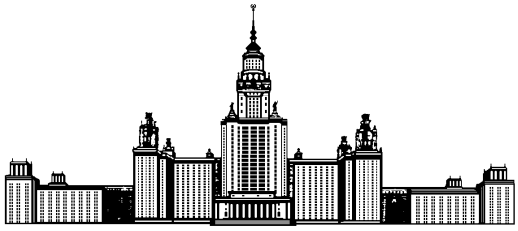
Московский государственный университет имени М. В. Ломоносова
Факультет Вычислительной Математики и Кибернетики
Кафедра Математических Методов Прогнозирования



Магистерская программа «Логические и комбинаторные методы анализа данных »

Магистерская диссертация

Использование нейросетевого подхода для аппроксимации одной гидродинамической модели

Выполнил:
студент 6 курса 617 группы
Темирчев Павел Георгиевич

Научный руководитель:
к.ф-м.н., доцент
Ветров Дмитрий Петрович

Москва, 2017

Содержание

Аннотация

Рассматривается задача аппроксимации гидродинамической модели нефтяного месторождения. Для аппроксимации используются методы машинного обучения: линейная регрессия, полносвязная нейронная сеть и рекуррентная нейронная сеть, обученные на результатах работы конечно-разностного гидродинамического симулятора. Задача аппроксимации возникает из-за низкой скорости работы конечно-разностного симулятора. Получена модель, применимая на практике и работающая с низкой ошибкой аппроксимации.

Предложен метод генерации реалистичной обучающей выборки, основанный на градиентном подъеме в пространстве признаков. В процессе градиентной оптимизации максимизируется функционал, характеризующий реалистичность объектов выборки.

1 Введение

В данной работе рассматривается задача аппроксимации гидродинамической модели нефтяного месторождения при помощи методов машинного обучения, в частности при помощи нейронных сетей.

Задача гидродинамического моделирования течения многофазной смеси в поровых каналах породы не нова и существующие методы ее решения пользуются большой популярностью в нефтегазовой отрасли. Здесь и далее будет рассматриваться моделирование процесса разработки нефтяной залежи.

Нефтяная залежь представляет собой пласт пористой горной породы, залегающий на глубине несколько сотен метров. Поры породы связные, что позволяет жидкостям и газам (флюидам) течь внутри породы. Порода нефтяной залежи насыщена собственно нефтью, а также часто водой и природным газом. Способность породы пропускать сквозь себя жидкость или, как говорят, способность породы к фильтрации жидкости определяется свойством, которое называется проницаемостью. Итак, нефтяная залежь - это пористый и проницаемый пласт горной породы, залегающий на глубине несколько сотен метров и насыщенный нефтью. В данной работе также будет использоваться термин "месторождение"в смысле аналогичном таковому для залежи. Вообще говоря, за границей данной работы, месторождение может состоять из нескольких гидродинамически не связанных залежей.

Основным способом извлечения нефти из залежи является разработка ее с использованием скважин. Скважина представляет собой цилиндрическую выемку, имеющую устье на поверхности и забой (нижняя часть скважины) в районе разрабатываемого горного пласта. Скважины устроены достаточно сложно и в них зачастую используется совершенно разное оборудование, но специфика задачи позволяет не вдаваться в подробности внутреннего устройства скважин.

Для большинства месторождений на сегодняшний день используется метод разработки с применением заводнения. Для реализации данного метода на месторождении бурятся скважины двух типов: нагнетательные и добывающие. Нагнетательные скважины закачивают в пласт воду. В добывающие скважины спускается насос, который из пласта откачивает жидкость и газ, которые приходят на забой скважины. Поскольку в пласт закачивается вода, то добывающая скважина добывает смесь воды и нефти. Кроме того, необходимо учесть, что из нефти при понижении давления выделяется так называемый попутный газ, который также попадает в скважину. Идея разработки месторождения с применением заводнения достаточно проста и основана на том, что давление в порах залежи нельзя понижать ниже давления насыщения нефти газом, иначе это приведет к выделению газа из нефти в поры. Мало того, что выделение газа из нефти делает ее более вязкой, при наличии второй более подвижной фазы фильтрация нефти сильно осложняется вплоть до полного ее прекращения.

Закачка воды в пласт позволяет решить проблему падения давления в порах залежи в процессе разработки, однако возбуждает другую проблему - прорыв воды из нагнетательной скважины в

добывающую по пути наименьшего гидродинамического сопротивления. Это приводит к добыче на добывающих скважинах практически только воды и к невозможности добыть большую часть запасов залежи. Подобные проблемы, а также некоторые другие, решаются грамотным выбором схемы расстановки скважин и режимов, на которых скважины работают - объемов закачки воды в нагнетательные скважины и объемов добычи жидкости из добывающих скважин (дебитов по жидкости).

В процессе разработки залежи ставится задача поиска компромисса между объемом добытой нефти, затраченным временем и экономическими издержками. Для поиска хорошего решения необходимо использовать компьютерную симуляцию процесса разработки. Наиболее распространенным способом моделирования сегодня является применение конечно-разностных методов:

- Пласт горной породы разбивается на ячейки по сетке, свойства во всем объеме ячейки считаются одинаковыми;
- Строится геологическая модель залежи, повторяющая форму реальной залежи, в ячейках заполняются характеристики породы и жидкости ее насыщающей;
- На модели залежи размещаются скважины и задаются временные ряды, характеризующие изменение режимов добычи и закачки во времени;
- Запускается итеративный пересчет перетоков жидкости между соседними ячейками, пересчитываются свойства для каждой ячейки, для каждой добывающей скважины вычисляется доля воды, нефти и газа в добываемой продукции.

Данный метод моделирования процесса разработки отличается высокой точностью, если все свойства породы и насыщающих жидкостей были заданы правильно. Однако, конечно-разностные методы обладают высокой вычислительной сложностью и малым потенциалом параллелизма. Чистовой расчет для крупного месторождения на мелкой сетке может длиться неделями на относительно больших вычислительных кластерах.

В данной работе рассматривается идея поиска быстрого и достаточно точного метода, аппроксимирующего поведение конечно-разностного симулятора. Рассматриваются методы машинного обучения с учителем, в частности нейронные сети. Задача решается для фиксированного месторождения. Это значит, что физические свойства породы и фильтруемых жидкостей остаются неизменными в процессе настройки модели машинного обучения. Также в данной работе мы считаем скважины зафиксированными и их расположение на залежи не меняется. Изменяемыми параметрами остаются режимы работы скважин.

Предлагаемый метод в тестовом режиме должен принимать на вход такие параметры режима, которые возможно регулировать с поверхности, замер этих параметров не должен требовать каких-либо специальных датчиков, которые могут быть не установлены на скважинах. Такими параметрами являются объем закачки воды в нагнетательные скважины и объем добытой жидкости из добывающей скважины в каждый момент времени.

В следующей главе данной работы будет проведен анализ существующих решений проблемы быстрого моделирования процессов разработки нефтяного месторождения. В третьей главе будет проведена постановка задачи аппроксимации. В главах 4-8 мной предлагается метод решения данной проблемы. В главах 8-9 проанализированы результаты, полученные в рамках данной работы, а также описаны возможные направления для продолжения научных изысканий в данном направлении.

2 Анализ литературы по теме работы

Проблема низкой скорости работы конечно-разностных гидродинамических симуляторов не нова. Основным способом борьбы с этой проблемой является настройка более простых моделей, обученных на истории разработки месторождения. При таком подходе обучаемые модели не требуют наличия собственно гидродинамического симулятора, однако требуют наличия истории разработки месторождения.

Большая часть работ, посвященных быстрому прогнозированию показателей разработки, посвящена задаче прогнозирования объемов добычи жидкости на добывающих скважинах по объемам закачки воды в нагнетательные и, возможно, по значению забойного давления в нагнетательных скважинах. Объем добываемой жидкости не является практически значимой характеристикой сам по себе. Более того, он регулируется с поверхности. Практически значимым результатом применения этих моделей является возможность определения гидродинамических связей между скважинами без использования гидродинамического симулятора.

Наиболее изученной и активно используемой моделью для решения этих задач является Capacitance Resistive Model (CRM) [4, 6, 5, 3]. Модель CRM является полуфизической моделью и построена по аналогии с электрической цепью. Представляет собой нелинейную регрессию, использующую как входы объемы закачки воды в нагнетательные скважины и давления на забое добывающих скважин. Наличие информации о забойных давлениях в скважинах возможно при установке в них специальных датчиков, отсутствующих в большинстве скважин. В случае отсутствия данных о забойных давлениях они интерпретируются как постоянные, что далеко от реального режима работы добывающей скважины.

Классическая модель CRM не делает различий между водой и нефтью. В работе [5] для прогнозирования обводненности продукции используется гибридная модель, являющаяся смесью CRM и моделью Коваля - непоршневой моделью несмешивающейся фильтрации жидкости в порах породы.

Поскольку CRM является полуфизической моделью, она зависит от физических характеристик залежи, принятых в классической модели неизменными в процессе разработки. При данном предположении классическая модель CRM может использоваться только на заключительных этапах разработки. Решение данной проблемы с помощью гибридизации CRM с физической моделью изменения нефтенасыщенности породы приведено в [4]. Другая проблема, связанная с CRM - предположение о замкнутости гидродинамической системы. То есть, предполагается, что в системе нет других

стоков и источников жидкости, кроме скважин. Решение данной проблемы достигается некоторым изменением модели CRM в работе [6].

Для решения аналогичной задачи, для которой используется модель CRM, применялись и нейронные сети [**crm5**, 3]. Вместо метода нелинейной регрессии в обоих работах используется полносвязная нейронная сеть с одним скрытым слоем. Поскольку ставится задача обучения на истории добычи внутри одного сценария, задача генерации репрезентивной выборки в этих работах не обсуждается. Объемы закачиваемой воды и добываемой жидкости сильно кореллируют, в обоих работах показано высокое качество прогнозирования. Для обучения моделей был использован метод градиентного спуска. Не рассматривается возможность расширения признакового пространства и более сложные архитектуры нейронных сетей. Нейронные сети обучаются на первых 80% продолжительности временных рядов, одним объектом ялвяется один конкретный момент времени, оставшиеся последние 20% временных рядов используются в качестве валидационных. В решении задачи прогнозирования объемов добычи жидкости по объемам закачки воды и задачи определения гидродинамической связи межжу скважинами нейронные сети показали себя лучше, чем модель CRM.

3 Постановка задачи аппроксимации

Как было сказано ранее, в этой главе мы будем рассматривать задачу обучения с учителем. Наша цель - создать алгоритм, позволяющий вычислять долю воды в добываемой продукции (Watercut - WCUT) и отношение объема добытого газа к объему добытой нефти (Well Gas Oil Ratio - WGOR) для каждой добывающей скважины. Входными для алгоритма переменными должны быть: объемы нагнетаемой в пласт воды (Well Water Injection Rate - WWIR) и объемы добываемой из скважины жидкости (Well Liquid Production Rate - WLPR).

Все входные и прогнозируемые переменные представляют собой временные ряды. Длина временного ряда определяется продолжительностью моделируемого сценария разработки месторождения.

Задача обучения с учителем предполагает наличие обучающих данных, с использованием которых настраивается алгоритм машинного обучения. Обучающие данные представляют собой пары: признаковое описание объекта - правильное значение прогнозируемой переменной, данное экспертом. Как было оговорено ранее, признаковое описание объекта должно быть получено с использованием только WWIR и WLPR для всех скважин. Прогнозируемая переменная - это вектор, состоящий из WCUT и WGOR для всех добывающих скважин.

В качестве объекта, в зависимости от конкретного реализуемого алгоритма, может использоваться либо целый сценарий разработки продолжительностью T, либо один момент времени конкретного сценария разработки $t : t \in \{0, 1, ..T\}$. В первом случае признаковым описанием объекта является матрица, имеющая размер $[T, d]$, где d - размерность признакового пространства в каждый момент времени. Во втором случае признаковым описанием объекта будет вектор длинны d. Признаковое описание одного объекта, вне зависимости от реализуемого алгоритма, здесь и далее будем

обозначать как x. Признаковое описание всей используемой выборки объектов, будем обозначать его X, - это либо трехмерный тензор размера $[N, T, d]$, либо матрица размера $[NT, d]$, где N - число сценариев разработки в обучающей выборке.

Каждому объекту выборки сопоставлено правильное значение прогнозируемой переменной y. Это либо вектор длины $2 \cdot pr$, в случае, если объект - момент времени, либо матрица размера $[T, 2 \cdot pr]$, если объект - целый сценарий. Здесь pr - количество добывающих скважин на месторождении.

Таким образом, решается задача поиска алгоритма a такого, что:

$$a : x \to y$$

Мы не будем делать различий между алгоритмом и реализуемым им отображением.

Предсказанное значение прогнозируемой переменной обозначим $\hat{y}$:

$$\hat{y} = a(x)$$

Для оценки несовершенства работы алгоритма используется функция потерь - корень из среднего квадратичного отклонения между предсказанными и истинными прогнозируемыми величинами y. Для модели, в которой объект представляет собой полный сценарий, функция потерь записывается в виде:

$$loss = \sqrt{\frac{1}{2TN \cdot pr} \sum_{t=0}^{T} \sum_{n=1}^{N} \sum_{k=1}^{2pr} (y_{nk}^t - \hat{y}_{nk}^t)^2}$$

В процессе обучения решается задача поиска алгоритма, минимизирующего введенную функцию потерь:

$$loss \to \min_a$$

Итак, нами была формализована задача машинного обучения с учителем, которая используется в данной работе. Решение поставленной задачи требует наличия набора объектов, с использованием которых и будет настроен метод машинного обучения. В рамках данной работы объекты - режимы работы скважин генерируются случайно с использованием схемы, приведенной в следующей главе.

В качестве эксперта, который для каждого случайно сгенерированного объекта вычисляет истинное значение прогнозируемой переменной, выступает конечно-разностный гидродинамический симулятор. Гидродинамический симулятор принимает на вход информацию о строении и физических свойствах залежи, о схеме расстановки скважин, а также о режимах работы скважин и позволяет вычислить состав добываемой продукции. В данной постановке задачи мы фиксируем все свойства, используемые симулятором как входы, кроме режимов работы скважин.

Таким образом, общая последовательность обучения модели машинного обучения может быть представлена в виде:

- Выбрать конкретную модель месторождения, разместить на ней скважины.

- Сгенерировать режимы работы нагнетательных и добывающих скважин.
- Для сгенерированных режимов работы скважин вычислить истинное значение прогнозируемой переменной с помощью гидродинамического симулятора.
- С помощью полученной выборки настроить алгоритм машинного обучения путем минимизации заданной функции потерь.

4 Генерация данных

В предыдущей главе была формализована задача машинного обучения с учителем, которая используется в данной работе. Решение поставленной задачи требует наличия набора объектов, с использованием которых и будет настроен метод машинного обучения. Как было оговорено ранее, признаковое описание объектов должно зависеть только от WWIR и WLPR, то есть только от режимов нагнетания и добычи на скважинах.

В данной главе ставится задача описания алгоритма для случайной генерации режимов работы скважин. Сгенерированные режимы должны обладать рядом свойств: они должны быть реалистичны, их должно быть достаточно много и они должны быть разнообразны.

Предположим, что существует некое вероятностное распределение на множестве всех возможных режимов работы скважин, такое что именно из этого распределения будут генерироваться режимы работы скважин исследуемые при применении разрабатываемого метода на практике. Тогда процедура генерации должна задавать распределение максимально близкое к описанному.

Поскольку оценить подобные распределения не представляется возможным, было решено генерировать режимы работы скважин, основываясь на традиционных для области шаблонах. Одним из таких шаблонов является так называемая 100%-я компенсация отборов закачкой. По сути, это требование, чтобы в каждый момент времени в сумме по всем скважинам в пласт нагнеталось столько же воды, сколько отбирается жидкости. Данное требование можно записать в виде:

$$\sum_{i=1}^{inj} WWIR_i(t) = \sum_{j=1}^{pr} WLPR_j(t)$$
$$\forall t \in \{0, .., T\}$$

Данное условие вводит зависимость между режимами работы нагнетательных и добывающих скважин. Мы будем генерировать режимы работы нагнетательных скважин, а затем, с учетом введенного условия, будут пересчитываться режимы работы добывающих скважин.

4.1 Генерация режимов работы нагнетательных скважин

Строго определить требования к виду временного ряда режима работы нагнетательных скважин не представляется возможным, однако следует отметить, что для сходимости конечно-разностного

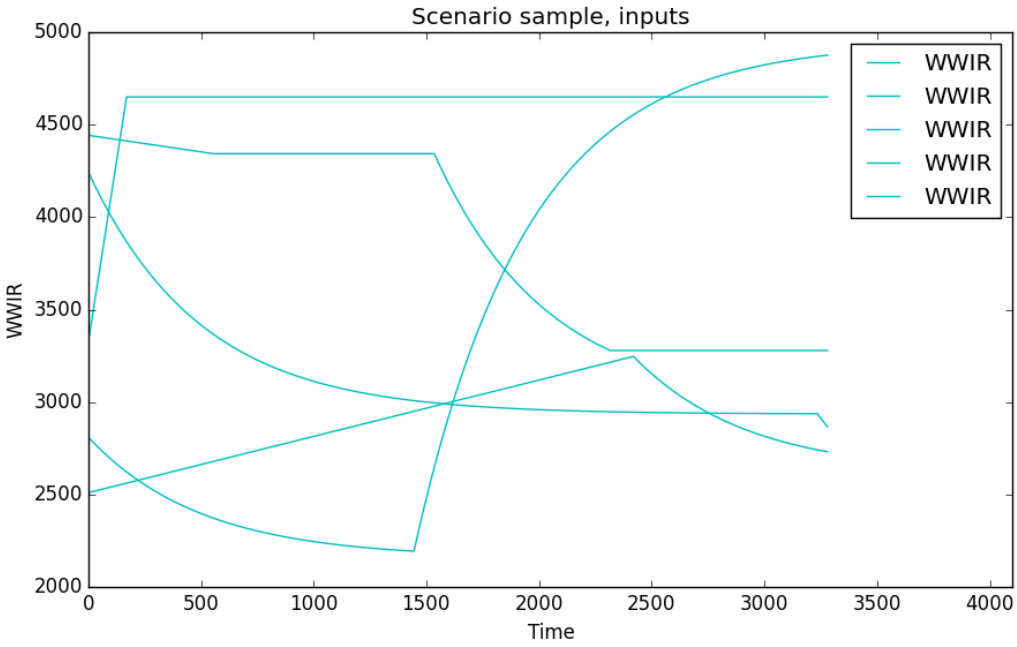


Рис. 1: Расписания закачки для 5ти нагнетательных скважин.

гидродинамического симулятора требуются временные ряды с малым количеством точек разрыва, а также почти всюду гладкие. Кроме того, заметим, что есть стандартные шаблоны закачки, используемые на промысле: закачка с постоянным расходом, с линейно либо экспоненциально изменяющимся расходом. Здесь расход - объем жидкости, протекающий через сечение за единицу времени $q = v/t$.

Режимы закачки для каждой нагнетательной скважины генерируются независимо. Генерация режимов закачки для одной скважины может быть представлена в виде 3-х этапов:

1. Разбить моделируемый промежуток времени на части случайной длины Δt, где $\Delta t \sim Unif[1, \delta]$, δ - параметр частоты смены режима.

2. Для каждой части выбрать один из трех типов нагнетания: константное значение расхода, линейно изменяющееся либо экспоненциально изменяющееся.

3. Случайно сгенерировать коэффициенты для конкретного типа режима закачки так, чтобы в точках смены режима временной ряд оставался неразрывным.

Пример сгенерированных временных рядов можно видеть на рис. 1. По оси абсцисс отложено время в сутках, по оси ординат - расход, измеряемый в баррелях в сутки.

4.2 Генерация режимов работы добывающих скважин

Следующим шагом в генерации данных будет вычисление режимов работы добывающих скважин. Одного условия 100%-й компенсации отборов закачкой недостаточно для определения режи-

ма работы конкретной добывающей скважины. Следовательно, необходимо ввести дополнительное условие. Требование 100%-й компенсации основано на том, чтобы в залежи сохранялось примерно постоянное давление жидкости в порах во времени. Однако, если залежь представлена двумя гидродинамически слабосвязанными участками и вся вода закачивается в один участок, а вся жидкость добывается из другого, то даже при 100%-й компенсации постоянства давления во времени наблюдаться не будет.

Отсюда логичное требование, чтобы сильно связанные гидродинамически скважины сильно влияли на режимы работы друг друга. Данное требование можно формально выразить так:

$$WLPR_j(t) = \sum_{i=1}^{inj} w_{ij} \cdot WWIR_i(t)$$
$$\forall t \in \{0, .., T\}$$

То есть объем добытой жидкости на конкретной скважине должен быть представлен как линейная комбинация объемов закачки во всех нагнетательных скважинах в каждый момент времени. Коэффициенты w_{ij} должны быть тем больше, чем более связанны скважины с индексами i и j. Одним из вариантов выбора коэффициентов w_{ij} может быть значение, пропорциональное средней проницаемости по участку залежи между скважинами. В данной работе успешно используется более простой вариант - обратное эвклидово расстояние между забоями скважин:

$$w_{ij} = 1/\sqrt{\rho(i,j)}$$

$$\rho(i,j) = (x_i - x_j)^2 + (y_i - y_j)^2$$

Для выполнения условия 100%-й компенсации отборов закачкой полученные коэффициенты необходимо отнормировать:

$$\sum_{j=1}^{pr} w_{ij} = 1, \forall i \in \{1, .., inj\}$$

То есть:

$$w_{ij} \leftarrow \frac{w_{ij}}{\sum_{j=1}^{pr} w_{ij}}, \forall i, j$$

Пример вычисленных таким образом режмов работы добывающих скважин приведен на рис. 2. На оси абсцисс отложено время в сутках, на оси ординат - дебит по жидкости, измеряемый в баррелях в сутки.

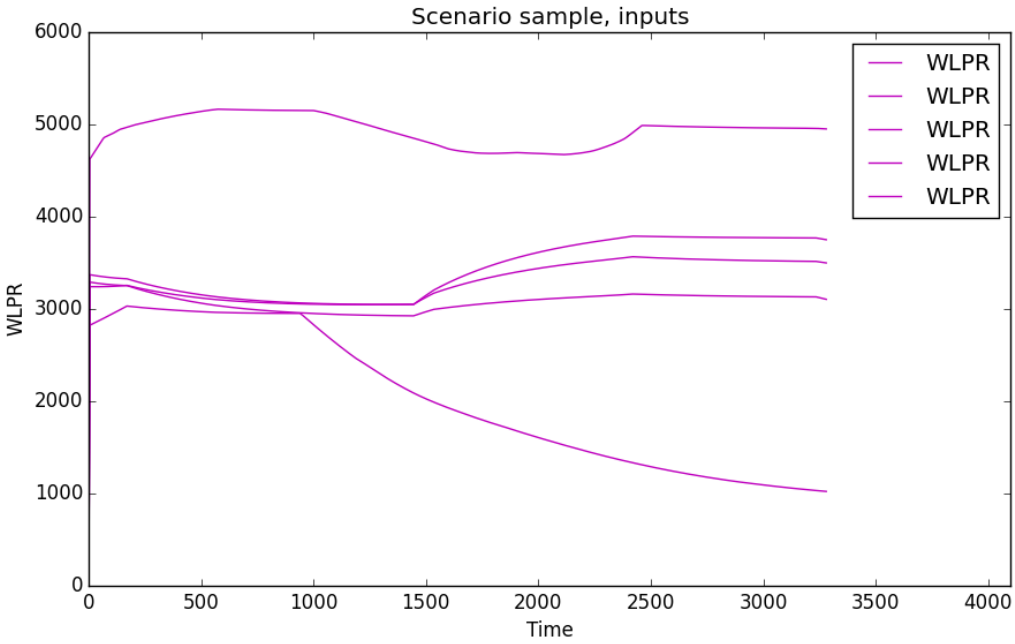


Рис. 2: Расписания добычи жидкости из 5ти добывающих скважин.

5 Используемые признаки и нормировка данных

5.1 Выбор признакового пространства

В предыдущих главах было оговорено, что признаковое пространство объектов должно зависеть только от режимов работы скважин: $WWIR$ и $WLPR$. В данной главе будет рассмотрен вопрос построения признакового пространства объектов по режимам работы скважин, а также методы нормировки полученных признаков.

Наиболее простым и очевидным способом выбора признакового пространства будет выбор в качестве признакового описания одного момента времени вектора, состоящего из объемов закачки WWIR и объемов добычи жидкости WLPR для всех скважин. Такой вектор имеет длину $inj + pr$. Однако, как показывает практика, для достижения высокого качества прогнозирования необходимы более сложные признаковые пространства.

Типичными производными из $WWIR$ и $WLPR$ для нефтегазовой промышленности являются накопленная закачка воды $WCWIR$ и накопленная добыча жидкости $WCLPR$. Данные характеристики можно подсчитать используя уравнения:

$$WCWIR_i(t^*) = \sum_{t=0}^{t^*} WWIR_i(t)$$

$$WCLPR_j(t^*) = \sum_{t=0}^{t^*} WLPR_j(t)$$

Кроме выбранных 4-х признаков было решено использовать значения этих признаков, возведенные в квадрат, а также обратные им. То есть, если признаковое описание объекта в конкретный момент времени - это вектор x, то x расширяется по схеме:

$$x \leftarrow [x^T, (x^2)^T, (x^{-1})^T]$$

Далее в данной работе будут рассмотрены два типа моделей машинного обучения: первые учитывают зависимости во времени, вторые обрабатывают каждый момент времени независимо. Для второго типа моделей вектор признаков в каждый момент времени должен содержать информацию о режимах работы скважин в прошлые моменты времени. Это необходимо поскольку рассматриваемая гидродинамическая система инертна - пара скважин оказывает влияние друг на друга с задержкой во времени, связанной как с пьезопроводностью породы, так и со способностью породы к деформации.

Для второго типа моделей - тех, которые обрабатывают каждый момент времени независимо, признаковое описание объекта расширяется следующим образом:

$$x(t) \leftarrow [x(t)^T, x(t-2^{i_0})^T, .., x(t-2^{i_L})^T]$$

$$\forall i_j \in \{1, 2, .., L\}$$

$$i_0 < i_1 < ... < i_L$$

Где $x(t)$ - это вектор, описывающий признаковое описание объекта в момент времени t.

В экспериментах используется значение $L = 10$.

5.2 Нормировка данных

Для устойчивости процесса обучения моделей машинного обучения данные, подаваемые модели на вход (X), и данные, используемые как прогнозируемые переменные (Y), должны быть отнормированы.

Стандартным подходом является нормировка на отрезок $[-1, 1]$. Данный тип нормировки используется для входных данных X:

$$X \leftarrow \frac{X - mean(X)}{max(abs(X))}$$

Хотя это не учтено в явном виде в функции потерь, для задачи моделирования процесса разработки с использованием заводнения очень важен момент прорыва воды в скважину. То есть тот момент времени, когда обводненность в скважине принимает ненулевое значение. В связи с этим прогнозируемые переменные нормируются на отрезок $[0, 1]$, сохраняющий нулевые значения в выборке нулевыми:

$$Y \leftarrow \frac{Y - min(Y)}{max(Y) - min(Y)}$$

6 Сравнительный анализ моделей машинного обучения

В данной главе рассматриваются предлагаемые модели машинного обучения, процедура выбора их параметров и результаты, полученные для каждой модели.

6.1 Полносвязная нейронная сеть

Основываясь на обзоре литературы, проведенном во 2-й главе данной работы, одним из используемых методов было решено выбрать полносвязную нейронную сеть, которая делает прогноз искомых переменных в каждый момент времени независимо. Судя по проанализированным работам подобная модель способна показывать высокое качество, однако в опубликованных работах совсем не обсуждаются ни процесс обучения нейронной сети, ни обучающая выборка, на которой сеть учится.

Искуственные нейронные сети появились как модели, имеющие некоторую аналогию с принципом работы головного мозга. Впервые понятие искуственной нейронной сети появляется в работе "Логическое исчисление идей, относящихся к нервной активности"Уоррена Мак-Каллока и Уолтера Питтса [9].

Полносвязная нейронная сеть представляет собой параметрическое семейство функций, обозначим его $f(x|W)$. Результат применения функции f к переменной x обозначим как $\hat{y} = f(x|W)$, где $x \in \mathbb{R}^D$, а $\hat{y} \in \mathbb{R}^d$, $W = \{W^0, .., W^l\}$ - множество весовых матриц. Здесь D - длина вектора признакового описания входов, d - длина прогнозируемого вектора, l - количество слоев в нейронной сети. Тогда $\hat{y}$ может быть вычислен по следующей схеме:

- Переобозначим x как $x^{(0)}$, так как он является входом для первого слоя сети.
- Вычислим $a^{(j)} = x^{(j-1)} \cdot W^{j-1}$ и $x^{(j)} = g_j(a^{(j)})$ для $\forall j \in \{1, .., l+1\}$. Здесь W^{j-1} - матрица весов линейной комбинации, число строк которой определено результатом линейной комбинации на предыдущем слое, а число столбцов - это число нейронов на этом слое, это регулируемый параметр, $g_j(\cdot)$ - некая нелинейная дифференцируемая функция.
- Выход последнего слоя нейронной сети и есть искомый результат применения f к x. То есть $\hat{y} = x^{(l+1)}$.

Принципиальная схема полносвязной нейронной сети изображена на рис. 3.

Считается, что в признаковом описании всех объектов есть одинаковый для всех, константный признак, равный единице. Это необходимо для того, чтобы не перегружать нотацию отдельным введением вектора сдвигов в линейную комбинацию.

Поскольку все составляющие части нейронной сети дифференцируемы, а также дифференцируема и введенная ранее функция потерь, можно вычислить градиент функции потерь по каждому элементу любой весовой матрицы $\delta loss/\delta W_{ij}^k$, $k \in \{0, .., l\}$. Эффективным алгоритмом вычисления градиента функции потерь по всем параметрам сети W, вытянутым в вектор, является алгоритм обратного распространения ошибки [11] [10] [7].

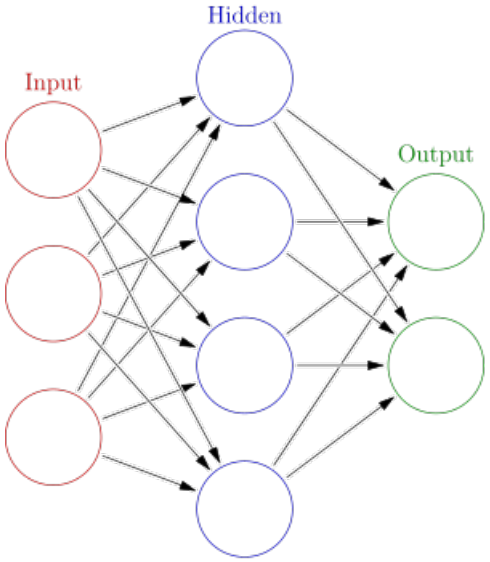


Рис. 3: Принципиальная схема полносвязной нейронной сети.

Зная градиент функции потерь по всем параметрам нейронной сети W можно провести итеративную градиентную минимизацию по параметрам, сведя таким образом задачу функциональной оптимизации в пространстве возможных алгоритмов аппроксимации к задаче параметрической оптимизации. В данной работе мы используем современный метод градиентной оптимизации ADAM [1], представляющий собой улучшенную версию стохастического градиентного спуска SGD.

В процессе градиентной оптимизации минимизируется функция потерь, введенная в главе 3, с добавленным к ней l-2 регуляризатором для уменьшения эффектов переобучения:

$$loss = \sqrt{\frac{1}{2TN \cdot pr} \sum_{t=0}^{T} \sum_{n=1}^{N} \sum_{k=1}^{2pr} (y_{nk}^t - \hat{y}_{nk}^t)^2} + \lambda \sum_{k=0}^{l} ||W^k||^2$$

Здесь коэффициент регуляризации λ выбирается равным малому числу и выступает как характеристика, определяющая соотношение важности минимизации значения исходной функции потерь и минимизации значения l-2 нормы весовых матриц нейронной сети.

Для проведения экспериментов было решено использовать следующую архитектуру полносвязной нейронной сети:

- Используется 3 скрытых слоя. То есть $l = 3$.
- На каждом слое нейронной сети располагается 64, 32 и 16 нейронов.
- Нелинейная функция $g_j(\cdot)$ на всех слоях, кроме выходного - гиперболический тангенс: $g_j(\cdot) = tanh(\cdot)$, $\forall j : 0 < j \leq l$.
- Нелинейная функция на последнем слое выбрана следующим образом: $g_{l+1}(\cdot) = ReLu(tanh(\cdot))$, где

$$ReLu(x) = \begin{cases} x & \text{если } x > 0 \\ 0 & \text{иначе} \end{cases}$$

Интуитивное объяснение такого выбора следующее: точка прорыва воды - момент времени, когда обводненность продукции WCUT становится больше нуля, обычно представляет собой точку, где функция $WWIR(t)$ недифференцируема по t. На практике получается лучше прогнозировать негладкий участок временного ряда используя негладкую же нелинейность на выходном слое. Кроме того, использование $ReLu(\cdot)$ позволяет бороться с отрицательными значениями прогнозируемых переменных, что соответствует физике процесса.

6.2 Линейная регрессия

Линейную регрессию можно считать наиболее простым случаем полносвязной нейронной сети. Полносвязная нейронная сеть с одним слоем и линейной функцией активации идентична линейной регрессии за исключением процедуры обучения. Для последней существует аналитическое решение задачи минимизации функции потерь, а для нейронной сети в общем случае - нет. Основываясь на данной логике мы также обучаем модель линейной регрессии, чтобы оценить, насколько сильно на конечный результат влияют сложность модели и возможность получить точное решение задачи минимизации вместо приближенного, полученного методами итеративной градиентной оптимизации.

Как и в случае с полносвязной нейронной сетью, предполагается наличие у всех объектов одинакового признака, равного единице, и реализующего роль вектора сдвигов в линейной комбинации.

При применении линейной регрессии матрица прогнозируемых переменных для всей выборки $\hat{Y}$ вычисляется следующим образом:

$$\hat{Y} = XW$$

Где $\hat{Y} \in \mathbb{R}^{N \times d}$, $X \in \mathbb{R}^{N \times D}$, $W \in \mathbb{R}^{D \times d}$ для N объектов в выборке и некоторых натуральных D и d, характеризующих размерность признакового описания объектов и размерность векторов прогнозируемых переменных.

Задача минимизации введенной функции потерь с l-2 регуляризацией может быть решена аналитически. Решение записывается в виде:

$$W^* = (X^T X + c\lambda I)^{-1} X^T Y$$

где $c = 2TNpr$, λ - коэффициент регуляризации, I - единичная матрица, $I \in \mathbb{R}^{D \times D}$, Y - матрица с истинными значениями прогнозируемых переменных для всей выборки, $Y \in \mathbb{R}^{N \times d}$.

6.3 Рекуррентная нейронная сеть

Рекуррентные нейронные сети - это семейство архитектур нейронных сетей для работы с последовательностями. Простые рекуррентные сети (Simple Recurrent Networks - SRN) впервые были описаны в работе Джеффри Элмана - "Поиск структуры во времени-[2]. Для простоты выкладок в данной секции будут рассмотрены рекуррентные нейронные сети только с одним скрытым слоем.

Существует ряд различных с точки зрения формата вывода нейронной сети архитектур. Так есть рекуррентные сети, результатом применения которых к последовательности произвольной длины является последовательность той же длины. Есть сети, результатом применения которых к последовательности произвольной длины является вектор фиксированного размера. Также существует еще ряд других архитектур, обсуждение которых выходит за рамки данной работы. Здесь и далее будут рассматриваться только рекуррентные нейронные сети, преобразующие одну последовательность произвольной длины в другую последовательность той же длины. Это интересующий нас случай, так как наша цель - прогнозировать одни временные ряды по другим временным рядам на том же самом временном интервале.

Принцип работы простой рекуррентной сети может быть определен следующим образом:

1. На вход нейронной сети подается последовательность векторов $X = [x_0, .., x_t, .., x_T]$ фиксированной размерности D. Длина самой последовательности T не является фиксированной.

2. Для первого члена последовательности x_0 вычисляется скрытое состояние сети: $h_0 = g_h(W_h x_0 + b_h)$, где $g_h(\cdot)$ - некая нелинейная дифференцируемая функция. Матрица W_h - матрица весов линейной комбинации, $W_h \in \mathbb{R}^{d_h \times D}$, b_h - вектор сдвигов размерности d_h.

3. Для всех последующих членов последовательности x_t скрытое состояние вычисляется следующим образом: $h_t = g_h(W_h x_t + U_h h_{t-1} + b_h$, где матрица U_h - матрица весов линейной комбинации над скрытым состоянием с прошлого шага, $U_h \in \mathbb{R}^{d_h \times d_h}$

4. В случае если используется архитектура с несколькими скрытыми слоями, то каждый новый слой повторяет рекуррентную процедуру, описанную в пунктах 1)-3), но уже не для последовательности X, а для последовательности $H^0 = [h_0, .., h_T]$.

5. Результирующая последовательность $Y = [y_0, .., y_T]$, $y_t \in \mathbb{R}^d \ \forall t \in \{0, .., T\}$ получается из скрытых состояний нейронной сети на последнем слое независимо для каждого члена последовательности: $y_t = g_y(W_y h_t + b_y)$.

Как и полносвязная нейронная сеть, рекуррентная составлена из дифференцируемых частей, что позволяет вычислить градиент дифференцируемой функции потерь по параметрам нейронной сети $W = \{W_h, U_h, b_h, W_y, b_y\}$. Тот факт, что в рекуррентной нейронной сети используются одни и те же весовые матрицы на всех итерациях внутреннего цикла, приводит к видоизмененной версии алгоритма обратного распространения ошибки - к алгоритму обратного распространения ошибки назад во времени.

Простые рекуррентные нейронные сети, по сути, представляют собой очень глубокую сеть. Как и в любой глубокой сети здесь проявляются проблемы затухающих и взрывающихся градиентов. Данная особенность не позволяет обучить простую рекуррентную сеть эффективно работать с последовательностями большой длины.

Наиболее успешной и активно применяющейся сегодня модификацией простой рекуррентной нейронной сети является рекуррентная нейронная сеть с долгосрочной и краткосрочной памятью (Long-Short Term Memory, LSTM) [8]. Данная архитектура показала себя высокоэффективной в задачах обработки текстов, аудио и экономических временных рядов. Отличие LSTM от простой рекуррентной сети заключается в скрытом слое, где вместо линейной комбинации входа и скрытого состояния в прошлый момент времени используется следующая схема:

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f)$$

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i)$$

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o)$$

$$c_t = f_t \circ c_{t-1} + i_t \circ \tanh(W_c x_t + U_c h_{t-1} + b_c)$$

$$h_t = o_t \circ \tanh(c_t)$$

$$\sigma(x) = \frac{1}{1 + \exp(-x)}$$

Символ $\circ$ обозначает Адамарово, почленное умножение.

В данной архитектуре вводится ячейка памяти c_t, отличная от h_t, а также три своего рода "клапана"(gates - ворота): forget gate - f_t, input gate - i_t и output gate - o_t. Каждый из них представляет собой вектор значений из интервала $[0, 1]$, конкретные значения которого зависят от контекста - входа сети на текущей итерации и скрытого состояния с прошлого момента времени. При умножении другого вектора на подобный "клапан"последний определяет долю информации, которая останется после умножения. Клапан f_t отвечает за то, какая доля доля информации будет сохранена от предыдущего значения ячейки памяти. Клапан i_t определяет, какая доля информации будет получена от текущего на этой итерации входа сети. Клапан o_t определяет, какая доля информации, содержащейся в ячейке памяти, будет подана на выход на данной итерации.

Принципиальная схема одной ячейки LSTM нейронной сети изображена на рис. 4.

При проведении экспериментов для LSTM нейронной сети было выбрано то же самое число скрытых слоев и нейронов на каждом слое, что и для полносвязной нейронной сети. А именно, число скрытых слоев выбрано равным 3, а число нейронов на каждом слое: 64, 32, 16. Для минимизации регуляризованной функции потерь также использовался метод градиентной оптимизации - ADAM.

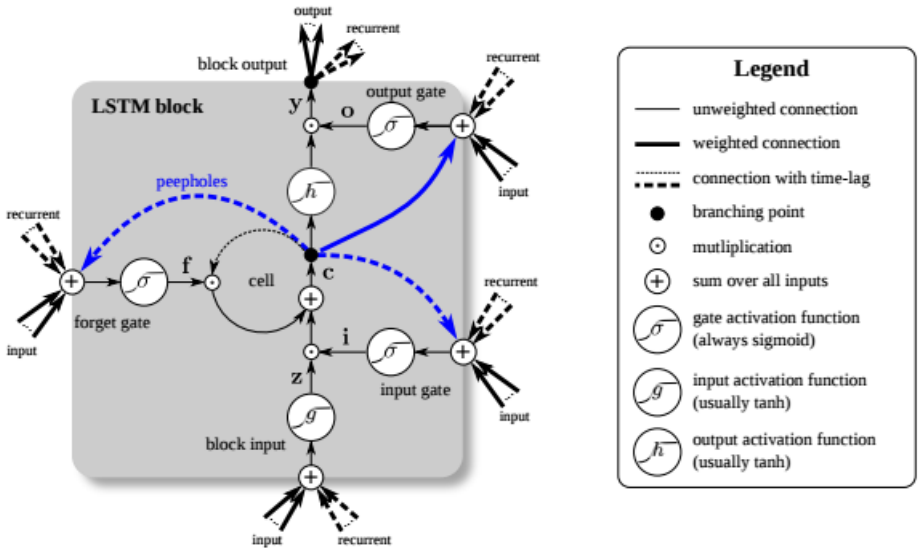


Рис. 4: Принципиальная схема ячейки рекуррентной LSTM нейронной сети.

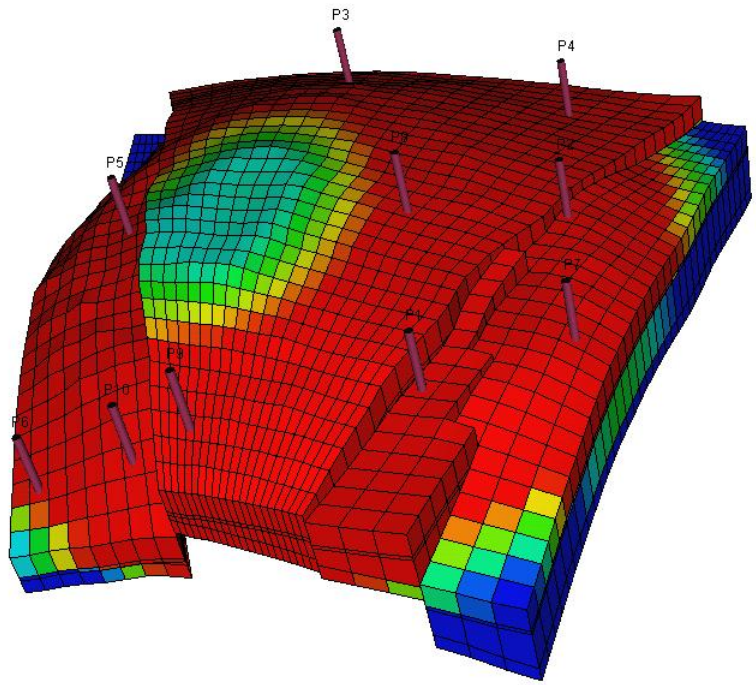


Рис. 5: Геологическая модель нефтяной залежи, используемой в экспериментах. Цветом обозначена доля нефти в порах породы.

6.4 Результаты экспериментов

В данной части работы рассматриваются результаты экспериментального сравнения предложенных моделей машинного обучения.

Методы были опробованы на модели реального участка нефтяной залежи. Гидродинамическая структура пласта достаточно сложная: имеются поверхности экранирования (поверхности, ток жидкости через которые невозможен или сильно затруднен), нефть в залежи подпирается водой снизу, то есть в залежи имеется внутренний источник воды, кроме той воды, которая нагнетается в скважины. Геологическая структура залежи изображена на рис [бла]. Залежь разбита на 35x35x5 ячеек: по 35 в двух горизонтальных направлениях и по 5 в вертикальном. На модели расположено 10 скважин: 5 добывающих и 5 нагнетательных. Схема расстановки скважин неравномерная по залежи, взаимное расположение добывающих и нагнетательных скважин близко к шахматному порядку. Геологическая модель залежи в процессе разработки изображена на рис. 5, цветом изображена доля нефти в порах породы.

Генерация режимов работы скважин проводилась в соответствии с алгоритмом, описанным в главе 4. Было сгенерировано 100 сценариев продолжительностью 9 лет. Дискретизация временных рядов - 5 суток. То есть один временной ряд продолжительностью 9 лет содержит в себе 657 точек.

В качестве конечно-разностного гидродинамического симулятора используется программное обеспечение Schlumberger ECLIPSE. Все вычисления гидродинамического симулятора проводились на высокопроизводительном компьютере, однако небольшой размер залежи не позволяет использовать более двух ядер процессора параллельно для моделирования одного сценария разработки залежи.

С помощью гидродинамического симулятора были вычислены истинные значения прогнозируемых переменных: WWIR и WGOR. Было построено признаковое пространство входов для алгоритмов машинного обучения и все переменные были отнормированны в соответствии с главой 5.

Для целей контроля за переобучением 15% выборки было отложено в качестве валидационных - модели машинного обучения не настраиваются на отложенных элементах выборки, на них лишь проверяется качество моделей в процессе и в результате обучения.

Для обучения нейронных сетей использовался графический ускоритель.

MODEL	LSTM-RNN	FCNN	Linear Regression
WWIR error	0.0036	0.0044	0.0040
WGOR error	0.104	0.113	0.105

Таблица 1: Качество работы алгоритмов машинного обучения на валидационной выборке

В таблице 1 приведены результаты тестирования моделей на валидационной подвыборке. Ошибка модели вычисляется отдельно для WWIR и для WGOR в масштабе до нормировки. В качестве ошибки используется корень из среднего квадратичного отклонения между предсказанным и истинным значениями (Rooted Mean Squared Error - RMSE):

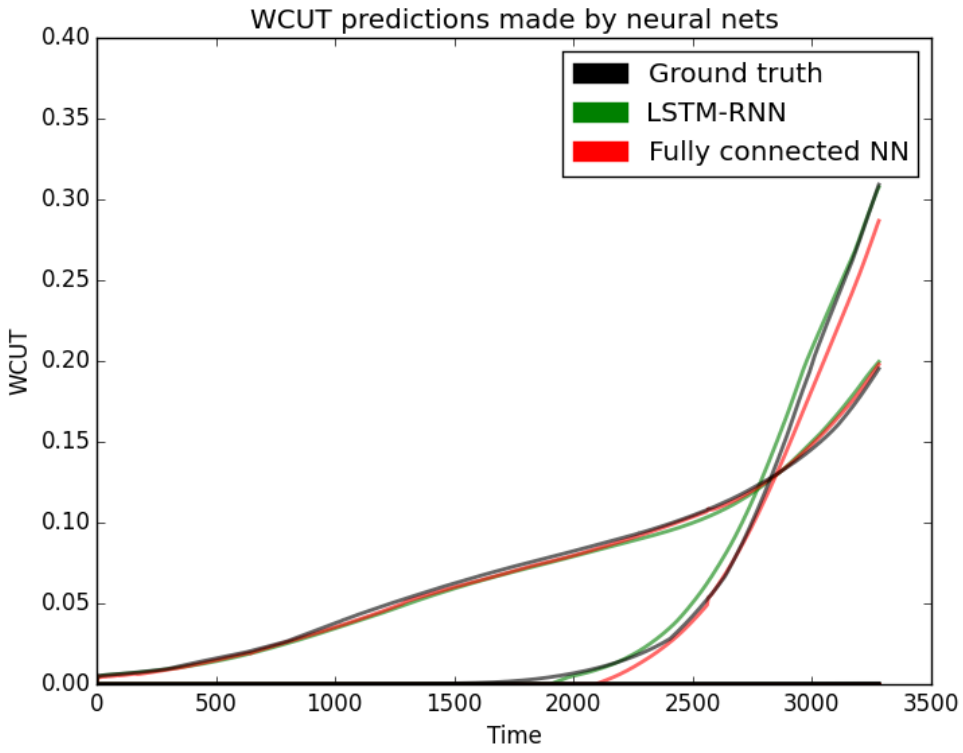


Рис. 6: Значение обводненности продукции WCUT на случайном сценарии, посчитанное гидродинамическим симулятором (ground truth), полносвязной и рекуррентной нейронными сетями.

Все три модели машинного обучения показали достаточно высокие показатели качества - погрешность определения свойств залежи при построении модели вносит намного большую ошибку в прогнозы, делаемые самим конечно-разностным симулятором.

Сравнение прогнозируемых нейронными сетями переменных WCUT и WGOR с их истинным значением можно провести по рисункам 6 и 7.

Однако, наилучшее качество было полученно для рекуррентной LSTM нейронной сети. Кроме прочего принцип работы рекуррентной нейронной сети во многом напоминает принцип работы самого гидродинамического симулятора. Как и рекуррентная нейронная сеть, гидродинамический симулятор в каждый момент времени получает информацию о режимах работы скважин в данный момент времени, на основе полученных данных пересчитывает вектор внутреннего состояния залежи (распределение давления и нефтенасыщенности по ячейкам залежи), а затем из этого внутреннего состояния пересчитывает прогнозируемые переменные для скважин. То есть рекуррентную нейронную сеть можно считать в чем-то соответствующей физике процесса.

Стоит также отметить, что между моделями, обрабатывающими каждый момент времени независимо - линейной регрессией и полносвязной нейронной сетью практически нет разницы в качестве прогнозирования, хотя нейронная сеть потенциально способна прогнозировать существенно более сложные зависимости, чем линейная регрессия. Данный результат говорит о том, что в данной задаче важно явно учитывать временные зависимости.

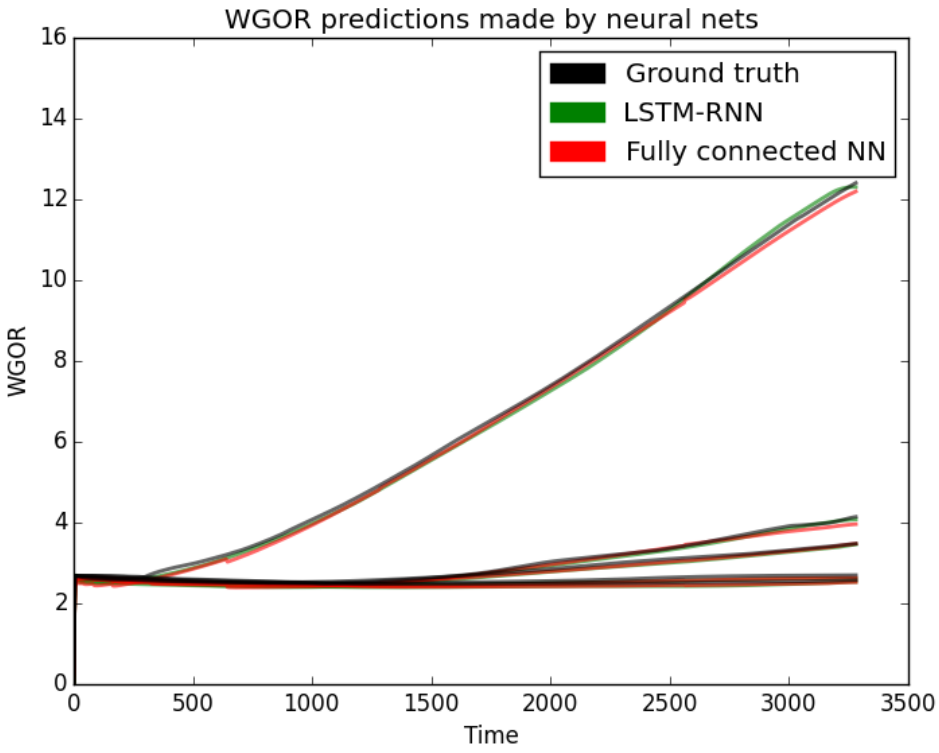


Рис. 7: Значение газо-нефтяного фактора WGOR на случайном сценарии, посчитанное гидродинамическим симулятором (ground truth), полносвязной и рекуррентной нейронными сетями.

По скорости обучения линейная регрессия является неоспоримым лидером - скорость обучения выше аналогичной для нейронных сетей на два порядка. Однако, в данной задаче доминирующее влияние оказывает время работы гидродинамического симулятора, поэтому скорость обучения модели машинного обучения не имеет решающего значения.

7 Генерация данных при помощи градиентной оптимизации

В данной главе будет рассмотрен способ повышения репрезентативности данных, на которых настраиваются алгоритмы машинного обучения.

Конечной целью разрабатываемых в данной работе моделей является прогнозирование искомых переменных с высокой точностью в условиях реальной разработки месторождения. Режимы работы скважин будут выбираться вручную инженером и для этих режимов будут запускаться предложенные в данной работе модели.

При обучении моделей к данным предъявляются стандартные для задач машинного обучения требования:

- Объекты выборки сгенерированы независимо из некого распределения.
- Объекты обучения и валидации сгенерированы из одного и того же распределения.

Второе требование предполагает, что обучение должно производиться на объектах, сгенерированных из того же самого распределения, из которого они будут выбраны при реальном использовании методов. Описать такое распределение не представляется возможным поскольку в реальности режимы работы скважин выбираются не случайно и определены физическими свойствами моделируемой системы.

Необходимо создать алгоритм для получения режимов работы скважин, максимально приближенных к реальным. При этом необходимо сохранить возможность случайной генерации относительно большого числа разнообразных режимов работы скважин.

Для решения поставленной задачи обратимся к логике, которой пользуется инженер, проводящий моделирование разработки конкретной залежи. Скорее всего инженер будет проверять те сценарии разработки, которые позволяют достичь высоких показателей разработки и низких экономических издержек. Сценарии, заведомо неудачные, скорее всего тестироваться не будут.

Можно ввести функцию качества сценария, которая будет иметь высокие значения для сценариев, выгодных с точки зрения показателей разработки и экономических издержек, и низкие для сценариев невыгодных.

В данной работе было решено в роли такой функции качества сценария выбрать суммарный объем добытой нефти из залежи за время разработки:

$$L = \sum_{j=1}^{pr} \sum_{t=0}^{T} WLPR_j(t)(1 - WCUT_j(t))$$

Поскольку все предложенные алгоритмы машинного обучения являются полностью дифференцируемыми, мы можем вычислить градиент введенной функции качества по вектору входов - $\nabla_x L$. Имея данный градиент мы можем изменить векторв входов x в направлении градиента так, чтобы значение функции качества увеличилось:

$$x \leftarrow x + \eta \nabla_x L$$

где η - длина шага градиентной максимизации, которая подбирается вручную.

Таким образом, если имеется выборка объектов - режимов работы скважин из произвольного распределения, то, применяя один шаг градиентной оптимизации к объектам данной выборки, можно получить новую выборку объектов, более реалистичных в смысле, оговоренном ранее.

Как только новая выборка режимов работы скважин получена, необходимо вычислить для каждого объекта этой выборки истинные значения прогнозируемых переменных с помощью гидродинамического симулятора. Далее на этой выборке возможно дообучить метод машинного обучения.

Данный процесс, состоящий из градиентного шага в пространстве сценариев и дообучении моделей машинного обучения, можно запустить итерационно. На каждой итерации модель машинного обучения будет настраиваться на все более реалистичной выборке объектов, тем самым качество прогнозирования в реальных условиях для таким образом обученной модели должно возрасти.

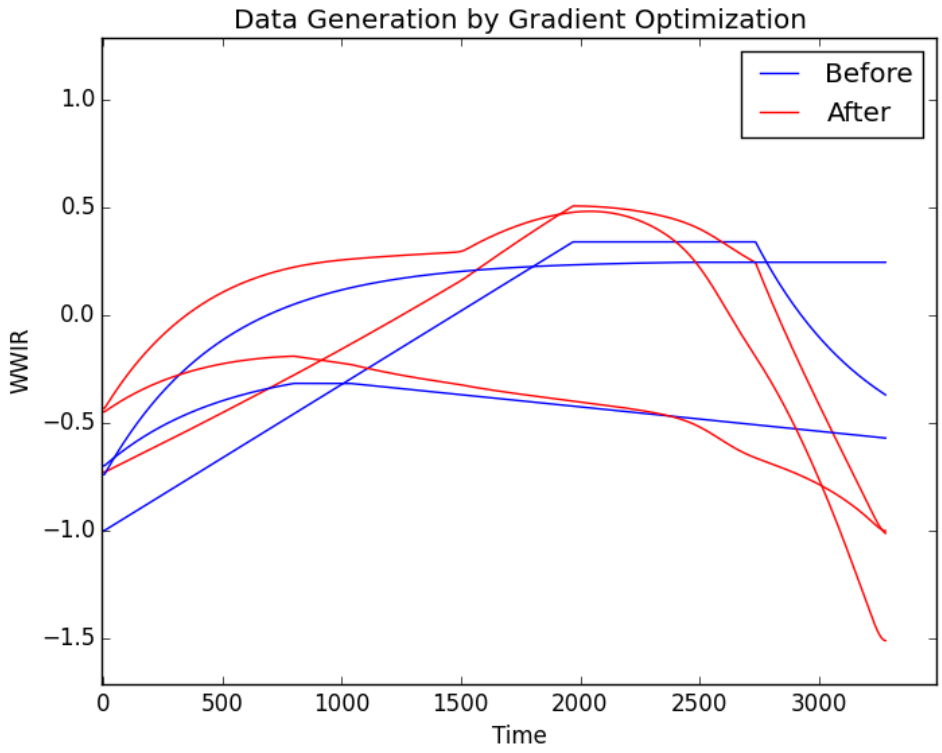


Рис. 8: Режимы работы трех нагнетательных скважин до и после применения градиентной оптимизации в пространстве сценариев.

Предложенный метод генерации данных при помощи градиентной оптимизации был опробован на рекуррентной LSTM нейронной сети, как на наиболее качественной из трех предложненных. Было сделано 3 шага градиентной оптимизации в пространстве режимов работы скважин. Длина градиентного шага выбиралась так, чтобы визуальные различия между сценариями до и после градиентного шага были заметны, но и не слишком велики. Результат применения метода можно увидеть на рис. 8.

Хотя это не было целью предложенного метода, но при расширении выборки новыми объектами ошибка прогноза рекуррентной нейронной сети уменьшилась с ??? на ???. При этом валидационная выборка, на которой вычислялась ошибка, никак не изменялась и никаким модификациям, описанным в этой главе, не подвергалась.

8 Выводы

В данной работе был предложен метод аппроксимации гидродинамической модели нефтяной залежи при помощи моделей машинного обучения, в частности нейронных сетей. Предложенные методы показали низкую ошибку прогнозирования при высокой скорости обучения и работы.

Предлагаемые методы обладают большей гибкостью в выборе признакового описания объектов и целевых переменных прогнозирования, чем полуфизические методы, рассмотренные в главе 2.

В данной работе проведен подробный обзор используемых архитектур нейронных сетей, описание способов настройки моделей машинного обучения. Подобный анализ отсутствует в других опубликованных работах по данной тематике, доступных для чтения.

Судя по анализу литературы по данной тематике, данная работа - единственная, предлагающая такой метод быстрой аппроксимации результатов работы гидродинамического симулятора, который в режиме эксплуатации использует только наблюдаемые на поверхности переменные и не накладывает никаких ограничений на сложность гидродинамической структуры залежи.

Был проведен сравнительный анализ моделей машинного обучения при решении поставленной задачи. Была показана эффективность рекуррентных нейронных сетей по сравнению с другими моделями.

Был предложен метод для случайной генерации реалистичных сценариев разработки при помощи методов градиентной оптимизации.

9 Заключение

Метод аппроксимации гидродинамической модели нефтяной залежи, предложенный в данной работе, может быть использован на практике для месторождений на средней и поздней стадии разработки, когда бурение новых скважин прекращено и имеется время для создания обучающей выборки.

Рассмотренный метод имеет широкий спектр возможных дополнительных направлений исследований. Данные исследования могут быть направлены на повышение практической применимости метода. Это могут быть:

- Расширение модели для возможности добавления новых скважин, которых не было в обучающей выборке.
- Анализ минимального достаточного для качественного прогнозирования количества объектов в обучающей выборке.
- Введение системы, позволяющей модели оценить уверенность в своем ответе.
- Прогнозирование не только характеристик добываемой из скважин смеси, но и распределения величин порового давления и нефтенасыщенности по ячейкам гидродинамической модели. Данные величины полностью характеризуют состояние залежи. На основе этого расширения можно будет строить гибридные модели, состоящие из медленных, но универсальных, конечно-разностных моделей и быстрых, но работающих в ограниченном множестве задач, моделей машинного обучения.

На данный момент ведутся исследования по последнему пункту.

Планируется публикация работы на одной из конференций Society of Petroleum Engineers - SPE.

Список литературы

[1] J. Ba D. P. Kingma. *Adam: A Method for Stochastic Optimization.* // Published as a conference paper at the 3rd International Conference for Learning Representations, San Diego, 2015.

[2] J. L. Elman. *Finding Structure in Time.* // Cognitive Science 14.2, с. 179–211, 1990.

[3] A. Emre. *Characterizing Reservoir Connectivity and Forecasting Waterflood Performance Using Data-Driven and Reduced-Physics Models.* // SPE Western Regional Meeting, 2016.

[4] L. W. Lake F. Cao H. Luo. *Development of a Fully Coupled Two-phase Flow Based Capacitance Resistance Model (CRM).* // SPE Improved Oil Recovery Symposium, 2014.

[5] L. W. Lake F. Cao H. Luo. *Oil Rate Forecast by Inferring Fractional Flow Models from Field Data.* // SPE Reservoir Simulation Symposium, 2015.

[6] K. Li H. Chen Z. Zhang. *Modification of Capacitance-Resistive Model for Estimating Waterflood Performance.* // CPS/SPE International Oil Gas Conference и Exhibition, 2010.

[7] Williams R.J. Rumelhart D.E. Hinton G.E. *Learning Internal Representations by Error Propagation.* // Parallel Distributed Processing, vol. 1, с. 318—362. Cambridge, MA, MIT Press, 1986.

[8] J. Schmidhuber S. Hochreiter. *Long Short-Term Memory.* // Neural Computation, vol. 9, с. 1735–1780, 1997.

[9] W. Pitts W. S. McCulloch. *A logical calculus of the ideas immanent in nervous activity.* // Bulletin of Mathematical Biology — New York: Springer New York, v. 5, № 4. — с. 115–133, 1943.

[10] P. J. Werbos. *Beyond regression: New tools for prediction and analysis in the behavioral sciences.* // Harvard University, Cambridge, MA, 1974.

[11] Галушкин А. И. *Синтез многослойных систем распознавания образов.* // М.: «Энергия», 1974.