

УДК 004.94

doi: 10.15622/rcai.2025.014

ИССЛЕДОВАНИЕ И РЕАЛИЗАЦИЯ МЕТОДОВ ИНТЕЛЛЕКТУАЛЬНОГО АНАЛИЗА ДАННЫХ ДЛЯ ПРОГНОЗИРОВАНИЯ ОБВОДНЁННОСТИ ПРОДУКЦИИ НЕФТЯНЫХ СКВАЖИН¹

И.Б. Фоминых (*igborfomin@mail.ru*)

И.С. Михайлов (*fr82@mail.ru*)

К.О. Сидоров (*kirill.sidoroff2014@yandex.ru*)

Мью Хлайн Вин (*myohlaingwin69287@gmail.com*)

Национальный исследовательский университет «МЭИ», Москва

В работе предлагается решение актуальной задачи прогнозирования обводнённости продукции нефтяных скважин с использованием методов интеллектуального анализа данных. Рассмотрены методы интеллектуального анализа данных: LSTM, BiLSTM, Prophet, ARIMA, XGBoost, NNAR и TBATS. Выявлены их сильные и слабые стороны. Выполнена реализация и тестирование данных методов с использованием реальных данных, полученных на нефтяных месторождениях. Показано, что для краткосрочного прогнозирования лучше использовать модели LSTM, BiLSTM и NNAR, для долгосрочного прогнозирования изменения обводнённости продукции скважин лучше использовать LSTM модель.

Ключевые слова: интеллектуальный анализ данных, искусственная нейронная сеть LSTM, прогнозирование, обводнённость продукции нефтяных скважин.

Введение

Уровень применения методов интеллектуального анализа данных в современных информационных системах неуклонно растёт. Данные методы используются для решения различных задач, связанных с классификацией, регрессией, прогнозированием, ассоциацией и другими. Наиболее сильно данная тенденция выражена в информационных системах, которые

¹ Работа выполнена при финансовой поддержке РФФ (проект № 24-11-00285), <https://rscf.ru/project/24-11-00285/>.

отвечают за анализ и управление различными технологическими процессами, поскольку объём обрабатываемой ими информации весьма велик и обычные математические модели требуют слишком детальной настройки на каждый отдельный случай [Барабанов и др., 2019]. В частности, к таким задачам относится задача анализа изменения обводнённости продукции скважин в нефтяной промышленности. Данный параметр наблюдается в режиме реального времени с помощью многофазного расходомера, устанавливаемого на устье нефтяных скважин. Необходимо отметить, что при увеличении обводнённости продукции скважины необходимо принимать управляющие воздействия, направленные на изменение режима добычи или технологии добычи нефти на ней, поскольку в противном случае скважина может стать нерентабельной. Также, в случае роста обводнённости может измениться режим течения продукции скважины, что приведёт к нештатной работе добывающего оборудования и возможной остановке добычи. Поэтому очень важно отслеживать данный параметр и качественно предсказывать его значение.

Однако этот процесс осложняется множеством факторов, включая свойства пласта, параметры оборудования и режим эксплуатации, что делает традиционные методы анализа недостаточно точными [Bian et al, 2022].

В связи с большим количеством обрабатываемых данных возникает необходимость в применении методов интеллектуального анализа, способных учитывать нелинейные зависимости и выявлять скрытые закономерности [Dong, 2022].

Данное исследование посвящено решению задачи прогнозирования изменения обводнённости продукции нефтяных скважин на основе методов машинного обучения. При этом необходимо выполнить анализ существующих подходов для решения задачи прогнозирования временных рядов, выявить их преимущества и ограничения, реализовать алгоритмы машинного обучения для решения поставленной задачи, выполнить тестирование и сравнение моделей на реальных данных измерительных систем, установленных на нефтяных скважинах [Sarker, 2021].

1. Измерение обводнённости продукции скважин

В качестве исходных данных для решения задачи прогнозирования обводнённости продукции скважин рассматриваются показания многофазных расходомеров, установленных на нефтяных скважинах. Они регистрируют в режиме реального времени следующий набор параметров: скорость нефте-водо-газового потока, газосодержание потока, скорость прохождения ультразвука в контролируемом объёме, температуру, давление. Расчётными параметрами являются: расход жидкости, расход газа, обводнённость продукции. Согласно разработанным математическим моделям

показано, что указанный набор параметров образует многомерное пространство, определяющее расходные параметры потока, в том числе обводненность потока.

Массив обучающих данных представляет собой набор результатов измерений расходомеров на скважинах с различными химическими параметрами продукции, с различным оборудованием добычи и различными режимами течения продукции.

Для повышения достоверности анализа проведена предобработка данных, включающая удаление следующих строк:

- 1) постоянных в течение длительного времени данных, поскольку они соответствуют остановке нефте-водо-газового потока на измерительном участке;
- 2) с нулевой скоростью ультразвука (ошибки измерений);
- 3) с одинаковой скоростью в разных точках измерения (некорректные данные);
- 4) с 100% содержанием газа, поскольку это означает, что измерительный участок заполнен попутным нефтяным газом;
- 5) с неизменной скоростью ультразвука при изменяющихся других параметрах (возможное зависание сенсора).

Очищенные данные сохранены в CSV-файл для дальнейшего анализа.

2. Методы интеллектуального анализа данных для решения задачи прогнозирования

В данном исследовании рассматриваются следующие наиболее популярные методы решения задачи прогнозирования.

2.1. LSTM

Long short-term memory (LSTM) – рекуррентная нейронная сеть (RNN), которая используется для решения задач обработки естественного языка, распознавания речи, музыкальной композиции и других задач, где есть последовательность входных данных. LSTM способна обрабатывать длинные последовательности данных с запоминанием предыдущих состояний. Данные сети были разработаны для решения проблемы исчезающего градиента, с которой можно столкнуться при обучении традиционных RNN [Hochreiter et al., 1997].

В отличие от других алгоритмов машинного обучения, рекуррентные нейронные сети с долгой кратковременной памятью способны автоматически выявлять признаки из временных последовательностей, обрабатывать многомерные данные, а также выводить последовательности переменной длины, благодаря чему их можно использовать для прогнозирования временных рядов.

LSTM-сети демонстрируют высокую эффективность при работе с временными рядами благодаря своей способности анализировать долгосрочные временные зависимости. Их архитектура специально разработана для решения задач классификации, обработки и прогнозирования последовательных данных, где критически важные события могут быть разделены произвольными временными интервалами. LSTM-архитектура была создана в качестве решения ключевой проблемы традиционных рекуррентных сетей – эффекта исчезающих градиентов. В отличие от стандартных RNN и скрытых марковских моделей, LSTM демонстрируют устойчивую работу с длинными временными зависимостями, что обеспечивает их превосходство в различных задачах обработки последовательностей.

2.2. BiLSTM

Бидирекциональная (двунаправленная) сеть с долгой кратковременной памятью (BiLSTM) представляет собой архитектуру рекуррентных нейронных сетей, которая расширяет традиционную LSTM за счёт одновременной обработки входных данных в двух направлениях: прямом и обратном. В отличие от классической LSTM, выполняющей анализ последовательности только в хронологическом порядке (с начала к концу), BiLSTM использует две параллельные LSTM-сети – одна обрабатывает данные в прямом направлении, другая – в обратном. Результаты обеих сетей объединяются, что позволяет модели учитывать контекст как из предыдущих, так и из последующих элементов последовательности.

Данная архитектура особенно эффективна в задачах, где значение текущего элемента зависит от информации, расположенной как в прошлом, так и в будущем, например, при обработке естественного языка, распознавании речи, анализе временных рядов и других последовательных данных. Благодаря двунаправленному подходу BiLSTM способна лучше выявлять долгосрочные зависимости и улучшать качество предсказаний по сравнению с однонаправленными моделями [Schuster et al., 1997].

Принцип работы BiLSTM заключается в следующем: входная последовательность сначала подаётся на прямую LSTM-сеть, которая формирует скрытые состояния, учитывающие контекст слева направо. Затем та же последовательность обрабатывается в обратном порядке другой LSTM-сетью, формирующей скрытые состояния с учётом контекста справа налево. Итоговые представления, полученные из обеих направлений, объединяются (например, посредством конкатенации), что обеспечивает более полное описание каждого элемента с учётом двунаправленного контекста.

2.3. Prophet

Prophet – это метод прогнозирования временных рядов, основанный на аддитивной модели, которая учитывает нелинейные тренды, отражающие годовые, недельные и суточные сезонные колебания, а также влияние

праздничных дней. Данный инструмент особенно эффективен при анализе временных рядов с выраженными сезонными паттернами и наличием нескольких циклов исторических данных. Prophet устойчив к пропущенным значениям и резким изменениям тренда, а также обычно хорошо справляется с аномалиями в данных [Taylor, Letham, 2017].

Prophet имеет несколько нижеприведенных преимуществ:

- Prophet, благодаря точности и скорости, используется во многих приложениях в Facebook для создания надежных прогнозов для планирования и постановки целей.
- Prophet создает настраиваемые прогнозы, представляет множество возможностей для пользователей создавать и корректировать прогнозы. Возможно использовать интерпретируемые человеком параметры для улучшения прогноза, добавляя знания о предметной области.
- Доступен в R или Python.
- Хорошо обрабатывает сезонные колебания. Prophet учитывает сезонность с несколькими периодами.
- Устойчив к выбросам (обрабатывает выбросы, удаляя их).

2.4. ARIMA

ARIMA – это аббревиатура, которая расшифровывается как Auto Regressive Integrated Moving Average (Авторегрессивное интегрированное скользящее среднее). Это алгоритм прогнозирования, основанный на идее, что информация в прошлых значениях временного ряда может использоваться сама по себе для прогнозирования будущих значений [Box et al., 1994].

Любой несезонный временной ряд, который демонстрирует закономерности и не является случайным белым шумом, может быть смоделирован с помощью моделей ARIMA.

Аббревиатура ARIMA является описательной и отражает ключевые аспекты самой модели. Вкратце, они таковы:

AR: Авторегрессия. Модель, которая использует зависимую связь между наблюдением и некоторым количеством наблюдений с лагом.

I: Интегрированная. Использование дифференциации сырых наблюдений (например, вычитание наблюдения из наблюдения на предыдущем временном шаге) для того, чтобы сделать временной ряд стационарным.

MA: Скользящее среднее. Модель, которая использует зависимость между наблюдением и остаточной ошибкой из модели скользящего среднего, применяемой к запаздывающим наблюдениям. Каждый из этих компонентов явно указан в модели как параметр. Используется стандартная нотация $ARIMA(p,d,q)$, где параметры заменяются целыми значениями для быстрого указания конкретной используемой модели ARIMA.

Параметры модели ARIMA определяются следующим образом: d - количество разностей, необходимых для того, чтобы сделать временной ряд стационарным, p - количество наблюдений запаздывания, включенных в модель, т.е. порядок члена AR, q - размер окна скользящего среднего, т.е. порядок члена MA.

Таким образом, цель обучения модели состоит в том, чтобы определить значения p , d и q .

2.5. XGBoost

XGBoost (eXtremeGradientBoosting) – это высокоэффективная реализация алгоритма градиентного бустинга на основе деревьев решений.

Бустинг – это метод, при котором модели (чаще всего деревья решений) обучаются последовательно, а не независимо, как в случайных лесах. Каждое последующее дерево концентрируется на исправлении ошибок, сделанных предыдущими моделями. Итоговый прогноз получается, как взвешенная сумма предсказаний всех деревьев [Chen et al., 2016].

К основным параметрам модели относятся: количество деревьев, максимальная глубина дерева, скорость обучения, минимальное уменьшение функции потерь для разбиения узла, доля данных для обучения каждого дерева, доля признаков для построения дерева, коэффициенты L1 и L2 регуляризации, функция потерь.

Основные особенности XGBoost, отличающие его от других алгоритмов градиентного бустинга:

- Адаптивный штраф при построении деревьев.
- Пропорциональное уменьшение узлов листьев.
- Метод Ньютона в оптимизации.
- Дополнительный параметр рандомизации.
- Автоматический отбор признаков.

2.6. NNAR

NNAR (NNETAR) (Neural Network AutoRegression) – это гибридная модель, сочетающая:

- Авторегрессию (AR) – использование лагов временного ряда как признаков. В классической модели AR прогноз строится на основе предыдущих значений ряда. Например, значение в момент t зависит от значений в моментах $t-1, t-2, \dots, t-p$.
- Нейронную сеть с одним скрытым слоем – для учета нелинейных зависимостей. Вместо простой линейной комбинации прошлых наблюдений, NNETAR использует многослойную перцептронную нейросеть, которая способна моделировать сложные нелинейные зависимости [Gregor et al., 2014].

2.7. TBATS

TBATS – это аббревиатура для обозначения тригонометрической сезонности (Trigonometric seasonality) преобразования Бокса-Кокса (Box-Cox transformation) ошибки ARMA (ARMA errors) тренда (Trend) сезонных компонент (Seasonal components). TBATS был разработан для прогнозирования временных рядов с несколькими сезонными периодами. Например, ежедневные данные могут иметь как недельный, так и годовой шаблон. Или часовые данные могут иметь три сезонных периода: ежедневный шаблон, недельный шаблон и годовой шаблон [De Livera et al., 2011].

В модели TBATS к исходному временному ряду сначала применяется преобразование Бокса-Кокса, после чего ряд описывается как линейное сочетание экспоненциально сглаженного тренда, сезонных компонентов и компонента ARMA. Для моделирования сезонности используются тригонометрические функции, основанные на разложении в ряды Фурье. Выбор и настройка гиперпараметров модели, таких как включение или исключение отдельных компонентов, осуществляется с помощью критерия информационного критерия Акаике (AIC).

3. Реализация рассмотренных методов и сравнение результатов тестирования

Рассмотренные методы были реализованы на языке Python. Также было выполнено их тестирование на подготовленных наборах данных и произведено их сравнение по основным метрикам для поиска оптимальной модели. Прогнозирование осуществлялось на интервалах различной длительности.

Результаты сравнения представлены в табл. 1.

Таблица 1

Сравнение оценок результатов прогнозирования с помощью различных моделей

Модель	Интервал	MAE	MSE	RMSE	R ²	MAPE (%)	WAPE (%)
LSTM	30 сек	1.2513	4.2588	2.0637	0.8829	1.51	1.38
	1 мин	1.8107	6.7384	2.5958	0.9584	2.58	2.29
	2 мин	1.8252	5.6176	2.3702	0.9456	2.53	2.41
	5 мин	1.9252	5.4583	2.3363	0.9614	2.56	2.49
	1 час	2.7792	13.2882	3.6453	0.9661	4.98	4.07
BiLSTM	30 сек	2.1495	8.6997	2.9495	0.7608	2.53	2.37
	1 мин	2.4367	9.5888	3.0966	0.9407	3.35	3.09
	2 мин	2.4726	8.9197	2.9866	0.9136	3.35	3.26
	5 мин	2.2178	7.0866	2.6621	0.9499	2.97	2.87
	1 час	2.7607	14.1673	3.7640	0.9639	5.09	4.04

Модель	Интервал	MAE	MSE	RMSE	R ²	MAPE (%)	WAPE (%)
Prophet	30 сек	28.4125	843.6725	29.0460	-22.2004	31.06	31.38
	1 мин	16.7951	444.0584	21.0727	-1.7440	19.20	21.27
	2 мин	13.7081	291.3234	17.0682	-1.8212	16.69	18.07
	5 мин	15.2648	373.0373	19.3142	-1.6370	17.92	19.72
	1 час	17.5568	428.0774	20.6900	-0.0919	32.25	25.68
ARIMA	30 сек	7.5700	99.8300	9.9900	0.7977	17.42	11.61
	1 мин	11.9700	256.3900	16.0100	0.4804	28.00	18.34
	2 мин	15.5000	388.7300	19.7200	0.2122	38.89	23.74
	5 мин	18.1800	498.1300	22.3200	-0.0096	44.78	27.85
	1 час	21.2600	632.9900	25.1600	-0.2829	42.63	32.57
XGBoost	30 сек	2.4243	7.9973	2.8280	0.7801	2.74	2.68
	1 мин	2.7073	13.1424	3.6252	0.9188	3.66	3.43
	2 мин	3.1110	14.4441	3.8005	0.8601	4.26	4.10
	5 мин	3.0611	14.4515	3.8015	0.8978	4.13	3.95
	1 час	3.6273	21.8579	4.6752	0.9442	6.55	5.31
NNAR	30 сек	2.6841	7.6938	2.7738	0.7884	3.02	2.96
	1 мин	2.6474	8.6734	2.9451	0.9464	3.50	3.35
	2 мин	2.1953	7.6499	2.7659	0.9259	2.93	2.89
	5 мин	2.3024	8.0152	2.8311	0.9433	3.07	2.97
	1 час	3.3613	20.0977	4.4830	0.9487	6.07	4.92
TBATS	30 сек	8.2331	87.3586	9.3466	-1.4023	9.47	9.09
	1 мин	11.9814	174.5704	13.2125	-0.0787	16.39	15.18
	2 мин	10.8501	162.5795	12.7507	-0.5744	14.54	14.30
	5 мин	11.8181	193.0253	13.8934	-0.3645	15.72	15.27
	1 час	17.6330	526.5015	22.9456	-0.3429	39.87	25.80

На основе полученных результатов можно сделать следующие выводы.

3.1. LSTM

Обладает хорошей производительностью (R^2 около 0.88–0.96), что указывает на сильную способность модели объяснять данные.

MAE и RMSE увеличиваются с увеличением интервала времени, что нормально для временных рядов.

По MAPE и WAPE модель демонстрирует хорошие результаты с низкими значениями данных оценок.

Высокая частота данных позволяет LSTM эффективно улавливать краткосрочные зависимости.

LSTM запоминает предыдущие значения, что важно для плавно меняющихся метрик.

Dropout и Dense слои помогают обобщить данные и сгладить влияние шума.

3.2. BiLSTM

R^2 выше для интервалов времени 1 минута и больше (0.94 и выше), что говорит о хорошей объясняющей способности.

Однако MAE и RMSE в некоторых случаях хуже, чем у LSTM, особенно на 30 секунд и 1 минуту.

MAPE и WAPE немного выше по сравнению с LSTM.

3.3. Prophet

Очень слабая производительность по R^2 (от -22 до -0.09), что указывает на плохую способность модели объяснять данные. Модель явно не подходит для данной задачи.

MAE, MSE, RMSE, MAPE и WAPE также имеют очень высокие значения, что делает эту модель непригодной для прогнозирования в данном контексте. Prophet плохо работает с высокочастотными данными, он ориентирован на дневной и недельный масштаб изменения наблюдаемых данных. Модель не учитывает краткосрочные зависимости, важные при интервалах 30 сек – 5 мин. Отсутствие автокорреляции в ядре модели делает её уязвимой к шуму и быстрым случайным отклонениям какой-либо величины.

3.4. ARIMA

R^2 ухудшается с увеличением интервала времени (от 0.80 до отрицательных значений), что указывает на плохую предсказательную способность модели на длительные интервалы.

MAE и RMSE также показывают большие ошибки на больших интервалах времени.

MAPE и WAPE также высоки, что делает модель менее эффективной по сравнению с нейронными сетями.

Приведение ряда к стационарному виду улучшает качество, но ARIMA всё ещё линейная модель, и не справляется с нелинейными и асимметричными паттернами в метрике.

При дифференцировании модель теряет часть долгосрочной динамики, что делает её прогноз менее устойчивым на горизонтах >2 минут.

Также, ARIMA не учитывает взаимодействие с другими переменными, и не имеет механизма адаптации к смене режима генерации данных (например, резкие падения).

3.5. XGBoost

XGBoost характеризуется хорошим значением R^2 (0.78–0.94), с улучшением на более длинных интервалах времени.

MAE и RMSE показывают стабильные результаты по сравнению с другими моделями.

Значения MAPE и WAPE невысокие, что делает эту модель достаточно эффективной для прогнозирования.

XGBoost использует бустинг деревьев решений, которые обнаруживают сложные нелинейные зависимости между лагами.

Хорошо обрабатывает высокочастотные шумные данные, если заранее заданы временные признаки (лаги, rollingwindow и др.).

Если нет предположений о стационарности, то модель легко адаптируется к изменяющимся паттернам.

Качество зависит от исходных признаков, поэтому при хорошем представлении признаков XGBoost близок к LSTM по точности.

3.6. NNAR

NNAR характеризуется хорошими значениями R^2 (0.95 и выше) на интервалах более одной минуты.

MAE и RMSE относительно стабильны и показывают хорошие результаты.

MAPE и WAPE также находятся на приемлемом уровне, с улучшением на более длительных интервалах времени.

Использует отложенные лаги как входы, что приближает данную модель к MLP-архитектуре, ориентированной на временные ряды.

Простая архитектура снижает риск переобучения и хорошо усваивает шумную, но структурированную динамику.

В отличие от ARIMA, не требует стационарности и может обучиться нелинейным трансформациям.

3.7. TBATS

TBATS показывает очень слабую производительность (R^2 находится в интервале от отрицательных значений до 0.70), что делает модель неэффективной для данной задачи.

MAE, MSE, RMSE, MAPE и WAPE также показывают очень высокие значения, что подтверждает её плохие результаты.

TBATS предназначен для выраженных сезонных рядов с длинными периодами (например, годовые/недельные циклы). В рассматриваемых данных не наблюдается сезонность.

Использует сложные компоненты (Box-Сох, ARMA, тройную экспоненту), которые приводят к переобучению при использовании на коротком и шумном временном ряде.

Высокая вариативность и отсутствие периодичности делают модель TBATS неэффективной для данной задачи.

Заключение

В работе отмечена актуальность разработки методов и программных средств, предназначенных для решения задачи изменения обводнённости продукции нефтяных скважин. Для решения указанной задачи были рассмотрены следующие методы интеллектуального анализа данных: LSTM, BiLSTM, Prophet, ARIMA, XGBoost, NNAR и TBATS. Выявлены сильные и слабые стороны данных методов. Выполнена реализация указанных методов. Выполнено тестирование реализованных методов с использованием реальных данных, полученных на нефтяных месторождениях. Показано, что при решении поставленной задачи для краткосрочного прогнозирования (от 1 до

5 минут) лучше использовать модели LSTM, BiLSTM и NNAR. Для долгосрочного прогнозирования изменения обводнённости продукции скважин лучше использовать LSTM модель. Также отмечено, что модели Prophet, XGBoost, ARIMA и TBATS не подходят для решения поставленной задачи из-за специфики изменения обводненности потока продукции скважин.

Таким образом, полученные результаты подтверждают целесообразность применения рекуррентных нейросетевых моделей для решения задач прогнозирования в нефтедобывающей отрасли. Дальнейшие исследования могут быть направлены на усовершенствование архитектур нейросетей с учётом специфики процессов обводнённости, а также на интеграцию гибридных подходов, способных повысить точность и надёжность прогнозов. Практическая значимость работы заключается в возможности использования разработанных моделей для оптимизации технологических процессов и повышения эффективности эксплуатации нефтяных скважин.

Список литературы

- [Барабанов и др., 2019] Барабанов А.О., Гужов С.В. Прогнозирование тепловой нагрузки на отопление с использованием ИНС // С.О.К. – 2019. – № 11. – С. 28-30.
- [Bian et al, 2022] Haihong Bian, Qian Wang, Guozheng Xu, Xiu Zhao. Load forecasting of hybrid deep learning model considering accumulated temperature effect // Energy Reports. – 2022. – Vol. 8. – P. 205-215. – DOI: 10.1016/j.egyr.2021.11.082.
- [Dong, 2022] Dong X., Deng S. & Wang D. A short-term power load forecasting method based on k-means and SVM // J Ambient Intell Human Comput. – 2022. – Vol. 13. – P. 5253-5267. – DOI: 10.1007/s12652-021-03444-x.
- [Sarker, 2021] Sarker I.H. Deep Learning: A Comprehensive Overview on Techniques, Taxonomy, Applications and Research Directions. // SN COMPUT. SCI. – 2021. – Vol. 2, 420. – <https://doi.org/10.1007/s42979-021-00815-1>.
- [Hochreiter et al., 1997] Hochreiter S., & Schmidhuber J. Long short-term memory // Neural Computation. – 1997. – Vol. 9, No. 8. – P. 1735-1780. – doi: 10.1162/neco.1997.9.8.1735.
- [Schuster, Paliwal, 1997] Schuster M., Paliwal K.K. Bidirectional recurrent neural networks // IEEE Transactions on Signal Processing. – 1997. – Vol. 45, No. 11. – P. 2673-2681.
- [Taylor et al., 2017] Taylor S.J., Letham B. Forecasting at scale // PeerJ Preprints. – 2017. – Vol. 5. – e3190v2. – DOI: 10.7287/peerj.preprints.3190v2.
- [Box et al., 1994] Box G.E.P., Jenkins G.M., Reinsel G.C. Time Series Analysis: Forecasting and Control, 3rd edition // Prentice Hall. – 1994.
- [Chen et al., 2016] Chen T., Guestrin C. XGBoost // Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. – 2016. – DOI: 10.1145/2939672.2939785.
- [Gregor et al., 2014] Gregor K., Danihelka I., Mnih A., Blundell C., Wierstra D. Deep AutoRegressive Networks // Proceedings of the 31st International Conference on Machine Learning (ICML), JMLR: W&CP. – 2014. – Vol. 32.
- [De Livera et al., 2011] De Livera A.M., Hyndman R.J., Snyder R.D. Forecasting time series with complex seasonal patterns using exponential smoothing // Journal of the American Statistical Association. – 2011. – Vol. 106, No. 496. – P. 1513-1527.