Научно-практическая конференция «Основы нефтегазового дела» для учащихся общеобразовательных организаций

Доклад на тему: “Машинное обучение для обработки геологических данных“

(слайд 3)

**Введение**

Геологические данные помогают в обосновании технологических решений проектирования разработки, регулировании процесса разработки, регулировании и учете фонда скважин, в принятии решений о переводе скважины из одного состояния в другое, в контроле добычи нефти, газа и воды и их динамики по скважине. Для их обработки, анализа и интерпретации требуется специалист-геолог. Зачастую ему приходится выполнять рутинную работу, которую можно было бы автоматизировать с помощью алгоритмов машинного обучения.

(слайд 4)

**Проблема**

Специалисты-геологи тратят много времени на рутинную работу по интерпретации геофизических данных, собранных со скважин

**Актуальность**

Несмотря на то, что такие решения уже есть, мы хотим использовать другой, новый быстроразвивающийся метод анализа данных, Топологический Анализ Данных (ТАД), и проверить его эффективность на этой задаче

**Цель**

Изучить целесообразность применения ТАД для интерпретации результатов ГИС

**Задачи**

* Узнать какие геофизические данные собирают и обрабатывают геологи
* Найти базу данных для обучения модели
* Создать алгоритм, использующий ТАД, для автоматизации работы геологов
* Оценить эффективность созданного алгоритма

(слайд 5)

**Методы геологических исследований скважин:**

Классификация методов ГИС может быть выполнена по виду изучаемых геофизических полей:

* Электрические методы - ~~измеряются удельное сопротивление, электропроводность и естественные потенциалы горных пород~~
* Радиометрические методы - ~~основаны на изучении естественного гамма-излучения и взаимодействия вещества горной породы с наведенным ионизирующим излучением~~
* Сейсмоакустические методы - ~~основаны на изучении среды посредством пробных акустических волн, распространяющихся в земных породах~~
* Акустический каротаж - ~~основаны на изучении свойств горных пород по измерениям в скважине характеристик упругих волн ультразвуковой (выше 20 кГц) и звуковой частоты~~

Самой распространенной разновидностью ГИС является каротаж, который, в зависимости от задачи исследования, объединяет в себе некоторые из этих методов.

(слайд 6)

**Кринж данных**

Поскольку чаще всего применяют электрические и радиометрические методы каротажа, то баз данных, где собраны соответствующие измерения, больше и они полнее.

Одной из основных задач ГИС является определение геологического разреза, поэтому нас будут интересовать базы данных, в которых указаны еще и литотипы.

В силу названных выше факторов, а также ограниченности вычислительных мощностей, используемого нами оборудования, мы выбрали базу данных, которая использовалась в конкурсе Geophysical Tutorial Machine Learning Contest 2016.

Соответственно, сравнивать наши результаты мы будем с результатами победителей конкурса, которые использовали модели машинного обучения, отличные от нашей.

(слайд 7)

База данных содержит в себе информацию о каротаже 10 скважин: литотипе, формации, названии скважины, глубине замера, измеренные значения ГК (GR), удельного сопротивления (ILD\_log10), разницы в пористости по плотности нейтронов (DeltaPHI), cредней пористости нейтронной плотности (PHIND), фотоэлектрического эффекта (PE), относительного положения (RELPOS) и др.

(слайд 8)

Из всех характеристик мы оставим только те, где заполнены все колонки. В дальнейшем, для обучения алгоритма мы будем использовать 9 из 10 скважин, а предсказывать будем литологию оставшейся скважины.

(слайд 9)

**Алгоритм Топологической Классификации**

Алгоритм топологической классификации был представлен в статье R.Kindelan (Роландо Кинделан), J.Frias (Хосе Фрайас), M.Cerda (Маурицио Черда), N.Hitchfield (Нэни Хичфельд), Classification based on Topological Data Analysis (Классификэйшн бэйзд он Тополоджикал Дата Анализиз), Feb. 9 2021, где также сравнивался с другими моделями машинного обучения, такими как k-NN (“ка” ближайших соседей) и wk-NN (его взвешенной версией), на стандартном наборе баз данных, и местами превзошёл их.

Алгоритм было решено реализовать на языке python (пайтон) с использованием библиотек pandas (пандас), numpy (нампай) и gudhi (гудхи).

(слайд 10)

Пример работы алгоритма.

Заданы три облака точек трех разных классов (красного, синего и зеленого)

На вход подается облако точек из круга и алгоритм классифицирует их:

(слайд 11)

**Применение Алгоритма Топологической Классификации**

Прежде чем тестировать алгоритм на базе данных, упомянем способ оценки верности предсказания.

Поскольку верно предсказать литотип сложно даже для специалиста и допускается некоторая погрешность в пределах схожих типов, то оценивать эффективность нашего алгоритма мы будем, руководствуясь таблицей, используемой в Geophysical Tutorial Machine Learning Contest 2016.

В контесте для оценки использовалась метрика F1-micro (эф один микро), которая, как известно, в задаче многоклассовой классификации совпадает с метрикой точности, поэтому её мы и будем вычислять.

(слайд 12)

Из всей базы данных для классификации мы использовали только четыре параметра: значения ГК (GR), удельного сопротивления (ILD\_log10), разницы в пористости по плотности нейтронов (DeltaPHI) и cредней пористости нейтронной плотности (PHIND).

Мы изменяли параметры filt\_value, mode, random\_label\_choice и change\_data и получили следующие графики зависимости метрики точности от этих параметров.

Параметр maxdim был при этом всегда равен единице, поскольку эмпирически было установлено, что при maxdim > 1 точность при тех же параметрах уменьшается, к тому же кратно возрастает время вычислений, т.к. появляется много симплексов размерности 2.

*(графики)*

(слайд 15)

Как видно из графиков, лучшим набором параметров является:

* filt\_value = 21
* maxdim = 1
* mode = 'star'
* random\_label\_choice = False
* change\_data = True

Пример предсказания литологии

(слайд 16)

**Результаты и обсуждение**

В таблице сравниваются две версии алгоритма: описанная в статье, использующая линк, и наша, использующая звезду. Представлены их метрики: точность, точность с погрешностью, учитывающая схожесть литотипов, и метрика эф один макро. Из таблицы следует, что версия, использующая звезду, справляется лучше.

(слайд 17)

По результатам контеста победил алгоритм, F1-micro метрика (то же, что и accuracy) которого приблизительно равна 0.64.

Учитывая тот факт, что adjacent\_accuracy чуть меньше чем вдвое больше accuracy, то можно ожидать, что accuracy этого алгоритма лежит в пределах 0.9 - 1 .

(слайд 18)

**Заключение**

Мы увидели, что алгоритм TopClassifier плохо справился с классификацией сырых геофизических данных, поэтому, возможно, стоит провести feature engineering (фича инжиринг), но для этого нужны хорошие знания в геофизике.

Мы испробовали совсем немного методов ТАД, и, чтобы в полной мере раскрыть его потенциал, потребуется больше реальных данных, вычислительных мощностей и консультирования со стороны специалистов-геологов.

В заключение можно сказать, что на данном этапе использование алгоритма TopClassifier для предсказания геологического разреза **не эффективно**.

(слайд 19)

**Добавление. Алгоритм Топологической Классификации. Термины**

**Симплекс** — геометрическая фигура, являющаяся n-мерным обобщением треугольника.

**Симплициальный комплекс** — топологическое пространство с заданной на нём триангуляцией, то есть, неформально говоря, склеенное из топологических симплексов по определённым правилам или обобщение графов на высшие размерности.

Пример симплициального комплекса: (на слайде показать 0-симплексы (вершины), 1-симплексы (отрезки), 2-симплексы (закрашенные треугольники НЕ ТЕТРАЭДР))

(слайд 20)

**Звезда симплекса** в симплициальном комплексе — это все симплексы в симплициальном комплексе, имеющие данный симплекс своей гранью (объясни, что такое грань симплекса (подсимплекс этого симплекса)): (на слайде)

**Линк симплекса** в симплициальном комплексе — обобщение окрестности вершины в графе. Линк вершины кодирует информацию о локальной структуре комплекса в её окрестности: (на слайде)

Зная звезду симплекса, можно легко вычислить и его линк.

(слайд 21)

**Комплекс Вьеториса-Рипса** или **VR-комплекс** — симплициальный комплекс, полученный из облака точек путем объединения его подмножеств в симплексы при условии, что диаметр этого подмножества меньше 2ε, где ε — заданный наперед параметр (можно понимать примерно как раздувание шариков из точек и объединение точек с пересекающимися шарами в симплексы, см. слайд)

(слайд 22)

**Фильтрация симплициального комплекса** — возрастающая последовательность его подкомплексов, т.е. каждый подкомплекс является подкомплексом следующего: (см. слайд, здесь можно видеть как растет комплекс, как появляются связи между вершинами и проч.)

Таким образом, мы можем построить фильтрацию VR-комплекса данного облака точек, занумерованную значениями параметра ε и каждому симплексу приписать значение фильтрации, при котором он впервые появляется.

(слайд 23, 24)

**Добавление. Алгоритм Топологической Классификации**

Перейдем к описанию самого алгоритма.

Его работу можно описать как последовательное применение трех функций к классифицируемой вершине.

Первая функция, Ассоциирующая, каждой вершине комплекса сопоставляет соответствующий её классу базисный вектор N-мерного пространства, где N — это кол-во классов. Т.е. мы ассоциировали базисные векторы N-мерного пространства и множество классов.

Эту функцию можно доопределить и для произвольных симплексов (см. слайд).

Таким образом, мы каждому симплексу при фиксированном значении эпсилон сопоставили эн-мерный вектор, компоненты которого — суть кол-во его вершин с данной маркой

(слайд 25, 26)

Вторая функция, функция Расширения, берет симплексы близкие к нашей точке, вычисляет ассоциированные с ними векторы, “взвешивает” их и суммирует результат.

В статье близкие симплексы берутся из линка вершины, мы же предлагаем брать их из звезды и посмотреть какой из способов будет лучше.

Здесь кси от сигма в комплексе К - значение фильтрации, при котором появился симплекс сигма.

Чем раньше наша вершина попала в комплекс сигма, тем меньше значение функции кси и тем больший вклад вносят вершины, которые он содержит.

Теперь мы имеем N-мерный вектор, компоненты которого характеризуют сколько и как близко находятся вершины данного класса к классифицируемой.

Заметим, что для вершин значение функции кси по определению равно нулю, но на ноль мы делить не можем, поэтому в реализации предполагается, что значение кси на вершинах равно одной миллионной (или любое другое достаточно малое число, это зависит от кол-ва точек)

(слайд 27)

Мы пришли к финальному этапу присвоения метки.

По заданной точке мы построили вектор или, проще говоря, набор весов, которые определяют влияние положения этой точки на образование связей с точками соответствующей метки.

Последнее, что остается сделать — это выбрать метку с максимальным весом и присвоить её точке. Если таких меток несколько, то можно всегда выбирать первую попавшуюся, а можно выбрать случайно.

Это, как и выбор линка или звезды для Функции Расширения, влияет на точность предсказания. Выбор этих параметров заложен в реализацию нашего алгоритма.

Также мы оставили возможность изменять или не изменять облако точек, которое мы используем для определения метки, т.е. использовать только что помеченные точки для предсказания меток новых.