Контроль по МО

Для контроля по практикам МО необходимо сдать и защитить 3 проекта по соответствующим темам «Классическое Машинное Обучение», «Рекомендательные системы» и «Основы Глубокого Обучения». Выполнять проекты можно в группах до 4 человек. Для каждого задания есть четкий бейзлайн, который необходимо полностью выполнить, в каких-то темах есть также дополнительная часть, направленная на расширение ваших практических навыков и знаний. Каждый проект необходимо защитить своей командой на семинарских занятиях. Вопросы по проектам можно задавать как на лекциях, так и на семинарах. При сдаче проектов могут быть заданы вопросы как по теоретической части, так и по практической, также могут быть вопросы не только по полученным результатам, но и по коду. Соответственно нужно уметь ориентироваться и объяснять написанный код. Также в коде должны присутствовать ваши пояснения по коду и комментарии по результатам. При копировании чужих материалов работа не засчитывается.

Тема «Классическое Машинное Обучение»

В качестве контроля по теме «Классическое МО» предлагается решить классическую задачу обучения с учителем по бинарной классификации. Данные и информацию по ним можно найти по ссылке, с примерами чужих решений вы также можете ознакомиться по вкладке Code в соревновании: https://www.kaggle.com/datasets/rashikrahmanpritom/heart-attack-analysis-prediction-dataset

Если коротко, то у вас в качестве признаков имеется некоторая информация по пациентам: общая (пол, возраст) и информация о здоровье (тип боли в груди, количество крупных сосудов и т.д.), а в качестве таргета необходимо научиться предсказывать вероятность сердечного приступа, после чего подобрать некое пороговое значение, которое предсказывает бинарный таргет - будет сердечный приступ или нет.

Бейзлайн выполнения задачи:

1. Обычно, когда мы хотим построить какую-либо модель, имеет смысл задуматься будут ли признаки релевантны для таргета. Соответственно нужно провести базовую аналитику по влиянию признаков как друг на друга, так и на целевую переменную. Здесь предлагается вам самим подумать как признаки этого датасета могут влиять на таргет. Например, нужно посмотреть корреляцию признаков между собой, различные сводные таблицы по влиянию признаков на таргет (средний возраст пациентов, у которых были сердечные приступы (таргет=1) и не было (таргет=0); среднее значение артериального давления в покое в зависимости от пола и целевого таргета, и т.д.), также можно посмотреть различные графики [2 балла]
2. Разделить выборку на обучающую и тестовую для проверки качества модели (обычно тестовая выборка размера 0.2 от общей). Тестовую выборку не трогать до последнего пункта, так как на ней мы пытаемся адекватно оценить качество модели. Для дальнейшего обучения и проверки на переобучение построенных моделей необходимо использовать кроссвалидацию
3. Сделать препроцессинг признаков [1 балл]
4. Построить качественную модель логистической регрессии (если какие-то признаки незначимы, то убрать их из модели; не забыть добавить bias (коэффициент смещения); вывести отчет по модели: p_value, t_st, критерий дарбина-уотсона и т.д (знать, что эти показатели демонстрируют)) [2 балла]
5. Построить качественную модель дерева и случайного леса, подобрать гиперпараметры моделей, чтобы модели не были переобучены (также если какие-то признаки не значимы, то их необходимо убрать из модели) [2 балла]
6. Сравнить качество полученных 3 моделей на тестовой выборке и также подобрать пороговые значения для моделей для перехода от вероятности к бинарному таргету (по предсказанной вероятности посмотреть ROC кривую, ROC-AUC, Precision-Recall кривую; по предсказанному таргету посмотреть Accuracy, Precision, Recall, F1-Score). При подборе порога в моделях подумайте, что в данном случае для нас

важнее из метрик Precision или Recall, поскольку цель модели научиться предсказывать приступы и предварительно отсекать группу риска, использую некоторые превентивные меры [3 балла]

Полученное решение необходимо оформить в юпитер-ноутбук с комментариями и заключением по каждому из пунктов, а также презентацию. Для защиты необходимо продемонстрировать полученные результаты и ответить на вопросы, в случае успешной защиты получаете максимальный балл - 10 баллов. Для успешных ответов на вопросы также необходимо знать в чем суть каждого из рассмотренных методов и уметь ориентироваться в своем решении, в том числе в коде.

Тема «Рекомендательные системы»

В проекте по RecSys мы рассматриваем стандартный датасет по рекомендательным системам MovieLens-1M, где на 3900 фильмов 6040 пользователей проставили примерно миллион рейтингов. Файлы и подробную информацию о датасете можно найти в readme: https://www.kaggle.com/datasets/odedgolden/movielens-1m-dataset?select=README

Если коротко, то у вас есть 3 файла: информация по фильмам (название фильма, жанр), информация по пользователям (пол, возраст, местоположение), информация по рейтингам (какой рейтинг поставил конкретный пользователь на конкретный фильм). Необходимо научиться предсказывать рекомендации пользователей.

Бейзлайн выполнения задачи:

1. Разделить выборку на обучающую и тестовую. Обосновать почему разделили выборку таким образом [1 балл]
2. В качестве базового решения попробовать взять в качестве рекомендации топ-10/20/50 популярных фильмов, посмотреть качество рекомендации на основе показателя accuracy для тестовой выборки (если рейтинг >= 4, считаем, что фильм понравился пользователю и его следует рекомендовать) [1 балла]

3. Попробовать рекомендации на основе KNN топ-10/20 ближайших соседей, посмотреть качество рекомендаций для тестовой выборки [2 балла]
4. Попробовать использовать метод als с implicit (значения рейтингов по фильму переходят в бинарные: 0 и 1) и explicit (рейтинг от 0 до 5) данными. Посмотреть на качество предсказаний на тестовой выборке [2 балла]
5. Подвести итоги по рассмотренным методам

Дополнительно:

1. Поэкспериментировать в рекомендациях на основе KNN с метрикой cosine, euclidian и т.д. Посмотреть как на тестовой выборке будет меняться показатель качества [1 балла]
2. Попробовать использовать метод lightfm для рекомендаций. Посмотреть на качество предсказаний на тестовой выборке [2 балла]
3. Посмотреть качество рекомендаций рассмотренных методов по метрике PRECISION@10/20/50 для каждого пользователя, и перейти от нее к MNAP@10/20/50 (Mean Normalized Average Precision), которая усреднена по всем пользователям [1 балл]

Полученное решение необходимо оформить в юпитер-ноутбук с комментариями и заключением по каждому из пунктов, а также презентацию. Для защиты необходимо продемонстрировать полученные результаты и ответить на вопросы, в случае успешной защиты получаете максимальный балл - 10 баллов . Для успешных ответов на вопросы также необходимо знать в чем суть каждого из рассмотренных методов и уметь ориентироваться в своем решении, в том числе в коде.

Полезные ссылки:

https://github.com/lyst/lightfm

https://github.com/benfred/implicit

https://medium.com/radon-dev/als-implicit-collaborative-filtering-5ed653ba39fe

https://github.com/abhilashhn1993/collaborative-filtering-using-ALS-for-movie-recommendation

Тема «Основы Глубокого Обучения»

В качестве проекта по основам ГО, необходимо выбрать один интересный для всей команды проект и защитить его. Требования по сдаче: необходимо предоставить юпитер-ноутбук и презентацию. При выступлении необходимо рассказать о своей задаче, о методе, который использовался в качестве решения и продемонстрировать полученные результаты. Также вам могут быть заданы различные вопросы, нужно разбираться и ориентироваться в сданном коде, поскольку решения по большинству проектов находятся в полезных ссылках. С вашей стороны нужно повторить результаты и разобраться в них.

Варианты проектов:

1. Взять датасет MovieLens-1M из задания выше и попробовать рассмотреть 3-5 методов из RECBOLE [https://recbole.io/], которые не были рассмотрены в задании выше. Также коротко рассказать в чем преимущества рассмотренных методов (graph-based, sequential-based и т.д.), проверить качество (MNAP, Accuracy) на тестовой выборке и подвести итоги [10 баллов]
2. Научиться распознавать людей из соревнования, более подробно изучить датасет можно по ссылке [Celeba]. Предполагается использовать либо обычную CNN, либо уже существующие архитектуры. Также после обучения нейронки, продемонстрировать качество распознания лиц на тестовой выборке. Рассказать о выбранном решении [10 баллов]
3. Классификация листьев по изображению. Более подробно датасет можно изучить по ссылке: [Leaf]. Необходимо научиться классифицировать листья по изображению, используя CNN и MLP, посмотреть качество классификации на тестовой выборке. Попробовать обучать с разными параметрами модель. Рассказать о решении, сравнить обучение с помощью CNN и MLP [10 баллов]
4. Классификация имен с помощью RNN. Разобраться, что такое RNN, как она устроена, рассказать как с помощью нее можно классифицировать

имена, проверить качество классификации. Данные находятся по ссылке [data] [10 баллов]

Полезные ссылки по 1 проекту:

https://github.com/RUCAIBox/RecBole

Полезные ссылки по 2 проекту:

https://github.com/ndb796/CelebA-HQ-Face-Identity-and-Attributes-Recognition-PyTorch

https://github.com/wowfun/face-recognition-on-celeba

Полезные ссылки по 3 проекту:

https://github.com/adl1995/leaf-classification

https://github.com/heytanay/leaf-disease-detection

https://www.kaggle.com/code/fatemehsharifi79/mlp-pytorch

Полезные ссылки по 4 проекту

https://pytorch.org/tutorials/intermediate/char_rnn_classification_tutorial.html