



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение высшего образования
«Дальневосточный федеральный университет»
(ДВФУ)
ИНСТИТУТ МАТЕМАТИКИ И КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ (ШКОЛА)

«СОГЛАСОВАНО»

Руководитель ОП

Подпись

Дремлюга Р.И.

УТВЕРЖДАЮ

И.о. директора Академии цифровой трансформации

Еременко А.С.

«26» января 2022 г.



РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Нейронные сети и глубокое обучение

Направление подготовки - 09.04.01 Информатика и вычислительная техника
(Искусственный интеллект и большие данные (совместно с ПАО Сбербанк))

Форма подготовки очная

курс 1 семестр 2
лекции 18 час.
практические занятия 36 час.
лабораторные работы 00 час.
в том числе с использованием МАО практические занятия 36 час.
всего часов аудиторной нагрузки 36 час.
самостоятельная работа 90 час.
в том числе на подготовку к экзамену 00 час.
контрольные работы (количество) не предусмотрены
курсовая работа / курсовой проект не предусмотрены
зачет с оценкой 2 семестр
экзамен не предусмотрен

Рабочая программа дисциплины разработана в соответствии с федеральным государственным образовательным стандартом (ФГОС) высшего образования (ВО) – магистратура по направлению подготовки 09.04.01 Информатика и вычислительная техника, утвержденного приказом Министерства образования и науки Российской Федерации от 19.09.2017 г. № 918 (с изменениями и дополнениями).

Рабочая программа обсуждена на заседании Академии цифровой трансформации протокол № 1 от 25 января 2022 г.

И.о. директора Академии цифровой трансформации

Еременко А.С.

Составитель (ли): к.т.н. Еременко А.С., Загуменнов А.А.

Владивосток
2022

Оборотная сторона титульного листа РПД

I. **Рабочая программа пересмотрена на заседании департамента:**

Протокол от « ____ » _____ 20__ г. № _____

Директор департамента _____
(подпись) (И.О.Фамилия)

II. **Рабочая программа пересмотрена на заседании департамента:**

Протокол от « ____ » _____ 20__ г. № _____

Директор департамента _____
(подпись) (И.О. Фамилия)

I. ЦЕЛИ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Цель курса - изучение методов «глубокого обучения» - нового поколения нейросетевых методов машинного обучения. В первую очередь курс направлен на формирования у студентов навыков решения прикладных задач при помощи глубоких нейронных сетей.

Задачи курса:

- изучить базовые понятия по следующим разделам курса: нейронные сети, их функциональные свойства, алгоритмы обучения, распознавания образов, прогнозирование временных рядов, оптимальные статистические решения;
- знать основные типы нейронных сетей и задач, для решения которых эти нейронные сети предназначены;
- иметь представление об основных сферах применения методов глубокого обучения, о концепциях и идеях, на которых основано многообразие современных нейросетевых технологий обработки информации;
- научиться использовать современные методы статистического анализа, прогнозирования и принятия решений на основе алгоритмов адаптации нейронных сетей.
- уметь использовать программные комплексы для создания и моделирования нейроподобных сетей, их применения в конкретных прикладных задачах.
- уметь разработать программную реализацию выбранного типа нейронной сети, произвести ее обучение и испытание.

В результате данной дисциплины у обучающихся формируются следующие профессиональные компетенции:

Код и наименование профессиональной компетенции (результат освоения)	Код и наименование индикатора достижения компетенции
ПК-1 Способен проектировать и разрабатывать системные и прикладные решения по анализу больших данных	ПК-1.1 Владеет инструментарием получения, хранения, передачи и обработки больших данных ПК-1.2 Формулирует и решает системные и прикладные задачи анализа больших данных для конкретных предметных областей ПК-1.3 Способен управлять разработкой продуктов, услуг и решений на основе больших данных
ПК-4 Способен ставить цели и принимать управленческие решения, основанные на анализе больших данных	ПК-4.1 Владеет навыками стратегического управления развитием методологической и технологической инфраструктуры анализа больших данных в организации ПК-4.2 Определяет необходимые ресурсы и инструменты для решения задач с использованием анализа данных;

Код и наименование профессиональной компетенции (результат освоения)	Код и наименование индикатора достижения компетенции
	руководит работой команды, вырабатывая командную стратегию на основе анализа данных

Код и наименование индикатора достижения компетенции	Наименование показателя оценивания (результата обучения)
ПК-1.1 Владеет инструментарием получения, хранения, передачи и обработки больших данных	Знает технологии, методы и инструментальные средства обработки больших данных
	Умеет использовать архитектуры и модели баз и хранилищ данных, адаптированные к технологиям больших данных
	Владеет навыками разработки предложений по развитию и совершенствованию системы получения, хранения, передачи, обработки больших данных
ПК-1.2 Формулирует и решает системные и прикладные задачи анализа больших данных для конкретных предметных областей	Знает существующий опыт разработки и использования продуктов и услуг на основе технологий больших данных
	Умеет разрабатывать проектную и рабочую документацию на разработку аналитических услуг на основе технологий больших данных
	Владеет навыками решения прикладных задач анализа больших данных для конкретных предметных областей
ПК-1.3 Способен управлять разработкой продуктов, услуг и решений на основе больших данных	Знает существующий опыт разработки и использования продуктов и услуг на основе технологий больших данных
	Умеет управлять исполнением проектных работ в области больших данных
	Владеет навыками создания прототипа сервиса на основе аналитики больших данных
	Владеет навыками управления проектными рисками в IT-проекте
ПК-4.1 Владеет навыками стратегического управления развитием методологической и технологической инфраструктуры анализа больших данных в организации	Знает существующие и перспективные методы и программный инструментарий технологий больших данных
	Умеет управлять развитием технологической инфраструктуры анализа больших данных
	Владеет навыками стратегического управления
ПК-4.2 Определяет необходимые ресурсы и инструменты для решения задач с использованием анализа данных; руководит работой команды, вырабатывая командную стратегию на основе анализа данных	Знает методы создания программного обеспечения для анализа и обработки данных
	Умеет использовать методы проектирования систем анализа и обработки данных
	Владеет навыками работы в распределенных командах

II. ТРУДОЁМКОСТЬ ДИСЦИПЛИНЫ И ВИДОВ УЧЕБНЫХ ЗАНЯТИЙ ПО ДИСЦИПЛИНЕ

Общая трудоемкость дисциплины составляет 4 зачётных единиц (144 академических часа).

Видами учебных занятий и работы обучающегося по дисциплине могут являться:

Обозначение	Виды учебных занятий и работы обучающегося
Лек	Лекционные занятия
Лаб	Лабораторные работы
Пр	Практические занятия

ОК	Онлайн курс
СР	Самостоятельная работа обучающегося в период теоретического обучения
Контроль	Самостоятельная работа обучающегося и контактная работа обучающегося с преподавателем в период промежуточной аттестации

Структура дисциплины:

Форма обучения – очная

	Наименование раздела дисциплины	Семестр	Количество часов по видам учебных занятий и работы обучающегося					Формы промежуточной аттестации	
			Лек	Лаб	Пр	ОК	СР		Контроль
1	Модуль 1. Нейронные сети и тематическое моделирование	2	4		4				УО-1, УО-4, ПР-6 зачет с оценкой
2	Модуль 2. Обучение с подкреплением и активное обучение	2	4		4				
3	Модуль 3. Сверточные нейронные сети. Рекуррентные нейронные сети. Генеративные модели	2	4		4		108		
4	Модуль 4. Кластеризация и частичное обучение	2	3		4				
5	Модуль 5. Автоматическое обучение машин (AutoML)	2	3		10				
	Итого:		18		36		108	зачет соценкой	

III. СТРУКТУРА И СОДЕРЖАНИЕ ТЕОРЕТИЧЕСКОЙ ЧАСТИ КУРСА

Лекции (18 часов)

Модуль 1. Нейронные сети и тематическое моделирование (4 часа)

Перцептрон, многослойный перцептрон. Функции активации. Методы оптимизации. Метод обратного распространения ошибки. Теория сложности и машинное обучение. Probably Approximately Correct learning. Задача тематического моделирования коллекции текстовых документов. Вероятностный латентный семантический анализ PLSA. Метод максимума правдоподобия. EM-алгоритм. Элементарная интерпретация EM-алгоритма. Латентное размещение Дирихле LDA. Метод максимума апостериорной

вероятности. Сглаженная частотная оценка условной вероятности. Небайесовская интерпретация LDA и её преимущества. Регуляризаторы разреживания, сглаживания, частичного обучения. Аддитивная регуляризация тематических моделей. Регуляризованный EM-алгоритм, теорема о стационарной точке (применение условий Каруша–Куна–Таккера). Рациональный EM-алгоритм. Онлайнный EM-алгоритм и его распараллеливание. Мультимодальная тематическая модель. Регуляризаторы классификации и регрессии. Регуляризаторы декоррелирования и отбора тем. Внутренние и внешние критерии качества тематических моделей.

Модуль 2. Обучение с подкреплением и активное обучение (4 часа)

Задача о многоруком бандите. Жадные и эpsilon-жадные стратегии. Метод UCSB (upper confidence bound). Стратегия Softmax. Среда для экспериментов. Адаптивные стратегии на основе скользящих средних. Метод сравнения с подкреплением. Метод преследования. Постановка задачи в случае, когда агент влияет на среду. Ценность состояния среды. Ценность действия. Жадные стратегии максимизации ценности. Уравнения оптимальности Беллмана. Метод временных разностей TD. Метод Q-обучения. Градиентная оптимизация стратегии (policy gradient). Связь с максимизацией log-правдоподобия. Постановка задачи при наличии информации о среде в случае выбора действия. Контекстный многорукий бандит. Линейная регрессионная модель с верхней доверительной оценкой LinUCB. Оценивание новой стратегии по большим историческим данным. Постановка задачи машинного обучения. Основные стратегии: отбор объектов из выборки и из потока, синтез объектов. Почему активное обучение быстрее пассивного. Сокращение пространства решений. Взвешивание по плотности. Оценивание качества активного обучения. Синтез объектов по критерию сокращения дисперсии. Применение обучения с подкреплением для активного обучения. Активное томпсоновское сэмплирование.

Модуль 3. Сверточные нейронные сети. Рекуррентные нейронные сети. Генеративные модели (4 часа)

Свёрточные нейронные сети (CNN). Свёрточный нейрон. Pooling нейрон. Выборка размеченных изображений ImageNet. Идея обобщения CNN на любые структурированные данные. Рекуррентные нейронные сети (RNN). Обучение рекуррентных сетей: Backpropagation Through Time (BPTT). Сети долгой кратковременной памяти (Long short-term memory, LSTM). Автокодировщики. Векторные представления дискретных данных. Моделирование случайных данных Неслучайные модельные данные. Ранговая классификация, OS-SVM. Парный подход: RankingSVM, RankNet, LambdaRank.

Модуль 4. Кластеризация и частичное обучение (3 часа)

Постановка задачи кластеризации. Примеры прикладных задач. Типы кластерных структур. Постановка задачи Semisupervised Learning, примеры приложений. Оптимизационные постановки задач кластеризации и частичного обучения. Алгоритм k-средних и EM-алгоритм для разделения гауссовской смеси.

Графовые алгоритмы кластеризации. Выделение связанных компонент. Кратчайший незамкнутый путь. Алгоритм ФОРЭЛ. Алгоритм DBSCAN. Агломеративная кластеризация, Алгоритм Ланса-Вильямса и его частные случаи. Алгоритм построения дендрограммы. Определение числа кластеров. Свойства сжатия/растяжения, монотонности и редуktivности. Псевдокод редуktivной версии алгоритма. Простые эвристические методы частичного обучения: self-training, co-training, co-learning. Трансдуктивный метод опорных векторов TSVM. Алгоритм Expectation-Regularization на основе многоклассовой регуляризированной логистической регрессии.

Модуль 5. Автоматическое обучение машин (AutoML) (3 часа)

Автоматическая подготовка данных и сбор и сохранение данных (из сырых данных и разнообразных форматов). Автоматическое конструирование признаков. Автоматический выбор модели. Оптимизация гиперпараметров алгоритма обучения и характеристики. Автоматический выбор каналов по времени, памяти и ограничений сложности. Автоматический выбор метрик оценки / процедур валидации. Автоматическая проверка задач. Автоматический анализ полученных результатов. Пользовательские результаты и визуализация для автоматического обучения машин

IV. СТРУКТУРА И СОДЕРЖАНИЕ ПРАКТИЧЕСКОЙ ЧАСТИ КУРСА И САМОСТОЯТЕЛЬНОЙ РАБОТЫ

Практические занятия (36 часов)

Наименование темы и содержание	Часы
Практическое задание №1. Методы активного обучения. Сэмплирование по неуверенности. Сэмплирование по несогласию в комитете. Сэмплирование по ожидаемому изменению модели. Сэмплирование по ожидаемому сокращению ошибки. Введение изучающих действий в стратегию активного обучения. Алгоритмы ϵ -active и EG-active.	2
Практическое задание №2. Методы селекции. Селекция: пропорционально качеству, универсальная выборка (stochastic universal sampling), с наследием (reward-based), турнир. Стратегия элитизма.	2

<p>Практическое задание №3. Методы кроссовера. Двух и многоточечный кроссовер, равномерный кроссовер (по подмножествам), для перестановок. Мутация. Влияние на скорость обучения.</p>	2
<p>Практическое задание №4. Управление популяцией. Сегрегация, старение, распараллеливание. Генетическое программирование.</p>	2
<p>Практическое задание №5. Наивный классификатор. Оценка плотности распределения для числовых признаков. Алгоритмические оптимизации. Алгоритм EM.</p>	2
<p>Практическое задание №6. Оптимизации с ограничениями Двойственная задача Лагранжа. Условия Каруша-Куна-Такера. Функция Лагранжа для линейного SVM. Опорный вектор. Типы опорных векторов. Kernel trick. Полиномиальное ядро. Радиально-базисное ядро (RBF).</p>	2
<p>Практическое задание №7. Алгоритм преобразования Хафа Написание алгоритма преобразования Хафа для прямых, кругов. Добавление оптимизации, требующих вычисление градиента на изображении.</p>	2
<p>Практическое задание №8. Алгоритмы нейронной сети (обработка изображений) Использование классических методов обработки изображений совместно с нейросетевым подходом (свертки, фильтры, описания текстур и пр.).</p>	2
<p>Практическое задание №9. Алгоритмы нейронной сети Реализация “ванильной” рекуррентной нейронной сети в TensorFlow. Реализация сети LSTM в TensorFlow.</p>	2
<p>Практическая работа №10. Использование классических методов обработки изображений совместно с нейросетевым подходом (свертки, фильтры, описания текстур и пр.).</p>	3
<p>Практическая работа №11. Построение сверточной сети средствами TensorFlow.</p>	3
<p>Практическое задание №12. Анализ последовательностей изображений в Google EarthEngine. Реализация классификатора на базе многослойного перцептрона для обработки последовательности изображений.</p>	3
<p>Практическое задание №13. Частотный анализ, представление bag-of-words, TF-IDF и его варианты. N-граммы, byte-pair encoding.</p>	3
<p>Практическое задание №14. Построение многослойного перцептрона для классификации изображений. Построение небольшой архитектуры сверточной нейронной сети для классификации. Построение одной из известных архитектур нейронных сетей (AlexNet, VGG, ResNet, Inception).</p>	4
<p>Практическое задание №15. Построение сверточной нейронной сети для задачи сегментации. Построение одной из известных архитектур (UNet).</p>	2

V. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ

Учебно-методическое обеспечение самостоятельной работы обучающихся по дисциплине «Нейронные сети и глубокое обучение» включает в себя:

- план-график выполнения самостоятельной работы по дисциплине, в том числе примерные нормы времени на выполнение по каждому заданию;
- характеристика заданий для самостоятельной работы обучающихся и методические рекомендации по их выполнению;
- требования к представлению и оформлению результатов самостоятельной работы;
- критерии оценки выполнения самостоятельной работы.

План-график выполнения самостоятельной работы по дисциплине

№ п/п	Дата/сроки выполнения	Вид самостоятельной работы	Примерные нормы времени на выполнение	Форма контроля
1	Февраль-май	Проработка лекционного материала по конспектам и учебной литературе	10	Собеседование, опрос
2	Февраль-май	Подготовка и выполнение практических заданий №1 - 7	25	Письменный отчет
3	Май	Выполнение и подготовка к защите лабораторных работ №8 - 15	25	Защита практических работ
4	Апрель	Самостоятельная проработка тем	20	Семинар
5	Июнь	Подготовка к зачету	10	Зачет
		Итого	90	

Перечень тем для самостоятельного изучения по дисциплине

Тема 1. Метрики

Примеси Джини (Gini impurity), добавленная информация (information gain). Деревья регрессии. Метрика вариации. Непрерывные признаки. Использование главных компонент вместо признаков. Сокращение дерева (pruning). Метрики, понятие центроида и представителя класса. Центроидные алгоритмы: k-means, k-medoid. Алгоритмы, основанные на плотности: DBSCAN, OPTICS. Алгоритмы, основанные на распределении: сумма гауссиан. Нечеткая кластеризация, алгоритм c-means. Метрики качества: leave-one-out, силуэт, индекс Дэвиса-Болдина (Davies-Bouldin), индекс Данна (Dunn).

Тема 2. Последовательности изображений

Анализ последовательностей изображений в Google EarthEngine. Классификатора на базе многослойного перцептрона для обработки последовательности изображений. Рекуррентные нейронные сети.

Тема 3. Алгоритмы компьютерного зрения

Классические алгоритмы обработки изображений.

Тема 4. Алгоритмы машинного обучения

Сверхточные нейронные сети. Классификация изображений. Сегментация. Локализация.

Вопросы к темам для самостоятельного изучения

1. Задачи обработки текста: извлечение, поиск, классификация (тематическая, эмоциональная), перевод
2. Разбиение на слова, пунктуация, лексический и морфологический анализ
3. Определение частей речи, имён, основ слов
4. Частотный анализ, представление bag-of-words, TF-IDF и его варианты
5. N-граммы, byte-pair encoding.
6. Векторные представления, семантическая интерпретация алгебраических операций
7. Унитарный код (One-hot encoding).
8. Алгоритмы Word2Vec и FastText.
9. Алгоритм GloVe*.
10. Постановка задачи, причины и цели снижения размерности.
11. Выбор и извлечение признаков.
12. Подходы к выбору признаков: filtering, wrapping, embedding.
13. Расстояние между распределениями. Расстояние Кульбака-Лейблера. Взаимная информация.
14. Алгоритмы выбора признаков: на основе корреляции (CFS), взаимной информации, Relief.
15. Метод главных компонент (PCA).
16. Нелинейные обобщения метода главных компонент. Kernel PCA.*
17. Неотрицательное матричное разложение (NMF).*
18. Стохастическое вложение соседей с t-распределением (t-SNE).

Вопросы для самостоятельной работы по дисциплине

1. Дайте определение объекта, образа и прецедента.
2. Приведите структурную схему системы распознавания образов.
3. Охарактеризуйте 3 способа минимизации среднего риска.
4. Дайте определение функционала риска.
5. Дайте определение функционала эмпирического риска.
6. Охарактеризуйте принцип минимизации эмпирического риска.
7. Дайте определение VC-измерения.
8. Охарактеризуйте понятие минимизации структурного риска.

9. Охарактеризуйте понятие вероятностно-корректной в смысле аппроксимации модели обучения.
10. Охарактеризуйте понятие байесовского классификатора.
11. Приведите структурные схемы байесовского классификатора на основе отношения правдоподобия и его логарифма.
12. Охарактеризуйте байесовский классификатор для Гауссовского распределения.
13. В чем заключаются сходство и различие персептрона и байесовского классификатора при решении задач классификации объектов.
14. Оптимальная гиперплоскость для линейно-разделимых образов.
15. Квадратичная оптимизация и поиск оптимальной гиперплоскости. Применение множителей Лагранжа.
16. Статистические свойства оптимальной гиперплоскости для линейно-разделимых образов.
17. Оптимальная гиперплоскость для неразделимых образов. Фиктивные переменные и множители Лагранжа.
18. В чем заключается идея машины опорных векторов для решения задачи распознавания образов?
19. Охарактеризуйте понятие ядра скалярного произведения.
20. Сформулируйте теорему Мерсера. Собственные функции и собственные значения.
21. Сформулируйте двойственную задачу условной оптимизации для машины опорных векторов.
22. Оптимальная селекция признаков.
23. Оптимальная селекция признаков на основе нейронной сети.

Рекомендации по самостоятельной работе студентов

Самостоятельная работа студента, безусловно – один из важнейших этапов в подготовке магистров. Она приобщает студентов к исследовательской работе, обогащает опытом и знаниями, необходимыми для дальнейшего их становления как специалистов, прививает навыки работы с литературой.

Цель самостоятельной работы – систематизация, закрепление и расширение теоретических и практических знаний с использованием современных информационных технологий и литературных источников. Данная цель может быть достигнута при решении следующего круга задач:

- изучение лекционного материала;
- изучение дополнительных источников информации;
- выполнение лабораторных работ;
- выполнение практических заданий.

Теоретическое обучение предполагает самостоятельную работу с литературными источниками. Вопросы, вынесенные на самостоятельное

изучение, дополняют сведения, полученные на лекциях, и также являются актуальными для будущей специальности. Некоторые из вопросов традиционно изучаются в рамках других дисциплин, поэтому они вынесены на самостоятельное изучение. В этом случае цель самостоятельного изучения заключается в том, чтобы студент получил на данном этапе общее представление о вопросе. Форма отчетности по проделанной работе – включение этих вопросов в экзаменационные билеты. Другая часть самостоятельной работы сводится к подготовке и защите в течение семестра лабораторных работ.

Руководство и контроль за самостоятельной работой студента осуществляется в форме индивидуальных консультаций.

При затруднении изучения отдельных тем, вопросов следует обращаться за консультациями к преподавателю.

IV. КОНТРОЛЬ ДОСТИЖЕНИЯ ЦЕЛЕЙ КУРСА

№ п/п	Контролируемые разделы/темы дисциплины	Код индикатора достижения компетенции	Результаты обучения	Оценочные средства	
				текущий контроль	промежуточная аттестация
1.	Модуль 1-3	ПК – 1.1 ПК – 1.2 ПК – 1.3	Знает способы получения оценок моделей и методов машинного обучения, а также критериев их сравнения.	Собеседование УО1, круглый стол УО-4	Зачет с оценкой Вопросы 1-13
			Умеет поставить задачу машинного обучения в заданной предметной области и предложить метод ее решения.		
			Владеет навыками работы с платформами, средами и программными средствами, в библиотеках которых реализованы методы машинного обучения.		
		ПК – 4.1 ПК -4.2	Знает способы выбора наиболее подходящего метода машинного обучения для заданной предметной области на основе оценки и анализа свойств существующих методов.		
		Умеет проанализировать заданную предметную область на предмет возможности и необходимости применения в ней методов машинного обучения.			
		Владеет навыками применения заданного метода машинного обучения (на основе существующих библиотек), а	Пр/работы 1-7, ПР-6		

			также получения результата (на основе заданного набора данных).		
2.	Модуль 4-5	ПК – 1.1 ПК – 1.2 ПК – 1.3 ПК – 4.1 ПК -4.2	Знает области применения, преимущества и недостатки основных алгоритмов и технологий обработки данных.	Собеседование УО1, круглый стол УО-4	Зачет с оценкой Вопросы 14-25 (практическая часть)
			Умеет оценить степень применимости, а также временную и вычислительную сложности выбранного алгоритма для решения конкретной прикладной задачи.	Пр/работы 8-15 ПР-6	
			Владеет способностью провести анализ степени применимости имеющихся алгоритмов для всех возможных (альтернативных) решений конкретной прикладной задачи.		
			Владеет способностью выбрать из имеющегося набора алгоритмов одного, наиболее подходящего с точки зрения экономии ресурсов, а также качества получаемого результата.		

Типовые контрольные задания, методические материалы, определяющие процедуры оценивания знаний, умений и навыков и (или) опыта деятельности, а также качественные критерии оценивания, которые описывают уровень сформированности компетенций, представлены в разделе VIII.

V. СПИСОК УЧЕБНОЙ ЛИТЕРАТУРЫ И ИНФОРМАЦИОННО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Основная литература

(электронные и печатные издания)

1. Горожанина Е.И. Нейронные сети [Электронный ресурс]: учебное пособие/ Горожанина Е.И.— Электрон. текстовые данные.— Самара: Поволжский государственный университет телекоммуникаций и информатики, 2017.— 84 с.— Режим доступа: <http://www.iprbookshop.ru/75391.html>.— ЭБС «IPRbooks»

2. Седов В.А. Введение в нейронные сети [Электронный ресурс]: методические указания к лабораторным работам по дисциплине «Нейроинформатика» для студентов специальности 09.03.02 «Информационные системы и технологии»/ Седов В.А., Седова Н.А.— Электрон. текстовые данные.— Саратов: Ай Пи Эр Медиа, 2018.— 30 с.— Режим доступа: <http://www.iprbookshop.ru/69319.html>.— ЭБС «IPRbooks»

3. Барский А.Б. Введение в нейронные сети [Электронный ресурс]/ Барский А.Б.— Электрон. текстовые данные.— М.: Интернет-Университет Информационных Технологий (ИНТУИТ), 2016.— 358 с.— Режим доступа: <http://www.iprbookshop.ru/52144.html>.— ЭБС «IPRbooks»

4. Гудфеллоу, Я. Глубокое обучение [Электронный ресурс] / Я. Гудфеллоу, И. Бенджио, А. Курвилль ; пер. с англ. А. А. Слинкина. — Электрон. дан. — Москва : ДМК Пресс, 2018. — 652 с. — Режим доступа: <https://e.lanbook.com/book/107901>. — Загл. с экрана.

5. Неделько В.М. Основы статистических методов машинного обучения [Электронный ресурс]: учебное пособие/ Неделько В.М.— Электрон. текстовые данные. — Новосибирск: Новосибирский государственный технический университет, 2010. — 72 с.— Режим доступа: <http://www.iprbookshop.ru/45418.html>. — ЭБС «IPRbooks»

6. Коэльо, Л.П. Построение систем машинного обучения на языке Python [Электронный ресурс] / Л.П. Коэльо, В. Ричарт; пер. с англ. Слинкин А. А.. — Электрон. дан. — Москва: ДМК Пресс, 2016. — 302 с. — Режим доступа: <https://e.lanbook.com/book/82818>. — Загл. с экрана.

7. Флах, П. Машинное обучение. Наука и искусство построения алгоритмов, которые извлекают знания из данных [Электронный ресурс] / П. Флах. — Электрон. дан. — Москва: ДМК Пресс, 2015. — 400 с. — Режим доступа: <https://e.lanbook.com/book/69955>. — Загл. с экрана.

8. Рашка, С. Python и машинное обучение: крайне необходимое пособие по новейшей предсказательной аналитике, обязательное для более глубокого понимания методологии машинного обучения [Электронный ресурс]: руководство / С. Рашка; пер. с англ. Логунова А.В. — Электрон. дан. — Москва: ДМК Пресс, 2017. — 418 с. — Режим доступа: <https://e.lanbook.com/book/100905>. — Загл. с экрана.

9. Шарден, Б. Крупномасштабное машинное обучение вместе с Python [Электронный ресурс]: учебное пособие / Б. Шарден, Л. Массарон, А. Боскетти; пер. с англ. А. В. Логунова. — Электрон. дан. — Москва: ДМК Пресс, 2018. — 358 с. — Режим доступа: <https://e.lanbook.com/book/105836>. — Загл. с экрана.

Дополнительная литература

(печатные и электронные издания)

1. Нейронные сети : основы теории / А. И. Галушкин. Москва : Горячая линия - Телеком, 2010. - 496 с. - Г 168 32.818.э

2. Кук, Д. Машинное обучение с использованием библиотеки H2O [Электронный ресурс] / Д. Кук; пер. с англ. Огурцова А.Б.. — Электрон. дан. — Москва: ДМК Пресс, 2018. — 250 с. — Режим доступа: <https://e.lanbook.com/book/97353>. — Загл. с экрана.

3. Информационные аналитические системы [Электронный ресурс]: учебник / Т. В. Алексеева, Ю. В. Амириди, В. В. Дик и др.; под ред. В. В. Дика. - М.: МФПУ Синергия, 2013. - 384 с. - (Университетская серия). - ISBN 978-5-4257-0092-6, <http://www.znaniyum.com/bookread.php?book=451186>

4. Домингос, П. Верховный алгоритм: как машинное обучение изменит наш мир [Электронный ресурс] Москва: Манн, Иванов и Фербер, 2016. 336 с. <https://e.lanbook.com/book/91645>.

5. Гаврилова, И.В. Основы искусственного интеллекта [Электронный ресурс]: учеб. пособие / И.В. Гаврилова, О.Е. Масленникова. Москва: ФЛИНТА, 2013. 282 с. <https://e.lanbook.com/book/44749>. 4. Ясницкий, Л.Н. Интеллектуальные системы [Электронный ресурс]: учеб. пособ./ Москва : Издательство 'Лаборатория знаний', 2016. 224 с. Режим доступа: <https://e.lanbook.com/book/90254>.

Перечень ресурсов

информационно-телекоммуникационной сети «Интернет»

1. Байесовские_методы_машинного_обучения_(курс_лекций)_/_2017 Д.П. Ветров - [http://www.machinelearning.ru/wiki/index.php?title=Байесовские_методы_машинного_обучения_\(курс_лекций\)_/_2017](http://www.machinelearning.ru/wiki/index.php?title=Байесовские_методы_машинного_обучения_(курс_лекций)_/_2017) Д.П. Ветров
2. Машинное обучение (курс лекций, Н.Ю. Золотых) - <http://www.uic.unn.ru/~zny/ml/>
3. Машинное_обучение_(курс_лекций_С.К.Воронцов). - [http://www.machinelearning.ru/wiki/index.php?title=Машинное_обучение_\(курс_лекций%2C_К.В.Воронцов\)](http://www.machinelearning.ru/wiki/index.php?title=Машинное_обучение_(курс_лекций%2C_К.В.Воронцов))
4. [Курс «Введение в машинное обучение», К.В.Воронцов \(ВШЭ и Яндекс\).Хабр об этом курсе.](#)
5. [Специализация «Машинное обучение и анализ данных» \(МФТИ и Яндекс\). Хабр об этом курсе.](#)
6. [Машинное обучение \(семинары,ФУПМ МФТИ\)](#)
7. [Машинное обучение \(семинары, ВМК МГУ\)](#)
8. [Машинное обучение \(курс лекций, Н.Ю.Золотых\)](#)
9. [Машинное обучение \(курс лекций, СГАУ, С.Лисицын\)](#)

VI. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

Планирование и организация времени, отведенного на изучение дисциплины. Приступить к освоению дисциплины следует незамедлительно в самом начале учебного семестра. Рекомендуется изучить структуру и основные положения Рабочей программы дисциплины. Обратит внимание, что кроме аудиторной работы (лекции, лабораторные занятия) планируется самостоятельная работа, итоги которой влияют на окончательную оценку по итогам освоения учебной дисциплины. Все задания (аудиторные и самостоятельные) необходимо выполнять и предоставлять на оценку в соответствии с графиком.

В процессе изучения материалов учебного курса предлагаются следующие формы работ: чтение лекций, лабораторные занятия, задания для самостоятельной работы.

Лекционные занятия ориентированы на освещение вводных тем в каждый раздел курса и призваны ориентировать студентов в предлагаемом материале, заложить научные и методологические основы для дальнейшей самостоятельной работы студентов.

Лабораторные занятия акцентированы на наиболее принципиальных и проблемных вопросах курса и призваны стимулировать выработку практических умений.

Особо значимой для профессиональной подготовки студентов является *самостоятельная работа* по курсу. В ходе этой работы студенты отбирают необходимый материал по изучаемому вопросу

Освоение курса способствует развитию навыков обоснованных и самостоятельных оценок фактов и концепций. Поэтому во всех формах контроля знаний, особенно при сдаче зачета, внимание обращается на понимание проблематики курса, на умение практически применять знания и делать выводы.

Работа с литературой. Рекомендуется использовать различные возможности работы с литературой: фонды научной библиотеки ДВФУ и электронные библиотеки (<http://www.dvfu.ru/library/>), а также доступные для использования другие научно-библиотечные системы.

Подготовка к экзамену/зачету. К сдаче экзамена/зачета допускаются обучающиеся, выполнившие все задания (лабораторные, самостоятельные), предусмотренные учебной программой дисциплины, посетившие не менее 85% аудиторных занятий.

VII. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Перечень материально-технического и программного обеспечения дисциплины приведен в таблице.

Материально-техническое и программное обеспечение дисциплины

Наименование специальных помещений и помещений для самостоятельной работы	Оснащенность специальных помещений и помещений для самостоятельной работы	Перечень лицензионного программного обеспечения. Реквизиты подтверждающего документа
<p>690922, Приморский край, г. Владивосток, остров Русский, полуостров Саперный, поселок Аякс, 10, корпус D, ауд. D 733, 733а. Учебная аудитория для проведения занятий лекционного типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации</p>	<p>Помещение укомплектовано специализированной учебной мебелью (посадочных мест – 13) Оборудование: ЖК-панель 47", Full HD, LG M4716 CCBA – 1 шт. Доска аудиторная, Моноблок Lenovo C360G-i34164G500UDK с лицензионными программами Microsoft Office 2013 (13 шт.) и аудиовизуальными средствами проектор Panasonic DLPPjectorPT-D2110XE</p>	<p>1С Предприятия8 (8.2), 7-Zip, ABBYY Lingvo12, Alice 3, Anaconda3, Autodesk, CodeBlocks, CorelDRAW X7, Dia, Directum4.8, DosBox-0.74, Farmanager, Firebird 2.5, FlameRobin, Foxit Reader, Free Pascal, Geany, Ghostscript, Git, Greenfoot, gsview, Inscapе0.91, Java, Java development Kit, Kaspersky, Lazarus, LibreOffice4.4, MatLab R2017b, Maxima 5.37.2, Microsoft Expression, Microsoft Office 2013, Microsoft Silverlight, Microsoft Silverlight 5 SDK-русский, Microsoft System Center, Microsoft Visual Studio 2012, MikTeX2.9, MySQL, NetBeans, Notepad++, Oracle VM VirtualBox, PascalABC.NET, PostgreSQL 9.4, PTC Mathcad, Putty, PyQt GPL v5.4.1 for Pythonv 3.4, Python2.7(3.4,3.6), QGIS Brighton, RStudio, SAM CoDeC Pack, SharePoint, Strawberry Perl, Tecnomatix, TeXnicCenter, TortoiseSVN, Unity2017.3.1f1, Veusz, Vim8.1, Visual Paradigm CE, Visual Studio2013, Windows Kits, Windows Phone SDK8.1, Xilinx Design Tools Acrobat ReaderDC, Adobe Bridge CS3, Adobe Device Central CS3, Adobe ExtendScript Toolkit 2, Adobe Photoshop CS3, DVD-студия Windows, Google Chrome, Internet Explorer, ITMO Proctor, Mozilla Firefox, Visual Studio Installer, Windows Media Center, WinSCP,</p>

В целях обеспечения специальных условий обучения инвалидов и лиц с ограниченными возможностями здоровья в ДВФУ все здания оборудованы пандусами, лифтами, подъемниками, специализированными местами, оснащенными туалетными комнатами, табличками информационно-навигационной поддержки.

VIII. ФОНДЫ ОЦЕНОЧНЫХ СРЕДСТВ

№ п/п	Контролируемые разделы/темы дисциплины	Код индикатора достижения компетенции	Результаты обучения	Оценочные средства	
				текущий контроль	промежуточная аттестация
1.	Модуль 1-3	ПК – 1.1 ПК – 1.2 ПК – 1.3	Знает способы получения оценок моделей и методов машинного обучения, а также критериев их сравнения.	Собеседование УО1, круглый стол УО-4	Зачет с оценкой Вопросы 1-13
			Умеет поставить задачу машинного обучения в заданной предметной области и предложить метод ее решения.		
			Владет навыками работы с платформами, средами и программными средствами, в библиотеках которых реализованы методы машинного обучения.		
		ПК – 4.1 ПК -4.2	Знает способы выбора наиболее подходящего метода машинного обучения для заданной предметной области на основе оценки и анализа свойств существующих методов.		
			Умеет проанализировать заданную предметную область на предмет возможности и необходимости применения в ней методов машинного обучения.		
Владет навыками применения заданного метода машинного обучения (на основе существующих библиотек), а также получения результата (на основе заданного набора данных).	Пр/работы 1-7, ПР-6				
2.	Модуль 4-5	ПК – 1.1 ПК – 1.2 ПК – 1.3 ПК – 4.1 ПК -4.2	Знает области применения, преимущества и недостатки основных алгоритмов и технологий обработки данных.	Собеседование УО1, круглый стол УО-4	Зачет с оценкой Вопросы 14-25 (практическая часть)
			Умеет оценить степень применимости, а также временную и вычислительную сложности выбранного алгоритма для решения конкретной прикладной задачи.		

			Владеет способностью провести анализ степени применимости имеющихся алгоритмов для всех возможных (альтернативных) решений конкретной прикладной задачи.		
			Владеет способностью выбрать из имеющегося набора алгоритмов одного, наиболее подходящего с точки зрения экономии ресурсов, а также качества получаемого результата.		

Методические материалы, определяющие процедуры оценивания результатов освоения дисциплины

Текущая аттестация студентов проводится в соответствии с локальными нормативными актами ДВФУ и является обязательной. Текущая аттестация проводится в форме контрольных мероприятий: собеседование, защита лабораторных работ.

Объектами оценивания выступают:

- учебная дисциплина (активность на занятиях, своевременность выполнения различных видов заданий, посещаемость всех видов занятий по аттестуемой дисциплине);
- степень усвоения теоретических знаний;
- уровень овладения практическими умениями и навыками по всем видам учебной работы;
- результаты самостоятельной работы.

Во время выполнения лабораторных работ преподаватель на основе серии контрольных вопросов проверяет теоретические знания студента по теме лабораторной работы. Для экзамена подготовлены 12 билетов. Оценка качества освоения дисциплины производится по результатам следующих контролирующих мероприятий:

Список вопросов к зачету

1. Обзор основных моделей нейросетей.
2. Сверточные нейронные сети (свертки и сверточные сети; часто используемые архитектуры).
3. Рекуррентные нейросети. Обработка последовательностей и “ванильная” рекуррентная сеть;
4. Использование Earthengine google для анализа спутниковой съемки.
5. Основы работы с TensorFlow.
6. Построение многослойного перцептрона в Google EarthEngine.

7. Построение сверточной сети средствами TensorFlow.
8. Анализ последовательностей изображений в Google EarthEngine.
9. Реализация “ванильной” рекуррентной нейронной сети в TensorFlow.
1. Методы оптимизации. Градиентный спуск.
2. Линейная регрессия.
3. Глобальная оптимизация. Генетический алгоритм.
4. Метод ближайших соседей (k-NN)
5. Наивный байесов классификатор
6. Логистическая регрессия
7. Сигмоид. Метод наибольшего правдоподобия.
8. Деревья решений
9. Кластеризация
10. Снижение размерности
11. Метод опорных векторов (SVM)
12. Кросс-валидация. Подбор гиперпараметров. Визуализация данных.
13. Методы селекции: пропорционально качеству, универсальная выборка (stochastic universal sampling), с наследием (reward-based), турнир. Стратегия элитизма.
14. Методы кроссовера. Двух и много-точечный, равномерный (по подмножествам), для перестановок.
15. Управление популяцией. Сегрегация, старение, распараллеливание. Генетическое программирование.
16. Наивный классификатор, предположение о независимости признаков. Оценка плотности распределения для числовых признаков. Алгоритмические оптимизации. Алгоритм EM.
17. Задача оптимизации с ограничениями. Двойственная задача Лагранжа. Условия Каруша-Куна-Такера.
18. Преобразование Хафа
19. Алгоритм Кэнни
20. Задача классификации изображений. Основные архитектуры нейронных сетей
21. Задача сегментации. Архитектура Encoder-Decoder. Основные виды нейронных сетей для задачи сегментации
22. Задача локализации. Алгоритм SSD, Алгоритм RCNN, Fast RCNN, Faster RCNN, YOLO.
23. Задача Instance Segmentation. Архитектура Mask RCNN
24. Автоэнкодеры
25. Генеративные модели.

Оценка предварительной подготовки

1. Стек вызова, работа с динамической памяти, время жизни переменных
2. Абстрактные типы данных (стек, очередь)
3. Работа с динамической памятью, разбор и реализация типа данных "Очередь". Работа с многофайловым проектом, применение и практика.
4. Введение в алгоритмы и структуры данных (сложность, классификации итд). Разбор процесса компиляции приложения (препроцессор, компиляция, линковщик). Разбор стандартной библиотеки Си.
5. Наследование и полиморфизм.
6. Шаблоны в языке C++, вывод типов, шаблонные функции, шаблонные классы, итераторы.
7. Обзор стандартной библиотеки C++. Практикум студентов на языке C++. Разбор задачи
8. Объектно-ориентированное программирование. Инкапсуляция, наследование, полиморфизм. Перегрузка операторов. Декораторы.

Критерии выставления оценки студенту на экзамене

Порядок начисления рейтинговых баллов по предмету

Выполнение практических работ – 100 баллов

Баллы (рейтинговой оценки)	Оценка экзамена	Требования к сформированным компетенциям
85-100	«отлично»	Оценка «отлично» выставляется студенту, если он глубоко и прочно усвоил программный материал, исчерпывающе, последовательно, четко и логически стройно его излагает, умеет тесно связывать теорию с практикой, свободно справляется с задачами, вопросами и другими видами применения знаний, причем не затрудняется с ответом при видоизменении заданий, использует в ответе материал монографической литературы, владеет разносторонними навыками и приемами выполнения лабораторных работ.
70-84	«хорошо»	Оценка «хорошо» выставляется студенту, если он твердо знает материал, грамотно и по существу

		излагает его, не допуская существенных неточностей в ответе на вопрос, правильно применяет теоретические положения при решении лабораторных работ вопросов и задач, владеет необходимыми навыками и приемами их выполнения.
50-69	«удовлетворительно»	Оценка «удовлетворительно» выставляется студенту, если он имеет знания только основного материала, но не усвоил его деталей, допускает неточности, недостаточно правильные формулировки, нарушения логической последовательности в изложении программного материала, испытывает затруднения при выполнении лабораторных работ.
0-49	«неудовлетворительно»	Оценка «неудовлетворительно» выставляется студенту, который не знает значительной части программного материала по, допускает существенные ошибки, неуверенно, с большими затруднениями выполняет практические работы. Как правило, оценка «неудовлетворительно» ставится студентам, которые не могут продолжить обучение без дополнительных занятий по соответствующей дисциплине.

Правила аттестации для студентов, не набравших необходимый минимум баллов по дисциплине

Если студент, в ходе изучения дисциплины набрал 70 и более баллов, то он имеет право на выставление соответствующей оценки по экзамену без его сдачи.

Если студент набрал менее 70 баллов, то он должен сдать экзамен (экзаменационный тест). Данный тест оценивается в диапазоне от 0 до 30 баллов. Полученные баллы суммируются к уже набранным и студенту выставляется итоговая оценка.

Типовые практические работы

Задача А. Градиентный спуск

Входной файл: Стандартный вход

Ограничение времени: 1 сек

Выходной файл: Стандартный выход

Ограничение памяти: 512
Мб

- **Условие**

Требуется реализовать класс на языке Python, который соответствует следующему интерфейсу.

```
class GradientOptimizer:
    def __init__(self, oracle, x0):
        self.oracle = oracle
        self.x0 = x0

    def optimize(self, iterations, eps, alpha):
        pass
```

В конструктор принимаются два аргумента — оракул, с помощью которого можно получить градиент оптимизируемой функции, а также точку, с которой необходимо начать градиентный спуск.

Метод `optimize` принимает максимальное число итераций для критерия остановки, L2-норму градиента, которую можно считать оптимальной, а также `learning rate`. Метод возвращает оптимальную точку.

Оракул имеет следующий интерфейс:

```
class Oracle:
    def get_func(self, x)
    def get_grad(self, x)
```

`x` имеет тип `np.array` вещественных чисел.

- **Формат выходных данных**

Код должен содержать только класс и его реализацию. Он не должен ничего выводить на экран.

Задача В. Линейная регрессия. Основы

Входной файл: Стандартный вход

Ограничение времени: 1 сек

Выходной файл: Стандартный выход

Ограничение памяти: 512
Мб

- **Условие**

Требуется реализовать следующие функции на языке Python.

```
def linear_func(theta, x)                # function value
def linear_func_all(theta, X)            # 1-d np.array of function values
of all rows of the matrix X
def mean_squared_error(theta, X, y)      # MSE value of current regression
def grad_mean_squared_error(theta, X, y) # 1-d array of gradient by theta
```

θ — одномерный `np.array`

x — одномерный `np.array`

X — двумерный `np.array`. Каждая строка соответствует по размерности вектору θ

y — реальные значения предсказываемой величины

Матрица XX имеет размер $M \times NM \times N$. MM строк и NN столбцов.

Используется линейная функция вида: $h_{\theta}(x) = \theta_1 x_1 + \theta_2 x_2 + \dots + \theta_n x_n$

Mean squared error (MSE) как функция от θ : $J(\theta) = \frac{1}{M} \sum_{i=1}^M (y_i - h_{\theta}(x^{(i)}))^2$

Где $x^{(i)}$ — i -я строка матрицы XX

Градиент функции MSE: $\nabla J(\theta) = \{\partial J / \partial \theta_1, \partial J / \partial \theta_2, \dots, \partial J / \partial \theta_N\}$

- **Пример**

```
X = np.array([[1, 2], [3, 4], [4, 5]])
```

```
theta = np.array([5, 6])
```

```
y = np.array([1, 2, 1])
```

```
linear_func_all(theta, X) # -> array([17, 39, 50])
```

```
mean_squared_error(theta, X, y) # -> 1342.0
```

```
grad_mean_squared_error(theta, X, y) # -> array([215.33333333, 283.33333333])
```

- **Формат выходных данных**

Код должен содержать только реализацию функций.

Задача С. Найти линейную регрессию

Входной файл: Стандартный вход

Ограничение времени: 10 сек

Выходной файл: Стандартный выход

Ограничение памяти: 512 Мб

- **Условие**

Требуется реализовать функцию на языке Python, которая находит линейную регрессию заданных векторов, используя метрику MSE.

```
def fit_linear_regression(X, y) # np.array of linear regression coeffs
```

X — двумерный `np.array`. Каждая строка соответствует отдельному примеру.

y — реальные значения предсказываемой величины

- **Формат выходных данных**

Код должен содержать только реализацию функций.

Распределение задач

Входной файл: `input.txt`

Ограничение времени: 1 сек

Выходной файл: `output.txt`

Ограничение памяти: 256 Мб

- **Условие**

Группа разработчиков работает над проектом. Весь проект разбит на задачи, для каждой задачи указывается ее категория сложности (1, 2, 3 или 4), а также оценочное время выполнения задачи в часах. Проект считается выполненным, если выполнены все задачи. Для каждого разработчика и для каждой категории сложности задачи указывается коэффициент, с которым, как ожидается, будет соотноситься реальное время выполнения задачи данным разработчиком к оценочному времени. Считается, что все разработчики начинают работать с проектом в одно и то же время и выделяют для работы одинаковое время. Необходимо реализовать программу, распределяющую задачи по разработчикам, с целью минимизировать время выполнения проекта (получить готовый проект за минимальный промежуток времени). Поиск решения необходимо реализовать с помощью генетического алгоритма.

- **Отправка решения и тестирование**

Данная задача будет проверяться на *ОДНОМ* входном файле. Этот файл можно скачать [ЗДЕСЬ](#).

В качестве решения принимается текстовый файл, содержащий ответ к задаче в требуемом формате (при его отправке следует выбрать в тестирующей системе среду разработки "Answer text").

Решение набирает количество баллов, вычисляемое по следующей формуле: $Score = 106 \cdot T_{max}$. T_{max} — наибольшее среди всех разработчиков время, затраченное на выполнение выданных соответствующему разработчику задач.

- **Формат входного файла**

Первая строка входного файла содержит целое число NN количество задач.

Вторая строка — NN целых чисел от 1 до 4 категорий сложности задач.

Третья строка — NN вещественных положительных чисел оценочного времени для задач.

Четвертая строка — целое число MM, количество разработчиков.

Следующие MM строк содержат по 4 вещественных положительных числа — коэффициенты каждого разработчика.

- **Формат выходного файла**

Первая и единственная строка выходного файла содержит NN целых чисел w_i — номер разработчика, назначенного на i -ю задачу.

- **Ограничения**

- **Примеры тестов**

№	Входной файл (input.txt)	Выходной файл (output.txt)
1	3 1 1 4 5.2 3.4 4 2 1 1 2 5 0.7 1 1.2 1.5	1 2 2

Логистическая регрессия. Основы

Входной файл: Стандартный вход

Ограничение времени: 1 сек

Выходной файл: Стандартный выход

Ограничение памяти: 512 МБ

- **Условие**

Требуется реализовать следующие функции на языке Python.

```
def logistic_func(theta, x) # function value
def logistic_func_all(theta, X) # 1-d np.array of function
values of all rows of the matrix X
def cross_entropy_loss(theta, X, y) # cross entropy loss value of
current regression
def grad_cross_entropy_loss(theta, X, y) # 1-d array of gradient by theta
```

theta — одномерный np.array

x — одномерный np.array

X — двумерный np.array. Каждая строка соответствует по размерности вектору theta

y — реальные значения предсказываемой величины

Матрица XX имеет размер $M \times N$. MM строк и NN столбцов.

Используется линейная функция вида: $h_{\theta}(x) = \theta_1 x_1 + \theta_2 x_2 + \dots + \theta_n x_n$

- **Формат выходных данных**

Код должен содержать только реализацию функций.

Найти логистическую регрессию

Входной файл: Стандартный вход

Ограничение времени: 10 сек

Выходной файл: Стандартный выход

Ограничение памяти: 512 Мб

- **Условие**

Требуется реализовать функцию на языке Python, которая находит логистическую регрессию заданных векторов, используя метрику cross entropy loss.

```
def fit_logistic_regression(X, y) # np.array of logistic regression coeffs
```

X — двумерный `np.array`. Каждая строка соответствует отдельному примеру.

y — реальные значения предсказываемой величины

- **Формат выходных данных**

Код должен содержать только реализацию функций.

News category

Входной файл: input.txt

Ограничение времени: 1 сек

Выходной файл: output.txt

Ограничение памяти: 256 Мб

- **Условие**

Требуется обучить модель определения категории новости. Обучающую выборку можно скачать [ЗДЕСЬ](#). Категория новости в обучающей выборке представлена столбцом `CAT`.

- `HEADER` — заголовок новости
- `MEDIANAME` — название СМИ
- `WEBSITE` — вебсайт СМИ
- `PTIME` — время публикации

Для определения качества модели будет использоваться тестовая выборка, доступная [ЗДЕСЬ](#).

В тестовой выборке требуется предсказать значения столбца `CAT`, соответствующие каждому тестовому примеру. Категории новостей кодируются одним символом, аналогично данным в обучающей выборке.

- **Отправка решения и тестирование**

Данная задача будет проверяться на *ОДНОМ* входном файле.

В качестве решения принимается текстовый файл, содержащий ответ к задаче в требуемом формате (при его отправке следует выбрать в тестирующей системе среду разработки "Answer text").

Решение набирает количество баллов, вычисляемое по следующей формуле: $Score = 105 \cdot AccuracyScore$. `AccuracyScore` — доля верно классифицированных новостей относительно всех новостей в тестовой выборке.

- **Формат выходного файла**

Каждая строка выходного файла должна содержать единственный символ, задающий категорию соответствующего тестового примера.

Качество вина

Входной файл: input.txt
Выходной файл: output.txt

Ограничение времени: 1 сек
Ограничение памяти: 256 Мб

- **Условие**

Требуется обучить модель определения качества вина. Качество вина определяется по 1010-балльной шкале. В данной задаче будем использовать бинарную модель и предсказывать, "хорошее" вино или "плохое". Хорошим будем считать вино с качеством строго выше 66. Обучающую выборку можно скачать [ЗДЕСЬ](#). Качество вина представлено столбцом `quality`.

Для определения качества модели будет использоваться тестовая выборка, доступная [ЗДЕСЬ](#).

В тестовой выборке требуется предсказать значения 11 или 00, "хорошее" вино или "плохое" соответственно, для каждого примера. Оценку качества по 1010-балльной шкале предсказывать не требуется.

- **Отправка решения и тестирование**

Данная задача будет проверяться на *ОДНОМ* входном файле.

В качестве решения принимается текстовый файл, содержащий ответ к задаче в требуемом формате (при его отправке следует выбрать в тестирующей системе среду разработки "Answer text").

Решение набирает количество баллов, вычисляемое по следующей формуле: $Score=10s \cdot F1$ $Score=105 \cdot F1$.

- **Формат выходного файла**

Каждая строка выходного файла должна содержать целое число 11 или 00. Количество строк должно быть равно количеству эл

Шкала оценивания

Менее 60 баллов	Незачтено	неудовлетворительно
От 61 до 75 баллов	Зачтено	удовлетворительно
От 76 до 85 баллов	Зачтено	хорошо
От 86 до 100 баллов	Зачтено	отлично