятся 30% студентов.

0,45 +

7. Какова будет апостериорная вероятность того, что студент готовился к экзамену, если он его сдал, вероятность сдать экзамен - $P(B)$, вероятность сдать при условии подготовки - $P(B|A)$, а всего к экзаменам готовятся $P(A)$ студентов.

$$P(A|B) = (P(B|A) * P(A)) / P(B) +$$

8. Каким функционалом порождается логистическая регрессия?

квадратичная функция потерь
сигмоидная функция потерь
логарифмическая функция потерь +
экспоненциальная функция потерь

9. Каким функционалом порождается нейронная сеть?

квадратичная функция потерь

логарифмическая функция потерь

экспоненциальная функция потерь

сигмоидная функция потерь+

10. Для совместной оценки полноты (Re) и точности (Pr) модели машинного обучения в случае их равнозначности используется мера:

$$F = \frac{Pr + Re}{Pr * Re}$$

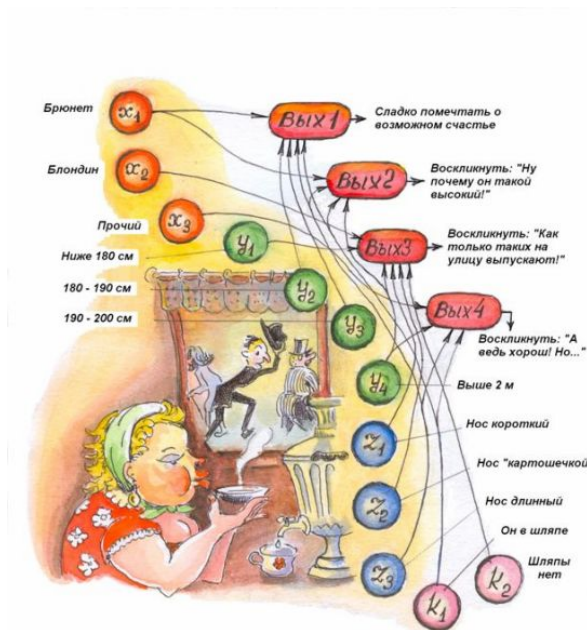
$$F = \frac{2 Pr + Re}{PR + Re} +$$

$$F = \frac{Pr * Re}{PR + Re}$$

$$F = \frac{2(Pr + Re)}{PR * Re}$$

7.2.3 Примерный перечень заданий для решения прикладных задач

1. Найдите предпочтительное решение по логической нейронной сети, представленной на рисунке, и по функции активации $f_i = \mathring{a} \sum_j w_j V_j$



$$x_1 = 0,4; x_2 = 0,5; x_3 = 0,1; y_1 = 0,9$$

$$y_2 = 0,1; y_3 = 0; y_4 = 0, z_1 = 0, z_2 = 0,1$$

$$z_3 = 0,9; k_1 = 0,9; k_2 = 0,1$$

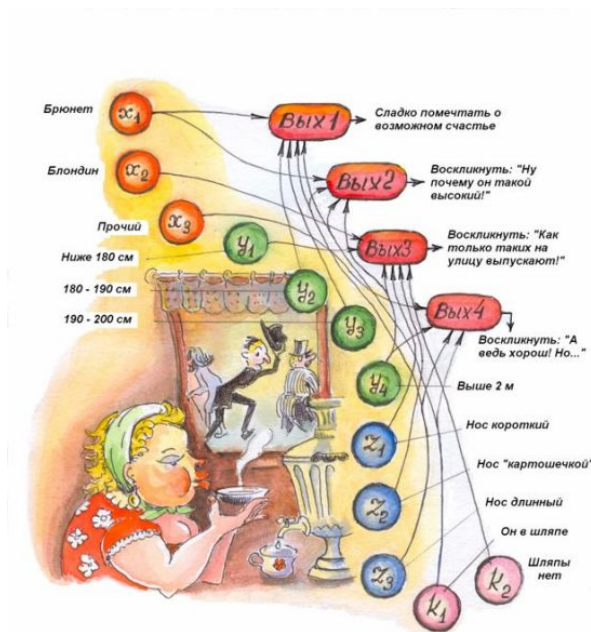
Ответ:

$R_1 \Leftarrow$ Сладко помечтать о +
возможном счастье >

$R_4 \Leftarrow$ Воскликнуть А ведь он хорош! Но >

$R_3 \Leftarrow$ Воскликнуть Как только таких на
улицу выпускают >

2. Найдите предпочтительное решение по логической нейронной сети, представленной на рисунке, и по функции активации $f_i = \mathring{a} \sum_j w_j V_j$



$x_1 = 0,5; x_2 = 0; x_3 = 0,5; y_1 = 0$
 $y_2 = 0; y_3 = 0,1; y_4 = 0,9; z_1 = 0,1, z_2 = 0,8$
 $z_3 = 0,1; k_1 = 0,1; k_2 = 0,9$

Ответ:

$R_1 \Leftarrow$ Сладко помечтать о
 возможном счастье >
 $R_4 \Leftarrow$ Воскликнуть А ведь он хорош! Но >
 +
 $R_3 \Leftarrow$ Воскликнуть Как только таких на
 улицу выпускают >

3. Создайте класс Neural Network на Python

Ответ:

```

class
NeuralNetwork:
    def __init__(self, x, y):
        self.input      = x
        self.weights1    = np.random.rand(self.input.shape[1],4)
        self.weights2    = np.random.rand(4,1)
        self.y           = y
        self.output      = np.zeros(y.shape)
  
```

4. Требуется натренировать регрессионную модель, представляющую линейную связь между признаком и вектором целей. Использовать линейную регрессию (в библиотеке scikit-learn это класс LinearRegression). Вставьте вместо многоточий недостающий код.

Используйте датасет (<https://www.kaggle.com/datasets/puxama/bostoncsv>):

```

# Загрузить библиотеки
from sklearn.linear_model import LinearRegression
from sklearn.datasets import load_boston
# Загрузить данные только с двумя признаками
boston = .....
features = boston.data[:,0:2]
target = .....
# Создать объект линейной регрессии
regression = .....
# Выполнить подгонку линейной регрессии
model = regression.fit(.....)
  
```

Ответ: `load_boston()`
 Ответ: `boston.target`
 Ответ: `LinearRegression()`
 Ответ: `features, target`

5. Требуется натренировать классификатор, используя дерево принятия решений. Использовать класс *DecisionTreeClassifier* библиотеки scikit-learn.

Датасет <https://www.kaggle.com/code/jchen2186/machine-learning-with-iris-dataset>

Вставьте вместо многоточий недостающий код.

```
# Загрузить библиотеки
from sklearn.tree import ..... Ответ: DecisionTreeClassifier
from sklearn import datasets
# Загрузить данные
iris = ..... Ответ: datasets.load_iris()
features = iris.data
target = iris.target
# Создать объект-классификатор дерева принятия решений
decisiontree = DecisionTreeClassifier(random_state=0)
# Натренировать модель
..... Ответ: model = decisiontree.fit(features, target)
```

6. Требуется найти к ближайших наблюдений (соседей) некоторого наблюдения. Использовать класс NearestNeighbors библиотеки scikit-learn.

Датасет <https://www.kaggle.com/code/jchen2186/machine-learning-with-iris-dataset>

Вставьте вместо многоточий недостающий код.

```
# Загрузить библиотеки
from sklearn import datasets
from sklearn.neighbors import NearestNeighbors
from sklearn.preprocessing import StandardScaler
# Загрузить данные

..... Ответ: iris = datasets.load_iris()
features = iris.data
# Создать стандартизатор
standardizer = StandardScaler()
# Стандартизировать признаки
.....
```

Ответ:

```
features_standardized = standardizer.fit_transform(features)
```

```
# Два ближайших соседа
```

```
nearest_neighbors =
```

```
NearestNeighbors(n_neighbors=2).fit(features_standardized)
```

```
# Создать наблюдение
```

```
..... Ответ: new_observation = [ 1, 1, 1, 1]
```

```
# Найти расстояния и индексы ближайших соседей наблюдения
```

```
distances, indices = nearest_neighbors.kneighbors([new_observation])
```

```
# Взглянуть на ближайших соседей
```

```
features_standardized[indices]
```

```
array([[1.03800476, 0.56925129, 1.10395287, 1.1850097 ],
       [0.79566902, 0.33784833, 0.76275864, 1.05353673]])
```

7. Требуется натренировать простую классификационную модель.

Натренировать логистическую регрессию в библиотеке scikit-learn с

помощью класса LogisticRegression. Вставьте вместо многоточий недостающий код. Датасет

<https://www.kaggle.com/code/jchen2186/machine-learning-with-iris-dataset>

```
# Загрузить библиотеки
from sklearn.linear_model import LogisticRegression
from sklearn import datasets
from sklearn.preprocessing import StandardScaler
# Загрузить данные только с двумя классами
..... Ответ: iris = datasets.load_iris()
features = iris.data[:100,:]
target = iris.target[:100]
# Стандартизировать признаки
scaler = ..... Ответ: StandardScaler()
features_standardized = scaler.fit_transform(features)
# Создать объект логистической регрессии
.....
Ответ: logistic_regression = LogisticRegression(random_state=0)
# Натренировать модель
.....
Ответ: model = logistic_regression.fit(features_standardized, target)
```

8. Требуется натренировать модель, чтобы классифицировать наблюдения. Использовать опорно-векторный классификатор (support vector classifier, SVC), чтобы найти гиперплоскость, которая максимизирует промежутки (зазоры) между классами. Вставьте вместо многоточий недостающий код. Датасет

<https://www.kaggle.com/code/jchen2186/machine-learning-with-iris-dataset>

```
# Загрузить библиотеки
from sklearn.svm import LinearSVC
from sklearn import datasets
from sklearn.preprocessing import StandardScaler
import numpy as np
# Загрузить данные всего с двумя классами и двумя признаками
iris = datasets.load_iris()
features = iris.data[:100,:2]
target = iris.target[:100]
# Стандартизировать признаки
scaler = ..... Ответ: StandardScaler()
features_standardized = scaler.fit_transform(features)
# Создать опорно-векторный классификатор
..... Ответ: svc = LinearSVC(C=1.0)
# Натренировать модель
..... Ответ: model = svc.fit(features_standardized, target)
```

9. Даны только непрерывные признаки, и требуется натренировать наивный байесов классификатор. Использовать наивный байесов классификатор (GaussianNB) в библиотеке scikit-learn

Вставьте вместо многоточий недостающий код. Датасет

<https://www.kaggle.com/code/jchen2186/machine-learning-with-iris-dataset>

Загрузить библиотеки

from sklearn import datasets

from sklearn.naive_bayes import GaussianNB

Загрузить данные

.....*Ответ: iris = datasets.load_iris()*

features = iris.data

target = iris.target

Создать объект гауссова наивного Байеса

.....*Ответ: classifier = GaussianNB()*

Натренировать модель

.....*Ответ: model = classifier.fit(features, target)*

10. Требуется сгруппировать наблюдения в k групп. Использовать кластеризацию методом k средних (KMeans). Вставьте вместо многоточий недостающий код. Датасет

<https://www.kaggle.com/code/jchen2186/machine-learning-with-iris-dataset>

Загрузить библиотеки

from sklearn import datasets

from sklearn.preprocessing import StandardScaler

from sklearn.cluster import*Ответ: KMeans*

Загрузить данные

iris = datasets.load_iris()

features = iris.data

Стандартизировать признаки

scaler = StandardScaler()

features_std = scaler.fit_transform(features)

Создать объект k средних

.....

Ответ:

cluster = KMeans(n_clusters=3, random_state=0, n_jobs=-1)

Натренировать модель

.....

Ответ:

model = cluster.fit(features_std)

7.2.4 Примерный перечень вопросов для подготовки к зачету

1. Основные задачи систем искусственного интеллекта. Классификация, кластеризация, регрессия.

2. Типы машинного обучения: с учителем, без учителя, с частичным привлечением учителя, обучение с подкреплением.

3. Классификация на примере алгоритма k -ближайших соседей (kNN).

4. Метрики оценки классификации: полнота, точность, F1, ROC, AUC.

5. Валидационная и тестовая выборка. Кросс-валидация.

6. Работа с категориальными признаками.

7. Регрессия. Метрики оценки регрессии: MSE, MAE, R2 - коэффициент детерминации.
8. Линейная регрессия, полиномиальная регрессия.
9. Переобучение и регуляризация, гребневая регрессия, LASSO, Elastic Net.
10. Линейные модели для классификации. Перцептрон, логистическая регрессия, полносвязные нейронные сети, стохастический градиентный спуск и обратное распространение градиента.
11. Регуляризация линейных моделей классификации.
12. Кластеризация. k-means, k-means++, DBSCAN, агломеративная кластеризация. Метрики оценки кластеризации.
13. Алгоритмы, основанные на применении решающих деревьев. Критерии разделения узла: информационный выигрыш, критерий Джини.
14. Ансамбли решающих деревьев: случайный лес, градиентный бустинг.
15. Метод опорных векторов. Прямая и обратная задача. Определение опорных векторов. Ядерный трюк.
16. Наивный байесовский классификатор.
17. ЕМ-алгоритм на примере смеси гауссиан.
18. Методы безградиентной оптимизации: случайный поиск, hill climb, отжиг, генетический алгоритм.
19. Нейронные сети. Функции ошибки нейронных сетей и обучение с помощью обратного распространения градиента. Понятие бэтча и эпохи.
20. Сверточные нейронные сети. Операции свертки, max-pooling. Популярные архитектуры сверточных нейронных сетей: AlexNet, VGG, Inception (GoogLeNet), ResNet.
21. Трансферное обучение. Обработка текстов. Работа с естественным языком с помощью нейронных сетей. Векторные представления для текста: word2vec, skipgram, CBOW, fasttext.
22. Рекуррентные нейронные сети, LSTM, GRU. Трансформеры, BERT, GPT.
23. Понятия агента, среды, состояния, действий и награды.
24. Функция ценности состояния (Value function) и функция качества действия (Q- function).
25. Оптимизация стратегии с помощью максимизации функций ценности и качества. Q-обучение.
26. Глубокое обучение с подкреплением. Deep Q-Networks, Actor-critic. Для уровня экспертный: REINFORCE, A2C, PPO, DDPG.

7.2.5 Примерный перечень заданий для подготовки к экзамену

Не предусмотрено учебным планом

7.2.6. Методика выставления оценки при проведении промежуточной аттестации

Экзамен проводится по тест-билетам, каждый из которых содержит 10 вопросов и задачу. Каждый правильный ответ на вопрос в тесте оценивается

1 баллом, задача оценивается в 10 баллов (5 баллов верное решение и 5 баллов за верный ответ). Максимальное количество набранных баллов – 20.

1. Оценка «Неудовлетворительно» ставится в случае, если студент набрал менее 6 баллов.

2. Оценка «Удовлетворительно» ставится в случае, если студент набрал от 6 до 10 баллов

3. Оценка «Хорошо» ставится в случае, если студент набрал от 11 до 15 баллов.

4. Оценка «Отлично» ставится, если студент набрал от 16 до 20 баллов)

7.2.7 Паспорт оценочных материалов

№ п/п	Контролируемые разделы (темы) дисциплины	Код контролируемой компетенции	Наименование оценочного средства
1	Введение в искусственный интеллект и основные методы машинного обучения для работы с табличными данными	ПК-4.8, ПК-4.9	Тест, защита лабораторных работ
2	Системы глубокого обучения	ПК-4.8, ПК-4.9	Тест, защита лабораторных работ, защита практических работ
3	Обучение с подкреплением	ПК-4.8, ПК-4.9	Тест, защита лабораторных работ, защита практических работ

7.3. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности

Тестирование осуществляется, либо при помощи компьютерной системы тестирования, либо с использованием выданных тест-заданий на бумажном носителе. Время тестирования 30 мин. Затем осуществляется проверка теста экзаменатором и выставляется оценка согласно методики выставления оценки при проведении промежуточной аттестации.

Решение стандартных задач осуществляется, либо при помощи компьютерной системы тестирования, либо с использованием выданных задач на бумажном носителе. Время решения задач 30 мин. Затем осуществляется проверка решения задач экзаменатором и выставляется оценка, согласно методики выставления оценки при проведении промежуточной аттестации.

Решение прикладных задач осуществляется, либо при помощи компьютерной системы тестирования, либо с использованием выданных задач на бумажном носителе. Время решения задач 30 мин. Затем осуществляется проверка решения задач экзаменатором и выставляется оценка, согласно методики выставления оценки при проведении промежуточной аттестации.

8 УЧЕБНО МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ

ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ)

8.1 Перечень учебной литературы, необходимой для освоения дисциплины

Основная литература

1. Платонов, А. В. Машинное обучение: учебное пособие для вузов / А. В. Платонов. — Москва: Издательство Юрайт, 2023. — 85 с. — (Высшее образование). — ISBN 978-5-534-15561-7. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/520544>

2. Саттон, Р. С. Обучение с подкреплением: введение / Р. С. Саттон, Э. Дж. Барто; перевод А. А. Слинкина. — 2-е изд. — Москва: ДМК Пресс, 2020. — 552 с. — ISBN 978-5-97060-097-9. — Текст: электронный // Цифровой образовательный ресурс IPR SMART: [сайт]. — URL: <https://www.iprbookshop.ru/125182.html>

3. Барский, А. Б. Введение в нейронные сети: учебное пособие / А. Б. Барский. — 3-е изд. — Москва, Саратов: Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2020. — 357 с. — ISBN 978-5-4497-0309-5. — Текст: электронный // Цифровой образовательный ресурс IPR SMART: [сайт]. — URL: <https://www.iprbookshop.ru/89426.html>.

Дополнительная литература

4. Воронов, М. В. Системы искусственного интеллекта: учебник и практикум для вузов / М. В. Воронов, В. И. Пименов, И. А. Небаев. — Москва: Издательство Юрайт, 2023. — 256 с. — (Высшее образование). — ISBN 978-5-534-14916-6. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/519916>

5. Лонца, А. Алгоритмы обучения с подкреплением на Python / А. Лонца; перевод А. А. Слинкин. — Москва: ДМК Пресс, 2020. — 286 с. — ISBN 978-5-97060-855-5. — Текст: электронный // Цифровой образовательный ресурс IPR SMART: [сайт]. — URL: <https://www.iprbookshop.ru/126211.html>

6. Ферлитш, Э. Шаблоны и практика глубокого обучения / Э. Ферлитш ; перевод А. В. Логунова. — Москва : ДМК Пресс, 2022. — 538 с. — ISBN 978-5-93700-113-9. — Текст: электронный // Цифровой образовательный ресурс IPR SMART: [сайт]. — URL: <https://www.iprbookshop.ru/125193.html>

7. Ю., (Х.) Обучение с подкреплением на PyTorch: сборник рецептов / Лю (Х.) Ю.; перевод А. А. Слинкин. — Москва: ДМК Пресс, 2020. — 282 с. — ISBN 978-5-97060-853-1. — Текст: электронный // Цифровой образовательный ресурс IPR SMART: [сайт]. — URL: <https://www.iprbookshop.ru/126247.html>

8.2 Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень лицензионного программного обеспечения, ресурсов

информационно-телекоммуникационной сети «Интернет», современных профессиональных баз данных и информационных справочных систем:

1. Open Machine Learning Course (<https://mlcourse.ai>)
2. Введение в машинное обучение от «Bioinformatic Institute» (<https://stepik.org/course/4852/promo>)
3. Специализация Машинное обучение и анализ данных от «Московский физико-технический институт» (<https://ru.coursera.org/specializations/machine-learning-data-analysis>)
4. Платформа для проведения соревнований по Data Science (<https://www.kaggle.com>).

9 МАТЕРИАЛЬНО-ТЕХНИЧЕСКАЯ БАЗА, НЕОБХОДИМАЯ ДЛЯ ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА

Аудитории с мультимедийными средствами, средствами звуковоспроизведения и имеющие выход в сеть «Интернет». Помещения для проведения аудиторных занятий, оборудованные учебной мебелью. Компьютерный класс с комплексом программных средств, позволяющих каждому студенту разрабатывать программные реализации практических задач в ходе выполнения лабораторных работ

10. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ)

По дисциплине «Системы искусственного интеллекта» читаются лекции, проводятся практические занятия и лабораторные работы.

Основой изучения дисциплины являются лекции, на которых излагаются наиболее существенные и трудные вопросы, а также вопросы, не нашедшие отражения в учебной литературе.

Практические занятия направлены на приобретение практических навыков программно-алгоритмического освоения материала. Занятия проводятся путем решения конкретных задач в аудитории.

Лабораторные работы выполняются на лабораторном оборудовании в соответствии с методиками, приведенными в указаниях к выполнению работ.

Вид учебных занятий	Деятельность студента
Лекция	Написание конспекта лекций: кратко, схематично, последовательно фиксировать основные положения, выводы, формулировки, обобщения; помечать важные мысли, выделять ключевые слова, термины. Проверка терминов, понятий с помощью энциклопедий, словарей, справочников с выписыванием толкований в тетрадь. Обозначение вопросов, терминов, материала, которые вызывают трудности, поиск ответов в рекомендуемой литературе. Если самостоятельно не удастся разобраться в материале, необходимо сформулировать вопрос и задать преподавателю на лекции или на практическом занятии.

Практическое занятие	Конспектирование рекомендуемых источников. Работа с конспектом лекций, подготовка ответов к контрольным вопросам, просмотр рекомендуемой литературы. Прослушивание аудио- и видеозаписей по заданной теме, выполнение расчетно-графических заданий, решение задач по алгоритму.
Лабораторная работа	Лабораторные работы позволяют научиться применять теоретические знания, полученные на лекции при решении конкретных задач. Чтобы наиболее рационально и полно использовать все возможности лабораторных для подготовки к ним необходимо: следует разобрать лекцию по соответствующей теме, ознакомиться с соответствующим разделом учебника, проработать дополнительную литературу и источники, решить задачи и выполнить другие письменные задания.
Самостоятельная работа	Самостоятельная работа студентов способствует глубокому усвоению учебного материала и развитию навыков самообразования. Самостоятельная работа предполагает следующие составляющие: - работа с текстами: учебниками, справочниками, дополнительной литературой, а также проработка конспектов лекций; - выполнение домашних заданий и расчетов; - работа над темами для самостоятельного изучения; - участие в работе студенческих научных конференций, олимпиад; - подготовка к промежуточной аттестации.
Подготовка к промежуточной аттестации	Готовиться к промежуточной аттестации следует систематически, в течение всего семестра. Интенсивная подготовка должна начинаться не позднее, чем за месяц-полтора до промежуточной аттестации. Данные перед зачетом три дня эффективнее всего использовать для повторения и систематизации материала.

ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

[illegible]