

**Федеральное государственное автономное образовательное  
учреждение высшего образования  
«Московский физико-технический институт  
(национальный исследовательский университет)»**

**УТВЕРЖДЕНО**

**Директор физтех-школы  
электроники, фотоники и  
молекулярной физики**

**В.В. Иванов**

**Рабочая программа дисциплины (модуля)**

**по дисциплине:** Теория и практика машинного обучения и нейросетевых алгоритмов  
**по направлению:** Прикладные математика и физика  
**профиль подготовки:** Физика перспективных технологий: электроника и квантовые технологии  
Физтех-школа Электроники, Фотоники и Молекулярной Физики  
кафедра электроники  
**курс:** 4  
**квалификация:** бакалавр

Семестр, формы промежуточной аттестации: 7 (осенний) - Дифференцированный зачет

Аудиторных часов: 60 всего, в том числе:

лекции: 30 час.

семинары: 30 час.

лабораторные занятия: 0 час.

Самостоятельная работа: 30 час.

Всего часов: 90, всего зач. ед.: 2

Программу составил: А.В. Хельвас

Программа обсуждена на заседании кафедры электроники 29.05.2020

## **Аннотация**

Курс "Теория и практика машинного обучения и нейросетевых алгоритмов" предусматривает формирование у студентов представления и знания о математических основах современных методов машинного обучения, существующих видах нейросетевых алгоритмов и особенностях их применения для различных классов задач. В процессе обучения студенты должны научиться разрабатывать и применять нейросетевые и генетические алгоритмы и уметь оценивать требования к используемым программно - аппаратным средствам.

Задачи курса:

Приобретение знаний о методах и основных подходах современного машинного обучения, о математических основах нейросетевых алгоритмов, видах нейронных сетей и особенностях их применения для решения различных классов прикладных задач.

Формирование умений разрабатывать и настраивать нейронные сети в соответствии с конкретными прикладными задачами.

Умение анализировать задачи, решение которых возможно на основе генетических алгоритмов и решать их.

Оценивать требования к платформе для реализации нейросетевых алгоритмов и проводить benchmarking для основных нейросетевых алгоритмов.

По результатам освоения курса студент должен:

Знать:

- возможности и сферы применения современной теории искусственных нейронных сетей;
- основные подходы решения задач на базе нейросетевых технологий.

Уметь:

- уметь поставить и представить конкретную задачу в нейросетевом логическом базисе;
- формализовать алгоритмы обработки информации в параллельных и распределенных представлениях;
- решать инженерные задачи с помощью инструментальных средств искусственных нейронных сетей.

Владеть:

- методами машинного обучения и создания нейросетевых алгоритмов.

Основное содержание курса изложено в следующих разделах:

1. Базовые элементы разработки на Python
2. Функции
3. Введение в ООП на языке Python
4. Отдельные главы разработки на Python
5. Приемы функционального программирования на языке Python
6. Работа с классами на примере сторонних библиотек
7. Практика программирования
8. Основы web технологий
9. Бэкенд на примере фреймворка flask
10. Тестирование программного обеспечения
11. Нейросети
12. Генетические алгоритмы

## **1. Цели и задачи**

## **Цель дисциплины**

- сформировать у студентов представления и знания о математических основах современных методов машинного обучения, существующих видах нейросетевых алгоритмов и особенностях их применения для различных классов задач. В процессе обучения студенты должны научиться разрабатывать и применять нейросетевые и генетические алгоритмы и уметь оценивать требования к используемым программно - аппаратным средствам.

### Задачи дисциплины

- приобретение знаний о методах и основных подходах современного машинного обучения, о математических основах нейросетевых алгоритмов, видах нейронных сетей и особенностях их применения для решения различных классов прикладных задач;
- формирование умений разрабатывать и настраивать нейронные сети в соответствии с конкретными прикладными задачами;
- умение анализировать задачи, решение которых возможно на основе генетических алгоритмов и решать их;
- оценивать требования к платформе для реализации нейросетевых алгоритмов и проводить benchmarking для основных нейросетевых алгоритмов.

## 2. Перечень формируемых компетенций

Освоение дисциплины направлено на формирование следующих компетенций:

Код и наименование компетенции	Индикаторы достижения компетенции
УК-1 Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	УК-1.1 Анализирует задачу, выделяя этапы ее решения, действия по решению задачи
	УК-1.3 Рассматривает различные варианты решения задачи, оценивает их преимущества и недостатки
ПК-1 Способен планировать и проводить научные эксперименты (в избранной предметной области) и (или) теоретические (аналитические и имитационные) исследования	ПК-1.1 Владеет фундаментальными понятиями, законами и теориями современной физики
	ПК-1.8 Владеет навыками работы с современными языками программирования и программными пакетами для научных расчетов
	ПК-1.4 Умеет строить математические модели для описания и исследования процессов и явлений в соответствующих научных областях

## 3. Перечень планируемых результатов обучения по дисциплине (модулю)

В результате освоения дисциплины обучающиеся должны

знать:

- возможности и сферы применения современной теории искусственных нейронных сетей;
- основные подходы решения задач на базе нейросетевых технологий.

уметь:

- уметь поставить и представить конкретную задачу в нейросетевом логическом базисе;
- формализовать алгоритмы обработки информации в параллельных и распределенных представлениях;
- решать инженерные задачи с помощью инструментальных средств искусственных нейронных сетей.

владеть:

- методами машинного обучения и создания нейросетевых алгоритмов.

## 4. Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий

### 4.1. Разделы дисциплины (модуля) и трудоемкости по видам учебных занятий

	Трудоемкость по видам учебных занятий, включая самостоятельную работу, час.
--	---

№	Тема (раздел) дисциплины	Лекции	Семинары	Лаборат. работы	Самост. работа
1	Базовые элементы разработки на Python	2	2		2
2	Функции	2	2		2
3	Введение в ООП на языке Python	2	2		2
4	Отдельные главы разработки на Python	2	2		2
5	Приемы функционального программирования на языке Python	2	2		2
6	Работа с классами на примере сторонних библиотек	2	2		2
7	Практика программирования	2	2		2
8	Основы web технологий	2	2		2
9	Бэкенд на примере фреймворка flask	2	2		2
10	Тестирование программного обеспечения	2	2		2
11	Нейросети	8	8		8
12	Генетические алгоритмы	2	2		2
Итого часов		30	30		30
Подготовка к экзамену		0 час.			
Общая трудоёмкость		90 час., 2 зач.ед.			

#### 4.2. Содержание дисциплины (модуля), структурированное по темам (разделам)

Семестр: 7 (Осенний)

##### 1. Базовые элементы разработки на Python

Динамичная типизация. Демонстрация примеров вытекающих из динамической типизации отличий от языка Java. Преимущества и минусы. Прочие особенности языка программирования Python такие как отсутствие констант, условное разделение на reference и value.

Полная классификация основных типов в Python. Строки, списки, множества, словари. Понятие неопределенной переменной. Иммутабельные типы: кортежи и замороженные множества

Основные методы строк, списков и словарей. Срезы. Конкатенация и дублирование. Суть иммутабельности строк

Условный оператор if-else-elif. Использование выражений невозвращающих логические выражения в качестве условных выражений. Логические операторы

Циклы while и do-while. Операторы break и continue

Цикл for. Отличие от языков C, Java. Понятие итератора и генератора. Генератор range

##### 2. Функции

Объявление функций. Ключевое слово def. Void и nonVoid функции. Оператор return. Типы возвращаемых значений в рамках динамической типизации. Иммутабельность возвращаемого значения функции.

Аргументы функции. Опциональные аргументы функции. Иммутабельность аргументов функции.

Аргументы функции списков и ключевые аргументы словарем.

Запуск программы с аргументами командной строки.

### 3. Введение в ООП на языке Python

Определение классов. Ключевое слово `class`. Поля класса и методы класса. Объекты и экземпляры классов. Условная инкапсуляция и псевдо-инкапсуляция в Python.

Понятие `self`. `Self` в качестве аргумента метода.

Полиморфизм методов. Рекурсивный полиморфизм в случае множественного наследования.

Конструкторы и деструкторы. Конструктор "по умолчанию". Примеры необходимости в деструкторах.

Наследование. Множественное наследование. Понятие `super` "Магические методы". Методы `__repr` и `str`. Перегрузка базовых операторов через магические методы класса.

### 4. Отдельные главы разработки на Python

Потоки. Работа с `.txt` файлами. Функции `open` и `close`. Опасность использования потоков.

Байтовые строки. Работа с байтовыми файлами.

Исключения. Оператор `try-except-finally`. Область использования. Разбор примеров неправильной работы с исключениями. Прочие способы реализации обработки ошибок.

`With-as`. Пример использования оператора `with-as` на примере безопасной работы с файлами.

Регулярные выражение. Модуль `re`. Примеры типичных задач с решениями.

### 5. Приемы функционального программирования на языке Python

Lambda-функции. Использование анонимных функций. Сортировка списка.

Использование `unzip`, `map`, `filter`, `reduce`.

Генераторы списков и словарей.

Декораторы. Примеры декораторов на задаче логирования. Различные примеры декораторов для решения обычных задач. Статические методы. Абстрактные классы.

Функции как переменные и аргументы функций. Функторы класса. Монады.

### 6. Работа с классами на примере сторонних библиотек

Пакетный менеджер `PIP`. Виртуального окружения. Демонстрация создания области с виртуальном окружением Python и запуск программ из под созданного виртуального окружения.

`Numpy`. Массивы и точные числовые переменные. `Matplotlib`. Примеры построения графиков `PyQt`. Создание десктопного приложения с GUI. Интеграция в приложение `Matplotlib`.

### 7. Практика программирования

Стандарт `PEP8`. Примеры неправильно оформленных программ.

Принцип `SOLID`. Причины использования в промышленной разработке. Демонстрация использования принципов в разработке.

Автодокументирование. Комментарии в коде. Примеры неправильного комментирования кода. Создание автодокументаций при разработке прикладных библиотек в проекте.

Архитектура. Понятия вертикальной и горизонтальной иерархии классов. Составные классы. Проблема пространства имен. Вложенные классы и функции. Шаблоны программирования. Примеры использования шаблонов `singleton`, `builder`, `factory`, `observer`.

Проблема связи компонентов продукта развернутых на различных элементах инфраструктуры проекта. Клиент-серверная архитектура. Роль бинарных и строковых типов при общении компонентов системы.

Промышленные стандарты разработки. Пространство имен. Условности и договоренности в команде проекта.

### 8. Основы web технологий

HTTP запросы. Заголовки, методы, данные. RESTfull сервисы. Области применения. JSON, Blob, Xml. Парсинг. Библиотека request. Фронтенд. xml + js. Шаблонизаторы на примере Jinja2.

## 9. Бэкенд на примере фреймворка flask

Написание простейшего REST сервиса с помощью фреймворка flask.

Паутинг и методы POST и GET d flask. Получения параметров из аргументов url для методов GET. Получения аргументов для методов POST. Передача и получения информации через html form.

Написание простейшего веб приложения используя html+js+jinja2 на flask.

SQL. Базы данных и системы управления базы данных. Sqlite3 через sqlite3-shell. Проблемы проектирования баз данных.

Использование библиотеки sqlite3. Написание content и context менеджеров для работы с локальной sqlite3.

## 10. Тестирование программного обеспечения

Понятие тестов. Юнит, модульные и интеграционные тесты. UI тестирование. Понятие покрытия кода. Принцип TDD. Написание Unit тестов средствами Python.

Библиотека Selenium. Использование для тестирования Web приложений. Дополнительные возможности использования Selenium для сбора информации из сети интернет. Написание веб ботов.

Тестирование веб-сервисов. Использование утилиты curl и программы postman для тестирования работоспособности REST-Api.

## 11. Нейросети

История нейросетей. Логистическая регрессия. Построение 2-слойной нейросети своими руками. Метрики в глубинном обучении. Построение глубокой нейросети. Многослойный перцептрон. Алгоритм обратного распространения ошибки. Алгоритмы оптимизации, используемые в глубинном обучении. Регуляризация и подбор гиперпараметров.

Отладка нейросетей. Метод главных компонент. Аппроксимация данных. Поиск ортогональных проекций с наибольшим рассеянием. Диагонализация ковариационной матрицы.

Сингулярное разложение матрицы данных. Матрица преобразования к главным компонентам.

Остаточная дисперсия. Отбор главных компонент по правилу Кайзера. Нормировка.

Сверточные нейросети. Введение в сверточные нейронные сети. Операция свертки.

Простой сверточный слой. Усложнение сверточного слоя. Пулинг слой. Архитектура первой сверточной сети. Обзор архитектур сверточных нейросетей. LeNet, VGG, ResNet, UNet AlexNet, ZFNet, GoogLeNet CNN.

## 12. Генетические алгоритмы

Эволюционные методы. Простой генетический алгоритм. Кроссовер. Мутации. Переупорядочение генов. Селекция в генетических алгоритмах. Теорема шаблонов.

Метод комбинирования эвристик. Эффективность генетических алгоритмов. Примеры применения генетических методов.

## 5. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю)

аудитория, оснащенная персональными компьютерами с выходом в сеть "Интернет".

## 6.Перечень рекомендуемой литературы

Основная литература

1. Хайкин, С. Нейронные сети: полный курс / пер. с англ. - 2-е изд., испр. - М.: ООО "И.Д.Вильямс", 2006. - 1104 с.
2. Уоссермен Ф. Нейрокомпьютерная техника: теория и практика. - М.: Мир, 1992.
3. Тархов Д. А. Нейронные сети. Модели и алгоритмы. – Радиотехника, 2005
4. Комарцова Л.Г., Максимов А.В. Нейрокомпьютеры. - 2-е изд. – МГТУ им. Баумана, 2004

#### Дополнительная литература

1. Кадури А., Архангельская Е. В., Николенко С. И. Глубокое обучение. Погружение в мир нейронных сетей. - Питер, 2017. - 481 с.
2. Каллан Р. Основные концепции нейронных сетей. –Вильямс, 2002

### **7. Перечень ресурсов информационно-телекоммуникационной сети "Интернет", необходимых для освоения дисциплины (модуля)**

Не используются

### **8. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень необходимого программного обеспечения и информационных справочных систем (при необходимости)**

- трекеры задач TRELLO и JIRA;
- систему документирования проектов Confluence;
- репозиторий Git;
- облако для разработки технической документации в Latex [url{http://overleaf.com}](http://overleaf.com);
- облако для разработки решений в области AI IBM Cloud.

### **9. Методические указания для обучающихся по освоению дисциплины (модуля)**

Студент, прослушавший курс, должен овладеть теоретической базой квантовой химии. Успешное освоение курса требует самостоятельной работы студента.

Самостоятельная работа включает в себя:

- проработку учебного материала (по конспектам лекций, учебной и научной литературе);
- чтение и конспектирование дополнительной литературы;
- подготовку ответов на вопросы, предназначенных для самостоятельного изучения;
- решение задач, предлагаемых студентам на лекциях;
- подготовку к дифференцированному зачету.

Руководство и контроль самостоятельной работы студента осуществляется в форме индивидуальных консультаций.

Важно добиться понимания изучаемого материала, а не механического его запоминания. При затруднении изучения отдельных тем, вопросов, следует обращаться за консультациями к лектору.

**ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)**

<b>по направлению:</b>	Прикладные математика и физика
<b>профиль подготовки:</b>	Физика перспективных технологий: электроника и квантовые технологии Физтех-школа Электроники, Фотоники и Молекулярной Физики кафедра электроники
<b>курс:</b>	4
<b>квалификация:</b>	бакалавр
Семестр, формы промежуточной аттестации: 7 (осенний) - Дифференцированный зачет	
<b>Разработчик:</b>	А.В. Хельвас



## 1. Компетенции, формируемые в процессе изучения дисциплины

Код и наименование компетенции	Индикаторы достижения компетенции
УК-1 Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	УК-1.1 Анализирует задачу, выделяя этапы ее решения, действия по решению задачи
	УК-1.3 Рассматривает различные варианты решения задачи, оценивает их преимущества и недостатки
ПК-1 Способен планировать и проводить научные эксперименты (в избранной предметной области) и (или) теоретические (аналитические и имитационные) исследования	ПК-1.1 Владеет фундаментальными понятиями, законами и теориями современной физики
	ПК-1.8 Владеет навыками работы с современными языками программирования и программными пакетами для научных расчетов
	ПК-1.4 Умеет строить математические модели для описания и исследования процессов и явлений в соответствующих научных областях

## 2. Показатели оценивания компетенций

В результате изучения дисциплины «Теория и практика машинного обучения и нейросетевых алгоритмов» обучающийся должен:

### знать:

- возможности и сферы применения современной теории искусственных нейронных сетей;
- основные подходы решения задач на базе нейросетевых технологий.

### уметь:

- уметь поставить и представить конкретную задачу в нейросетевом логическом базисе;
- формализовать алгоритмы обработки информации в параллельных и распределенных представлениях;
- решать инженерные задачи с помощью инструментальных средств искусственных нейронных сетей.

### владеть:

- методами машинного обучения и создания нейросетевых алгоритмов.

## 3. Перечень типовых (примерных) вопросов, заданий, тем для подготовки к текущему контролю

С целью контроля освоения обучающимися учебного материала проводится устный опрос в начале занятия по теме прошлой лекции или в конце занятия по пройденной теме.

## 4. Перечень типовых (примерных) вопросов и тем для проведения промежуточной аттестации обучающихся

Вопросы для подготовки к дифференцированному зачету:

1. Биологический и искусственный нейрон. Основные функции активации нейронов. Преимущества нейронных сетей. Сопоставление традиционных ЭВМ и нейрокомпьютеров.
2. Классификации нейронных сетей, области применения и решаемые задачи. Основные направления развития нейрокомпьютинга.
3. Персептрон Розенблата. Алгоритм обучения персептрона и правило Хебба. Теорема о сходимости алгоритма обучения персептрона для линейно-разделимых множеств. Проблема исключающего «или».
4. Многослойный персептрон. Представление булевых функций. Преодоление ограничения линейной разделимости и решение проблемы исключающего «или».
5. 13-ая проблема Гильберта, теорема Колмогорова-Арнольда и её связь с нейронными сетями. Теорема Хехт-Нильсена.
6. Нейронные сети как универсальные аппроксиматоры. Теорема Стоуна-Вейерштрасса и теорема Цыбенко.

7. Общая идея градиентных методов решения задач безусловной оптимизации. Метод наискорейшего спуска.
8. Общая идея градиентных методов решения задач безусловной оптимизации. Метод Ньютона-Рафсона.
9. Функции потерь, сети типа ADALINE и дельта-правило Видроу-Хопфа.
10. Алгоритм обратного распространения ошибки. Достоинства и недостатки алгоритма. Понятие паралича сети и причины его возникновения.
11. Проблема овражности поверхности функционала ошибки и её частичное преодоление с помощью введения момента (инерциальной поправки). Физический смысл момента. Обобщенное дельта-правило.
12. Эвристические приемы улучшения сходимости и качества градиентного обучения (нормализация, выбор функции активации, выбор начальных значений весов, порядок предъявления обучающих примеров, выбор величины шага, сокращение числа весов, выбивание из локальных минимумов, проблема переобучения и разделение выборки). Методы упрощения структуры нейронной сети. Общие принципы обучения.
13. Этапы нейросетевого анализа. Методы кодирования категориальных и ординальных данных. Методы нормализации.
14. Методы восстановления пропусков в данных и возможные причины их возникновения. Обработка дубликатов и противоречий.
15. Методы выявления аномальных значений в данных (анализ графиков и гистограмм, ящичковые диаграммы, статистический тест, критерий Титъена-Мура, критерий Хи-квадрат для многомерных данных, вычисление расстояний в многомерном пространстве признаков, методы выявления аномальных значений для временных рядов).
16. Примеры возникновения хаоса в динамических системах. Показатели Ляпунова и их связь с максимальным горизонтом прогноза. Погружение временного ряда в лаговое пространство. Скользящее окно. Теорема Такенса. Фрактальная размерность Хаусдорфа-Безиковича и её связь с теоремой Такенса. Информационный подход Шеннона к вычислению максимального горизонта прогноза.
17. Аддитивная и мультипликативная модели временных рядов. Компоненты временного ряда. Исследование временных рядов на основе коррелограммы
18. Специфика прогнозирования финансовых временных рядов (выбор входных сигналов, метод искусственных примеров, выбор функционала ошибки и оценка величины капитала игрока).
19. Неравенство Коши-Буняковского. Неравенство между средним арифметическим и средним квадратическим. Преимущества использования комитетов нейронных сетей.
20. Задачи, решаемые без учителя. Идея метода главных компонент. Задача кластеризации данных. Основные метрики для количественных и не количественных переменных.
21. Сети Кохонена, правила жесткой, справедливой и мягкой конкуренции. Алгоритм обучения. Задача квантования данных. Задача многомерной визуализации и самоорганизующиеся карты Кохонена.

Кейсы для работы

1. Распознавание продуктов на подносе
2. Классификация жестов
3. Укладка складской паллеты.
4. Классификация людей на видео по вектору признаков

Критерии оценивания

Оценка отлично 10 баллов - выставляется студенту, показавшему всесторонние, систематизированные, глубокие знания учебной программы дисциплины, проявляющему интерес к данной предметной области, продемонстрировавшему умение уверенно и творчески применять их на практике при решении конкретных задач, свободное и правильное обоснование принятых решений.

Оценка отлично 9 баллов - выставляется студенту, показавшему всесторонние, систематизированные, глубокие знания учебной программы дисциплины и умение уверенно применять их на практике при решении конкретных задач, свободное и правильное обоснование принятых решений.

Оценка отлично 8 баллов - выставляется студенту, показавшему всесторонние, систематизированные, глубокие знания учебной программы дисциплины и умение уверенно применять их на практике при решении конкретных задач, правильное обоснование принятых решений, с некоторыми недочетами.

Оценка хорошо 7 баллов - выставляется студенту, если он твердо знает материал, грамотно и по существу излагает его, умеет применять полученные знания на практике, но недостаточно грамотно обосновывает полученные результаты.

Оценка хорошо 6 баллов - выставляется студенту, если он твердо знает материал, грамотно и по существу излагает его, умеет применять полученные знания на практике, но допускает в ответе или в решении задач некоторые неточности.

Оценка хорошо 5 баллов - выставляется студенту, если он в основном знает материал, грамотно и по существу излагает его, умеет применять полученные знания на практике, но допускает в ответе или в решении задач достаточно большое количество неточностей.

Оценка удовлетворительно 4 балла - выставляется студенту, показавшему фрагментарный, разрозненный характер знаний, недостаточно правильные формулировки базовых понятий, нарушения логической последовательности в изложении программного материала, но при этом он освоил основные разделы учебной программы, необходимые для дальнейшего обучения, и может применять полученные знания по образцу в стандартной ситуации.

Оценка удовлетворительно 3 балла - выставляется студенту, показавшему фрагментарный, разрозненный характер знаний, допускающему ошибки в формулировках базовых понятий, нарушения логической последовательности в изложении программного материала, слабо владеет основными разделами учебной программы, необходимыми для дальнейшего обучения и с трудом применяет полученные знания даже в стандартной ситуации.

Оценка неудовлетворительно 2 балла - выставляется студенту, который не знает большей части основного содержания учебной программы дисциплины, допускает грубые ошибки в формулировках основных принципов и не умеет использовать полученные знания при решении типовых задач.

Оценка неудовлетворительно 1 балл - выставляется студенту, который не знает основного содержания учебной программы дисциплины, допускает грубейшие ошибки в формулировках базовых понятий дисциплины и вообще не имеет навыков решения типовых практических задач.

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

При проведении устного дифференцированного зачета обучающемуся предоставляется 30 минут на подготовку. Опрос обучающегося по билету на устном дифференцированном зачете не должен превышать одного астрономического часа.