

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

УТВЕРЖДАЮ

Проректор по учебной работе и
образовательным инновациям

«30» сентября 2020 г. 011.34рок

Регистрационный № УД- 8928 уч.

МАШИННОЕ ОБУЧЕНИЕ

Учебная программа учреждения высшего образования
по учебной дисциплине для специальности:

1-31 80 03 Математика и компьютерные науки

профилизация Компьютерная математика и системный анализ

2020 г.

Учебная программа составлена на основе образовательного стандарта ОСВО 1-31 80 03-2019 и учебных планов: № G313-090/уч., № G31-049/уч. от 11.04.2019.

СОСТАВИТЕЛИ:

Л.Л. Голубева, доцент кафедры дифференциальных уравнений и системного анализа Белорусского государственного университета, кандидат физико-математических наук, доцент.

В.А. Воробей, ассистент кафедры дифференциальных уравнений и системного анализа Белорусского государственного университета.

РЕЦЕНЗЕНТ:

А.И. Смольская, инженер-программист ООО "ПЦИ Солюшенс".

РЕКОМЕНДОВАНА К УТВЕРЖДЕНИЮ:

Кафедрой дифференциальных уравнений и системного анализа БГУ
(протокол № 9 от 14.05.2020);

Научно-методическим советом БГУ
(протокол № 5 от 17.06.2020).

Зав. кафедрой дифференциальных уравнений
и системного анализа, профессор



В. И. Громак

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Целью дисциплины «Машинное обучение» является подготовка специалистов, способных использовать фундаментальные математические знания в качестве основы при выполнении прикладных исследований в области обработки данных и искусственного интеллекта.

Задачи учебной дисциплины:

- формирование у магистрантов способностей самостоятельно разработки алгоритмов решения задач и их анализа;
- развитие навыков применения теоретических знаний при решении прикладных задач;
- использование математических и компьютерных методов исследований при анализе современных естественнонаучных, экономических, социально-политических процессов;
- приобретение способностей самостоятельного расширения компьютерных математических знаний и их использованием при анализе математических моделей широкого круга прикладных задач.

Место учебной дисциплины в системе подготовки специалиста с высшим образованием (магистра).

Учебная дисциплина является дисциплиной компонента учреждения высшего образования и входит в состав **модуля** «Анализ данных».

Дисциплина «Машинное обучение» посвящена изучению класса методов искусственного интеллекта, способных обучаться. Для построения таких методов используются средства математической статистики, теории вероятностей, теории графов, численных методов, методов оптимизации, искусственных нейронных сетей, различные техники работы с данными в цифровой форме, связанные с извлечением и интеллектуальным анализом данных. Поэтому преподавание данной дисциплины связано с перечисленными дисциплинами, а также с дисциплинами «Эволюционные алгоритмы», «Моделирование и оптимизация», «Современные алгоритмы в теории информации» и «Прикладной искусственный интеллект».

Требования к компетенциям

Освоение учебной дисциплины «Машинное обучение» должно обеспечить формирование следующей **специализированной компетенции**:

СК-4. Быть способным эффективно использовать алгоритмы обработки данных и нейронные сети.

В результате изучения учебной дисциплины обучающийся должен:

знать:

- типологию задач обучения по прецедентам;
- базовые понятия теории машинного обучения;
- подходы и методы к решению задач машинного обучения;
- принципы устройства, обучения и работы нейронных сетей;
- основные методы глубокого обучения;

уметь:

- использовать современные методы компьютерного моделирования для исследования информационных систем и анализа данных;
- находить в представленных данных скрытые естественные закономерности или внутренние структуры, которые генерируют понимание и помогают принимать лучшие решения и прогнозы;
- проектировать, обучать и использовать искусственные нейронные сети для решения практических задач;
- использовать различные алгоритмы машинного обучения для решения практических задач;
- делать оценки и сравнивать качество обучения и функционирования различных моделей, построенных на базе искусственных нейронных сетей и других алгоритмов;
- самостоятельно расширять компьютерные математические знания с дальнейшим их использованием при построении и анализе математических и компьютерных моделей широкого круга теоретических и прикладных задач;

владеть:

- методами построения и обучения искусственных нейронных сетей;
- методами построения других алгоритмов машинного обучения;
- методами исследований и решения проблем извлечения, анализа и обработки данных с использованием математических компьютерных приложений;
- языками программирования и пакетами для машинного обучения.

Структура учебной дисциплины

Форма получения высшего образования очная (дневная) и заочная.

Дисциплина изучается в третьем семестре. Всего на изучение учебной дисциплины «Машинное обучение» отведено:

– для очной формы получения высшего образования – 206 часов, в том числе 72 аудиторных часа, из них: лекции – 24 часа (из них: 10 часов ДО), лабораторные занятия – 16 часов (из них: 8 часов ДО), семинарские занятия – 16 часов (из них: 16 часов ДО), управляемая самостоятельная работа – 16 часов.

– для заочной формы получения высшего образования – 206 часов, в том числе 16 аудиторных часов, из них 6 часов лекции, 6 часов лабораторные занятия, 4 часа – семинарские занятия.

Трудоемкость учебной дисциплины составляет 6 зачетных единиц.

Формой текущей аттестации по учебной дисциплине является экзамен.

СОДЕРЖАНИЕ УЧЕБНОГО МАТЕРИАЛА

Раздел 1. Задачи обучения по прецедентам

Тема 1.1. Введение в машинное обучение (2 часа)

AI vs ML vs DL. Искусственный интеллект (Artificial Intelligence), машинное обучение (Machine Learning), глубокое обучение (Deep Learning) и искусственные нейронные сети (Artificial Neural Networks). Определения, основные понятия машинного обучения (МО). Задачи интеллектуального анализа данных (Data Mining). Большие данные (Big Data).

Тема 1.2. Задачи обучения по прецедентам (2 часа)

Задачи обучения по прецедентам. Типология задач. Постановка задачи обучения по прецедентам. Объект, признаковое описание объекта. Множество объектов, обучающее множество. Матрица «объекты-признаки». Множество допустимых ответов. Модель алгоритмов восстанавливаемой зависимости. Метод обучения. Функция ошибок. Функционал качества. Метод минимизации эмпирического риска. Обобщающая способность ИНС. Проблема переобучения.

Тема 1.3. Подходы и методы к задачам машинного обучения (2 часа)

Процесс решения задач методами МО. Подходы и методы МО: обучение с учителем и обучение без учителя. Типология задач МО, выбор алгоритма в зависимости от типа решаемой задачи.

Тема 1.4. Задачи обучения с учителем (4 часа)

Задачи классификации. Постановка задачи. Бинарная и мультиклассовая классификация. Линейная разделимость и неразделимость классов. Алгоритмы классификации: логистическая регрессия (Logistic Regression), метод ближайших соседей (k Nearest Neighbor, k NN), метод опорных векторов (Support Vector Machine, SVM), наивный байесовский классификатор (Naive Bayes Classifier, NBC), дерево принятия решений (Decision Tree), алгоритмические композиции (Bagging and Boosting).

Задачи регрессии. Постановка задачи. Алгоритмы регрессии: линейная и нелинейная регрессии (Linear Regression, Nonlinear Regression), метод опорных векторов (Support Vector Machine, SVM-Regression), обобщенная линейная модель (Generalized Linear Model), дерево принятия решений (Regression Tree).

Совершенствование моделей. Выбор и преобразование признаков, настройка гиперпараметров.

Тема 1.5. Задачи обучения без учителя (2 часа)

Задачи кластеризации. Постановка задачи. Типы кластерных структур. Меры сходства в задачах кластеризации. Типы метрик. Жесткая и мягкая кластеризации. Алгоритмы МО для задач кластеризации.

Нейронные сети Кохонена, архитектура и обучение. Слой Кохонена. Самоорганизующаяся карта Кохонена, архитектура и правила обучения.

Обзор: Сети векторного квантования *Ivqnet*.

Раздел 2. Глубокое обучение

Тема 2.1. Искусственные нейронные сети (4 часа)

Искусственные нейронные сети (ИНС). Основные понятия и определения теории ИНС. Биологический прототип нейрона. Математическая модель нейрона. Области применения ИНС. Классификация ИНС. Этапы построения ИНС. Компьютерная модель нейрона и архитектура ИНС. Однослойные и многослойные ИНС. Неглубокие (Shallow Networks) и глубокие (Deep Networks) нейронные сети.

Обучение нейронных сетей. Градиент, метод градиентного спуска. Обучение многослойных ИНС методом обратного распространения ошибки (backpropagation).

Тема 2.2. Сверточные нейронные сети (4 часа)

Свертка. Свертка функций. Свертка изображений. Сверточные вычисления.

Сверточные нейронные сети (СНС). Биологический прототип СНС. Задачи СНС. Архитектура СНС: блок выделения признаков и блок принятий решений. Сверточный слой. Одноканальная и многоканальная свертки. Ядро, фильтр, коллекция фильтров. Функции активации. Пулинг. Batch-нормализация. Полносвязный слой. Обратное распространение ошибки *back-propagation*.

Модификации СНС. Использование предобученных нейронных сетей для решения задачи классификации изображений (Transfer Learning). Расширение обучающего датасета с использованием техники аугментации изображений (Image Augmentation).

Тема 2.3. Рекуррентные нейронные сети (2 часа)

Рекуррентные нейронные сети (РНС). Назначение, структура РНС. Проблема долгосрочных зависимостей.

Сети LSTM. Назначение, структура сетей LSTM. Структура модуля сети LSTM. Типы информации, проходящие через модули сети LSTM. Вентиль, типы вентиляей. Пошаговый алгоритм работы сети LSTM. Вариации сетей LSTM. Архитектура LSTM-сетей в зависимости от типа решаемых задач (классификации, регрессии, видео-классификации).

Тема 2.4. Практические задачи (2 часа)

Хранилища данных. Типичные задачи машинного обучения. Эксперименты на реальных данных и на модельных данных.

УЧЕБНО-МЕТОДИЧЕСКАЯ КАРТА УЧЕБНОЙ ДИСЦИПЛИНЫ

Дневная форма получения образования с применением дистанционных образовательных технологий

Номер раздела, темы	Название раздела, темы	Количество аудиторных часов					Количество часов УСР	Форма контроля знаний
		Лекции	Практические занятия	Семинарские занятия	Лабораторные занятия	Иное		
1	2	3	4	5	6	7	8	9
	Машинное обучение	24		16	16		16	
1.	Задачи обучения по прецедентам	12		6	8		10	
1.1.	Введение в машинное обучение	2						Собеседование
1.2.	Задачи обучения по прецедентам	2		1 (ДО)	2		5	Письменный отчет с устной защитой, устный опрос, доклад на семинарском занятии
1.3.	Подходы и методы к задачам машинного обучения	2 (ДО)		1 (ДО)	2		5	Письменный отчет с устной защитой, устный опрос, доклад на семинарском занятии
1.4.	Задачи обучения с учителем	4		2 (ДО)	2 (ДО)			Письменный отчет с устной защитой, устный опрос, доклад на семинарском занятии

1.5.	Задачи обучения без учителя	2 (ДО)		2 (ДО)	2 (ДО)			Письменный отчет с устной защитой, устный опрос, доклад на семинарском занятии
2.	Глубокое обучение	12		10	8		6	
2.1.	Искусственные нейронные сети	4		2 (ДО)	2 (ДО)			Письменный отчет с устной защитой, устный опрос, доклад на семинарском занятии
2.2.	Сверточные нейронные сети	4 (ДО)		4 (ДО)	2		6	Письменный отчет с устной защитой, устный опрос, доклад на семинарском занятии
2.3.	Рекуррентные нейронные сети	2 (ДО)		2 (ДО)	2 (ДО)			Письменный отчет с устной защитой, устный опрос, доклад на семинарском занятии
2.4.	Практические задачи	2		2 (ДО)	2			Письменный отчет с устной защитой, устный опрос, доклад на семинарском занятии

УЧЕБНО-МЕТОДИЧЕСКАЯ КАРТА УЧЕБНОЙ ДИСЦИПЛИНЫ

Заочная форма получения образования

Номер раздела, темы	Название раздела, темы	Количество аудиторных часов					Количество часов УСР	Форма контроля знаний
		Лекции	Практические занятия	Семинарские занятия	Лабораторные занятия	Иное		
1	2	3	4	5	6	7	8	9
	Машинное обучение	6		4	6			
1.	Задачи обучения по прецедентам	3		2	4			
1.1.	Введение в машинное обучение							Собеседование
1.2.	Задачи обучения по прецедентам	1			1			Письменный отчет с устной защитой, устный опрос
1.3.	Подходы и методы к задачам машинного обучения	1			1			Письменный отчет с устной защитой, устный опрос
1.4.	Задачи обучения с учителем	0.5		1	1			Письменный отчет с устной защитой, устный опрос, доклад на семинарском занятии
1.5.	Задачи обучения без учителя	0.5		1	1			Письменный отчет с устной защитой, устный опрос, доклад на семинарском занятии
2.	Глубокое обучение	3		2	2			

2.1.	Искусственные нейронные сети	1		0.5	0.5			Письменный отчет с устной защитой, устный опрос, доклад на семинарском занятии
2.2.	Сверточные нейронные сети	1		0.5	0.5			Собеседование, доклад на семинарском занятии
2.3.	Рекуррентные нейронные сети	1		0.5	0.5			Собеседование, доклад на семинарском занятии
2.4.	Практические задачи			0.5	0.5			Письменный отчет с устной защитой, устный опрос, доклад на семинарском занятии

ИНФОРМАЦИОННО-МЕТОДИЧЕСКАЯ ЧАСТЬ

Перечень основной литературы

1. *Флах, П.* Машинное обучение. Наука и искусство построения алгоритмов, которые извлекают знания из данных. М.: ДМК Пресс, 2015. 400 с.
2. *Хайкин, С.* Нейронные сети: Полный курс, 2-е издание. М.: «Вильямс», 2006. 1104 с.
3. *Haykin, S.* Neural Networks and Learning Machines Third Edition. Copyright © 2009 by Pearson Education, Inc., Upper Saddle River, New Jersey 07458, 2009. 938 p.
4. *Траск, Э.* Грокаем глубокое обучение (Серия «Библиотека программиста»). СПб.: Питер, 2019. 352 с.: ил.
5. *Daumé, H.* A course in Machine Learning. http://ciml.info/dl/v0_9/ciml-v0_9-all.pdf
6. *Воронцов, К.В.* Машинное обучение. Курс лекций. <http://www.machinelearning.ru>
7. *Гудфеллоу, Я.* Глубокое обучение / Я. Гудфеллоу, И. Бенджио, А. Курвилль. М.: ДМК Пресс, 2017. 652 с.
8. *Николенко, С.* Глубокое обучение. Погружение в мир нейронных сетей (Серия «Библиотека программиста»). / С. Николенко. А. Кадури, Е. Архангельская. СПб.: Питер, 2018. 480 с.: ил.

Перечень дополнительной литературы

1. *Beale, M.D.* Deep Learning Toolbox™. Getting Started Guide / M.D. Beale, M.T. Hagan, H.D. Demuth. © COPYRIGHT 1992–2020 by The MathWorks, Inc., 2020. 148 p.
2. *Beale, M.D.* Deep Learning Toolbox™. User's Guide / M.D. Beale, M.T. Hagan, H.D. Demuth. © COPYRIGHT 1992–2020 by The MathWorks, Inc., 2020. 2626 p.
3. *Мюллер, А.* Введение в машинное обучение с помощью Python. Руководство для специалистов по работе с данными / А. Мюллер, С. Гвидо. М.: «Вильямс», 2017. 480 с.

Перечень рекомендуемых средств диагностики и методика формирования итоговой оценки

Согласно Положению о рейтинговой системе оценки знаний обучающихся по дисциплине в БГУ контроль знаний по дисциплине «Машинное обучение» происходит в форме текущего контроля и текущей аттестации.

Итоговая оценка по дисциплине «Машинное обучение» формируется на основе 3-х документов:

1. Правила проведения аттестации (Постановление № 53 от 29.05.2012 г.).
2. Положение о рейтинговой системе оценки знаний обучающихся по учебной дисциплине в БГУ (Приказ ректора № 189-ОД от 31.03.2020 г.).
3. Критерии оценки знаний и компетенций студентов по 10-балльной шкале.

Оценка текущего контроля знаний по дисциплине «Машинное обучение» формируется в результате систематической проверки знаний обучающихся во время занятий и по итогам их самостоятельной работы. Текущий контроль знаний проходит в форме опроса на лекциях, семинарских и лабораторных занятиях, во время устной защиты отчета по лабораторным работам, выполняемым в учебной лаборатории и самостоятельно вне аудитории, а также во время выступления с докладом на семинарских занятиях. Выдаваемые обучающимся задания составляются согласно содержанию учебного материала. Во время самостоятельной работы обучающийся выполняет задания, полученные на лабораторных занятиях, а также изучает рекомендуемую литературу. При защите лабораторных работ оценивается полнота ответа, аргументация выбранных решений, последовательность и оригинальность изложения материала, оригинальность кода, корректность оформления, самостоятельность выполнения заданий. Для совершенствования способностей учиться самостоятельно обучающимся выдаются темы докладов, с которыми они выступают на семинарских занятиях. Формирование оценки текущего контроля знаний:

- отчеты по лабораторным работам – 40 %;
- доклады на семинарских занятиях – 40 %;
- опросы на лекциях – 20%.

Формой текущей аттестации по дисциплине «Машинное обучение» учебным планом предусмотрен экзамен. Экзамен по дисциплине проходит в устной форме.

Итоговая оценка по дисциплине «Машинное обучение» рассчитывается на основе оценок текущего контроля знаний и текущей аттестации (экзаменационной оценки) с учетом их весовых коэффициентов. На основе Положения о рейтинговой системе оценки знаний обучающихся по учебной дисциплине в БГУ по дисциплинам, закрепленным за кафедрой ДУиСА, установлены следующие весовые коэффициенты, определяющие вклад текущего контроля и текущей аттестации в итоговую оценку знаний обучающихся: весовой коэффициент текущего контроля – 0.4, весовой коэффициент текущей аттестации – 0.6 (Протокол №8 от 16.04.2020 г. заседания кафедры ДУиСА БГУ).

Примерный перечень заданий для управляемой самостоятельной работы студентов

Тема 1.2. Задачи обучения по прецедентам (5 часов)

Лабораторная работа: Прогнозирование времени нахождения животного в приюте

Цель: построить несколько моделей для определения времени, за которое животные могут забрать новых хозяев: на основе набора числовых и категориальных признаков, а также текста профиля животного.

План:

1. Подготовка данных
2. Моделирование на основе числовых признаков
3. Обработка текстовых данных
4. Построение прогнозов

За основу взяты данные с соревнования на Kaggle (<https://www.kaggle.com/c/petfinder-adoption-prediction/data>), там же приведено их описание. Целевая переменная – AdoptionSpeed. Метрика качества – Accuracy.

Перед выполнением работы рекомендуется ознакомиться со следующими материалами:

1. Python Numpy Tutorial (<http://cs231n.github.io/python-numpy-tutorial/>)
2. Numpy quickstart (<https://docs.scipy.org/doc/numpy/user/quickstart.html>)
3. Matplotlib tutorial (https://matplotlib.org/users/pyplot_tutorial.html)
4. Python Plotting With Matplotlib (<https://realpython.com/python-matplotlib-guide/>)
5. Pandas tutorials (<https://pandas.pydata.org/pandas-docs/stable/tutorials.html>)
6. SKLearn: Working With Text Data (https://scikit-learn.org/0.21/tutorial/text_analytics/working_with_text_data.html)

Задание 1. Установка дополнительных пакетов

Для установки LightGBM необходимо запустить Anaconda Prompt от имени администратора и выполнить команду `conda install -c conda-forge lightgbm`.

Задание 2. Подготовка данных

1. Скачайте архив с данными (<https://drive.google.com/file/d/1B3ZOIksQoIOWVRRJocPtqfI0XBvFI8LT/view?usp=sharing>).
2. Проведите разведочный анализ данных: постройте гистограммы числовых признаков, найдите наиболее популярные значения для категориальных признаков, посмотрите статистику по тексту профиля и др.
3. Разбейте исходные данные на матрицу признаков и целевую переменную.

Задание 3. Моделирование на основе числовых и категориальных признаков

1. Постройте новые числовые признаки на основе категориальных (рекомендуется использовать функцию `pd.get_dummies()`).

Замечание: для категориальных признаков, которые содержат большое количество значений, может иметь смысл использовать только наиболее популярные значения.

2. Создайте матрицу, на которой будет обучаться модель: для этого объедините числовые признаки в исходных данных, а также признаки, построенные в предыдущем пункте.

3. Подумайте, какие признаки можно дополнительно использовать для обучения модели, и при необходимости добавьте их к матрице признаков.

4. Стандартизируйте данных при помощи метода `sklearn.preprocessing.StandardScaler`.

5. При помощи кросс-валидации оцените работу следующих моделей и прокомментируйте результаты: `LogisticRegression`, `SVC` (SVM), `RandomForest`, `XGBoost/LightGBM`. Для проведения кросс-валидации рекомендуется использовать `sklearn.model_selection.cross_val_score` (обратите внимание на параметры функции).

6. Работа каких алгоритмов зависит от стандартизации данных?

7. Для древесных алгоритмов постройте графики важности признаков (см. свойство `feature_importances_` у обученных моделей).

8. Попытайтесь улучшить результат модели, которая показала себя лучше остальных при кросс-валидации, например, при помощи подбора гиперпараметров, создания/удаления признаков и др.

Задание 4. Моделирование на основе текста профиля

1. Используя `CountVectorizer`, постройте графики наиболее часто используемых выражений длиной до 3-х слов для признака `Description` тренировочного и тестового набора данных.

2. Какие классификаторы имеет смысл использовать в первую очередь?

3. Зафиксируйте классификатор и обучите модели на основе `CountVectorizer`, `TfidfVectorizer` на наборах слов и n-граммах символов (размер окна выберите самостоятельно). Какая модель работает лучше остальных?

4. Выберите оптимальное, на ваш взгляд, сочетание шагов препроцессинга и классификатора, и сравните результат с лучшей моделью, построенной во 2-м задании. Какой можно сделать вывод?

5. Попробуйте улучшить результат модели из предыдущего пункта, подбирая ее гиперпараметры.

Задание 5. Построение прогнозов

1. Используя результаты предыдущих пунктов, предскажите значение целевой переменной для тестовых и данных. Например, вы можете усреднить предсказание моделей, построенных ранее, или использовать

предсказание на тексте как еще один числовой признак и обучить одну модель (имейте ввиду, что в таком случае модель должна делать предсказания на данных, которые не использовались в процессе обучения: удобнее это сделать при помощи функции `sklearn.model_selection.cross_val_predict`).

2. Сохраните результат в файле `submission_ФамилияИмя.csv`.

Форма контроля – письменный отчет с устной защитой, устный опрос.

Тема 1.3. Подходы и методы к задачам машинного обучения (5 часов)

Лабораторная работа: Обработка временных рядов

Цель: научиться работать с временными рядами. Данные представляют собой курс иностранной валюты (евро) за последние несколько лет. Задача – проанализировать эти данные и научиться прогнозировать будущие значения. Метрика качества – Mean Absolute Error.

План:

1. Сглаживание временного ряда методом скользящего среднего
2. Выделение тренда, сезонности, шума
3. Подготовка данных для кросс-валидации
4. Построение матрицы признаков
5. Прогнозирование на основе матрицы признаков: линейные и древесные модели
6. Построение модели авторегрессии
7. Удаление тренда
8. Построение ARIMA-модели и подбор ее параметров
9. Построение прогнозов

Установите библиотеку для работы с временными рядами Statsmodels (<https://www.statsmodels.org/stable/index.html>) и скачайте архив с данными (<https://drive.google.com/file/d/1fdjCswQ64LnhdNPiATE5HbmSZuKDsCSf/vi ew?usp=sharing>).

Перед выполнением работы рекомендуется ознакомиться со следующими материалами:

1. Анализ временных рядов с помощью Python (<https://habr.com/ru/company/ods/blog/327242/>).
2. Анализ временных рядов с помощью Python (2) (<https://habr.com/ru/post/207160/>).

Задание 1. Разведочный анализ данных

1. Загрузите данные.
2. Используя метод скользящего среднего, сгладьте временной ряд. Ширину окна выбирайте равной одной недели, месяцу, кварталу.
3. На одной графике изобразите исходный временной ряд, а также его сглаженные версии. Есть ли у ряда тренд, сезонность?

4. Является ли заданный временной ряд стационарным? Почему? Проверьте своё предположение с помощью теста Дики-Фуллера (https://ru.wikipedia.org/wiki/Тест_Дики_—_Фуллера).

Задание 2. Кросс-валидация

Реализуйте функцию `cross_val_on_time_series`, которая разбивает обучающие данные на группы, затем для каждой из них обучает переданный регрессор, предсказывает значения для тестовых данных и возвращает список из значений по целевой метрике для предсказанных и реальных значений. Функция должна иметь следующие входные параметры:

- `regressor` – регрессор, для которого реализованы методы `fit` и `predict`
- `X` – матрица признаков для примеров (упорядоченных по времени)
- `y` – значения целевой переменной (упорядоченные по времени)
- `n_splits` – количество разбиений для временного ряда
- `metric` – целевая метрика.

Примечание: для разбиения данных на группы рекомендуется использовать `TimeSeriesSplit` из `sklearn.model_selection`.

Задание 3. Прогнозирование с использованием матрицы признаков

1. Напишите функцию `create_time_series_features`, которая создает матрицу признаков для переданного временного ряда на основе времени наблюдения целевой переменной.
2. Создайте матрицу признаков для заданного временного ряда.
3. Сравните производительность следующих алгоритмов, используя кросс-валидацию (нужно ли использовать функцию из предыдущего задания?): `SGDRegressor`, `LinearRegression`, `RandomForestRegressor`, `XGBRegressor`/`LGBMRegressor`.
4. Обучите бустинг-модель и постройте график важности признаков (`feature_importances`).
5. Разбейте временной ряд на тренировочный и тестовый так, чтобы в тестовой части были данные за 2 последних месяца. Обучите по одной линейной и древесной модели, которые показали себя лучше всего в 3-м пункте, и спрогнозируйте значения для тестовых данных и тренировочных данных.
6. На одном графике изобразите исходный временной ряд и предсказания всех моделей, построенных в предыдущем пункте. Используйте разные цвета для истинных и предсказанных значений. Прогнозы разных моделей также изображайте разными цветами. Добавьте на график легенду. Выделите участок, на котором изображаются предсказания для тестовых данных (например, сделайте фон темнее).
7. Попробуйте улучшить работу алгоритмов, подбирая параметры моделей или изменяя исходный ряд, например, при помощи преобразования Бокса-Кокса (http://www.machinelearning.ru/wiki/index.php?title=Метод_Бокса-Кокса).

Задание 4. Авторегрессия

1. Постройте модель авторегрессии и, используя результаты 3-го задания, оцените работу алгоритма при помощи кросс-валидации.
 2. Удалите из ряда тренд и вновь проведите кросс-валидацию модели. Прокомментируйте результаты, сравните их с результатами, полученными во 2-м пункте лабораторной работы.
- Примечание: тренд можно удалять явно, обучая функцию от времени, например, линейную квадратичную и т.д., или дифференцируя исходный ряд (в этом случае подумайте, как строить прогноз для исходного ряда).
3. Действуя по аналогии с пунктом 6 задания 3, предскажите значения для всего временного промежутка и добавьте их на график с другими моделями.
 4. Попробуйте улучшить качество модели.

Задание 5. Алгоритм ARIMA

1. Постройте модель ARIMA и подберите её параметры (например, основываясь на информационном критерии Акаике AIC (https://ru.wikipedia.org/wiki/Информационный_критерий_Акаике)).
2. Имеет ли смысл для данного временного ряда использовать модель SARIMA? Почему?
3. Оцените качество построенной модели.
4. Действуя по аналогии с предыдущими пунктами, сравните результаты разных моделей, построьте совместный график для предсказанных значений на всем временном промежутке.

Задание 6. Построение прогнозов

1. Используя результаты предыдущих пунктов, предскажите значения временного ряда на период с 28.10.2019 по 11.11.2019.
2. Сохраните результат в файле submission_ФамилияИмя.csv. В файле должно быть два столбца: Date – дата, для которой строится прогноз, в формате Year-Month-Day (например, 2019-01-31); Rate – значение целевой переменной (см. файл с данными, а также пример записи предсказаний https://nbviewer.jupyter.org/github/uladzislau-varabei/machine-learning-course-bsu-mm-2019-autumn/blob/master/LB_02/Submission_example.ipynb).

Форма контроля – письменный отчет с устной защитой, устный опрос.

Тема 2.2. Сверточные нейронные сети (6 часов)

Лабораторная работа: Сверточные нейронные сети

Цель: научиться предсказывать класс изображений из датасет CIFAR-10. Целевая метрика – accuracy.

План:

1. Подготовка данных
2. Аугментация изображений
3. Создание модели нейронной сети

4. Выбор функции потерь и оптимизатора
5. Обучение модели
6. Контроль над ходом обучения модели
7. Эксперименты с архитектурой модели
8. Выбор лучшей модели

Задания:

1. Скачайте версию данных для Python:
<https://www.cs.toronto.edu/~kriz/cifar.html>
2. Загрузите данные и сохраните изображения в png-файлы, предварительно изменив размеры массива на (32, 32, 3) для каждого конкретного примера. Создайте отдельный словарь вида {filename:label} и сохраните его в виде json-файла (или создайте DataFrame и сохранить его в csv-файл).
3. Загрузите и отобразите несколько из созданных изображений. Сколько примеров доступно для каждого класса в тренировочном множестве?
4. Разделите файлы для обучения на два множества: тренировочное (80%) и валидационное (20%), используя `sklearn.model_selection.train_test_split` с фиксированным параметром `random_state`.
5. Закодируйте классы изображения one-hot векторами (например, при помощи `keras.utils.to_categorical` или `sklearn.preprocessing.OneHotEncoder`).
6. Напишите генератор, который на входе принимает список файлов и размер батча, а на выходе выдает батч из загруженных изображений заданного размера и one-hot вектора, соответствующие их классам.
7. Напишите функцию, которая будет случайным образом аугментировать батч изображений. Приведите примеры ее работы. В генератор из предыдущего пункта добавьте параметр, отвечающий за необходимость аугментирования изображений.
8. Создайте модель нейронной сети, которая состоит из двух свёрточных блоков, за которыми идут несколько полносвязных слоёв с батч-нормализацией и дропаутом.
 - Свёрточный блок должен иметь вид: Conv2D -> Conv2D -> MaxPooling2D.
 - Для свёрточных слоев использовать число фильтров равное 32/64/128/256.
 - Выход последнего свёрточного слоя необходимо распрямить в вектор.
 - Для полносвязных слоев используйте 64-512 нейронов.
 - Рекомендуются использовать 1-3 полносвязных слоя.
 - Для свёрточных и полносвязных слоёв можно, например, начать с функции активации `relu`.
 - На выходе сети использовать softmax-слой с количеством нейронов, равным количеству классов.
9. Задайте функцию потерь и оптимизатор модели.

10. Обучите модель, используя генератор, написанный в 6-м задании (изображения из валидационного набора аугментировать не нужно).
11. Постройте совместные графики метрики и функции потерь для тренировочных и валидационных данных.
12. Предскажите результаты для тестового множества и вычислите значений целевой метрики. Какое качество показывает модель на разных классах?
13. Поэкспериментируйте с архитектурой сети и процессом её обучения:
 - Замените MaxPooling2D в сверточных блоках на AveragePooling2D. Как изменился результат/время обучения?
 - Обучите сеть без аугментации изображений. Как сильно она влияет на результат?
 - Измените количество нейронов в свёрточных слоях. Лучше их постепенно увеличивать/уменьшать?
 - Поэкспериментируйте с параметрами свёрточных слоев.
 - Примените другие функции активации в свёрточных/полносвязных слоях.
 - Уменьшите/увеличьте количество полносвязных слоев, как это влияет на качество?
 - Увеличивается ли точность сети при добавлении батч-нормализации/дропаута в свёрточные блоки? Насколько это замедляет процесс обучения?
14. Опишите модель, которая показала лучший результат.

Форма контроля – письменный отчет с устной защитой, устный опрос.

Описание инновационных подходов и методов к преподаванию учебной дисциплины

При организации образовательного процесса используется **эвристический подход**, который предполагает демонстрацию многообразия решений большинства профессиональных задач и жизненных проблем.

При организации образовательного процесса используется **практико-ориентированный подход**, который предполагает освоение содержания через решения практических задач.

При организации образовательного процесса **используются методы и приемы развития критического мышления**, которые представляют собой систему, формирующую навыки работы с информацией в процессе чтения и письма; понимания информации как отправного, а не конечного пункта критического мышления.

Методические рекомендации по организации самостоятельной работы обучающихся

Для организации самостоятельной работы обучающихся по учебной дисциплине рекомендовано разместить на образовательном портале или сайте кафедры учебно-методические материалы: курсы лекций и лабораторные практикумы, методические указания к лабораторным занятиям, вопросы для подготовки к экзамену, перечень рекомендуемой литературы, информационные ресурсы.

Самостоятельная работа обучающегося включает в себя работу с учебной литературой по заданным разделам дисциплины, поиск в Интернете новейшей учебной и научной информации в указанных областях знаний и знакомство с ней, а также выполнение задач, поставленных на занятиях.

ПРОТОКОЛ СОГЛАСОВАНИЯ УЧЕБНОЙ ПРОГРАММЫ УВО

Название учебной дисциплины, с которой требуется согласование	Название кафедры	Предложения об изменениях в содержании учебной программы УВО по учебной дисциплине	Решение, принятое кафедрой, разработавшей учебную программу (с указанием даты и номера протокола) ¹
Эволюционные алгоритмы	Кафедра дифференциальных уравнений и системного анализа	нет	Вносить изменения не требуется (№ 9 от 14.05.2020)

¹ При наличии предложений об изменениях в содержании учебной программы УВО.

ДОПОЛНЕНИЯ И ИЗМЕНЕНИЯ
К УЧЕБНОЙ ПРОГРАММЕ ПО ИЗУЧАЕМОЙ УЧЕБНОЙ
ДИСЦИПЛИНЕ НА _____ / _____ УЧЕБНЫЙ ГОД

№ п/п	Дополнения и изменения	Основание

Учебная программа пересмотрена и одобрена на заседании кафедры
 _____ (название кафедры) (протокол № _____ от _____ 200__ г.)

Заведующий кафедрой
 доктор физ.-мат. наук, профессор _____
 (ученая степень, ученое звание) (подпись)

В.И. Громак
 (И.О.Фамилия)

УТВЕРЖДАЮ
 Декан факультета
 доктор физ.-мат. наук, доцент _____
 (ученая степень, ученое звание) (подпись)

С.М. Босяков
 (И.О.Фамилия)