# ТЕХНОЛОГИЧЕСКАЯ КАРТА ЗАНЯТИЯ

**Тема занятия:** Пайплайн машинного обучения

**Аннотация к занятию:** в первой части занятия обучающиеся знакомятся со схемой пайплайна машинного обучения, обработкой данных и data leakage, обработкой признаков и выбросами, кросс-валидацией. Во второй части занятия закрепляют изученные термины в игровой форме.

**Цель занятия:** изучение схемы пайплайна машинного обучения, приобретение навыка обработки данных и оценки качества моделей с помощью кросс-валидации.

**Задачи занятия:**

- рассмотреть диаграмму пайплайна, изучить принцип разделения данных;
- способствовать приобретению навыка обработки данных (валидации, удаления утечки и обработки пропущенных значений, навыка обработки категориальных и численных признаков);
- изучить метод оценки качества модели — кросс-валидацию.

## Ход занятия

| Этап занятия | Время | Деятельность педагога | Комментарии, рекомендации для педагогов |
|---|---|---|---|
| **Организационный этап** | 2 мин. | Здравствуйте! Сегодня нас ждёт продуктивный день. Все готовы его начать? | Обеспечение полной готовности аудитории к работе на занятии |
| **Постановка цели и задач занятия. Мотивация учебной деятельности обучающихся** | 10 мин. | Давайте вспомним, как выглядит базовый пайплайн, то есть схема создания алгоритма машинного обучения?<br><br>**Возможный ответ учеников**<br>Это документ, визуализирующий процесс разработки продукта. Он представляет собой последовательность этапов, расположенных так, что конец предыдущего является началом следующего. Благодаря этому создается эффект производственного конвейера или трубопровода, по которому проект движется от первоначальной идеи до конкретного продукта.<br><br>Пайплайны используются для организации и контроля рабочего процесса. С их помощью вырабатываются новые идеи, продумываются решения и конкурентные преимущества будущего продукта. | Создание мотивации учебной деятельности. Подведение к теме занятия |

| | | | |
|---|---|---|---|
| | | Отлично, сегодня вы научитесь правильно обрабатывать данные, а также оценивать качество моделей с помощью кросс-валидации | |
| **Изучение нового материала** | 55 мин. | Познакомимся с самой важной частью этого занятия — схемой пайплайна машинного обучения. Эта диаграмма расскажет, что нужно делать, чтобы успешно построить хорошую модель машинного обучения. Стоит оговориться, что пайплайн, который мы рассматриваем, очень базовый и в реальности для построения хороших моделей нужно добавлять в эту схему какие-то действия.<br><br>Машинное обучение начинается с сырых данных, которые есть у нас на входе. То, как их получить и превратить в таблицы, мы оставляем за рамками занятия, так как действия очень сильно зависят от природы данных и от того, где они хранятся. С самого начала данные нужно разделить на обучающую часть, на которой будет учиться алгоритм, валидационную часть, на которой мы будем проверять качество во время обучения, и тестовую часть, по которой мы будем принимать финальное решение о том, насколько хорошая наша модель. Сделать разделение в самом начале критически важно, потому что иначе во время обработки данных мы можем создать утечку и каким-то образом закодировать информацию из тестового множества в обучающее. Например, мы можем посчитать среднее значение признака и вычесть его из всей колонки. Если при подсчёте среднего использовались тестовые данные, то это утечка, которая может повлиять на результат.<br><br>После разделения данных необходимо их обработать. Как именно это делать, мы обсудим немного позже. | Активные действия учащихся с объемом изучения.<br>Вызов познавательного интереса к предмету. |

| | | | |
|---|---|---|---|
| | | Затем мы делаем самое интересное — обучаем модель машинного обучения.<br><br>Полученную модель мы валидируем, то есть вычисляем качество на валидационной выборке. На этом этапе можно сравнить различные построенные модели и выбрать наилучшую. Если качество нас не устраивает, придётся вернуться к обработке данных и обучению новой модели. Если качество устраивает, то мы проводим финальное тестирование на тестовой выборке. После этого модель готова, и можно начинать её применять.<br><br>Теперь подробно поговорим об этапе обработки данных. Обрабатывать данные можно по-разному. Пока что мы будем пользоваться максимально простой схемой. Сначала удалим из признаков утечки и обработаем пропущенные значения. Если признак категориальный, необходимо превратить его в числовой. Если признак уже числовой, то, возможно, потребуется почистить его от выбросов.<br>Пройдёмся по каждому пункту в отдельности.<br><br>Первая проблема, на которую нужно обратить внимание, это так называемые утечки данных, по-английски data leakage. Объясню, что это такое, на примере.<br>Пусть нам поступил заказ от госпиталя на разработку модели медицинской диагностики. Модель будет определять, есть ли у человека пневмония. Перед нами задача бинарной классификации — болен человек или нет.<br><br>Госпиталь предоставил нам данные для обучения. На слайде вы видите пример обучающей выборки. Обратите внимание, если у пациента есть пневмония, то он, скорее всего, принимает | |

| | | | |
|---|---|---|---|
| | | антибиотики, и наоборот, если пациент принимает антибиотики, то у него намного выше вероятность болеть пневмонией. Если признак «принимает антибиотики» добавить в обучающую выборку, то наша модель выучит, что класс можно предсказывать всего лишь по одному признаку — принимает ли пациент антибиотики. Мы получим отличные метрики на валидационном и тестовом сете, но как только нашу модель попытаются применить в жизни, окажется, что её применяют на пациентах, которые ещё не были у врача и не получали назначения на антибиотики. То есть все значения в колонке окажутся False, и наша модель, опирающаяся на один признак, сломается.<br><br>Итак, проблема в том, что модель опиралась в предсказаниях на данные, которые оказались недоступны во время применения модели. В данном случае информация о целевой переменной «протекла» в один из признаков. Такой признак использовать для построения модели нельзя.<br>Поверьте, здесь нужно быть очень внимательными.<br><br>Следующая сложность, с которой вы можете встретиться при обработке данных, это пропущенные значения.<br>Сбор данных для обучения — сложная задача, и не всегда данные бывают идеальными. Иногда значения некоторых признаков оказываются пропущены.<br><br>Как раз такая задача — это задача предсказания выживших пассажиров «Титаника»: когда собирали данные, как-то не рассчитывали, что кто-то на них будет обучать машины. Большинство алгоритмов машинного обучения совсем не умеют работать с пропусками, поэтому нам придётся их каким-то образом обработать вручную. Мы рассмотрим три группы | |

| | | | |
|---|---|---|---|
| | | простых методов: для категориальных признаков, для числовых признаков и для тех и других. Существуют и намного более тонкие и сложные методы, чем те, которые мы рассмотрим, но они выходят за рамки нашего курса.<br><br>Если у нас есть колонки с пропущенными числовыми значениями, то мы можем их заполнить средним значением по колонке или медианой по колонке. Для многих алгоритмов такое заполнение пропусков будет неплохо работать. Например, для колонки с возрастом наше пропущенное значение равно 31,2.<br><br>Для категориальных пропущенных значений есть два основных варианта: заполнение самым часто встречающимся значением или же добавление нового значения, которое будет соответствовать пропуску. Какой именно вариант выбрать, зависит от типа категориальной переменной, типа алгоритма и здравого смысла. Здравый смысл, напоминаю, очень важен! Например, в данной задаче мы заполняем значение кабины лейблом unknown.<br><br>Иногда мы можем прийти к выводу, что у нас не получится хорошо заполнить пропуски. В таких ситуациях можно просто удалить плохие данные. Если пропусков много, то стоит удалить всю колонку с признаком. Если пропусков мало, стоит удалить строчки, в которых признак пропущен.<br><br>Если же во время применения модели вам точно не встретятся пропуски и у вас получается от них избавиться в обучающем датасете, удалив не слишком много данных, то так и поступите. Заполнение пропусков средним, медианой или самым часто | |

| | | | |
|---|---|---|---|
| | | встречающимся значением может исказить данные, поэтому применяется только в крайних случаях.<br><br>Перейдём к обработке категориальных признаков. Чтобы подавать их в модель машинного обучения, категориальные признаки необходимо превратить в числа. На практике мы рассматривали метод one hot encoding. Давайте повторим ещё раз, что именно мы делаем.<br><br>Пусть у нас есть признак «город», который принимает несколько значений. Пока наш категориальный признак закодирован в виде строк («Дубай», «Москва», «Амстердам»), машина не сможет понять его смысл. Мы можем попробовать закодировать все уникальные значения переменной своим целым числом. В данном случае кодируем Дубай нулём, Москву единицей, Амстердам двойкой.<br><br>Такие данные можно было бы использовать, но здесь мы немного обманываем модель. Так как она не знает, что признак изначально категориальный, ей кажется, что Дубай и Москва в два раза ближе друг к другу, чем Дубай и Амстердам, потому что расстояние от 0 до 1 в два раза меньше, чем расстояние от 0 до 2. Я имею в виду, конечно, не расстояние в километрах, а просто какую-то абстрактную близость. На самом деле у нас априори нет такой информации.<br><br>Чтобы перестать обманывать модель, вместо одной старой колонки мы создаем три новых — по количеству возможных значений признака. Для каждого объекта мы выставляем единичку в той колонке, которая соответствует нужному городу. | |

| | | | |
|---|---|---|---|
| | | Рассмотрим пример с практического занятия. У нас были две категориальные переменные с тремя и пятью различными значениями, которые превратились в 3 и 5 колонок соответственно.<br><br>Следующий шаг — обработка численных признаков. С ними можно проводить много различных преобразований: видоизменять, комбинировать между собой. Мы рассмотрим очистку данных от выбросов — то, что нужно делать всегда. Выброс — это такое значение признака, которое сильно выбивается из остальных. Обычно выбросы — это какие-то численные ошибки, например, когда температура, измеренная с утра, оказалась равна 9999 градусам по цельсию, вместо положенных 21.<br><br>Чтобы находить и убирать выбросы, существуют специальные математические методы. Они очень сложные и требуют точной калибровки, поэтому их мы использовать не будем.<br><br>С другой стороны, для поиска выбросов часто достаточно построить хорошую визуализацию. Например, бокс-плот с усами, гистограмму или обычный график. Эти методы хороши своей простотой и отсутствием калибровки. При этом находить глазами выбросы, когда переменных, по которым могут быть выбросы, слишком много, становится почти невозможно.<br>Итак, мы перечислили несколько важных методов предварительной обработки признаков. В предстоящей практике мы постараемся применить все изученные техники.<br><br>Поговорим о ещё одной важной технике — кросс-валидации. Это метод оценки качества модели, который не требует разделения на обучающую и валидационную выборки и | |

| | | | |
|---|---|---|---|
| | | позволяет сэкономить на данных. Кроме того, он позволяет повысить точность оценки качества модели.<br>В чём состоит проблема? Очень часто бывает жалко выделять 10% доступных данных на валидацию, чтобы просто оценивать на них метрики, ведь эти данные могли послужить нам для обучения.<br><br>Алгоритм кросс-валидации позволяет избавиться от этого недостатка. Вместо того, чтобы один раз разделить данные на обучающую и валидационную части, мы будем делать это n раз. Скажем, n = 5. В самом начале мы делим все обучающие данные на 5 непересекающихся частей. После этого запускаем обучение пять раз. Для каждого запуска мы выбираем одну из частей, которую будем использовать в качестве валидации для конкретно этого запуска. Затем мы обучаемся на оставшихся четырёх частях. У полученного алгоритма мы вычисляем качество предсказания только на выделенной пятой части. Проделав эту процедуру 5 раз, мы получим 5 чисел, показывающих качество предсказаний. Теперь мы их усредним, чтобы получить финальную оценку качества.<br><br>Финальную модель можно обучать на всех доступных данных.<br><br>У такого подхода много плюсов. Во-первых, мы получаем более точную и сбалансированную оценку качества, поскольку наше итоговое качество — это среднее значение из пяти чисел. Во-вторых, мы потратили ноль данных на валидационный датасет, поскольку итоговая модель обучена на всех доступных данных. | |

| | | | |
|---|---|---|---|
| | | Единственный минус — нам необходимо проводить обучение несколько раз, поэтому такой способ не подойдёт для нейронных сетей, обучение которых занимает много времени. | |
| **Закрепление изученного материала** | 10 мин. | **Игра «Расшифровка терминов»**<br><br>**Вопросы:**<br>1. Метод оценки аналитической модели и её поведения на независимых данных с наиболее равномерным использованием имеющихся данных.<br>2. Последовательные стадии преобразования данных, предшествующие их загрузке в модель.<br>3. Значение признака, которое сильно выбивается из остальных.<br>4. Обработанная и структурированная информация в табличном виде.<br>5. Использование информации в процессе обучения модели, которая, как ожидается, не будет доступна во время прогнозирования, в результате чего прогнозные оценки будут переоценивать полезность модели при запуске в производственной среде.<br><br>**Ответы:**<br>1. Кросс-валидация<br>2. Пайплайн<br>3. Выброс<br>4. Датасет<br>5. Утечка данных | Выполнение практического задания.<br><br>Рекомендации учителю для проведения игры<br>Учитель читает значение термина, обучающиеся высказывают предположения. |

| | | | |
|---|---|---|---|
| **Этап подведения итогов занятия (рефлексия)** | 8 мин. | **Подведём итоги.** Мы повторили схему пайплайна машинного обучения для построения алгоритмов и оценки качества. Научились обрабатывать входные данные. Наконец, узнали о кросс-валидации для оценки качества моделей.<br><br>**Продолжите фразы:**<br>**Вопросы:**<br>1. Мне было интересно узнать, что...<br>2. Мне было трудно понять...<br>3. Теперь я могу...<br>4. Я научился...<br>5. Меня удивило...<br>6. Мне захотелось...<br>7. Особенно интересно было на уроке... | Способствование осознания ценности выполненной работы. Достижение полного понимания изученного материала. Самооценка детей, сравнение результатов собственной деятельности с другими, осмысление результатов |
| **Информация о домашнем задании, инструктаж по его применению** | 5 мин. | В этом домашнем задании вам предстоит создать и протестировать ваш первый алгоритм машинного обучения --- метод ближайших соседей.<br><br>Метод ближайших соседей (k Nearest Neighbors, или kNN) — очень популярный метод классификации, также иногда используемый в задачах регрессии. Это один из самых понятных подходов к классификации. На уровне интуиции суть метода такова: посмотри на соседей; какие преобладают --- таков и ты. Формально основой метода является гипотеза компактности: если метрика расстояния между примерами введена достаточно удачно, то схожие примеры гораздо чаще лежат в одном классе, чем в разных. | |

| | | | |
|---|---|---|---|
| | | Для классификации каждого из объектов тестовой выборки необходимо последовательно выполнить следующие операции:<br><br>• Вычислить расстояние до каждого из объектов обучающей выборки<br>• Отобрать объектов обучающей выборки, расстояние до которых минимально<br>• Класс классифицируемого объекта — это класс, наиболее часто встречающийся среди $k$ ближайших соседей | |

**Рекомендуемые ресурсы для дополнительного изучения:**

1. Пайплайн (Pipeline) [Электронный ресурс] – Режим доступа: https://www.helenkapatsa.ru/paiplain/.
2. Кросс-валидация (Cross-validation) [Электронный ресурс] – Режим доступа: https://wiki.loginom.ru/articles/cross-validation.html.