

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное
учреждение высшего образования
«Кубанский государственный университет»
Факультет компьютерных технологий и прикладной математики

УТВЕРЖДАЮ

Проректор по учебной работе,
качеству образования – первый
проректор

Хагуров Т.А.

подпись

«02» июня 2026 г.

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ

Б1.В.20 «ИНТЕЛЛЕКТУАЛЬНЫЕ МЕТОДЫ ОПТИМИЗАЦИИ»

Направление подготовки 02.03.03 Математическое обеспечение и
администрирование информационных систем

Направленность (профиль) Искусственный интеллект и аналитика данных

Форма обучения очная

Квалификация бакалавр

Краснодар 2026

Рабочая программа дисциплины «Интеллектуальные методы оптимизации» составлена в соответствии с Федеральным государственным образовательным стандартом высшего образования (ФГОС ВО) по направлению подготовки 02.03.03 Математическое обеспечение и администрирование информационных систем.

Программу составил(и):

В.А. Акиньшина, канд. пед. наук, доцент


подпись

Рабочая программа дисциплины «Интеллектуальные методы оптимизации» на заседании центра искусственного интеллекта протокол № 4 от «14» мая 2026 г.

Руководитель центра ИИ Коваленко А.В.


подпись

Утверждена на заседании учебно-методической комиссии факультета компьютерных технологий и прикладной математики протокол №3 от «15» мая 2026 г.

Председатель УМК факультета
Н. Ю. Добровольская


подпись

Рецензенты:

Мостовой Евгений Викторович, генеральный директор ООО «Портал-Юг»,
e-mail: mostovoy@portal-yug.ru

Луценко Евгений Вениаминович, доктор экономических наук, кандидат технических наук, профессор кафедры компьютерных технологий и систем Федерального государственного бюджетное образовательное учреждение высшего образования «Кубанский государственный аграрный университет имени И.Т. Трубилина», e-mail: prof.lutsenko@gmail.com

1 Цели и задачи изучения дисциплины (модуля)

1.1 Цель освоения дисциплины

Цели изучения дисциплины определены государственным образовательным стандартом высшего образования и соотнесены с общими целями ООП ВО по направлениям подготовки, в рамках которых преподается дисциплина.

Дисциплина посвящена овладению студентами знаний по основам теории оптимизации и знаний об основных подходах к практическому решению оптимизационных задач, что позволит применять методы анализа прикладной области на концептуальном, логическом, математическом и алгоритмическом уровнях, а также применять системный подход и математические методы в формализации решения прикладных задач.

1.2. Задачи дисциплины. В ходе изучения дисциплины ставятся задачи научить студентов:

- знать содержание программы курса, формулировки задач, методы их исследования;
- выбирать подходящие методы для решения экстремальных задач;
- уметь применять на практике конкретные вычислительные методы к анализу и решению оптимизационных задач;
- изучать самостоятельно научную и учебно-методическую литературу по профилю из различных источников для решения профессиональных и социальных задач.

1.3. Место учебной дисциплины в структуре образовательной программы

Дисциплина «Интеллектуальные методы оптимизации» относится к вариативной части Блока 1 «Дисциплины (модули)» учебного плана.

Данная дисциплина («Интеллектуальные методы оптимизации») тесно связана с дисциплинами: «Математический анализ», «Алгебра и геометрия», «Методы математической физики». В совокупности изучение этой дисциплины готовит обучаемых как к различным видам практической экономической деятельности, так и к научно-теоретической и исследовательской деятельности.

1.4. Профессиональные роли в структуре образовательной программы

Роль 1: Data Analyst (Аналитик данных)

Задачи:

1. Статистический анализ, визуализация данных, предварительная обработка.
2. Создание прогнозных моделей
3. Построение аналитических моделей для поддержки бизнес-решений.

Роль 2: MLOps (Специалист по эксплуатации ИИ)

Задачи:

1. DevOps для ML.
2. Автоматизация, мониторинг ML-систем.
3. Операционное управление жизненным циклом ML-моделей.

Роль 3: AI PM (Менеджер проектов ИИ)

Задачи:

1. Управление ИИ-проектами от идеи до внедрения
2. Анализ бизнес-требований и постановка задач
3. Оценка эффективности и ROI ИИ-решений

1.5. Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами освоения образовательной программы

Изучение данной учебной дисциплины направлено на формирование у обучающихся следующих компетенций:

Код, уровень и формулировка компетенции	Индикаторы	Уровни освоения индикаторов компетенции
DL-1 Способен применять и (или) разрабатывать архитектуры глубоких нейронных сетей	DL-1.1 Способен объяснять и применять математические основы нейронных сетей, включая расчет градиентов, методы оптимизации и алгоритм обратного распространения ошибки (backpropagation), для эффективного обучения моделей.	(П) Имеет навыки реализации и обучения простых полносвязных нейронных сетей с использованием высокоуровневых API (keras, Sequential), может использовать встроенные функции для автоматического дифференцирования (autograd) в PyTorch или TensorFlow. Владеет базовыми навыками отладки процесса обучения: анализ кривых обучения (loss/accuracy). Способен провести сравнительный анализ различных методов оптимизации и архитектурных решений на конкретной задаче, аргументировав выбор наилучшего подхода.
DL-1 Способен применять и (или) разрабатывать архитектуры глубоких нейронных сетей	DL-1.2 Способен реализовывать неглубокие нейронные сети (перцептроны, MLP), выбирать количество и размер слоёв, подходящие функции активации и функции потерь для решения задач классификации и регрессии DL-1.3 Способен применять современные архитектуры глубоких сетей для решения различных задач, понимая их внутреннюю структуру и особенности обучения.	(П) Реализовывает неглубокие нейронные сети (перцептроны, MLP), выбирает количество и размер слоёв, подходящие функции активации и функции потерь для решения задач классификации и регрессии. Владеет продвинутыми техниками регуляризации и их интеграцией в модель для ее улучшения. Оптимизирует гиперпараметры для улучшения качества. Вносит структурные изменения в существующие архитектуры (добавлять механизмы внимания в CNN, создавать гибридные модели CNN+RNN), реализовывать собственные слои или блоки. Может анализировать динамику градиентов, активаций и

Код, уровень и формулировка компетенции	Индикаторы	Уровни освоения индикаторов компетенции
		весов в глубоких сетях, идентифицировать специфические проблемы и применять targeted-решения. Проводит глубокий сравнительный анализ архитектур, аргументировать выбор не только по точности, но и по вычислительной сложности, объему памяти и пригодности для практического применения.
MF-3 Способен применять современные методы оптимизации для обучения моделей машинного обучения настройки гиперпараметров и решения задач ИИ	MF-3.2 Применяет методы оптимизации для настройки гиперпараметров моделей машинного обучения, включая использование методов поиска (grid search, random search) и байесовской оптимизации	(Б) Умеет автоматизировать процесс настройки моделей машинного обучения, эффективно используя различные методы оптимизации - от простого перебора до интеллектуального байесовского поиска.
MF-3 Способен применять современные методы оптимизации для обучения моделей машинного обучения настройки гиперпараметров и решения задач ИИ	MF-3.1 Применяет методы оптимизации для разработки и исследования обучающих алгоритмов МО (градиентные методы, Adam, RMSprop, оптимизация гиперпараметров)	(Б) Выбрать оптимизатор под задачу, провести grid/random search (Optuna, GridSearchCV), настроить расписание LR.. Реализовать кастомный оптимизатор, настроить распределенный поиск гиперпараметров (Ray Tune, Optuna с БД), применить ASHA.
О-3 Способен применять и (или) разрабатывать интеллектуальные методы оптимизации	О-3.2 Обосновывает способы и варианты применения интеллектуальных методов в задачах оптимизации	(Б) Умеет анализировать сложность задачи и обосновывать выбор конкретного интеллектуального метода. Модифицировать стандартные алгоритмы под специфику конкретной задачи, сравнивать эффективность разных методов на реальных кейсах, оценивать вычислительную сложность и сходимость методов, комбинировать интеллектуальные методы с классическими подходами
О-3 Способен применять и	О-3.3 Эффективно применяет интеллектуальные ме-	Самостоятельно выбирает и настраивает метод под конкретную задачу.

Код, уровень и формулировка компетенции	Индикаторы	Уровни освоения индикаторов компетенции
(или) разрабатывать интеллектуальные методы оптимизации	тоды оптимизации для обеспечения функциональных характеристик систем ИИ	Применяет методы сжатия моделей для баланса между точностью и производительностью. Проводит А/В-тестирование для сравнения исходной и оптимизированной версии. Управляет компромиссами между вычислительной сложностью, точностью и энергоэффективностью на уровне архитектуры ИИ-системы. Проводит бенчмаркинг и выбирает оптимальную стратегию оптимизации для крупных промышленных систем.
FC-1 Способен проводить фронтальные исследования в области архитектур, алгоритмов МО, оптимизации и математики	FC-1.1 Разрабатывает фундаментальные основы и новые алгоритмы машинного обучения	(Б) Способен выявить фундаментальную проблему, сдерживающую развитие целой области, и сформулировать программу исследований для ее решения. Способен синтезировать принципиально новые математические формулировки задач и архитектуры алгоритмов, а не просто комбинировать существующие блоки.

Результаты обучения по дисциплине «Интеллектуальные методы оптимизации» достигаются в рамках осуществления всех видов контактной и самостоятельной работы обучающихся в соответствии с утвержденным учебным планом.

Индикаторы достижения компетенций считаются сформированными при достижении соответствующих им результатов обучения.

2. Структура и содержание дисциплины

2.1 Распределение трудоёмкости дисциплины по видам работ

Общая трудоёмкость дисциплины составляет 3 зач. ед. (108 часов), их распределение по видам работ представлено в таблице.

Вид работы	Трудовое время, часов
	6 семестр
Контактная работа, в том числе:	68,3
Аудиторная работа:	64
<i>Лекции (Л)</i>	32
<i>Практические занятия (ПЗ)</i>	
<i>Лабораторные работы (ЛР)</i>	32

Иная контактная работа:		
Контроль самостоятельной работы (КСР)		4
Промежуточная аттестация (ИКР)		0,3
Самостоятельная работа (СР):		4
Курсовой проект (КП), курсовая работа (КР)		-
Самоподготовка (проработка и повторение лекционного материала и материала учебников и учебных пособий, подготовка к лабораторным и практическим занятиям, коллоквиумам и т.д.)		-
Выполнение индивидуальных заданий (подготовка сообщений, презентаций)		4
Реферат		-
Подготовка к текущему контролю		-
Контроль:		
Подготовка и сдача экзамена ¹		35,7
Общая трудоемкость	час.	108
	в том числе контактная работа	68,3
	зач. ед	4
Вид итогового контроля		экзамен

2.2 Структура дисциплины:

Распределение видов учебной работы и их трудоемкости по разделам дисциплины.
Разделы (темы) дисциплины, изучаемые в 6 семестре

№	Наименование разделов (тем)	Количество часов				
		Всего	Аудиторная работа			Внеаудиторная работа
			Л	ПЗ	ЛР	
	1 Безусловная одномерная оптимизация					
1.	Формулировка математической задачи оптимизации. Классические методы решения задач одномерной оптимизации	4	2		2	
2.	Численные методы решения задач одномерной оптимизации	4	2		2	
	2 Безусловная многомерная оптимизация					
3.	Классические методы решения задач многомерной оптимизации.	2	2		0	
4.	Классификация и обзор методов безусловной оптимизации	2	2		0	
5.	Численные методы безусловной оптимизации функции многих переменных. Методы первого порядка.	10	4		6	
6.	Численные методы безусловной оптимизации функции многих переменных. Методы второго порядка.	10	4		6	
	3 Нелинейное программирование					
7.	Классификация задач нелинейного программирования.	4	4		0	

¹ При наличии экзамена по дисциплине

8.	Задачи линейного программирования	6	2		4	
	4 Специальные интеллектуальные методы оптимизации					
9.	<i>Методы решения задач с применением интеллектуальных методов оптимизации</i>	10	4		6	
10.	<i>Примеры задач с применением интеллектуальных методов оптимизации</i>	16	6		6	4
	ИТОГО по разделам дисциплины	68	32		32	4
	Контроль самостоятельной работы (КСР)	4				
	Промежуточная аттестация (ИКР)	0,3				
	Подготовка к текущему контролю	35,7				
	Общая трудоемкость по дисциплине	108				

Примечание: Л – лекции, ЛР – лабораторные занятия, СРС – самостоятельная работа студента, КСР -контролируемая самостоятельная работа студентов

2.3. Содержание разделов дисциплины

2.3.1 Занятия лекционного типа

№	Наименование раздела	Содержание раздела	Форма текущего контроля
1	Безусловная одномерная оптимизация	Тема 1. Формулировка математической задачи оптимизации. Классические методы решения задач одномерной оптимизации Тема 2. Численные методы решения задач одномерной оптимизации	Т, ЛР
2	Безусловная многомерная оптимизация	Тема 1. Классические методы решения задач многомерной оптимизации. Тема 2. Классификация и обзор методов безусловной оптимизации Тема 3. Численные методы безусловной оптимизации функции многих переменных. Методы первого порядка. Тема 4. Численные методы безусловной оптимизации функции многих переменных. Методы второго порядка.	Т, ЛР
3	Нелинейное программирование	Тема 1. Классификация задач нелинейного программирования. Тема 2. Задачи нелинейного программирования	Т, ЛР
4	Специальные интеллектуальные методы оптимизации	Тема 1. Методы решения задач с применением интеллектуальных методов оптимизации Тема 2. Примеры задач с применением ин-	Т, ЛР

	мизации	теллектуальных методов оптимизации	
--	---------	------------------------------------	--

Примечание: ЛР – отчет/защита лабораторной работы, КП – выполнение курсового проекта, КР – курсовой работы, РГЗ – расчетно-графического задания, Р – написание реферата, Э – эссе, К – коллоквиум, Т – тестирование, РЗ – решение задач.

2.3.2 Занятия семинарского типа

Семинарские занятия – не предусмотрены

2.3.3 Лабораторные занятия

№ п/п	Темы лабораторных занятий	Наименование лабораторных работ	Форма текущего контроля
1	Численные методы решения задач одномерной оптимизации	Постановка задачи методов первого порядка; стратегия поиска минимума; разработка алгоритмов; оценка сходимости 1. Алгоритм Свенна 2. Метод Дихотомии 3. Метод Фибоначчи 4. Метод золотого сечения. 5. Метод Розенброка 6. Отчет по результатам индивидуального задания.	Т, ЛР
2	Численные методы безусловной оптимизации функции многих переменных. Методы первого порядка.	Постановка задачи методов первого порядка; стратегия поиска минимума; разработка алгоритмов; оценка сходимости 1. Метод градиентного спуска 2. Метод наискорейшего градиентного спуска 3. Метод Флетчера-Ривса 4. Метод Давидона-Флетчера-Пауэлла Отчет по результатам индивидуального задания.	Т, ЛР
3	Численные методы безусловной оптимизации функции многих переменных. Методы второго порядка.	Постановка задачи методов первого порядка; стратегия поиска минимума; разработка алгоритмов; оценка сходимости 1. Метод Ньютона 2. Метод Ньютона-Равсона 3. Метод Маквардта 4. Отчет по результатам индивидуального задания.	Т, ЛР
4	Задачи нелинейного программирования	1. Метод множителей Лагранжа. 2. Метод штрафных функций. 3. Метод барьерных функций. 4. Метод проекции градиента. 5. Отчет по результатам индивидуального задания.	Т, ЛР
5	Методы решения задач с применением интеллектуальных методов оптимизации	1. Методы Монте-Карло 2. Отчет по результатам индивидуального задания.	Т, ЛР

6	Примеры задач с применением интеллектуальных методов оптимизации.	1. Генетические алгоритмы. 2. Нейронные сети 3. Отчет по результатам индивидуального задания.	Т, ЛР
---	---	---	-------

Примечание: ЛР – отчет/защита лабораторной работы, КП - выполнение курсового проекта, КР - курсовой работы, РГЗ - расчетно-графического задания, Р - написание реферата, Э - эссе, К - коллоквиум, Т – тестирование, РЗ – решение задач.

2.3.4 Примерная тематика курсовых работ (проектов)

Курсовые работы (КР) – не предусмотрены.

2.4 Перечень учебно-методического обеспечения для самостоятельной работы обучающегося по дисциплине

Целью самостоятельной работы студента является углубление знаний, полученных в результате аудиторных занятий. Вырабатываются навыки самостоятельной работы. Закрепляются опыт и знания полученные во время лабораторных занятий.

№	Вид СРС	Перечень учебно-методического обеспечения дисциплины по выполнению самостоятельной работы
1	2	3
1	Изучение теоретического материала	Методические указания по организации самостоятельной работы студентов, утвержденные УСФ, протокол №1 от 30.06.2025
2	Решение задач	Методические указания по организации самостоятельной работы студентов, утвержденные УСФ, протокол №1 от 30.06.2025

Учебно-методические материалы для самостоятельной работы обучающихся из числа инвалидов и лиц с ограниченными возможностями здоровья (ОВЗ) предоставляются в формах, адаптированных к ограничениям их здоровья и восприятия информации:

Для лиц с нарушениями зрения:

- в печатной форме увеличенным шрифтом,
- в форме электронного документа,
- в форме аудиофайла,
- в печатной форме на языке Брайля.

Для лиц с нарушениями слуха:

- в печатной форме,
- в форме электронного документа.

Для лиц с нарушениями опорно-двигательного аппарата:

- в печатной форме,
- в форме электронного документа,
- в форме аудиофайла.

Данный перечень может быть конкретизирован в зависимости от контингента обучающихся.

3. Образовательные технологии

В соответствии с требованиями ФГОС в программа дисциплины предусматривает использование в учебном процессе следующих образовательные технологии: чтение лекций с использованием мультимедийных технологий; метод малых групп, разбор практических задач и кейсов.

При обучении используются следующие образовательные технологии:

– Технология коммуникативного обучения – направлена на формирование коммуникативной компетентности студентов, которая является базовой, необходимой для адаптации к современным условиям межкультурной коммуникации.

– Технология разноуровневого (дифференцированного) обучения – предполагает осуществление познавательной деятельности студентов с учётом их индивидуальных способностей, возможностей и интересов, поощряя их реализовывать свой творческий потенциал. Создание и использование диагностических тестов является неотъемлемой частью данной технологии.

– Технология модульного обучения – предусматривает деление содержания дисциплины на достаточно автономные разделы (модули), интегрированные в общий курс.

– Информационно-коммуникационные технологии (ИКТ) – расширяют рамки образовательного процесса, повышая его практическую направленность, способствуют интенсификации самостоятельной работы учащихся и повышению познавательной активности. В рамках ИКТ выделяются 2 вида технологий:

– Технология использования компьютерных программ – позволяет эффективно дополнить процесс обучения языку на всех уровнях.

– Интернет-технологии – предоставляют широкие возможности для поиска информации, разработки научных проектов, ведения научных исследований.

– Технология индивидуализации обучения – помогает реализовывать личностно-ориентированный подход, учитывая индивидуальные особенности и потребности учащихся.

– Проектная технология – ориентирована на моделирование социального взаимодействия учащихся с целью решения задачи, которая определяется в рамках профессиональной подготовки, выделяя ту или иную предметную область.

– Технология обучения в сотрудничестве – реализует идею взаимного обучения, осуществляя как индивидуальную, так и коллективную ответственность за решение учебных задач.

– Игровая технология – позволяет развивать навыки рассмотрения ряда возможных способов решения проблем, активизируя мышление студентов и раскрывая личностный потенциал каждого учащегося.

– Технология развития критического мышления – способствует формированию разносторонней личности, способной критически относиться к информации, умению отбирать информацию для решения поставленной задачи.

Комплексное использование в учебном процессе всех вышеназванных технологий стимулируют личностную, интеллектуальную активность, развивают познавательные процессы, способствуют формированию компетенций, которыми должен обладать будущий специалист.

Основные виды интерактивных образовательных технологий включают в себя:

– работа в малых группах (команде) – совместная деятельность студентов в группе под руководством лидера, направленная на решение общей задачи путём творческого сложения результатов индивидуальной работы членов команды с делением полномочий и ответственности;

– проектная технология – индивидуальная или коллективная деятельность по отбору, распределению и систематизации материала по определенной теме, в результате которой составляется проект;

– анализ конкретных ситуаций – анализ реальных проблемных ситуаций, имевших место в соответствующей области профессиональной деятельности, и поиск вариантов лучших решений;

– развитие критического мышления – образовательная деятельность, направленная на развитие у студентов разумного, рефлексивного мышления, способного выдвинуть новые идеи и увидеть новые возможности.

Подход разбора конкретных задач и ситуаций широко используется как преподавателем, так и студентами во время лекций, лабораторных занятий и анализа результатов самостоятельной работы. Это обусловлено тем, что при исследовании и решении каждой конкретной задачи имеется, как правило, несколько методов, а это требует разбора и оценки целой совокупности конкретных ситуаций.

Занятия, проводимые с использованием интерактивных технологий

№	Наименование разделов (тем)	Количество часов	
		всего ауд. часов	интерактивные часы
1.	Безусловная одномерная оптимизация	16	2
2.	Безусловная многомерная оптимизация	22	6
3.	Нелинейное программирование	14	4
4.	Специальные интеллектуальные методы оптимизации	12	2
	<i>Итого по дисциплине:</i>	64	14

Темы, задания и вопросы для самостоятельной работы призваны сформировать навыки поиска информации, умения самостоятельно расширять и углублять знания, полученные в ходе лекционных и практических занятий.

Подход разбора конкретных ситуаций широко используется как преподавателем, так и студентами при проведении анализа результатов самостоятельной работы.

Для лиц с ограниченными возможностями здоровья предусмотрена организация консультаций с использованием электронной почты.

Для лиц с нарушениями зрения:

- в печатной форме увеличенным шрифтом,
- в форме электронного документа.

Для лиц с нарушениями слуха:

- в печатной форме,
- в форме электронного документа.

Для лиц с нарушениями опорно-двигательного аппарата:

- в печатной форме,
- в форме электронного документа.

Для лиц с ограниченными возможностями здоровья предусмотрена организация консультаций с использованием электронной почты.

Данный перечень может быть конкретизирован в зависимости от контингента обучающихся.

4. Оценочные и методические материалы

4.1 Оценочные средства для текущего контроля успеваемости и промежуточной аттестации

Оценочные средства предназначены для контроля и оценки образовательных достижений обучающихся, освоивших программу учебной дисциплины «название дисциплины».

Оценочные средства включает контрольные материалы для проведения **текущего контроля** в форме тестовых заданий, кейсов и **промежуточной аттестации** в форме вопросов и заданий к **экзамену**.

Оценочные средства для инвалидов и лиц с ограниченными возможностями здоровья выбираются с учетом их индивидуальных психофизических особенностей.

– при необходимости инвалидам и лицам с ограниченными возможностями здоровья предоставляется дополнительное время для подготовки ответа на экзамене;

– при проведении процедуры оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья предусматривается использование технических средств, необходимых им в связи с их индивидуальными особенностями;

– при необходимости для обучающихся с ограниченными возможностями здоровья и инвалидов процедура оценивания результатов обучения по дисциплине может проводиться в несколько этапов.

Процедура оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья по дисциплине (модулю) предусматривает предоставление информации в формах, адаптированных к ограничениям их здоровья и восприятия информации:

Для лиц с нарушениями зрения:

- в печатной форме увеличенным шрифтом,
- в форме электронного документа.

Для лиц с нарушениями слуха:

- в печатной форме,
- в форме электронного документа.

Для лиц с нарушениями опорно-двигательного аппарата:

- в печатной форме,
- в форме электронного документа.

Данный перечень может быть конкретизирован в зависимости от контингента обучающихся.

Структура оценочных средств для текущей и промежуточной аттестации

№ п/п	Контролируемые разделы (темы) дисциплины*	Код контролируемой компетенции (или ее части)	Наименование оценочного средства	
			Текущий контроль	Промежуточная аттестация
1	Тема 1. Формулировка математической задачи оптимизации. Классические методы решения задач одномерной оптимизации Тема 2. Численные методы решения задач одномерной оптимизации	MF-3.1; MF-3.2; FC-1.1; O-3.2; O-3.3; DL-1.1; DL-1.2; DL-1.3	<i>Тест, Лабораторная работа №1 Лабораторная работа №2</i>	<i>Вопросы к экзамену 1-11</i>
2	Тема 1. Классические методы решения задач многомерной оптимизации. Тема 2. Классификация и обзор методов безусловной оптимизации Тема 3. Численные методы безусловной оптимизации функции многих переменных. Методы первого порядка. Тема 4. Численные методы безусловной оптимизации функции многих переменных. Методы второго порядка.	MF-3.1; MF-3.2; FC-1.1; O-3.2; O-3.3; DL-1.1; DL-1.2; DL-1.3	<i>Лабораторная работа №3 Лабораторная работа №4,</i>	<i>Вопросы к экзамену 12-39</i>

3	Тема 1. Классификация задач нелинейного программирования. Тема 2. Задачи нелинейного программирования	MF-3.1; MF-3.2; FC-1.1; O-3.2; O-3.3; DL-1.1; DL-1.2; DL-1.3	Лабораторная работа №5 Лабораторная работа №6,	Вопросы к экзамену 40-45
4	Тема 1. Методы решения задач с применением интеллектуальных методов оптимизации Тема 2. Примеры задач с применением интеллектуальных методов оптимизации	MF-3.1; MF-3.2; FC-1.1; O-3.2; O-3.3; DL-1.1; DL-1.2; DL-1.3	Лабораторная работа №7, Лабораторная работа №8	Вопросы к экзамену 46-64

Показатели, критерии и шкала оценки сформированных компетенций

Соответствие **пороговому уровню** освоения компетенций планируемым результатам обучения и критериям их оценивания (оценка: **удовлетворительно /зачтено**):

MF-3 Способен применять современные методы оптимизации для обучения моделей машинного обучения настройки гиперпараметров и решения задач ИИ

MF- 3.2

Применяет методы оптимизации для настройки гиперпараметров моделей машинного обучения, включая использование методов поиска (grid search, random search) и байесовской оптимизации

Знать

Определение гиперпараметров модели и их отличие от обучаемых параметров.

Назначение методов поиска по сетке (Grid Search) и случайного поиска (Random Search).

Понятие кросс-валидации и её роли в настройке гиперпараметров.

Основные гиперпараметры популярных моделей (например, max_depth, n_estimators, C, gamma).

Уметь

Запускать GridSearchCV и RandomizedSearchCV из библиотеки Scikit-learn на небольшом наборе гиперпараметров.

Интерпретировать результаты поиска (лучшие параметры, лучшую метрику).

Ограничивать перебор параметров вручную на основе здравого смысла.

Выявлять переобучение при настройке (разница между валидационной и тестовой метрикой).

Владеть

Простейшим пайплайном настройки гиперпараметров с использованием фиксированной сетки.

Навыком чтения документации для подбора диапазонов параметров.

Работой с готовыми примерами кода (ноутбуками) для настройки моделей.

MF-3.1

Применяет методы оптимизации для разработки и исследования обучающих алгоритмов МО (градиентные методы, Adam, RMSprop, оптимизация гиперпараметров)

Знать

Что такое градиентный спуск (batch, стохастический, mini-batch) и его цель — минимизация функции потерь.

Уметь	Основные гиперпараметры градиентного спуска: скорость обучения (learning rate), количество эпох, размер батча. Названия популярных оптимизаторов: SGD, Adam, RMSprop Запустить обучение модели с использованием SGD или Adam в готовом фреймворке (scikit-learn, PyTorch, TensorFlow). Изменить скорость обучения в коде и увидеть изменение сходимости (по графику потерь).
	Перечислить шаги для подбора гиперпараметров вручную (попробовать несколько значений learning rate).
Владеть	Навыком установки и базовой настройки оптимизатора при обучении простой модели (линейная регрессия, логистическая регрессия, мелкая нейросеть). Способностью читать график сходимости (loss curve) и определять переобучение/недообучение. Базовой терминологией для обсуждения проблем оптимизации (расходящийся градиент, затухание градиента).
DL-1 II Способен применять и (или) разрабатывать архитектуры глубоких нейронных сетей	
DL-1.1	<i>Способен объяснять и применять математические основы нейронных сетей, включая расчет градиентов, методы оптимизации и алгоритм об-ратного распространения ошибки (backpropagation), для эффективного обучения моделей.</i>
Знать	Основные понятия и определения нейронных сетей (нейроны, слои, активация). Простейшие виды функций активации (например, сигмоида, ReLU). Что такое прямой проход (forward pass) и обратный проход (backpropagation).
Уметь	Выполнять расчеты прямого прохода для простых однослойных нейронных сетей вручную. Вычислять частные производные простейших функций активации. Использовать стандартные библиотеки Python (NumPy, TensorFlow/Keras) для построения небольших нейронных сетей.
Владеть	Навыком расчета простых частных производных вручную. Умением создавать небольшие модели нейронных сетей с использованием стандартных библиотек.
DL-1.2 Способен реализовывать неглубокие нейронные сети (перцептроны, MLP), выбирать количество и размер слоёв, подходящие функции активации и функции потерь для решения задач классификации и регрессии	
Знать	Базовую структуру нейрона и MLP. Назначение функций активации (ReLU, Softmax) и потерь (MSE, Cross-Entropy). Понимать разницу между задачами классификации и регрессии.
Уметь	Реализовать готовую архитектуру MLP по примеру, используя Keras/PyTorch. Провести базовую предобработку данных (масштабирование). Обучить модель и оценить её качество по стандартной метрике.
Владеть	Навыком работы с фреймворком для запуска обучения. Навыком поиска и применения типовых решений для стандартных задач (бинарная классификация, прогноз значения).
DL-1.3 Способен применять современные архитектуры глубоких сетей для решения различных задач, понимая их внутреннюю структуру и особенности обучения.	
Знать	Основное назначение и область применения ключевых архитектур (CNN — изображения, RNN — последовательности, Transformer — текст/NLP). Базовые компоненты (свертки, пулинг, механизм внимания).

Уметь	Загрузить предобученную модель (например, из torchvision или transformers) и использовать ее для инференса или извлечения признаков. Провести тонкую настройку (fine-tuning) на своих данных по инструкции.
Владеть	Навыком использования библиотек с предобученными моделями. Умением подготовить данные в требуемый для архитектуры формат (например, токенизация для Transformer).
О-3 Б Способен применять и (или) разрабатывать интеллектуальные методы оптимизации	
О-3.2	<i>Обосновывает способы и варианты применения интеллектуальных методов в задачах оптимизации</i>
Знать	Основные понятия и принципы интеллектуальных методов оптимизации (например, генетические алгоритмы, метод градиентного спуска).
Уметь	Применять готовые инструменты и библиотеки для решения простых оптимизационных задач.
Владеть	Базовыми методами анализа результатов оптимизации и способностью интерпретировать полученные результаты.
О-3.3	<i>Эффективно применяет интеллектуальные методы оптимизации для обеспечения функциональных характеристик систем ИИ</i>
Знать	Основные функциональные характеристики систем ИИ (точность, время работы, объём памяти, робастность). Понятие интеллектуальной оптимизации и её отличие от классических методов (градиентный спуск vs эвристики). Типовые задачи оптимизации в ИИ: подбор гиперпараметров, выбор признаков, настройка архитектуры нейронной сети. Принципы работы простейших метаэвристик (случайный поиск, полный перебор).
Уметь	Запускать готовую реализацию генетического алгоритма или случайного поиска для настройки параметров модели ИИ. Читать и интерпретировать графики сходимости оптимизации (значение целевой функции от итерации). Формулировать целевую функцию (например, максимизация точности при ограничении по времени). Следовать шаблону кода для оптимизации характеристик под руководством наставника.
Владеть	Простейшим скриптом на Python с использованием sklearn.model_selection.RandomizedSearchCV. Навыком визуализации одномерных и двумерных зависимостей параметров от функциональной характеристики. Использованием готовых решений (например, оптимизация порога классификации для баланса precision/recall).
FC-1	Способен проводить фронтальные исследования в области архитектур, алгоритмов МО, оптимизации и математики
FC-1.1	Разрабатывает фундаментальные основы и новые алгоритмы машинного обучения
Знать	Фундаментальные основы машинного обучения: типы задач (обучение с учителем, без учителя, с подкреплением), основные алгоритмы (линейная регрессия, SVM, k-means, деревья решений), базовые концепции (переобучение/недообучение, кросс-валидация,

Уметь	метрики качества). Анализировать научную литературу по машинному обучению. Реализовывать и тестировать существующие алгоритмы на стандартных наборах данных. Проводить сравнительный анализ алгоритмов по заданным критериям.
Владеть	Навыками программирования (Python/R) и владеть основными библиотеками (scikit-learn, NumPy). Навыками предобработки данных и оценки моделей.

Соответствие **базовому уровню** освоения компетенций планируемым результатам обучения и критериям их оценивания (оценка: **хорошо /зачтено**):

MF-3	Способен применять современные методы оптимизации для обучения моделей машинного обучения настройки гиперпараметров и решения задач ИИ
MF-3.1	<i>Применяет методы оптимизации для настройки гиперпараметров моделей машинного обучения, включая использование методов поиска (grid search, random search) и байесовской оптимизации</i>
Знать	Как Random Search эффективнее Grid Search при большом числе гиперпараметров. Принцип работы байесовской оптимизации (суррогатная модель, функция приобретения). Компромисс между скоростью и точностью при выборе числа итераций. Особенности настройки разных типов гиперпараметров (непрерывные, категориальные, целочисленные).
Уметь	Самостоятельно выбирать между Grid, Random и простой байесовской оптимизацией для конкретной задачи. Реализовывать байесовскую оптимизацию с помощью библиотек (Optuna, Hyperopt, Scikit-Optimize). Задавать распределения параметров для случайного поиска и байесовской оптимизации. Ограничивать время и ресурсы на подбор параметров (ранняя остановка, уменьшение фолдов).
Владеть	Построением воспроизводимого конвейера настройки с логированием и сохранением лучшей модели. Использованием методов прогресса (pruning) в Optuna для ускорения перебора. Оценкой устойчивости полученных гиперпараметров к изменениям данных (например, повторная настройка на разных сплитах). Базовой визуализацией процесса оптимизации (learning curves, importance of parameters).
MF-3.1	<i>Применяет методы оптимизации для разработки и исследования обучающих алгоритмов МО (градиентные методы, Adam, RMSprop, оптимизация гиперпараметров)</i>
Знать	Математическую суть градиентного спуска: вычисление градиента функции потерь по параметрам, правило обновления. Отличия SGD от Adam и RMSprop: учёт моментов (momentum), адаптивная скорость обучения для каждого параметра, коррекция смещения (bias correction) в Adam. Методы оптимизации гиперпараметров: grid search, random search, ручной подбор, базовые приёмы (log-шкала для learning rate,

Уметь	уменьшение LR при плато).
	Самостоятельно реализовать SGD с нуля (numpy) для линейной модели для демонстрации понимания.
Владеть	Выбрать оптимизатор (SGD с momentum, Adam, RMSprop) в зависимости от задачи (размер данных, разреженность градиентов, необходимость быстрой сходимости).
	Провести grid search или random search с помощью библиотек (scikit-learn GridSearchCV, Optuna, Hyperopt) для подбора learning rate, размера батча, момента.
Владеть	Настроить расписание скорости обучения (learning rate schedule: step decay, exponential, cosine annealing).
	Навыком сравнения оптимизаторов на одном наборе данных: графики сходимости, итоговая метрика, время сходимости.
Владеть	Приёмами диагностики проблем оптимизации (плохой локальный минимум, осцилляции, застревание) и коррекции (изменение learning rate, добавление момента, сброс оптимизатора).
	Инструментами для автоматического логирования гиперпараметров (TensorBoard, MLflow, Weights & Biases).
Владеть	Методикой выбора критерия остановки (early stopping) с контролем валидационной метрики.
	DL-1 П Способен применять и (или) разрабатывать архитектуры глубоких нейронных сетей
DL-1.1	<i>Способен объяснять и применять математические основы нейронных сетей, включая расчет градиентов, методы оптимизации и алгоритм об-ратного распространения ошибки (backpropagation), для эффективного обучения моделей.</i>
Знать	Подробности алгоритма backpropagation (формулы для вычисления градиента).
	Понятие регуляризации и её роль в обучении нейронных сетей.
Знать	Методы оптимизации (SGD, Adam, RMSProp).
	Типичные архитектуры нейронных сетей (полносвязная сеть, сверточная сеть).
Уметь	Проводить полный цикл расчетов backpropagation вручную для многослойных нейронных сетей.
	Применять различные методы оптимизации в реальных проектах машинного обучения.
Уметь	Объяснять преимущества и недостатки различных методов оптимизации.
	Разрабатывать небольшие нейронные сети самостоятельно с применением оптимизационных методов.
Владеть	Техническими аспектами реализации основных видов нейронных сетей.
	Практическими методами оценки качества полученных моделей.
Владеть	DL-1.2 Способен реализовывать неглубокие нейронные сети (перцептроны, MLP), выбирать количество и размер слоёв, подходящие функции активации и функции потерь для решения задач классификации и регрессии
	<i>Знать</i> Связь между сложностью задачи и архитектурой сети (глубина, ширина). Сильные и слабые стороны разных функций активации. Причины переобучения и методы регуляризации (Dropout, L2).
Уметь	Осознанно выбирать количество слоёв, нейронов, функции активации и потерь. Анализировать графики обучения для выявления переобучения. Проводить эксперименты для подбора гиперпараметров.
	<i>Владеть</i> Полным циклом разработки модели: от данных до валидации.

	Навыком итеративного улучшения модели на основе результатов валидации.
DL-1.3 Способен применять современные архитектуры глубоких сетей для решения различных задач, понимая их внутреннюю структуру и особенности обучения.	
Знать	Внутреннюю структуру и особенности обучения основных архитектур (как работает сверточный блок, механизм внимания, LSTM). Их сильные и слабые стороны, компромиссы (точность/скорость).
Уметь	Выбирать оптимальную архитектуру для новой задачи. Модифицировать готовые сети (например, заменить головной классификатор). Готовить данные и настраивать процесс обучения с учетом особенностей архитектуры (длина последовательности для RNN).
Владеть	Навыком осмысленного тонкого настраивания и оценки нескольких архитектур для сравнения. Умением диагностировать типичные проблемы обучения (взрывающиеся градиенты в RNN).
О-3 Б Способен применять и (или) разрабатывать интеллектуальные методы оптимизации	
О-3.2 <i>Обосновывает способы и варианты применения интеллектуальных методов в задачах оптимизации</i>	
Знать	Типовые подходы и методики интеллектуальной оптимизации (нейронные сети, эвристики), области их применимости.
Уметь	Самостоятельно выбирать подходящий метод оптимизации для конкретной ситуации, настраивать параметры метода, оценивать качество полученных решений.
Владеть	Навыком разработки простейших моделей оптимизации с использованием специализированных инструментов и библиотек.
О-3.3 <i>Эффективно применяет интеллектуальные методы оптимизации для обеспечения функциональных характеристик систем ИИ</i>	
Знать	Классические метаэвристики: генетический алгоритм (ГА), эволюционные стратегии, метод роя частиц (PSO), имитация отжига. Критерии сравнения методов оптимизации (скорость сходимости, устойчивость к локальным оптимумам, масштабируемость). Понятие многоцелевой оптимизации (Pareto-фронт) и её роль для баланса конфликтующих характеристик (например, точность vs быстродействие). Основы работы библиотек оптимизации: Optuna, DEAP, Hyperopt, pyswarm.
Уметь	Самостоятельно реализовывать генетический алгоритм или PSO для отбора признаков или настройки гиперпараметров модели ИИ. Настраивать параметры самого оптимизатора (размер популяции, скорость мутации, коэффициенты PSO) для достижения сходимости. Применять штрафные функции для учёта ограничений (например, время вывода < 50 мс). Сравнивать две метаэвристики на одной задаче и выбирать лучшую по заданному критерию. Обеспечивать воспроизводимость результатов (фиксация random seed, логирование).
Владеть	Построением полноценного пайплайна оптимизации: целевая функция — поиск — перекрестная проверка — выбор лучшего решения. Работой с Optuna для настройки гиперпараметров нейронной сети в

	Keras/PyTorch.
	Применением многоцелевой оптимизации (например, Optuna с мониторингом двух метрик).
	Анализом компромиссов между функциональными характеристиками с помощью графиков Pareto front.
FC-1	Способен проводить фронтальные исследования в области архитектур, алгоритмов МО, оптимизации и математики
FC-1.1	Разрабатывает фундаментальные основы и новые алгоритмы машинного обучения
Знать	Глубокую математическую основу алгоритмов МО (теория оптимизации, теория вероятностей, линейная алгебра). Сильные и слабые стороны различных подходов, современные архитектуры нейронных сетей.
Уметь	Модифицировать существующие алгоритмы для улучшения их характеристик (скорость, точность, устойчивость). Предлагать новые комбинации методов для решения специфических прикладных задач. Формализовать задачи и ставить корректные эксперименты.
Владеть	Навыками проведения независимых исследований, включая постановку задачи, эксперимент и анализ результатов. Навыками работы с фреймворками глубокого обучения (PyTorch, TensorFlow).

Соответствие **продвинутому уровню** освоения компетенций планируемым результатам обучения и критериям их оценивания (оценка: **отлично /зачтено**):

MF-3	Способен применять современные методы оптимизации для обучения моделей машинного обучения настройки гиперпараметров и решения задач ИИ
MF-3.2	<i>Применяет методы оптимизации для настройки гиперпараметров моделей машинного обучения, включая использование методов поиска (grid search, random search) и байесовской оптимизации</i>
Знать	Математические основы байесовской оптимизации: GP (гауссовские процессы), TPE (Tree-structured Parzen Estimator), EI (Expected Improvement). Стратегии многоцелевой и многоусловной оптимизации (условные гиперпараметры, несколько метрик). Распределённые и параллельные подходы к настройке гиперпараметров (Ray Tune, Optuna с DB backend). Продвинутые методы: BOHB (Bayesian Optimization + HyperBand), мультифидельность (настройка на подвыборках данных, уменьшенных эпохах).
Уметь	Проектировать сложное пространство поиска с зависимыми гиперпараметрами (например, параметры для dropout только если есть dropout layer). Интегрировать настройку гиперпараметров в CI/CD/ML пайплайны (например, с использованием MLflow, Kubeflow). Применять мультифидельную оптимизацию для ускорения настройки глубоких моделей или больших данных. Адаптировать методы оптимизации под нестационарные целевые функции (например, меняющиеся со временем).
Владеть	Разработкой и внедрением кастомных стратегий оптимизации (комбинация байесовской оптимизации с экспертными правилами). Диагностикой и устранением проблем сходимости оптимизации

	(застывание в локальных оптимумах, неинформативные параметры). Сравнительным бенчмаркингом разных методов оптимизации под конкретную задачу (затраты времени против выигрыша в метрике). Обучением команды выбору стратегии настройки и автоматизацией этого процесса для повторяющихся задач.
MF-3.1	Применяет методы оптимизации для разработки и исследования обучающих алгоритмов МО (градиентные методы, Adam, RMSprop, оптимизация гиперпараметров)
Знать	Теоретические основы адаптивных методов: экспоненциально взвешенное скользящее среднее градиентов и квадратов градиентов, роль параметров β_1, β_2, ϵ в Adam, влияние коррекции смещения на первых итерациях. Продвинутое методы оптимизации: AdamW, NAdam, RAdam, Lookahead, Ranger. Методы многокрасочной оптимизации гиперпараметров: байесовская оптимизация (BO), последовательные методы (Hyperband, BOHB), дифференцируемая оптимизация гиперпараметров (DRO), популяционные методы (PBT). Учёт вычислительных ограничений при оптимизации (асимптотическая сходимость, trade-off время-качество).
Уметь	Реализовать кастомный оптимизатор (например, Adam с изменёнными правилами обновления) в фреймворке для исследования его поведения на специфических функциях потерь. Настроить распределённую оптимизацию гиперпараметров с использованием кластера (Optuna с базой данных, Ray Tune, Ax) для большого пространства поиска. Применять методы early stopping с прогнозированием (Median Stopping Rule, ASHA) для ускорения перебора гиперпараметров. Оптимизировать сложные составные функции потерь с несколькими компонентами (балансировка метрик) через настройку весов или многоцелевой поиск (MOBO).
Владеть	Экспертным подходом к выбору стратегии оптимизации в зависимости от бюджета (время, вычислительные ресурсы) и сложности модели. Навыком создания воспроизводимых пайплайнов оптимизации гиперпараметров с версионированием, логированием и последующим анализом. Методологией бенчмаркинга оптимизаторов и методов подбора гиперпараметров для научных исследований. Способностью коучинга коллег по продвинутым техникам оптимизации и внедрения стандартов в команде (репозитории конфигураций, чек-листы).
DL-1 П Способен применять и (или) разрабатывать архитектуры глубоких нейронных сетей	
DL-1.1	<i>Способен объяснять и применять математические основы нейронных сетей, включая расчет градиентов, методы оптимизации и алгоритм обратного распространения ошибки (backpropagation), для эффективного обучения моделей.</i>
Знать	Принципы современных архитектур глубоких нейронных сетей (ResNet, DenseNet, Transformer).

	Сложные типы функций потерь и регуляризацию (L1/L2, dropout, batch normalization).
	Современные подходы к обучению больших моделей (transfer learning, fine-tuning).
	Алгоритмы ускоренного обучения и параллельные вычисления (GPU/TPU).
Уметь	Реализовывать сложные нейронные сети и настраивать процесс обучения вручную.
	Оценивать качество моделей с точки зрения эффективности и устойчивости.
	Создавать собственные модели с нуля и эффективно обучать их на больших объемах данных.
	Оптимизировать обучение нейронных сетей путем выбора правильных гиперпараметров и методов оптимизации.
Владеть	Высокоуровневыми инструментами глубокого обучения (TensorFlow, PyTorch, Keras).
	Современными методиками анализа производительности и масштабирования нейронных сетей.
DL-1.2 Способен реализовывать неглубокие нейронные сети (перцептроны, MLP), выбирать количество и размер слоёв, подходящие функции активации и функции потерь для решения задач классификации и регрессии	
Знать	Нюансы работы оптимизаторов (Adam, SGD с моментом). Передовые методы инициализации и регуляризации. Влияние различных техник предобработки данных на сходимость обучения.
Уметь	Проектировать сложные MLP-архитектуры под специфичные данные. Использовать продвинутые техники (например, callback'и для планирования скорости обучения) для тонкой настройки. Глубоко интерпретировать ошибки модели для направленного улучшения.
Владеть	Навыком создания высокоточных и эффективных моделей для нестандартных задач. Экспертизой для выбора оптимальной и надежной конфигурации сети перед внедрением.
DL-1.3 Способен применять современные архитектуры глубоких сетей для решения различных задач, понимая их внутреннюю структуру и особенности обучения.	
Знать	Детали реализации, нюансы обучения (например, различные схемы позиционного кодирования в Transformer) и state-of-the-art модификации основных архитектур.
Уметь	Комбинировать компоненты разных архитектур (например, использовать CNN как энкодер для RNN). Адаптировать структуру сети под специфические требования (кастомные слои, ограничения). Глубоко настраивать гиперпараметры обучения для конкретной архитектуры.
Владеть	Навыком проектирования и реализации эффективных гибридных решений для сложных, нестандартных задач. Экспертизой для выбора и оптимизации всей цепочки обучения под выбранную архитектуру.
О-3 Б Способен применять и (или) разрабатывать интеллектуальные методы оптимизации	
О-3.2 <i>Обосновывает способы и варианты применения интеллектуальных методов в задачах оптимизации</i>	
Знать	Современные достижения и тенденции развития методов интеллектуальной оптимизации, ограничения существующих подходов, возможности их комбинирования.
Уметь	Разрабатывать новые методы оптимизации либо существенно улучшать существующие, адаптировать стандартные модели к не-

	стандартным условиям, анализировать эффективность разработанных методик.
Владеть	Высокими компетенциями в проектировании сложных систем и алгоритмов интеллектуальной оптимизации, включая умение создавать уникальные решения, эффективные в условиях неопределенности и многомерности решаемых задач.
О-3.3	<i>Эффективно применяет интеллектуальные методы оптимизации для обеспечения функциональных характеристик систем ИИ</i>
Знать	Современные интеллектуальные методы: дифференцируемая эволюция, коэволюционные алгоритмы, роевые методы с памятью, адаптивные и самонастраивающиеся эвристики. Теоретические основы сходимости метаэвристик (марковские цепи, теория оптимального поиска). Продвинутые стратегии многоцелевой оптимизации (NSGA-III, MOEA/D, индикаторные алгоритмы). Методы ускорения оптимизации: мультифидельность (многоуровневая оценка), параллельные и распределённые вычисления, использование суррогатных моделей. Интеграция интеллектуальной оптимизации в MLOps и жизненный цикл ИИ-систем (непрерывная оптимизация характеристик).
Уметь	Проектировать сложные целевые функции с несколькими ограничениями и динамическими условиями (например, оптимизация модели под разные аппаратные платформы). Разрабатывать кастомные операторы мутации и скрещивания для специфических пространств поиска (структурированные, графовые, условные гиперпараметры). Применять мета-оптимизацию (настройка параметров самого оптимизатора) для адаптации под класс задач. Внедрять оптимизацию в CI/CD/CT-пайплайн с автоматическим выбором метода в зависимости от характеристики (например, если время обучения > порога — переключиться на мультифидельный поиск). Оценивать устойчивость полученного решения к изменениям данных или окружения.
Владеть	Разработкой и внедрением гибридных алгоритмов (например, байесовская оптимизация + эволюционный поиск) для обеспечения заданных характеристик. Диагностикой проблем масштабирования (размерность > 100 параметров) и применением методов уменьшения размерности/аппроксимации. Сравнительным бенчмаркингом с публичными наборами задач и обоснованием выбора метода в научно-технической документации. Обучением команды выбору и настройке интеллектуальных методов под конкретные функциональные требования системы ИИ. Применением распределённых фреймворков (Ray Tune, Optuna с RDB backend, Spark + evolutionary) для оптимизации крупномасштабных ИИ-систем.
FC-1	Способен проводить фронтирные исследования в области архитектур, алгоритмов МО, оптимизации и математики
FC-1.1	Разрабатывает фундаментальные основы и новые алгоритмы машинного обучения

Знать	Передовой край науки в области машинного обучения (последние исследования, тренды, нерешённые проблемы). Методологию создания принципиально новых алгоритмических парадигм.
Уметь	Разрабатывать принципиально новые алгоритмы и теоретические основы машинного обучения. Доказывать теоретические свойства алгоритмов (сходимость, сложность, устойчивость). Формировать новые исследовательские направления.
Владеть	Навыком публикации результатов в ведущих научных журналах и выступления на международных конференциях (NeurIPS, ICML, ICLR). Навыком руководства научными группами и генерации инновационных идей.

Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы

Практические кейсы по тематике лабораторных работ

Лабораторная работа №1: Классические методы одномерной оптимизации

Кейс: «Поиск оптимальной цены для максимизации прибыли»

Постановка задачи: Исследовать унимодальную функцию прибыли от цены и найти оптимальную цену с заданной точностью.

Требования:

1. Реализовать алгоритм Свенна для поиска интервала неопределенности
2. Применить методы дихотомии и золотого сечения для точного поиска экстремума
3. Сравнить скорость сходимости методов по количеству итераций
4. Проверить выполнение достаточных условий экстремума

Инструменты: Python, NumPy, Matplotlib, Jupyter Notebook

Лабораторная работа №2: Градиентные методы многомерной оптимизации

Кейс: «Оптимизация портфеля инвестиций»

Постановка задачи: Минимизировать риск портфеля при заданной доходности using gradient methods.

Требования:

1. Реализовать метод наискорейшего спуска
2. Реализовать метод сопряженных градиентов Флетчера-Ривса
3. Исследовать влияние выбора шага на сходимость
4. Визуализировать траекторию движения к optimum в пространстве переменных

Инструменты: Python, NumPy, Matplotlib, Pandas

Лабораторная работа №3: Методы второго порядка

Кейс: «Калибровка параметров финансовой модели»

Постановка задачи: Найти параметры модели, минимизирующие среднеквадратичную ошибку прогноза.

Требования:

1. Реализовать метод Ньютона-Рафсона
2. Реализовать метод Дэвидона-Флетчера-Пауэлла (DFP)
3. Сравнить сходимость с градиентными методами
4. Исследовать влияние обусловленности гессиана на скорость сходимости

Инструменты: Python, NumPy, SciPy, Jupyter Notebook

Лабораторная работа №4: Условная оптимизация

Кейс: «Оптимизация производственных мощностей с ограничениями»

Постановка задачи: Максимизировать выпуск продукции при ограничениях на ресурсы.

Требования:

1. Реализовать метод множителей Лагранжа
2. Реализовать метод проекции градиента
3. Сравнить эффективность методов для разных типов ограничений
4. Проанализировать выполнение условий Куна-Таккера

Инструменты: Python, NumPy, CVXPY, Jupyter Notebook

Лабораторная работа №5: Методы штрафных функций

Кейс: «Оптимизация формы конструкций с прочностными ограничениями»

Постановка задачи: Найти оптимальную геометрию конструкции, удовлетворяющую constraints.

Требования:

1. Реализовать метод внутренних штрафных функций (барьеров)
2. Реализовать метод внешних штрафных функций
3. Исследовать влияние параметра штрафа на сходимость
4. Сравнить оба подхода по точности и вычислительной сложности

Инструменты: Python, NumPy, Matplotlib

Лабораторная работа №6: Генетические алгоритмы

Кейс: «Оптимизация маршрута доставки товаров»

Постановка задачи: Решить задачу коммивояжера для 10 городов using genetic algorithm.

Требования:

1. Разработать схему кодирования решения (хромосому)
2. Реализовать операторы кроссинговера и мутации
3. Исследовать влияние различных стратегий отбора
4. Провести параметрическую настройку алгоритма

Инструменты: Python, DEAP library, Matplotlib

Лабораторная работа №7: Методы Монте-Карло

Кейс: «Оценка риска инвестиционного портфеля»

Постановка задачи: Оценить Value-at-Risk портфеля методом Монте-Карло.

Требования:

1. Реализовать метод Монте-Карло для численного интегрирования
2. Применить метод для оценки вероятности больших потерь
3. Исследовать зависимость точности от количества испытаний
4. Сравнить с аналитическими методами оценки риска

Инструменты: Python, NumPy, Pandas, Matplotlib

Лабораторная работа №8: Нейросетевые методы оптимизации

Кейс: «Прогнозирование и оптимизация энергопотребления»

Постановка задачи: Построить нейросеть для прогноза и оптимизации энергопотребления.

Требования:

1. Построить нейросеть для прогнозирования временного ряда
2. Использовать сеть как целевую функцию в оптимизационной задаче
3. Решить задачу оптимизации комбинированным методом
4. Сравнить с традиционными подходами оптимизации

Инструменты: Python, TensorFlow/PyTorch, Scikit-learn

Пример лабораторной работы

Лабораторная работа №2: «Градиентные методы многомерной оптимизации»

Цель: найти минимум заданной функции аналитически и численно методами наискорейшего градиентного спуска и Флетчера-Ривса и сравнить их. Выбор средств реализации остается за обучаемым.

Исходные данные:

- Начальное приближение $x^0 = (x_1, x_2)$.
- Заданные значения точности $\varepsilon_1, \varepsilon_2 > 0$.
- Предельное число итераций M .
- Функция $f(x) = x_1^2 + 8x_2^2 - x_1x_2 + x_1$

Необходимо:

1. Найти количество итераций k , необходимых для достижения заданной точности.
2. Найти точку минимума x^* .
3. Вычислить значение функции в точке минимума $f(x^*)$.
4. Определить, по какому условию остановки алгоритм завершил работу.
5. Построить график.

Алгоритм метода наискорейшего градиентного спуска.

Шаг 1: задается $x^0, \varepsilon_1, \varepsilon_2, M, \text{grad}(f(x)), H(x)$.

Шаг 2: $k = 0$

Шаг 3: находим $\text{grad}(f(x^k))$

Шаг 4: если $\|\text{grad}(f(x^k))\| < \varepsilon_1 \Rightarrow x^* = x^k$, иначе шаг 5

Шаг 5: если $k \geq M$, то $x^* = x^k$, иначе шаг 6

Шаг 6: $t_k = \varphi(t_k) = f(x^k - t_k * \text{grad}(f(x^k))) \rightarrow \min$

Шаг 7: $x^{k+1} = x^k - t_k * \text{grad}(f(x^k))$

Шаг 8: если $\|x^{k+1} - x^k\| < \varepsilon_2$ и $|f(x^{k+1}) - f(x^k)| < \varepsilon_2$ для текущего k и $k-1 \Rightarrow x^* = x^{k+1}$, иначе $k = k+1$ и шаг 3

Алгоритм метода Флетчера-Ривса.

Шаг 1: задается $x^0, \varepsilon_1, \varepsilon_2, M, \text{grad}(f(x)), H(x)$.

Шаг 2: $k = 0$

Шаг 3: находим $\text{grad}(f(x^k))$

Шаг 4: если $\|\text{grad}(f(x^k))\| < \varepsilon_1 \Rightarrow x^* = x^k$, иначе шаг 5

Шаг 5: если $k \geq M$, то $x^* = x^k$, иначе шаг 6

Шаг 6: $d^0 = -\|\text{grad}(f(x^k))\|$

Шаг 7: $\beta_{k-1} = \frac{\|\text{grad}(f(x^k))\|^2}{\|\text{grad}(f(x^{k-1}))\|^2}$

Шаг 8: $d^k = -\text{grad}(f(x^k)) + \beta_{k-1} * d_{k-1}$

Шаг 9: $t_k = \varphi(t_k) = f(x^k + t_k * d_k) \rightarrow \min$

Шаг 10: $x^{k+1} = x^k + t_k * d_k$

Шаг 11: если $\|x^{k+1} - x^k\| < \varepsilon_2$ и $|f(x^{k+1}) - f(x^k)| < \varepsilon_2$ для текущего k и $k-1 \Rightarrow x^* = x^{k+1}$, иначе $k = k+1$ и шаг 3

Критерии оценки:

Отлично: Полное выполнение всех заданий, качественный код и отчет.

Хорошо: Незначительные недочеты в коде или отчете.

Удовлетворительно: Выполнены базовые задания, но с ошибками.

Неудовлетворительно: Критические ошибки или невыполнение работы.

Проверяемые компетенции комплексом практических заданий: MF-3.1; MF-3.2; FC-1.1; O-3.2; O-3.3; DL-1.1; DL-1.2; DL-1.3

Тестовые вопросы по дисциплине

В1

Вставьте пропущенные слова:

Точка x_0 будет производить _____ отрезка, если отношение длины всего отрезка к длине большей его части равно отношению длин _____ части отрезка к _____ его части.

В2

Вставьте пропущенные слова:

Алгоритм метода _____ совпадает с алгоритмом метода Фибоначчи, исключая то, что в качестве внутренних точек будем выбирать точки _____.

В3

Вставьте пропущенные слова:

Метод золотого сечения сводится к последовательному уменьшению отрезка локализации на основании анализов значений функции в двух точках. Причем, в отличие от метода дихотомии, на первой итерации требуется вычисление _____ точек, а на каждой последующей – только _____.

В4

Вставьте пропущенные слова:

В методе _____, также, как и в методе Фибоначчи, количество вычислений сокращается _____, и каждая новая итерация потребует расчетов только одного нового значения функции.

В5

Вставьте пропущенные слова:

Длина отрезка локализации в методе _____ больше, чем в методе Фибоначчи на 15%.

В6

Сопоставьте метод с его свойством:

- 1) Численно неустойчив при неправильном задании числа N (количество вычислений функции)
- 2) Численно устойчив, если после некоторого количества шагов произвести пересчет приближения по соответствующей формуле

- А) Метод Фибоначчи
- Б) Метод золотого сечения
- В) Метод дихотомии

В7

Сопоставьте метод с его свойством:

- 1) Необходимость вычисления на каждой итерации двух значений функции
- 2) Длина отрезка локализации больше, чем в методе Фибоначчи

- А) Метод золотого сечения
- Б) Метод дихотомии
- В) Метод Фибоначчи

В8

Сопоставьте метод с его свойством:

- 1) Наиболее прост в реализации (по сравнению с остальными методами одномерной оптимизации)
- 2) При одном и том же количестве вычислений дает наименьший отрезок локализации

- А) Метод Фибоначчи
- Б) Метод дихотомии
- В) Метод золотого сечения

В9

Сопоставьте метод с его свойством:

- 1) Всегда численно устойчив

- 2) В расчетных формулах метода присутствует множитель $\frac{3 - \sqrt{5}}{2}$

$$\frac{3 - \sqrt{5}}{2}$$

- А) Метод дихотомии
- Б) Метод золотого сечения
- В) Метод Фибоначчи

B10

Сопоставьте метод с его свойством:

- 1) Основан на вычислении чисел Фибоначчи
 - 2) Предполагает наибольшее количество итераций относительно остальных методов одномерной оптимизации
- А) Метод дихотомии
 Б) Метод Фибоначчи
 В) Метод золотого сечения

B11

Выберите один вариант ответа:

Комбинированное применение данного метода и метода последовательной параболической интерполяции используется для минимизации функции одной переменной в пакетах прикладных программ вроде MATLAB.

- А) Метод Фибоначчи Б) Метод золотого сечения В) Метод дихотомии

B12

Выберите один вариант ответа:

Критерий останова методов одномерной оптимизации:

- А) Длина отрезка не превосходит заданного числа эпсилон, умноженного на 2
 Б) Длина отрезка не превосходит заданного числа эпсилон
 В) Длина отрезка не превосходит заданного числа эпсилон, поделенного на 2

B13

Выберите один вариант ответа:

Данной сходимостью

$$R(N) = \left(1 - \frac{3 - \sqrt{5}}{2}\right)^N$$

$$R(N) = \left(1 - \frac{3 - \sqrt{5}}{2}\right)^N$$

обладает:

- А) Метод дихотомии Б) Метод Фибоначчи В) Метод золотого сечения

B14

Выберите один вариант ответа:

Данная формула

$$\frac{b_0 - a_0}{b_0 - x_0} = \frac{b_0 - x_0}{x_0 - a_0}$$

$$\frac{b_0 - a_0}{b_0 - x_0} = \frac{b_0 - x_0}{x_0 - a_0}$$

связана с:

- А) Методом золотого сечения Б) Методом Фибоначчи В) Методом дихотомии

B15

Выберите один вариант ответа:

Один из методов одномерной оптимизации будет численно устойчивым, если после некоторого количества шагов производить пересчет приближения по следующей формуле:

А)

$$x_k = a_k + (3 - \sqrt{5})(b_k - a_k)$$

$$x_k = a_k + (3 - \sqrt{5})(b_k - a_k)$$

Б)

$$x_k = a_k + (3 + \sqrt{5})(b_k - a_k)$$

$$x_k = a_k + (3 + \sqrt{5})(b_k - a_k)$$

В)

$$x_k = a_k + \frac{3 - \sqrt{5}}{2}(b_k - a_k)$$

$$x_k = a_k + \frac{3 - \sqrt{5}}{2}(b_k - a_k)$$

Время выполнения: 20-30 минут.

Проверяемые компетенции комплексом практических заданий: FC-1.1; О-3.2; О-3.3; DL-1.1; DL-1.2; DL-1.3

Зачетно-экзаменационные материалы для промежуточной аттестации (экзамен)

Вопросы для подготовки к экзамену

1. Общая классификация методов оптимизации.
2. Общая постановка задачи безусловной оптимизации. Теорема Вейерштрасса.
3. Оптимизация функции одной переменной. Теорема (Ферма). Геометрический смысл необходимого условия экстремума.
4. Теорема о необходимом и достаточном условии. Классический алгоритм.
5. Общая схема отыскания экстремума. Теорема (второй достаточный признак экстремума).
6. Унимодальные функции. Свойства унимодальных функций (самостоятельно доказать св-ва 2 и 3).
7. Алгоритм Свенна.
8. Метод дихотомии. Алгоритм. Сходимость.
9. Метод Фибоначчи. Алгоритм. Сходимость.
10. Метод золотого сечения (самостоятельно доказать св-ва).
11. Сравнительный анализ методов первого порядка (преимущества, недостатки).
12. Многомерная оптимизация функций многих переменных. Аксиомы нормы. Неравенство Коши-Буняковского.
13. Метод сопряжённых градиентов.
14. Метод Бroyдена.
15. Метод Дэвидона-Флетчера-Пауэлла
16. Методы переменной метрики.
17. Норма матрицы. Квадратичные формы. Лемма о положительно определенных квадратичных формах.
18. Критерий Сильвестра.
19. Необходимое и достаточное условие неотрицательной определенности квадратичной формы.
20. Линии уровня. Примеры.
21. Градиент функции. Стационарная точка. Необходимое условие экстремума 1 порядка.
22. Теоремы о необходимом и достаточном условии безусловного минимума функции многих переменных.
23. Общая схема отыскания безусловного экстремума функций многих переменных. Примеры.
24. Численная многомерная безусловная оптимизация. Примеры.

25. Леммы (1) о направлении спуска (безусловная оптимизация). Геометрическая интерпретация.
26. Леммы (2) о направлениях спуска (безусловная оптимизация). Способы выбора матриц .
27. Способы выбора матриц . Алгоритмы определения шага .
28. Алгоритмы определения шага .
29. Метод градиентного спуска. Теоремы о сходимости.
30. Алгоритм метода наискорейшего спуска. Геометрическая интерпретация.
31. Метод сопряженных градиентов Флетчера–Ривса. Геометрическая интерпретация.
32. Методы Ньютона и Ньютона–Равсона.
33. Постановка задачи нелинейного программирования. Определения. Примеры.
34. Классификация задач нелинейного программирования.
35. Лемма 1 (необ. усл. экстремума) с замечаниями. Лемма 2 (дост. усл.). Примеры.
36. Алгоритм метода множителей Лагранжа.
37. Леммы 3 и 4 о возможных направлениях (условная оптимизация).
38. Теорема 1 о необходимом условии условного минимума.
39. Теорема Фаркаша. Теорема 2 о необходимом условии условного минимума.
40. Численные методы поиска условного экстремума
41. Метод возможных направлений.
42. Метод проекций градиента.
43. Определение внутренних функций. Метод внутренних штрафных функций
44. Определение внешних функций. Метод внешних штрафных функций
45. Сравнительная оценка и общие свойства традиционных методов штрафных функций.
46. Что такое генетический алгоритм?
47. Опишите структуру и принцип работы генетического алгоритма.
48. Какие компоненты входят в стандартный генетический алгоритм?
49. Какова роль мутации и кроссинговера в генетическом алгоритме?
50. Отбор особей в генетическом алгоритме.
51. Приведите пример практического применения генетических алгоритмов.
52. Объясните понятие «хромосома», «аллель» и «фенотип» применительно к генетическим алгоритмам.
53. Перечислите преимущества и недостатки генетических алгоритмов перед классическими методами оптимизации.
54. Ограничения генетических алгоритмов.
55. Метод Монте-Карло.
56. Назовите основные области применения методов Монте-Карло.
57. Расскажите о применении метода Монте-Карло в численном интегрировании.
58. В чём состоит задача оценки вероятности события методом Монте-Карло.
59. Приведите пример задачи из физики, которую удобно решать методом Монте-Карло.
60. Какими преимуществами обладают методы Монте-Карло перед детерминированными методами.
61. Какой недостаток является основным ограничением для применения методов Монте-Карло.
62. Определение искусственной нейронной сети.
63. Типы нейронных сетей для решения оптимизационных задач.
64. Примеры оптимизационных задач, решаемых нейросетями.

Перечень компетенций (части компетенции), проверяемых оценочным средством MF-3.1; MF-3.2; FC-1.1; O-3.2; O-3.3; DL-1.1; DL-1.2; DL-1.3

4.2 Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

Методические рекомендации, определяющие процедуры оценивания на экзамене:

Процедура промежуточной аттестации проходит в соответствии с Положением о текущем контроле и промежуточной аттестации обучающихся ФГБОУ ВО «КубГУ».

Итоговой формой контроля сформированности компетенций у обучающихся по дисциплине является экзамен. Студенты обязаны сдать экзамен в соответствии с расписанием и учебным планом.

ФОС промежуточной аттестации состоит из вопросов к экзамену и результатов текущего контроля.

Экзамен по дисциплине преследует цель оценить работу студента за курс, получение теоретических знаний, их прочность, развитие творческого мышления, приобретение навыков самостоятельной работы, умение применять полученные знания для решения практических задач.

Форма проведения экзамена: устно.

Экзаменатору предоставляется право задавать студентам дополнительные вопросы по всей учебной программе дисциплины.

Результат сдачи экзамена заносится преподавателем в экзаменационную ведомость и зачетную книжку.

Оценивание уровня освоения дисциплины основывается на качестве выполнения студентом заданий текущего контроля и ответов на вопросы экзамена.

Критерии оценки:

1. Оценка ответов на экзаменационные вопросы (40% итоговой оценки)

Отлично (5)

Полные, развернутые ответы с демонстрацией глубокого понимания темы.

Использование примеров, формул, корректных терминов.

Умение анализировать и сравнивать методы.

% выполнения: 90–100% (допускаются незначительные неточности).

Хорошо (4)

Ответы содержат основные идеи, но без углубленного анализа.

Возможны небольшие ошибки в деталях или формулировках.

% выполнения: 75–89%.

Удовлетворительно (3)

Ответы поверхностные, с существенными пробелами.

Отсутствие примеров или некорректное применение терминов.

% выполнения: 60–74%.

Неудовлетворительно (2)

Отсутствие понимания ключевых концепций.

Грубые ошибки или неспособность ответить на большую часть вопросов.

% выполнения: <60%.

Критерии оценки лабораторных работ и промежуточного тестирования

Отлично: Полное выполнение всех шагов, проведение сравнительного анализа и ответы на все вопросы, прохождение теста $\geq 90\%$ правильных ответов.

Хорошо: Корректная работа алгоритма, но без его оптимизации, при ответах на вопросы получены незначительные замечания, прохождение теста $\geq 75\%$.

Удовлетворительно: Алгоритм работает с ошибками, устраняемыми в процессе приема задачи, при ответах на вопросы получен ряд замечаний, прохождение теста =>60%.

Неудовлетворительно: Невыполнение ключевых этапов.

Для допуска к экзамену необходимо выполнить **все лабораторные работы на минимум "удовлетворительно"**.

"Отлично" требует высоких результатов во всех лабораторных работах.

Методические рекомендации, определяющие процедуры оценивания лабораторных работ:

Процедура оценивания лабораторных работ проходит в соответствии с Положением о текущем контроле и промежуточной аттестации обучающихся ФГБОУ ВО «КубГУ».

По каждой лабораторной работе оформляется отчет. Отчеты сдаются на проверку руководителю в течение курса по мере их выполнения, и защищаются студентами в установленном порядке.

При защите отчета студенту могут быть заданы вопросы и дополнительные задания по сути лабораторной работы, в том числе из списка контрольных вопросов к данной лабораторной работе. При неудовлетворительной оценке знаний студента по теме данного отчета, студент возвращается к повторному изучению соответствующих материалов, после чего допускается к повторной защите. Неудовлетворительно выполненный отчет также возвращается на доработку.

Отчет должен содержать заголовок, тему лабораторной работы, цель, задание, индивидуальную тему, описание хода выполнения работы, необходимые прикладные материалы (схемы, макеты документов и т.п.), в соответствии с требованиями к содержанию, и выводы по работе.

Оценочные средства для инвалидов и лиц с ограниченными возможностями здоровья выбираются с учетом их индивидуальных психофизических особенностей.

– при необходимости инвалидам и лицам с ограниченными возможностями здоровья предоставляется дополнительное время для подготовки ответа на экзамене;

– при проведении процедуры оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья предусматривается использование технических средств, необходимых им в связи с их индивидуальными особенностями;

– при необходимости для обучающихся с ограниченными возможностями здоровья и инвалидов процедура оценивания результатов обучения по дисциплине может проводиться в несколько этапов.

Процедура оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья по дисциплине предусматривает предоставление информации в формах, адаптированных к ограничениям их здоровья и восприятия информации:

Для лиц с нарушениями зрения:

- в печатной форме увеличенным шрифтом,
- в форме электронного документа.

Для лиц с нарушениями слуха:

- в печатной форме,
- в форме электронного документа.

Для лиц с нарушениями опорно-двигательного аппарата:

- в печатной форме,
- в форме электронного документа.

Данный перечень может быть конкретизирован в зависимости от контингента обучающихся.

4.3. Методические указания по организации вычислительной инфраструктуры

Условия применения:

- Курс рассчитан на студентов 3-го года обучения.
- Наличие доступа к вычислительным ресурсам (GitLab, Google Colab или Yandex DataSphere, JupyterHub, Hugging Face).
- Разработаны лабораторные работы;
- Инфраструктура для приёма задач (gitlab, CI/CD) согласована с лабораторными работами.

Цели, задачи и ожидаемые результаты

Цели организации вычислительной инфраструктуры:

- дать начальное представление о работе в IT инфраструктуре (приучить пользоваться гитом, jupyter-ноутбуками).

Задачи преподавателя:

- Организация регистрации студентов в Google Colab и Yandex DataSphere
- Создание учетных записей студентов в gitlab вуза;
- Настройка GitLab Runner для автоматического тестирования кода.
- Разработка шаблонного репозитория для лабораторных работ с предустановленными зависимостями (PyTorch, Keras, TensorFlow, Hugging Face Transformers).
- Написание автотестов для проверки корректности выполнения заданий (например, fine-tuning моделей).
- Визуализация результатов тестирования через HTML-отчеты.
- Подготовка инструкций по работе с Git и облачными ресурсами.

Ожидаемые результаты студентов:

- начальное представление о работе в IT инфраструктуре (гит, нейминг).
- Навыки запуска и тестирования глубоких нейросетей в облачных средах.
- Понимание CI/CD-процессов в контексте разработки глубоких нейросетей.

Порядок реализации

Задача №1: Организация регистрации студентов в Google Colab и Yandex DataSphere

Задача №2: Создание учетных записей студентов в gitlab вуза

Задача №3: Настройка GitLab Runner:

Для автоматического тестирования кода используется Docker-образ с предустановленными библиотеками (PyTorch, Keras, TensorFlow, Hugging Face Transformers).

Для выполнения CI/CD пайплайна был настроен GitLab Runner на удалённой виртуальной машине с ОС Ubuntu 24.04.

Последовательность настройки включала следующие шаги:

- Настройка системы – установка необходимых компонентов, таких, как Docker.
- Установка GitLab Runner по официальной инструкции.
- Регистрация Runner для частного сервера GitLab.

Задача №4: Шаблонный репозиторий:

Включает:

.gitlab-ci.yml для CI/CD.

Скрипты для предобработки текста и обучения моделей.

Примеры кода для работы с BERT, GPT и другими архитектурами.

Задача №5: Автотесты:

Проверяют корректность fine-tuning моделей (например, ассигасу на тестовом датасете).

Задача №6: Визуализация результатов:

Генерация HTML-отчетов с результатами тестирования, включая метрики качества моделей.

Порядок проверки корректности:

- Наличие Git-репозитория у всех студентов.
- Шаблонный репозиторий с подключенными автотестами.

- Инструкция по работе с Git и CI/CD в формате README.md.
- Вся структура максимально адаптирована для копирования студентами и минимизации порога входа при выполнении лабораторных

4.4. Методические указания по организации лабораторных работ

Условия применения:

- Курс рассчитан на студентов 3-го года обучения.
- Наличие доступа к вычислительным ресурсам (GitLab) и к GPU/CPU (Kaggle, локальные серверы).
- Разработана инфраструктура для приёма задач (Gitlab, CI/CD) и согласована с лабораторными работами и настроена на всех студентов образовательной программы;
- Использование открытых датасетов и библиотек.

Цели, задачи и ожидаемые результаты

Цели организации лабораторных работ:

- Закрепление теоретических знаний о классических и интеллектуальных методах оптимизации на практике.
- Развитие навыков алгоритмического мышления и реализации оптимизационных алгоритмов для решения прикладных задач.
- Подготовка к решению реальных задач в таких областях, как финансы, логистика, машинное обучение и проектирование.

Задачи преподавателя:

- Обеспечить студентов структурированными лабораторными работами, охватывающими спектр методов от классических до современных интеллектуальных.
- Предоставить доступ к необходимым программным средствам и вычислительным ресурсам (Python, библиотеки для оптимизации, среды для визуализации).
- Организовать проверку и обратную связь по выполненным работам, акцентируя внимание на корректности реализации, анализе результатов и качестве выводов.

Ожидаемые результаты студентов:

- Умение применять и сравнивать различные классы оптимизационных алгоритмов (градиентные, методы второго порядка, эволюционные, swarm-интеллект) на практике.
- Владение библиотеками и фреймворками для научных вычислений и оптимизации (NumPy, SciPy, Matplotlib, DEAP, Scikit-learn).
- Опыт решения прикладных задач оптимизации из разных предметных областей (поиск экстремума функций, условная оптимизация, комбинаторные задачи).
- Навыки анализа поведения алгоритмов, включая сходимость, устойчивость и вычислительную сложность, а также визуализации процесса оптимизации.

Порядок реализации

Задача №1: Подготовка лабораторных работ (в соответствии с п. 2.3.3 РПД)

1) Определение тем:

1. Классические методы одномерной оптимизации
2. Градиентные методы многомерной безусловной оптимизации
3. Методы второго порядка (Ньютон, BFGS, DFP)
4. Методы условной оптимизации и теорема Куна-Таккера
5. Методы штрафных функций для задач с ограничениями
6. Генетические алгоритмы для комбинаторной оптимизации
7. Методы Монте-Карло для стохастической оптимизации и оценки рисков
8. Нейросетевые подходы к решению оптимизационных задач

Лабораторная работа №1: Классические методы одномерной оптимизации

Кейс: «Поиск оптимальной цены для максимизации прибыли»

Постановка задачи: Исследовать унимодальную функцию прибыли от цены и найти оптимальную цену с заданной точностью.

Требования:

5. Реализовать алгоритм Свенна для поиска интервала неопределенности
6. Применить методы дихотомии и золотого сечения для точного поиска экстремума
7. Сравнить скорость сходимости методов по количеству итераций
8. Проверить выполнение достаточных условий экстремума

Инструменты: Python, NumPy, Matplotlib, Jupyter Notebook

Лабораторная работа №2: Градиентные методы многомерной оптимизации

Кейс: «Оптимизация портфеля инвестиций»

Постановка задачи: Минимизировать риск портфеля при заданной доходности using gradient methods.

Требования:

5. Реализовать метод наискорейшего спуска
6. Реализовать метод сопряженных градиентов Флетчера-Ривса
7. Исследовать влияние выбора шага на сходимость
8. Визуализировать траекторию движения к optimum в пространстве переменных

Инструменты: Python, NumPy, Matplotlib, Pandas

Лабораторная работа №3: Методы второго порядка

Кейс: «Калибровка параметров финансовой модели»

Постановка задачи: Найти параметры модели, минимизирующие среднеквадратичную ошибку прогноза.

Требования:

5. Реализовать метод Ньютона-Рафсона
6. Реализовать метод Дэвидона-Флетчера-Пауэлла (DFP)
7. Сравнить сходимость с градиентными методами
8. Исследовать влияние обусловленности гессиана на скорость сходимости

Инструменты: Python, NumPy, SciPy, Jupyter Notebook

Лабораторная работа №4: Условная оптимизация

Кейс: «Оптимизация производственных мощностей с ограничениями»

Постановка задачи: Максимизировать выпуск продукции при ограничениях на ресурсы.

Требования:

5. Реализовать метод множителей Лагранжа
6. Реализовать метод проекции градиента
7. Сравнить эффективность методов для разных типов ограничений
8. Проанализировать выполнение условий Куна-Таккера

Инструменты: Python, NumPy, CVXPY, Jupyter Notebook

Лабораторная работа №5: Методы штрафных функций

Кейс: «Оптимизация формы конструкций с прочностными ограничениями»

Постановка задачи: Найти оптимальную геометрию конструкции, удовлетворяющую constraints.

Требования:

5. Реализовать метод внутренних штрафных функций (барьеров)
6. Реализовать метод внешних штрафных функций
7. Исследовать влияние параметра штрафа на сходимость

8. Сравнить оба подхода по точности и вычислительной сложности
Инструменты: Python, NumPy, Matplotlib

Лабораторная работа №6: Генетические алгоритмы

Кейс: «Оптимизация маршрута доставки товаров»

Постановка задачи: Решить задачу коммивояжера для 10 городов using genetic algorithm.

Требования:

5. Разработать схему кодирования решения (хромосому)
6. Реализовать операторы кроссинговера и мутации
7. Исследовать влияние различных стратегий отбора
8. Провести параметрическую настройку алгоритма

Инструменты: Python, DEAP library, Matplotlib

Лабораторная работа №7: Методы Монте-Карло

Кейс: «Оценка риска инвестиционного портфеля»

Постановка задачи: Оценить Value-at-Risk портфеля методом Монте-Карло.

Требования:

5. Реализовать метод Монте-Карло для численного интегрирования
6. Применить метод для оценки вероятности больших потерь
7. Исследовать зависимость точности от количества испытаний
8. Сравнить с аналитическими методами оценки риска

Инструменты: Python, NumPy, Pandas, Matplotlib

Лабораторная работа №8: Нейросетевые методы оптимизации

Кейс: «Прогнозирование и оптимизация энергопотребления»

Постановка задачи: Построить нейросеть для прогноза и оптимизации энергопотребления.

Требования:

5. Построить нейросеть для прогнозирования временного ряда
6. Использовать сеть как целевую функцию в оптимизационной задаче
7. Решить задачу оптимизации комбинированным методом
8. Сравнить с традиционными подходами оптимизации

Инструменты: Python, TensorFlow/PyTorch, Scikit-learn

Наименование Лабораторной работы	Содержание Лабораторной работы	Распределение часов
2	3	
Классические методы одномерной оптимизации	Сравнение методов Свенна, дихотомии и золотого сечения для поиска минимума унимодальной функции. Визуализация процесса сужения интервала неопределённости.	4 часа (4 часа ЛР)
Градиентные методы многомерной оптимизации	Реализация и сравнение методов наискорейшего спуска и сопряжённых градиентов. Анализ траектории движения к минимуму функции Розенброка. Проанализировать результаты и сделать выводы о производительностях методов.	6 часов (6 часов ЛР)
Методы второго порядка	Реализация и исследование методов Ньютона и BFGS. Сравнение скорости сходимости с градиентными методами на плохо обусловленных функциях	6 часов (6 часов ЛР)
Условная оптимизация	Решение задач с ограничениями методом множителей Лагранжа и проекции градиента. Проверка	2 часа (2 часа ЛР)

Наименование Лабораторной работы	Содержание Лабораторной работы	Распределение часов
2	3	
	условий Куна-Таккера.	
Методы штрафных функций	Реализация методов внешних штрафов и внутренних барьеров. Исследование влияния параметра штрафа на сходимость. Проанализировать результаты и сделать выводы о производительности методов.	2 часа (2 часа ЛР)
Генетические алгоритмы	Реализовать решение задачи коммивояжёра с реализацией операторов кроссовера, мутации и отбора. Проанализировать влияния параметров на результат. Оценить точность модели на тестовом наборе данных. Изучить результаты и сделать выводы о производительности модели.	3 часа (3 часа ЛР)
Методы Монте-Карло	Применение стохастического моделирования для численного интегрирования и поиска глобального минимума. Расчёт Value-at-Risk для портфеля. Проанализировать результаты и сделать выводы о производительности модели.	3 часа (3 часа ЛР)
Нейросетевые методы оптимизации	Использование нейросетей как моделей для оптимизации сложных функций. Решение простой задачи оптимального управления. Проанализировать результаты и сделать выводы о сходимости и точности.	10 часов (6 часов ЛР, 4 часа СР)

2) Разработка заданий:

- Пошаговые инструкции.
- Примеры кода.

Контрольные вопросы.

2. Пошаговые инструкции

Шаг 1: Установка библиотек

Шаг 2: Подготовка тестовых функций

Шаг 3: Реализация метода Ньютона

Шаг 4: Реализация метода BFGS

Шаг 5: Сравнительный анализ методов

3. Примеры кода на Python

python

1. Импорт библиотек

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.optimize import minimize
from autograd import grad, hessian
```

2. Подготовка тестовых функций

```
def rosenbrock(x):
    """Функция Розенброка - тестовая функция для оптимизации"""
    return sum(100.0*(x[1:] - x[:-1])**2.0)**2.0 + (1-x[:-1])**2.0

def quadratic(x):
```

```

        """Квадратичная функция для демонстрации"""
        return x[0]**2 + 2*x[1]**2 + x[0]*x[1]

```

3. Реализация метода Ньютона

```

def newton_method(f, x0, max_iter=100, tol=1e-6):
    """
    Реализация метода Ньютона для оптимизации
    """
    x = np.array(x0, dtype=float)
    f_grad = grad(f)
    f_hess = hessian(f)

    trajectory = [x.copy()]

    for i in range(max_iter):
        grad_val = f_grad(x)
        hess_val = f_hess(x)

        # Проверка положительной определенности гессиана
        if np.all(np.linalg.eigvals(hess_val) > 0):
            delta_x = -np.linalg.solve(hess_val, grad_val)
        else:
            # Регуляризация при необходимости
            delta_x = -np.linalg.solve(hess_val + 0.1*np.eye(len(x)), grad_val)

        x = x + delta_x
        trajectory.append(x.copy())

        if np.linalg.norm(grad_val) < tol:
            break

    return x, trajectory

```

4. Реализация BFGS метода

```

def bfgs_method(f, x0, max_iter=100, tol=1e-6):
    """
    Реализация BFGS метода с помощью scipy
    """
    result = minimize(f, x0, method='BFGS',
                      options={'maxiter': max_iter, 'gtol': tol})
    return result.x, result.nit

```

5. Сравнительный анализ

```

def compare_methods():

```

```

# Начальная точка
x0 = [-1.5, 2.0]

# Метод Ньютона
newton_result, newton_traj = newton_method(rosenbrock, x0)

# Метод BFGS
bfgs_result, bfgs_iters = bfgs_method(rosenbrock, x0)

print("Метод Ньютона:")
print(f"Результат: {newton_result}")
print(f"Количество итераций: {len(newton_traj)}")

print("\nМетод BFGS:")
print(f"Результат: {bfgs_result}")
print(f"Количество итераций: {bfgs_iters}")

# Визуализация траекторий
visualize_trajectory(newton_traj, "Метод Ньютона")

def visualize_trajectory(trajecctory, title):
    """Визуализация траектории оптимизации"""
    trajectory = np.array(trajecctory)

    plt.figure(figsize=(10, 6))
    plt.plot(trajectory[:, 0], trajectory[:, 1], 'bo-', markersize=4)
    plt.xlabel('x1')
    plt.ylabel('x2')
    plt.title(f'Траектория оптимизации - {title}')
    plt.grid(True)
    plt.show()

# Запуск сравнения
if __name__ == "__main__":
    compare_methods()

```

4. Контрольные вопросы

1. Теоретические:

- В чем основное отличие методов первого и второго порядка?
- Какие преимущества и недостатки у метода Ньютона?
- Почему метод BFGS называется квази-ньютоновским?

2. Практические:

- Реализуйте вычисление гессиана численными методами
- Модифицируйте код для работы с ограничениями типа неравенств
- Добавьте критерий остановки по изменению значения функции

3. Аналитические:

- В каких случаях метод Ньютона может не сходиться?
- Как влияет обусловленность гессиана на скорость сходимости?
- Почему BFGS часто предпочтительнее метода Ньютона на практике?

Критерии оценки

Отлично: Полная реализация обоих методов, анализ сходимости, сравнение эффективности на разных функциях

Хорошо: Корректная реализация методов, но без глубокого анализа

Удовлетворительно: Реализован один метод с незначительными ошибками

Неудовлетворительно: Критические ошибки в реализации или невыполнение работы

Дополнительные задания

1. Исследование сходимости:

- Протестируйте методы на функциях с разной обусловленностью
- Постройте графики сходимости для каждого метода

2. Модификация методов:

- Добавьте регуляризацию гессиана
- Реализуйте метод доверительных областей

3. Прикладная задача:

- Примените методы для оптимизации параметров машинного обучения
- Сравните с градиентными методами первого порядка

python

Дополнительное задание: Сравнение с градиентным спуском

```
def gradient_descent(f, x0, lr=0.01, max_iter=1000):
    f_grad = grad(f)
    x = np.array(x0)
    trajectory = [x.copy()]

    for i in range(max_iter):
        grad_val = f_grad(x)
        x = x - lr * grad_val
        trajectory.append(x.copy())

    return x, trajectory
```

Требования к отчету:

- Описание реализованных алгоритмов
- Сравнительные графики сходимости
- Анализ результатов для разных тестовых функций
- Выводы о применимости методов

Порядок проверки корректности

Чек-лист для проверки лабораторных работ:

1. Выполнение заданий:
 - Код запускается без ошибок.
 - Достигнуто целевое качество модели.
2. Качество кода:
 - Соблюдение PEP-8.
 - Наличие комментариев.
3. Отчет:

инструкция по работе с гитом с подробным описанием именования методов и коммитов;
Описание хода работы.
Анализ результатов.

4. Своевременность:

Работа сдана в установленный срок.

Критерии оценки:

Отлично: Полное выполнение всех заданий, качественный код и отчет.

Хорошо: Незначительные недочеты в коде или отчете.

Удовлетворительно: Выполнены базовые задания, но с ошибками.

Неудовлетворительно: Критические ошибки или невыполнение работы.

4.5. Методические указания по организации проектной деятельности студентов

Условия применения:

Курс рассчитан на студентов 3-го года обучения,

Общее время на проект – не более 16 часов на каждого студента.

Имеется доступ к кейсам индустриальных партнеров.

Цели, задачи и ожидаемые результаты

Цели организации вычислительной инфраструктуры:

дать начальное представление о реальных задачах, решаемых с помощью глубоких нейронных сетей и возникающих проблемах.

Задачи преподавателя:

- сбор кейсов индустриальных партнеров;
- сбор кейсов преподавателей практиков и лабораторий в вузе;
- формирование ТЗ на зачетный проект на основе кейсов;
- разработка системы учёта результатов проекта в итоговой оценке зачета

Ожидаемые результаты студентов:

начальное представление о реальных задачах, решаемых с помощью глубоких нейросетей и возникающих проблемах.

Порядок реализации

Задача №1: Сбор кейсов индустриальных партнеров

1. Оптимизация логистики строительных материалов (Construction Logistics

Optimization)

Описание: Необходимо разработать систему оптимизации доставки строительных материалов на multiple объектов строительной компании с учетом пробок, графика работы поставщиков и приоритетов объектов. Это позволит сократить простои рабочих и снизить логистические затраты.

Цель: Минимизировать суммарные затраты на логистику при соблюдении сроков поставки материалов для критических строительных операций.

Ожидаемые результаты:

- Модель оптимизации маршрутов доставки с временными окнами
- Алгоритм динамической перепланировки при изменении условий
- Расчет экономического эффекта от внедрения системы

2. Оптимизация штатного расписания строительных бригад (Workforce Scheduling Optimization)

Описание: Разработать систему оптимального распределения строительных бригад между объектами компании с учетом квалификации рабочих, сроков выполнения задач и трудового законодательства.

Цель: Обеспечить равномерную загрузку бригад и выполнение проектов в срок при минимизации сверхурочных работ.

Ожидаемые результаты:

- Алгоритм оптимального распределения человеческих ресурсов
- Модель прогнозирования сроков выполнения работ

- Инструмент для анализа "узких мест" в строительных процессах

Задача №2: Кейсы преподавателей-практиков и лабораторий в вузе

1. Оптимизация параметров бетонной смеси

Постановка задачи: Исследовательская лаборатория вуза изучает возможность оптимизации состава бетонной смеси для достижения максимальной прочности при минимальной стоимости. Необходимо применить методы нелинейной оптимизации для решения этой задачи.

Требования:

- Построить математическую модель зависимости прочности от состава смеси
- Применить методы условной оптимизации с ограничениями по стоимости
- Провести верификацию результатов экспериментальными данными

2. Оптимизация энергопотребления строительного объекта

Постановка задачи: Кафедра энергоэффективности разрабатывает систему оптимального управления энергопотреблением строящегося объекта. Необходимо создать модель минимизации энергозатрат при соблюдении технологических требований.

Требования:

- Реализовать генетический алгоритм для планирования энергопотребления
- Учесть суточные графики работы оборудования и тарифы на электроэнергию
- Обеспечить соблюдение технологических ограничений

Задача №3: Формирование ТЗ на зачетный проект на основе кейсов

1. Оптимизация графика строительства жилого комплекса

Используя методы интеллектуальной оптимизации, разработайте оптимальный календарный план строительства жилого комплекса из 3 домов. Учесть ограничения по ресурсам, сезонность и технологическую последовательность работ.

Проект выполняется в командах от 2 до 3 человек. Оценивается вся команда одной оценкой.

Индивидуальное задание состоит в анализе качества разработанных алгоритмов оптимизации. Для выполнения задания необходимо:

1. Подготовить тестовые данные для проверки алгоритмов
2. Реализовать минимум 2 различных метода оптимизации (например, генетический алгоритм и метод роевого интеллекта)
3. Провести сравнительный анализ методов по критериям:
 - Скорость сходимости
 - Качество решения
 - Устойчивость к изменению начальных условий
4. Построить графики сходимости для каждого метода
5. Рассчитать метрики эффективности для полученных решений
6. Сделать выводы о применимости методов для данного класса задач

Критерии оценки:

Критерии оценки:

Отлично: Полное выполнение задачи, поставленной в кейсе. Важно оценить каждый критерий.

- Корректность математической постановки задачи
- Качество реализации алгоритмов оптимизации
- Глубина сравнительного анализа методов
- Практическая значимость полученных результатов
- Качество оформления отчета и презентации

Хорошо: Незначительные недочеты в коде или отчете.

Удовлетворительно: Выполнены базовые задания, но с ошибками.

Неудовлетворительно: Критические ошибки или невыполнение работы.

Дополнительные требования ко всем проектам:

- Использование Python и специализированных библиотек оптимизации

- Визуализация процесса оптимизации и результатов
- Экономическое обоснование предлагаемых решений
- Соответствие строительным нормативам и стандартам

Требования для повышения оценки и итоговую оценку формирует преподаватель.

Задача №4: разработка системы учёта результатов проекта в итоговой отметке за зачет

Выполнено в РПД, п 4.2

Порядок проверки корректности

Чек-лист для проверки лабораторных работ:

Набор кейсов индустриальных партнеров – 21 шт;

Набор кейсов преподавателей практиков и лабораторий ВУЗа – 8 шт;

Набор ТЗ в количестве 15 штук.

5. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины (модуля)

5.1 Учебная литература

Основная литература:

1. Кундышева, Е. С. Математические методы и модели в экономике : учебник / Е. С. Кундышева ; под науч. ред. Б. А. Сулакова. – 3-е изд. – Москва : Дашков и К°, 2022. – 286 с. : ил., табл., граф. – (Учебные издания для бакалавров). – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=684490> (дата обращения: 01.12.2024). – ISBN 978-5-394-04621-6. – Текст : электронный.

2. Поляков, В. М. Интеллектуальные методы оптимизации : учебное пособие / В. М. Поляков, З. С. Агаларов. – 3-е изд. – Москва : Дашков и К°, 2024. – 86 с. : ил., табл. – (Учебные издания для вузов). – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=720347> (дата обращения: 01.12.2024). – Библиогр. в кн. – ISBN 978-5-394-05718-2. – Текст : электронный.

3. Количественные методы и модели в теории управления : учебник для магистратуры : [16+] / Л. А. Каргина, О. Е. Михненко, А. И. Фроловичев [и др.] ; под ред. Л. А. Каргиной ; Российский университет транспорта (РУТ (МИИТ)). – Москва : Прометей, 2022. – 274 с. : ил. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=690752> (дата обращения: 01.12.2024). – Библиогр. в кн. – ISBN 978-5-00172-299-1. – Текст : электронный.

5.2. Дополнительная литература:

1. Sun, X., Li, J., Kovalenko, A.V., Feng, W., Ou, Y. Integrating Reinforcement Learning and Learning From Demonstrations to Learn Nonprehensile Manipulation //IEEE Transactions on Automation Science and Engineering, 2023, 20(3), 1735–1744, DOI: 10.1109/TASE.2022.3185071, Q1
2. Petukhova, A.V.; Kovalenko, A.V.; Ovsyannikova, A.V. Algorithm for Optimization of Inverse Problem Modeling in Fuzzy Cognitive Maps. Mathematics 2022, 10, 3452. DOI: 10.3390/math10193452, Q1
3. Kirillova, E.; Kovalenko, A.; Urtenov, M. Study of the Current–Voltage Characteristics of Membrane Systems Using Neural Networks. AppliedMath 2025, 5, 10. <https://doi.org/10.3390/appliedmath5010010>.
4. Kadurin, Artur, et al. "The cornucopia of meaningful leads: Applying deep adversarial autoencoders for new molecule development in oncology." Oncotarget 8.7 (2016): 10883.
5. Kadurin, Artur, et al. "druGAN: an advanced generative adversarial autoencoder model for de novo generation of new molecules with desired molecular properties in silico." Molecular pharmacology 14.9 (2017): 3098-3104.

6. Polykovskiy, Daniil, et al. "Molecular sets (MOSES): a benchmarking platform for molecular generation models." *Frontiers in pharmacology* 11 (2020): 565644.
7. Khrabrov, Kuzma, et al. " ∇^2 DFT: A Universal Quantum Chemistry Dataset of Drug-Like Molecules and a Benchmark for Neural Network Potentials." *Advances in Neural Information Processing Systems* 37 (2024): 36869-36889.
8. Polykovskiy, Daniil, et al. "Entangled conditional adversarial autoencoder for de novo drug discovery." *Molecular pharmaceutics* 15.10 (2018): 4398-4405.
9. Николенко, Сергей, Кадури, Артур и Архангельская Екатерина. Глубокое обучение. "Издательский дом" Питер", 2017.
10. Рабчевский, А. Н. Синтетические данные и развитие нейросетевых технологий : учебное пособие для вузов / А. Н. Рабчевский. — Москва : Издательство Юрайт, 2024. — 187 с. — (Высшее образование). — ISBN 978-5-534-17716-9. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/545036> (дата обращения: 19.07.2025).
11. Елисеев А. И., Минин Ю. В. Разработка программных интерфейсов веб-приложений с использованием фреймворка FastAPI : учебное пособие. Тамбов: ТГТУ, 2024. 81 с. <https://e.lanbook.com/book/472310> (дата обращения: 19.07.2025).
12. Чернышев, С. А. Основы программирования на Python : учебное пособие для вузов / С. А. Чернышев. — Москва : Издательство Юрайт, 2022. — 286 с. — (Высшее образование). — ISBN 978-5-534-14350-8. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/496893>. (дата обращения: 19.07.2025).
13. Златопольский Д. М. Основы программирования на языке Python. 2-е изд. Москва: ДМК Пресс, 2018.
14. Зайцев, Михаил Григорьевич. Интеллектуальные методы оптимизации управления и принятия решений: примеры, задачи, кейсы/ Зайцев, Михаил Григорьевич, С. Е. Варюхин; М. Г. Зайцев, С. Е. Варюхин; Рос. акад. народного хоз-ва и гос. службы при Президенте Рос. Федерации. - [3-е изд., испр. и доп.]. - М.: Дело, 2011. - 639 с.: ил. - (Учебники Президентской Академии). - ISBN 9785774904921.
15. Далингер, В. А. Информатика и математика. Решение уравнений и оптимизация в mathcad и maple [Электронный ресурс] : учебник и практикум для прикладного бакалавриата / В. А. Далингер, С. Д. Симонженков. - 2-е изд., испр. и доп. - М. : Юрайт, 2018. - 161 с. - <https://biblio-online.ru/book/373E27B2-F2B8-4BC9-9D66-EFFA2353B4D1>.
16. Интеллектуальные методы оптимизации [Электронный ресурс] : учебник и практикум для бакалавриата и магистратуры / Ф. П. Васильев, М. М. Потапов, Б. А. Будак, Л. А. Артемьева ; под ред. Ф. П. Васильева. - М. : Юрайт, 2018. - 375 с. - <https://biblio-online.ru/book/CAA9AF22-E3BB-454A-BE5C-BB243EAAE72A>.

5.3. Периодические издания:

1. Базы данных компании «Ист Вью» <http://dlib.eastview.com>
2. Электронная библиотека GREBENNIKON.RU <https://grebennikon.ru/>

Конференции А*:

3. <https://openreview.net/forum?id=FMMF1a9ifL>
4. <https://openreview.net/forum?id=ElUrNM9U8c#discussion>
5. <https://openreview.net/forum?id=JoO6mtCLHD>
6. <https://aclanthology.org/2024.findings-emnlp.760/>
7. <https://aclanthology.org/2020.coling-main.588/>
8. https://link.springer.com/chapter/10.1007/978-3-030-72113-8_30
9. https://link.springer.com/chapter/10.1007/978-3-031-42448-9_10
10. <https://aclanthology.org/2024.findings-naacl.288/>

5.4. Интернет-ресурсы, в том числе современные профессиональные базы данных и информационные справочные системы

Электронно-библиотечные системы (ЭБС):

1. ЭБС «ЮРАЙТ» <https://urait.ru/>
2. ЭБС «УНИВЕРСИТЕТСКАЯ БИБЛИОТЕКА ОНЛАЙН» <http://www.biblioclub.ru/>
3. ЭБС «BOOK.ru» <https://www.book.ru>
4. ЭБС «ZNANIUM.COM» www.znanium.com
5. ЭБС «ЛАНЬ» <https://e.lanbook.com>

Профессиональные базы данных

1. Scopus <http://www.scopus.com/>
2. ScienceDirect <https://www.sciencedirect.com/>
3. Журналы издательства Wiley <https://onlinelibrary.wiley.com/>
4. Научная электронная библиотека (НЭБ) <http://www.elibrary.ru/>
5. Полнотекстовые архивы ведущих западных научных журналов на Российской платформе научных журналов НЭИКОН <http://archive.neicon.ru>
6. Национальная электронная библиотека (доступ к Электронной библиотеке диссертаций Российской государственной библиотеки (РГБ) <https://rusneb.ru/>
7. Президентская библиотека им. Б.Н. Ельцина <https://www.prilib.ru/>
8. База данных CSD Кембриджского центра кристаллографических данных (CCDC) <https://www.ccdc.cam.ac.uk/structures/>
9. Springer Journals: <https://link.springer.com/>
10. Springer Journals Archive: <https://link.springer.com/>
11. Nature Journals: <https://www.nature.com/>
12. Springer Nature Protocols and Methods: <https://experiments.springernature.com/sources/springer-protocols>
13. Springer Materials: <http://materials.springer.com/>
14. Nano Database: <https://nano.nature.com/>
15. Springer eBooks (i.e. 2020 eBook collections): <https://link.springer.com/>
16. "Лекториум ТВ" <http://www.lektorium.tv/>
17. Университетская информационная система РОССИЯ <http://uisrussia.msu.ru>

Бесплатные образовательные ресурсы

1. Jupyter Notebook – интерактивные вычисления
2. Visual Studio Code – редактор кода с поддержкой Python
3. Google Scholar/arXiv – доступ к научным публикациям

Ресурсы свободного доступа

1. Официальная документация по TensorFlow <https://www.tensorflow.org/?hl=ru>
2. Официальная документация по Keras <https://www.tensorflow.org/guide/keras?hl=ru>
3. Официальная документация по pyTorch <https://docs.pytorch.org/docs/stable/index.html>
4. КиберЛенинка <http://cyberleninka.ru/>;
5. Американская патентная база данных <http://www.uspto.gov/patft/>
6. Министерство науки и высшего образования Российской Федерации <https://www.minobrnauki.gov.ru/>;
7. Федеральный портал "Российское образование" <http://www.edu.ru/>;
8. Информационная система "Единое окно доступа к образовательным ресурсам" <http://window.edu.ru/>;
9. Единая коллекция цифровых образовательных ресурсов <http://school-collection.edu.ru/> .
10. Проект Государственного института русского языка имени А.С. Пушкина "Образование на русском" <https://pushkininstitute.ru/>;
11. Справочно-информационный портал "Русский язык" <http://gramota.ru/>;
12. Служба тематических толковых словарей <http://www.glossary.ru/>;

13. Словари и энциклопедии <http://dic.academic.ru/>;
14. Образовательный портал "Учеба" <http://www.ucheba.com/>;
15. Законопроект "Об образовании в Российской Федерации". Вопросы и ответы http://xn--273--84d1f.xn--p1ai/voprosy_i_otvety

Собственные электронные образовательные и информационные ресурсы КубГУ

1. Электронный каталог Научной библиотеки КубГУ <http://megapro.kubsu.ru/MegaPro/Web>
2. Электронная библиотека трудов ученых КубГУ <http://megapro.kubsu.ru/MegaPro/UserEntry?Action=ToDb&idb=6>
3. Среда модульного динамического обучения <http://moodle.kubsu.ru>
4. База учебных планов, учебно-методических комплексов, публикаций и конференций <http://infoneeds.kubsu.ru/>
5. Библиотека информационных ресурсов кафедры информационных образовательных технологий [http://mschool.kubsu.ru/](http://mschool.kubsu.ru;);
6. Электронный архив документов КубГУ <http://docspace.kubsu.ru/>
7. Электронные образовательные ресурсы кафедры информационных систем и технологий в образовании КубГУ и научно-методического журнала "ШКОЛЬНЫЕ ГОДЫ" <http://icdau.kubsu.ru/>

Перечень ресурсов информационно-телекоммуникационной сети «Интернет», в том числе современные профессиональные базы данных и информационные справочные системы, необходимые для освоения дисциплины.

1. Справочно-правовая система «Консультант Плюс» (<http://www.consultant.ru>)
2. Портал открытых данных Российской Федерации <https://data.gov.ru>
3. База открытых данных Министерства труда и социальной защиты РФ <https://rosmintrud.ru/opendata>
4. База данных Научной электронной библиотеки eLIBRARY.RU <https://elibrary.ru/>
5. База данных Всероссийского института научной и технической информации (ВИНИТИ) РАН <http://www2.viniti.ru/>
6. Базы данных в сфере интеллектуальной собственности, включая патентные базы данных www.rusnano.com
7. Базы данных и аналитические публикации «Университетская информационная система РОССИЯ» <https://uisrussia.msu.ru/>
8. Википедия, свободная энциклопедия. [Электронный ресурс]. – Wikipedia <http://ru.wikipedia.org>
9. <http://math.nsc.ru/LBRT/k5/opt.html> (Интеллектуальные методы оптимизации, учебное пособие)

6. Методические указания для обучающихся по освоению дисциплины

Учебная деятельность проходит в соответствии с графиком учебного процесса. Процесс самостоятельной работы контролируется во время аудиторных занятий и индивидуальных консультаций. Самостоятельная работа студентов проводится в форме изучения отдельных теоретических вопросов по предлагаемой литературе и выполнении практических заданий по разобранным во время аудиторных занятий примерам.

Фонд оценочных средств дисциплины состоит из средств текущего контроля (см. список лабораторных работ, задач и вопросов) и итоговой аттестации (зачета, экзамена).

В качестве оценочных средств, используемых для текущего контроля успеваемости, предлагается перечень вопросов, которые прорабатываются в процессе освоения курса. Данный перечень охватывает все основные разделы курса, включая знания, получаемые во время самостоятельной работы. Кроме того, важным элементом технологии являются самостоятельное решение студентами и сдача заданий. Это полностью индивидуаль-

ная форма обучения. Студент рассказывает свое решение преподавателю, отвечает на дополнительные вопросы.

Для студентов с ограниченными возможностями здоровья предусмотрены индивидуальные консультации и адаптированные материалы. Преподаватель помогает осваивать интерфейсы взаимодействия с ИИ, объясняет ключевые понятия в доступной форме, предоставляет инструкции с альтернативным форматированием. При необходимости используются голосовые интерфейсы, увеличенный масштаб экрана, сопровождение при выполнении заданий. Индивидуальный подход обеспечивает равные условия участия в образовательном процессе и достижения запланированных результатов обучения.

Подход, определяющий установление соответствия кейсов ИП и УГТ (5-7), позволяет четко соотносить этапы развития технологии с вовлеченностью партнера и снижать риски при переходе от лабораторных испытаний к промышленному внедрению.

Кейсы ПАО «Сбербанк»

1. Оптимизация сети отделений банка (Branch Network Optimization)

Описание: Используя данные о клиентском потоке, арендных ставках, демографии и транспортной доступности, необходимо разработать модель оптимизации сети отделений Сбербанка. Учесть затраты на содержание, потенциальную выручку и стратегические цели охвата регионов.

Цель: Определить оптимальную конфигурацию сети отделений (открытие, закрытие, переезд) для максимизации прибыли при сохранении качества обслуживания.

Ожидаемые результаты:

- Модель оптимизации размещения отделений с учетом пространственных данных
- Алгоритм оценки экономической эффективности каждого отделения
- План реструктуризации сети с расчетом ROI

2. Оптимизация капитала банка (Capital Allocation Optimization)

Описание: Разработать систему оптимального распределения капитала между различными направлениями бизнеса (кредитование, инвестиции, лизинг) с учетом риск-аппетита, регуляторных требований и рыночных условий.

Цель: Максимизировать доходность использования капитала при соблюдении нормативов ЦБ и ограничений по рискам.

Ожидаемые результаты:

- Многокритериальная модель оптимизации распределения капитала
- Алгоритм динамической корректировки allocation в зависимости от изменения рынка
- Инструмент для сценарного анализа и стресс-тестирования

3. Оптимизация ликвидности банка (Liquidity Management Optimization)

Описание: Создать систему управления ликвидностью, которая оптимально распределяет денежные средства между различными инструментами (межбанк, ОФЗ, корпоративные облигации) с учетом сроков, доходности и рисков.

Цель: Обеспечить выполнение нормативов ликвидности при минимизации альтернативных издержек.

Ожидаемые результаты:

- Модель оптимизации портфеля ликвидности

- Алгоритм прогнозирования cash flow и стресс-сценариев
- Система рекомендаций по размещению временно свободных средств

4. Оптимизация кредитного портфеля (Loan Portfolio Optimization)

Описание: Разработать модель оптимизации структуры кредитного портфеля по отраслям, регионам и типам заемщиков с учетом риск-профиля, доходности и корреляции дефолтов.

Цель: Максимизировать риск-скорректированную доходность кредитного портфеля в рамках установленных лимитов.

Ожидаемые результаты:

- Модель оптимизации Markowitz для кредитного портфеля
- Алгоритм расчета оптимальных кредитных лимитов
- Инструмент для анализа концентрации рисков

5. Оптимизация процессов колл-центра (Call Center Workforce Optimization)

Описание: Создать систему оптимального распределения операторов колл-центра по сменам и навыкам на основе прогноза входящих звонков, сложности запросов и SLA.

Цель: Минимизировать затраты на персонал при обеспечении целевых показателей обслуживания.

Ожидаемые результаты:

- Модель оптимизации расписания операторов
- Алгоритм прогнозирования нагрузки в разрезе типов запросов
- Расчет экономии фонда оплаты труда

6. Оптимизация кампаний кросс-продаж (Cross-Selling Campaign Optimization)

Описание: Разработать систему оптимального планирования маркетинговых кампаний по кросс-продажам банковских продуктов с учетом клиентского профиля, истории взаимодействий и каналов коммуникации.

Цель: Максимизировать конверсию кросс-продаж при ограниченном маркетинговом бюджете.

Ожидаемые результаты:

- Модель оптимизации маркетингового микса
- Алгоритм персонализированных предложений
- Инструмент оценки эффективности кампаний

7. Оптимизация процессинговой инфраструктуры (Payment Processing Optimization)

Описание: Создать модель оптимального распределения транзакционной нагрузки между процессинговыми центрами с учетом их мощностей, задержек и стоимости обработки.

Цель: Минимизировать операционные затраты на процессинг при обеспечении SLA по времени обработки транзакций.

Ожидаемые результаты:

- Модель балансировки нагрузки между дата-центрами
- Алгоритм прогнозирования пиковых нагрузок
- Расчет оптимальной конфигурации процессинговой инфраструктуры

8. Оптимизация валютной позиции банка (Foreign Exchange Position Optimization)

Описание: Разработать систему управления валютной позицией банка, оптимально распределяющей активы и пассивы в различных валютах с учетом курсовых рисков и стоимости хеджирования.

Цель: Минимизировать валютные риски при заданных ограничениях по капиталу.

Ожидаемые результаты:

- Модель оптимизации валютной позиции
- Алгоритм определения оптимальных объемов хеджирования
- Система мониторинга валютных рисков в реальном времени

9. Оптимизация залогового портфеля (Collateral Portfolio Optimization)

Описание: Создать систему оптимального управления залоговым портфелем ипотечных и корпоративных кредитов с учетом ликвидности залогов, их стоимости и рисков обесценения.

Цель: Максимизировать качество залогового покрытия при минимизации затрат на управление.

Ожидаемые результаты:

- Модель оптимизации структуры залогового портфеля
- Алгоритм приоритезации залогов для регулярной переоценки
- Инструмент для стресс-тестирования залоговой базы

10. Оптимизация IT-инфраструктуры банка (IT Infrastructure Optimization)

Описание: Разработать модель оптимального распределения вычислительных ресурсов между бизнес-приложениями банка с учетом их критичности, нагрузки и требований к производительности.

Цель: Минимизировать затраты на IT-инфраструктуру при обеспечении бесперебойности бизнес-процессов.

Ожидаемые результаты:

- Модель оптимизации распределения вычислительных ресурсов
- Алгоритм автоматического масштабирования мощностей
- Расчет экономии на капитальных и операционных затратах

Кейсы от «АВАЛАБ»

1. Оптимизация графика строительства жилого комплекса (Construction Schedule Optimization)

Описание: Используя данные о последовательности работ, ресурсах, погодных условиях и логистике, необходимо разработать оптимальный календарный план строительства жилого комплекса. Учесть взаимозависимости работ, ограничения по подрядчикам и сезонные факторы.

Цель: Минимизировать сроки строительства при соблюдении бюджетных ограничений и качества работ.

Ожидаемые результаты:

- Модель оптимизации критического пути с учетом ресурсных ограничений
- Алгоритм динамической корректировки графика при сбоях
- Расчет экономии от сокращения сроков строительства

2. Оптимизация закупки строительных материалов (Procurement Optimization)

Описание: Разработать систему оптимального планирования закупок материалов с учетом колебания цен, логистики, условий хранения и графика строительства. Учесть риски срыва поставок и альтернативных поставщиков.

Цель: Минимизировать затраты на материалы при обеспечении бесперебойности строительства.

Ожидаемые результаты:

- Модель оптимизации объемов и сроков закупок
- Алгоритм выбора оптимальных поставщиков

- Система управления складскими запасами

3. Оптимизация использования строительной техники (Equipment Utilization Optimization)

Описание: Создать систему оптимального распределения строительной техники между объектами компании с учетом их характеристик, стоимости аренды/эксплуатации и требований проектов.

Цель: Максимизировать коэффициент использования техники при минимизации затрат.

Ожидаемые результаты:

- Модель оптимизации парка техники
- Алгоритм планирования ремонтов и обслуживания
- Расчет оптимального состава собственного и арендуемого парка

4. Оптимизация проектных решений (Design Solutions Optimization)

Описание: Разработать систему многокритериальной оптимизации проектных решений для жилых комплексов с учетом стоимости материалов, энергоэффективности, сроков реализации и рыночной привлекательности.

Цель: Найти оптимальные проектные решения, балансирующие между затратами и качеством.

Ожидаемые результаты:

- Модель оптимизации конструктивных решений
- Алгоритм выбора отделочных материалов
- Инструмент для оценки trade-off между различными параметрами

5. Оптимизация продаж квартир (Apartment Sales Optimization)

Описание: Создать систему оптимального управления продажами квартир в строящемся доме с учетом динамики спроса, ценовой эластичности и конкурентной среды.

Цель: Максимизировать выручку от продажи квартир в течение жизненного цикла проекта.

Ожидаемые результаты:

- Модель динамического ценообразования
- Алгоритм определения оптимального момента для запуска продаж
- Система управления скидками и акциями

6. Оптимизация финансирования строительства (Construction Financing Optimization)

Описание: Разработать модель оптимального структурирования финансирования строительных проектов с учетом стоимости капитала, графиков освоения средств и рисков.

Цель: Минимизировать стоимость финансирования при обеспечении ликвидности проекта.

Ожидаемые результаты:

- Модель оптимизации структуры финансирования
- Алгоритм управления денежными потоками
- Инструмент для сценарного анализа финансовых рисков

7. Оптимизация управления подрядчиками (Contractor Management Optimization)

Описание: Создать систему оптимального отбора и распределения подрядчиков по объектам с учетом их специализации, загрузки, репутации и стоимости услуг.

Цель: Минимизировать риски срывов сроков и качества работ при оптимизации затрат на подрядчиков.

Ожидаемые результаты:

- Модель оптимизации пула подрядчиков
- Алгоритм оценки надежности подрядчиков
- Система мониторинга выполнения обязательств

8. Оптимизация земельного банка (Land Bank Optimization)

Описание: Разработать модель оптимального управления земельным банком компании с учетом перспектив развития территорий, инфраструктурных проектов и рыночного спроса.

Цель: Максимизировать стоимость земельного банка и эффективность его использования.

Ожидаемые результаты:

- Модель оптимизации портфеля земельных участков
- Алгоритм определения приоритетов освоения
- Инструмент для оценки инвестиционной привлекательности участков

9. Оптимизация эксплуатационных расходов (Operational Expenses Optimization)

Описание: Создать систему оптимального управления эксплуатационными расходами сданных объектов с учетом энергоэффективности, сервисных затрат и требований жителей.

Цель: Минимизировать эксплуатационные расходы при сохранении качества обслуживания.

Ожидаемые результаты:

- Модель оптимизации затрат на содержание ЖК
- Алгоритм планирования ремонтов и обновлений
- Система мониторинга эффективности управляющей компании

10. Оптимизация маркетингового бюджета (Marketing Budget Optimization)

Описание: Разработать модель оптимального распределения маркетингового бюджета между каналами продвижения, проектами и этапами продаж с учетом их эффективности и целевой аудитории.

Цель: Максимизировать ROI маркетинговых инвестиций.

Ожидаемые результаты:

- Модель оптимизации медиамикса
- Алгоритм определения оптимального бюджета для каждого проекта
- Инструмент для атрибуции маркетинговых результатов

Каждый кейс содержит конкретную бизнес-задачу девелоперской компании, требующую применения интеллектуальных методов оптимизации, и четкие измеримые результаты для оценки эффективности решений.

7. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю)

7.1 Перечень информационных технологий.

1. Облачные платформы и сервисы

Google Colab – облачная среда для выполнения кода на Python с GPU/TPU

Kaggle – платформа для работы с датасетами и соревнований по ML

Hugging Face Spaces – развертывание CV-моделей в виде демо

cloud.ru, YandexCloud, AWS/GCP/Azure – облачные вычисления

2. Системы управления версиями и коллаборации

Git/GitHub/GitLab – контроль версий кода и совместная разработка

3. Инструменты для работы с данными

Label Studio – разметка датасетов
DVC (Data Version Control) – управление версиями данных
Apache Spark – обработка больших текстовых корпусов

4. Система управления обучением
Moodle – сдача работ

7.2 Перечень необходимого программного обеспечения.

1. Лицензионное ПО

VSCoде – IDE для Python (свободнораспространяемое)
LibreOffice– оформление отчетов (свободнораспространяемое)

2. Свободное ПО (Open Source)

Hugging Face Transformers – предобученные модели (BERT, GPT, визуальные трансформеры)
Hugging Face Optimum – оптимизация и развёртывание моделей

Фреймворки и библиотеки для DL:

PyTorch/TensorFlow/Keras– разработка нейросетей

Инструменты для визуализации:

Streamlit/Gradio – создание веб-интерфейсов для моделей
Matplotlib/Seaborn – графики и анализ данных

СУБД:

SQLite/PostgreSQL – хранение структурированных данных
FAISS/Annoy – векторный поиск

8. Материально-техническое обеспечение по дисциплине (модулю)

Виртуальные машины и ресурсы GPU в облаке предоставляется индустриальным партнером ПАО «Сбербанк»:

№	Продукт	Параметры продукта	Кол-во	Кол-во конфигураций	Ед. изм.
1	Виртуальная машина	Виртуальная машина 10% vCPU 2 vCPU 4 RAM	1	60	Шт
		ОС Ubuntu 22.04	1		Шт
		Системный диск SSD	1		Шт
			10		Гб
		Аренда публичного IP	1		Шт
2	Виртуальная машина с GPU	Виртуальная машина с GPU NVIDIA® Tesla® V100 2 GPU 8 vCPU 128 Гб RAM	1	1	Шт
		ОС Ubuntu_24.04	1		Шт
		Системный диск SSD	1		Шт

3	K8S		2000		Гб
		Диск SSD	1		Шт
			4096		Гб
		Диск SSD	1		Шт
			4096		Гб
		Аренда публичного IP	1		Шт
		Master node 8 vCPU 16 RAM	1	1	Шт
		Worker node 10% доля 4 vCPU 32 RAM	5		Шт
		Worker node SSD-NVME	64		Гб
		Аренда публичного IP	1		Шт
4	ML Inference Instance Type GPU	Время работы в месяц	40	1	Ч
		Инстанс 8 x NVIDIA® H100 NVLink PCIe 160 vCPU 1520 GB RAM	1		Шт
		Количество запросов к ML-моделям	1		Млн. Шт
		Кэш ML-моделей	160		Гб
		Токены GigaChat 2 Max	50		Млн. Шт
5	LLM	Токены Embeddings	400		Млн. Шт

Дополнительные облачные ресурсы предоставляются технологическим партнером Yandex Cloud.

№	Вид работ	Наименование учебной аудитории, ее оснащенность оборудованием и техническими средствами обучения
1.	Лекционные занятия	Аудитория, укомплектованная специализированной мебелью и техническими средствами обучения
2.	Лабораторные занятия	Аудитория, укомплектованная специализированной мебелью и техническими средствами обучения, компьютерами, проектором, программным обеспечением
3.	Практические занятия	Аудитория, укомплектованная специализированной мебелью и техническими средствами обучения
4.	Групповые (индивидуальные) консультации	Аудитория, укомплектованная специализированной мебелью и техническими средствами обучения, компьютерами, программным обеспечением
5.	Текущий контроль, промежуточная аттестация	Аудитория, укомплектованная специализированной мебелью и техническими средствами обучения, компьютерами, программным обеспечением
6.	Самостоятельная работа	Кабинет для самостоятельной работы, оснащенный компьютерной техникой с возможностью подключения к сети «Интернет», программой экранного увеличения и обеспеченный доступом в электронную информационно-образовательную среду университета.

Примечание: Конкретизация аудиторий и их оснащение определяется ОПОП.