Содержание

ВВЕДЕНИЕ

Промышленные предприятия в России являются важным элементом экономики страны. Они производят широкий спектр продукции, включая металлы, нефтепродукты, химические вещества, машины и оборудование, продукты питания и многое другое. Промышленность является одной из основных отраслей экономики России и вносит значительный вклад в ВВП страны. При этом промышленные предприятия – один из крупнейших потребителей электрической энергии в Российской Федерации.

Промышленные предприятия (компании), потребляющие значительное количество электрической энергии, обязаны каждый день подавать заявки с информацией, сколько планируется закупить электроэнергии на ближайшие сутки. Отклонение фактического потребления электроэнергии – в меньшую или большую сторону – приводит к дополнительным расходам. Превышение заявленного потребления ведет к нехватке мощностей для работы на предприятии, что решается дополнительной закупкой по завышенному тарифу, а потребление ниже заявленного сводится к нецелесообразно израсходованным средствам.

С другой стороны, прогнозирование потребления электроэнергии требуется и генерирующим компаниям. Перепроизводство электроэнергии может привести к проблемам с ее хранением, а недостаточное производство – к поломкам в результате запуска электростанций в ударном режиме.

История вопроса разработки алгоритма прогнозирования потребления электроэнергии промышленным предприятием начинается с появления потребности в точном прогнозировании потребления электроэнергии регулярно и в реальном времени, особенно в условиях быстро развивающихся технологий, которые требуют все большего количества энергии для обеспечения эффективной работы систем. Первые алгоритмы, используемые для прогнозирования потребления электроэнергии, были простыми статистическими методами, которые основывались на анализе прошлых

данных о потреблении энергии. С развитием компьютерных технологий и возможностей обработки больших данных, начали создаваться более сложные алгоритмы прогнозирования. Тем не менее такие методы требуют значительного объема данных и времени для надстройки и обучения, что не всегда подходит для промышленных предприятий, где гибкость работы модели также важна, как и точность прогноза. Эти условия делают проблему прогнозирования потребления электроэнергии до сих пор актуальной.

Алгоритмы прогнозирования потребления электроэнергии индустриальными предприятиями были разработаны еще в 1960-х годах с развитием компьютерных технологий и расширением применения электроники в энергетике. В последующие годы алгоритмы прогнозирования стали совершенствоваться с учетом новых технологий и изменения в структуре потребителей электроэнергии. Анализ проблемы и пути решения данной задачи в последнее десятилетие были рассмотрены И. Д. Моргоевым, И. С. Даубом, П. В. Матрениным, М. М. Мигрrunовым, Д. Р. Григорьевой и другими.

Объект исследования: ООО «Харовсклеспром», лесопромышленное предприятие Вологодской области.

Предмет исследования: потребление электроэнергии промышленным предприятием, модели и методы краткосрочного прогнозирования потребления электроэнергии.

Цель исследования: повышение эффективности промышленного предприятия за счет снижения ошибки при прогнозировании посуточного потребления электроэнергии для ее закупки.

Для достижения поставленной цели требуется решить следующие задачи:

1) изучить особенности потребления электроэнергии промышленными предприятиями;

2) проанализировать существующие методы прогнозирования и отобрать наиболее подходящие для конкретного предприятия с учетом времени подбора параметров, обучения и прогнозирования;

3) применить и сравнить отобранные методы на данных по потреблению электроэнергии;

4) обобщить полученные результаты и на их основе выработать алгоритм прогнозирования;

5) сделать выводы по достигнутым результатам с целью установить целесообразность внедрения алгоритма.

Полный исходный код представлен в публичном репозитории по ссылке: https://github.com/MKJenna/Graduation-work

Работа состоит из введения, трех глав, заключения, списка использованной литературы, который представлен 40 источниками, в том числе 12 на иностранном языке, и 1 приложения.

Глава 1. Анализ современного опыта прогнозирования потребления электроэнергии

1.1. Электроэнергетика в России: прогнозирование потребления на промышленных предприятиях

Начало электроэнергетики в Российской Федерации было положено еще в XIX веке. В 1913 году на душу населения вырабатывалось примерно 14 кВт ч – почти в 16 раз меньше, чем в США. Однако несмотря на очевидную разницу в количественных характеристиках, по качеству дореволюционная Россия не уступала зарубежным странам, хотя и имелись некоторые проблемы [10].

Электростанции в то время имели ограниченное число потребителей и не были между собой связаны, а значения величин их тока и частот имели колоссальный разброс из-за отсутствия какой-либо единой системы при разработке станций. В январе 1918 года состоялась I Всероссийская конференция работников электропромышленности, на которой было предложено создать орган для руководства энергетическим строительством, в результате чего уже к концу 1920 года был подготовлен «План электрификации РСФСР». План ГОЭЛРО состоял из шести глав, таких как электрификация и план государственного хозяйства, электрификация и промышленность и другие. Уже в 1926 году план, направленный на восстановление энергетического хозяйства страны, был перевыполнен, а к 1935 году было выработано более 26 млрд кВт ч.

Теперь, почти сотню лет спустя, электроэнергетический комплекс России остается одним из самых развитых в мире. Согласно данным Международного энергетического агентства, в 2019 году Россия занимала 4-е место в мире по производству электроэнергии после Китая, США и Индии. Во время пандемии коронавируса потребление электроэнергии значительно выросло из-за того, что многие люди были вынуждены работать из дома, а из-за санкций, вызванных сложной геополитической обстановкой, возникли проблемы в осуществлении поставок основных средств на предприятия по

производству электроэнергии. Политика развития электроэнергетических систем по 2035 год включает в себя пересмотр ценовых зон. Некоторые предприятия по производству электрической энергии устанавливают системы генерации энергии, пытаясь снизить затраты на оплату покупаемой у государства электроэнергии. Ненужную предприятию энергию приходится либо перенаправлять на обычных пользователей, либо повышать стоимость услуг по производству и распределению электроэнергии, что является проблемой как для потребителей электроэнергии, так и для компаний по её производству [17].

Нормативная правовая база электроэнергетики состоит из большого количества актов юридической силы, в то числе:

- Федерального закона от 26.03.2003 № 35-ФЗ «Об электроэнергетике»;
- Федерального закона от 27.07.2010 N 190-ФЗ «О теплоснабжении»;
- Гражданского кодекса Российской Федерации (ГК РФ);
- Постановления Правительства РФ от 27.12.2010 N 1172 (ред. от 29.03.2023) «Об утверждении Правил оптового рынка электрической энергии и мощности и о внесении изменений в некоторые акты Правительства Российской Федерации по вопросам организации функционирования оптового рынка электрической энергии и мощности»;
- Постановления Правительства РФ от 04.05.2012 N 442 (ред. от 07.04.2023) «О функционировании розничных рынков электрической энергии, полном и (или) частичном ограничении режима потребления электрической энергии»;
- Постановления Правительства РФ от 29.12.2011 N 1178 (ред. от 06.02.2023) «О ценообразовании в области регулируемых цен (тарифов) в электроэнергетике»;
- Приказов Минэнерго и так далее.

Кроме этого, хотя договор о присоединении к торговой системе оптового рынка (ДОП) не является нормативным актом, его условия остаются обязательными для всех субъектов оптового рынка [21].

Генерирующие компании осуществляют выработку и реализацию электрической энергии и мощности на оптовом (или розничных) рынках конечным потребителям, либо же сбытовым компаниям, которые затем предоставляют электрическую энергию потребителям, выполняя роль посредника.

Одним из ключевых потребителей электрической энергии Российской Федерации являются промышленные предприятия.

Для получения электроэнергии промышленные предприятия должны подавать заявку генерирующим компаниям с указанием планового почасового потребления электроэнергии на следующие сутки. Таким образом компании, занимающиеся генерацией электрической энергии, пытаются избавиться от ситуации, когда сгенерировано слишком много или слишком мало электрической энергии. Однако это создаёт проблемы для промышленных предприятий: зачастую невозможно точно предсказать, сколько именно электроэнергии будет израсходовано в будущем. Электроэнергия – это не тот ресурс, который можно хранить, поэтому, если предприятие закажет слишком много электрической энергии, это приведёт к пустой трате ресурсов. Если же заказанной энергии не хватит, то предприятию придётся докупать электрическую энергию по более высокому тарифу [13].

Почему же так сложно предсказать потребление электрической энергии на промышленном предприятии?

Потребление электрической энергии промышленным предприятием зависит от множества факторов: от планируемого объёма производства, плановых остановах и пусках производства, плановых и внеплановых ремонтах, погодных условиях и так далее. Кроме того, в 2021 году компания Sedmax провела эксперимент по планированию потребления электроэнергии с помощью машинного обучения. Для этого было отобрано шесть крупнейших

энергоёмких предприятий из разных отраслей: нефтедобывающая и нефтеперерабатывающая компания; мебельный завод крупнейшего европейского ритейла; завод по производству стали и труб; добыча и обработка драгоценных камней; крупнейший металлургический комплекс России, горно-обогатительный комбинат.

Это исследование [13] позволило выявить, что точность прогноза потребления электроэнергии зависит, в том числе от отрасли и оборудования компании. Например, для добычи и обработки драгоценных камней используются специальные установки, потребление электроэнергии которых сильно зависит от погодных условий. Благодаря Циклическим и сезонным процессом Sedmax получила наилучший результат по работе с этой компанией. А если в производстве и обработки задействованы газотурбинные или газопоршневые агрегаты, сжигающие попутный газ, то получить достоверный прогноз не получится. Непостоянство объёмов параметров газа, а также резкие скачки, создают сильные выбросы на данных, которые не может учесть ни человек, ни компьютер.

ООО «Харовсклеспром» с 2002 года входит в состав Группы компаний «Вологодские лесопромышленники». Компания выпускает широкий ассортимент пиломатериалов из сосны и ели, а также технологическую щепу. Продукция реализуется в различные регионы России и в страны ближнего и дальнего зарубежья [26].

Промышленные предприятия по обработке древесины потребляют значительное количество электроэнергии в процессе своей деятельности. Одной из особенностей потребления электроэнергии этими предприятиями является необходимость использования большого количества энергоемких оборудований, таких как электродвигатели, насосы, компрессоры и т.д. Эти оборудования используются для обработки и переработки древесины.

Кроме того, потребление электроэнергии промышленными предприятиями по обработке древесины может меняться в зависимости от времени суток, дня недели и сезона. Например, в зимний период потребление

электроэнергии может быть выше из-за использования дополнительного оборудования для отопления и обогрева помещений.

Также следует учитывать, что процессы обработки древесины могут быть связаны с различными факторами, которые могут влиять на потребление электроэнергии. Например, влажность древесины, ее качество и состав, а также условия окружающей среды, такие как температура и влажность воздуха, могут оказывать влияние на процессы обработки и, соответственно, на потребление электроэнергии. Поэтому для прогнозирования потребления электроэнергии промышленными предприятиями по обработке древесины необходимо проанализировать влияние данных факторов (при возможности), и попробовать применить полученные выводы для повышения точности прогноза.

1.2 Современные методы и модели прогнозирования потребления электрической энергии промышленным предприятием

Существует множество методов и способов прогнозировать потребление электроэнергии. Самый простой способ – ручной: некоторый человек на промышленном предприятии, опираясь на историю потребления электроэнергии, графика ремонтных работ, планы о потреблении электроэнергии и свой опыт делает приблизительный прогноз на следующий период времени. Другой вариант прогнозирования временного ряда – использование методов сглаживания временного ряда. Более классическим способом прогнозировать временной ряд являются статистические модели: ARMA, ARIMA, GARCH и другие. Линейная регрессия и её разновидности не являются стандартными моделями прогнозирования временных рядов, однако они также могут быть применены для этой задачи. С ростом популярности искусственного интеллекта, для задачи прогнозирования потребления электроэнергии стали пробовать применять нейронные сети. Так, например, наиболее популярными в вопросе прогнозирования временных рядов стал многослойный перцептрон и рекуррентные нейронные сети. Кроме того, для

решения этой задачи часто применяют деревья решений и случайный лес, которые изначально были моделями классификации.

Эти методы могут давать различные результаты в зависимости от структуры исходных данных и области применения. Если рассмотреть научные работы, посвящённые прогнозированию электроэнергии или другим задачам энергетики, то можно обнаружить, что некоторые модели дают результат хуже, чем ручной подсчёт.

Мигранов М.М., Устинов А.А., Мельников А.В. в своей работе [15] провели сравнение нескольких моделей прогнозирования для двух вариантов прогноза: на сутки вперёд и на неделю вперёд. Все модели дали результат не ниже 4,5% по метрике MAPE. Наилучший результат для суточного прогноза дал Huber Regressor (0,47%), DecisionTreeRegressor (1,71%), ExtraTreeRedressor (2,03%); для недельного — Lasso Lars CV (1,37%), модель Хольта-Винтерса (1,62%) и Elastic Net CV (1,96%). Авторы статьи не привели результаты прогнозирования всех используемых моделей для обоих случаем дальности горизонта прогнозирования, что не дает в полной мере сделать вывод о том, какая модель лучше всего подходит для суточного или недельного прогноза. Кроме того, входными признаками для моделей машинного обучения являлся не только сам временной ряд, но и признаки, основанные на значениях даты и времени: час, день в году, День недели, неделя в году, месяц, будни /выходной, а также лаги — сдвиги временного ряда, которые позволяют учитывать данные о потреблении за прошлые периоды. Это позволяет предположить, что без использования дополнительных признаков модели машинного обучения на самом деле могут уступать по качеству эконометрическим моделям и скользящим средним.

В статье [19] проводилось сравнение между регрессиями, деревьями решений, бустингами и многослойным перцептроном. Как и в предыдущей статье, авторы ввели дополнительные признаки, основанные на дате и времени. При этом по корреляционной матрице видно, что большинство введённых признаков не имеют сильной линейной зависимости с целевой

переменной, а также наблюдается сильная линейная зависимость между самими признаками. Это может привести к ухудшению работы модели. Если признаки сильно коррелируют между собой, то модель может иметь проблему мультиколлинеарности, когда ей трудно выделить важные признаки и присвоить им веса, что может привести к нестабильности модели, низкой точности и трудностям в интерпретации результатов. Возможно, этим и объясняются довольно низкие результаты прогнозирования. Лучше всего с задачей справились CatBoostRegressor (7,95%), DecisionTreeRegressor (9,89%) и XGBRegressor (9,97%).

В статье Кирилычева [11] рассказывается о преимуществах использования нейронных сетей в задачах прогнозирования цен на электрическую энергию. Автор отмечает такие свойства нейронных сетей, как возможность обучаться и учитывать сложные зависимости между входными и выходными данными. Кирилычев указывает, что для использования данного метода необходимо обладать знаниями о том, какие признаки следует сохранять, как подготовить данные для обучения модели, как выбрать архитектуру и как интерпретировать результаты, а это значит, что для применение нейронных сетей требует меньше знаний в этой области, нежели при работе с классическими статистическими моделями.

В статье [6] автор не только приводит свои доводы о необходимости применения искусственного интеллекта в данной задаче, но и приводит результаты своего эксперимента, где точность прогноза превышает 90%. Также автор статьи отмечает, что использование искусственных нейронных сетей не только позволяет решить задачу повышения точности прогнозных расчётов, но и также проверить корректность передаваемых показаний приборов учета. Однако в своей статье автор не упоминает архитектуру используемой нейронной сети и не указывает по какой формуле проводился подсчет точности, что является существенным недостатком этой исследовательской работы.

Компания Sedmax провела полномасштабный эксперимент [13], в котором сравнивала качество прогноза нейронных сетей на разных типах промышленного производства. Наилучшие результаты были получены при работе с компанией, чья деятельность завязана на обработке драгоценных камней: ошибка составляла от 1 до 3% за счет использования метеорологических данных. Также высокие результаты были получены для металлургической компании. Остальные рассматриваемые компании (нефть, мебель, трубы, ГОК) дали не столь успешные результаты. Для обучения нейронных сетей авторы использовали не только данные о временном ряде потребления, но и данные о планируемых работах, метеоданные, планируемой загрузке производства и сделали вывод, что чем качественнее процесс планирования на предприятии, тем точнее можно получить прогноз.

Также среди работ, посвященных прогнозированию именно с помощью нейронных сетей следует отметить [8], в которой авторам не удалось добиться низкой ошибки прогнозных значений.

Довольно интересный эксперимент по сравнению различных моделей прогнозирования временных рядов был приведен в статье [29]. Нейронные сети дали наихудший результат прогнозирования (7745,15 по метрике MAE), а наилучший дал случайный лес (2750,21). Довольно хорошие результаты дали и статистические модели: seasonal ARIMA (2923,35), ARIMA (2916,03) и градиентный бустинг на основе случайного леса (2907,69).

В статье [20] авторы проводят сравнение между линейной регрессией, деревьями решений, многослойным перцептроном и бустингами. Результаты прогнозирования довольно неудовлетворительные (коэффициент детерминации почти везде отрицательный). Авторы предполагают, что подобный неудовлетворительный результат получен в связи с загруженностью предприятий и технологическим процессом производства. Тем не менее многослойный перцептрон дал положительный коэффициент детерминации для промышленного предприятия с ярко выраженной сезонностью.

Анализ литературы, посвящённый прогнозированию потребления электрической энергии промышленным предприятием, а также прогнозированию временных рядов, показал, что наилучшие результаты дают бустинги деревьев и рекуррентные нейронные сети. Статистические модели, которые считаются классическими моделями прогнозирования временных рядов, практически не применялись для решения проблемы потребления электрической энергии. Нейронные сети, а именно рекуррентные нейронные сети, позволяют получить качественный прогноз, хотя для промышленного предприятия этот прогноз всего на пару процентов точнее ручного подсчёта. Практически во всех научных работах для обучения моделей помимо самого временного ряда потребления электрической энергии вводились дополнительные признаки на основе даты и времени, а также данные о погоде. При этом для статистических моделей использовались только данные о самом временном ряде без использования дополнительных признаков, что, на мой взгляд, не дает возможности полноценно сравнить методы. Ещё одним важным недостатком этих научных работ является отсутствие замера времени на обучение, подбор параметров и построение прогноза, имеющих важное значение для предприятий, для которых важную роль в прогнозе имеет также отведенное на него время.

Хальясмаа А.И., Матренин П.В. И Ерошенко С.А. посвятили анализу ошибок применению алгоритмов машинного обучения задачах электроэнергетики целую статью [25]. В ней уделяется внимание выбору источников данных, важности предобработки данных, принципам формирования выборок и ошибкам моделирования и тестирования моделей. Авторы статьи обращают внимание на необходимость разделения задач по времени, которое требуется на её решение и времени и требуется непосредственно на обучение данной модели. Задачи электроэнергетики делятся на три типа:

- оперативные, которые требуют большого объёма заранее определённых данных и малого времени обучения модели (например, для решения проблем в онлайн режиме);
- среднесрочные задачи, которые требуют большого объёма данных и разумного времени выполнения;
- долгосрочные задачи, в которых огромную роль играет точность прогноза, а время обучения модели является второстепенной проблемой.

В задаче прогнозирование потребления электроэнергии промышленным предприятием в первую очередь важно получить модель, которая делает прогноз за разумное время; время обучения и подбор параметров играет меньшую, но также важную, роль.

С учётом всего вышесказанного было решено провести сравнительный анализ трёх видов моделей: статистические, многослойного перцептрона и деревьев.

Первые две были выбраны из-за малой освещённости применения этих моделей в данной конкретной задаче энергетики, что приводит к вопросу, не является ли усложнение моделей прогнозирования неоправданным фактором. Многие эксперты по машинному обучению считают, что простая модель часто лучше сложной, особенно если у нее достаточно данных, чтобы справляться с поставленной задачей. Поэтому следует проверить, действительно ли нельзя получить высокое качество прогноза без использования рекуррентных нейронных сетей, требующих значительного времени на настройку и обучение.

Деревья решений и случайный лес давали хорошие результаты, пусть и уступали по качеству бустингам (CatBoost, Light GBM и XGBoost). При этом в исследованиях не проводится замер времени, хотя можно предположить, что случайный лес должен обучаться быстрее бустингов, поскольку во втором случае обучение отдельных моделей происходит постепенно, а в первом – одновременно (независимо друг от друга). Это позволяет предположить, что

случайный лес может оказаться более релевантным инструментом для решения поставленной задачи.

В рамках выпускной квалифицированной работы отобранные модели будут сравниваться по времени работы, качеству посуточного и недельного прогноза. Кроме этого, во второй главе при обучении моделей будет использоваться только один признак — сам временной ряд. Это сделано с целью выяснить насколько хорошо данные модели справляются с задачей прогнозирования потребления энергии промышленным предприятием в условиях ограниченной информации.

1.2.1. Статистические модели

Статистические модели ARMA, ARIMA и Seasonal ARIMA находят широкое применение в анализе временных рядов. Они позволяют предсказывать будущие значения временного ряда на основе его прошлых значений.

Модель ARMA (autoregressive moving average) — это комбинация авторегрессионной и скользящей средней моделей. Модель ARMA(p,q) имеет два параметра: p обозначает порядок авторегрессии, а q — порядок скользящей средней. Авторегрессионная модель использует прошлые значения временного ряда для прогнозирования его будущих значений. Скользящая средняя модель учитывает случайные ошибки в прошлых значениях временного ряда. Авторегрессия AR(p) описывает зависимость текущего значения ряда от его предыдущих значений. Иными словами, авторегрессионная компонента модели ARIMA использует p предыдущих значений ряда для прогнозирования следующего значения. Скользящее среднее MA(q) описывает зависимость текущего значения ряда от ошибок предыдущих прогнозов. Компонента скользящего среднего MA(q) модели ARIMA использует q предыдущих ошибок прогнозов для прогнозирования следующего значения.

Модель ARIMA (autoregressive integrated moving average) — это расширение модели ARMA для учета тенденций и сезонности в данных.

Модель ARIMA(p,d,q) имеет дополнительный параметр d, который обозначает порядок интегрирования – то есть указывает, сколько раз необходимо продифференцировать исходный ряд, чтобы получить стационарный. Если d=0, то имеем просто модель ARMA(p,q).

Оператор лага L – это оператор, который каждому члену временного ряда ставит в соответствие предыдущий: $LY(t) = Y(t-1)$

Тогда модель ARIMA в лаговой форме будет иметь вид:

$$\left(1 - \sum_{k=1}^{p} \rho_k L^k\right) \Delta^d Y(t) = \mu + \left(1 - \sum_{k=1}^{q} \beta_k L^k\right) \varepsilon(t)$$

Модель ARIMA широко используется в экономике, финансах, метеорологии и других областях, где важно прогнозировать будущие значения временных рядов, однако она требует дополнительных проверок данных для получения достоверного прогноза.

Стационарность — это свойство временного ряда, когда его статистические характеристики не меняются со временем. Другими словами, стационарный временной ряд имеет постоянное среднее значение, постоянную дисперсию и постоянную автокорреляцию независимо от времени. Модель ARIMA работает только со стационарными временными рядами. Если ряд не стационарен, то его необходимо привести к стационарному виду путем различных преобразований (дифференцирование, то есть взятие разностей).

Существуют различные методы для проверки стационарности временного ряда, такие как визуальный анализ графиков, тест Дики-Фуллера и KPSS тест, которые могут быть использованы для определения нестационарных свойств временного ряда.

Один из ключевых требований к остаткам модели ARIMA состоит в том, что они должны быть белым шумом. Белый шум — это случайный процесс, который имеет нулевое среднее значение и постоянную дисперсию.

Желательно, чтобы, значения белого шума должны быть независимыми и одинаково распределенными. Если остатки модели ARIMA не являются белым шумом, то это означает, что модель не может полностью объяснить изменчивость ряда и остается некоторая систематическая ошибка в прогнозировании значений. Это может привести к неверным выводам и ошибкам в прогнозировании будущих значений ряда.

Проверка на белый шум можно выполнить с помощью графиков автокорреляции и частной автокорреляции остатков, а также статистических тестов, таких как тест Льюнга-Бокса.

Таким образом, алгоритм прогнозирования с помощью модели ARIMA выглядит следующим образом:

1) Временной ряд проверяется на стационарность с помощью теста Дикки-Фуллера на наличие единичного корня. Если ряд нестационарный, то ряд дифференцируется d раз, пока он не окажется стационарным;

2) По ACF и PACF полученного продифференцированного ряда оценивают порядок модели;

3) Оцениваются коэффициенты модели и проверяется их значимость;

4) Вычисляются остатки модели и проверяются на обладание свойств белого шума. Если остатки по результатам тестирования не ведут себя как белый шум, то требуется подобрать другие p и q;

5) От разностей возвращаются к исходному ряду;

6) С помощью полученных коэффициентов вычисляется прогноз.

ARIMA-модель используется для моделирования временных рядов, которые не имеют ярко выраженной сезонности. Однако в некоторых случаях сезонность может оказывать значительное влияние на поведение временного ряда. В этом случае используется модель SARIMA, которая включает в себя дополнительные параметры, отвечающие за сезонность.

Модель SARIMA имеет следующие параметры:

- p: порядок авторегрессии (AR)

- d: порядок интеграции (I)

- q: порядок скользящего среднего (MA)

- P: порядок сезонной авторегрессии (SAR)

- D: порядок сезонной интеграции (SI)

- Q: порядок сезонного скользящего среднего (SMA)

- s: период сезонности

s (иногда пишут m) – это количество наблюдений за цикл. Так, если данные ежегодные, то s = 1, если ежеквартальные, то s = 4 (так как 4 наблюдения в год), ежемесячные – s = 12 и так далее. Если же данные собираются ежедневно, то частоту можно интерпретировать по-разному. Например, ежедневные данные могут иметь недельную сезонность. В этом случае частота равна s = 7, поскольку за полный цикл будет семь наблюдений.

SARIMA-модель может быть полезна при прогнозировании экономических показателей, продаж, трафика на сайте и т.д.

ARIMA или SARIMA используют только для краткосрочных прогнозов, поскольку долгосрочные, строго говоря, по адекватности не сильно отличаются от прогнозирования с помощью среднего значения.

Перечислим основные преимущества данного метода:

1. Меньшая потребность в данных: статистическим моделям нужно не так много исторических данных для прогнозирования. Так, например, если мы располагаем почасовыми данными, то для успешного применения статистической модели достаточно пары месяцев наблюдения.

2. Интерпретируемость: в большинстве случаев статистические модели можно интерпретировать. Таким образом, мы не только получаем прогноз, но и по рассчитанным коэффициентам модели можем определить влияние той или иной переменной.

3. Учет сезонности и цикличности.

Тем не менее, статистические модели имеют и недостатки:

1. Ограниченность: статистические модели могут быть ограничены в своей способности учитывать сложные зависимости в данных. Например, они

могут не учитывать нелинейные зависимости или взаимодействия между различными переменными.

2. Необходимость дополнительной обработки данных и проверки модели: перед использованием модели данные следует проверить на стационарность, при необходимости – выявить тренд, сезонность, цикличность. Полученные после применение модели остатки требуется также проверить на нормальность, и в случае невыполнения этого требования модель нельзя использовать для прогноза.

3. Чувствительность к выбросам: наличие выбросов приводит к искажению результатов, в связи с чем подготовительный этап обязательно должен включать поиск аномалий и их замену.

4. Статистические модели могут делать прогноз только на одно-два наблюдение вперед. Таким образом, чтобы получить прогноз на сутки или неделю вперед, требуется заново обучать модель на новых данных, которые формируются исходные данные с добавлением предыдущего прогноза. Это может привести к накоплению ошибки.

1.2.2. Многослойный перцептрон

Многослойный перцептрон (англ. Multilayer Perceptron, MLP) — это один из наиболее распространенных типов нейронных сетей, которые используются для решения широкого спектра задач, от распознавания речи до анализа финансовых рынков.

История создания MLP началась в 1943 году, когда Уоррен Маккаллок и Уолтер Питтс представили первую модель искусственного нейрона. Однако, первые многослойные перцептроны были созданы только в конце 1960-х годов, когда Бернард Видроу и Роберт Хофф опубликовали свою работу "Глубокие нейронные сети и распознавание образов", в которой они описали алгоритм обучения многослойного перцептрона.

MLP состоит из нескольких слоев нейронов, каждый из которых обрабатывает информацию и передает ее на следующий слой. Входной слой получает данные и передает их на скрытые слои, где они обрабатываются с

помощью весов и функций активации. Затем выходной слой получает результаты и выдает ответ.

Функция активации определяет, какой будет выход нейрона в зависимости от его входных данных. Наиболее распространенными функциями активации являются сигмоидальная функция:

$$x_{new} = \frac{1}{1 + e^{-x}}$$

и функция ReLU (Rectified Linear Unit):

$$x_{new} = max(0, x)$$

Одной из главных проблем многослойных перцептронов является переобучение. Это происходит, когда модель слишком хорошо запоминает обучающие данные и не может обобщать на новые данные. Для борьбы с этой проблемой используются различные методы, такие как регуляризация и дропаут.

MLP широко применяется в различных областях, включая обработку изображений, распознавание речи, анализ финансовых рынков, прогнозирование временных рядов, обработку естественного языка и многое другое. Например, в обработке изображений MLP может использоваться для распознавания лиц, определения объектов на изображении и сегментации изображения.

В отличие от статистических моделей, для работы с многослойным персептроном имеющейся данные следует поделить на признаки (входные значения, которые используются для обучения модели) и целевые значения (выходные значения, которые модель должна предсказать).

Один из наиболее распространенных подходов — это использование метода скользящего окна (sliding window). Суть метода заключается в том, чтобы выбрать определенное количество последовательных точек данных в качестве признаков, а следующие точки данных – в качестве целей (то, что

нужно предсказать). Затем окно сдвигается на одну точку вправо, и процесс повторяется до конца временного ряда.

Допустим, мы имеем временной ряд: x1, x2, x3, …, x71, x72, где xi — это посуточное потребление электрической энергии, и мы хотим, чтобы модель сделала предсказание на основе данных, полученных за последние 24 часа. Тогда при первом проходе получаем: массив признаков = [x1, x2, x3, …, x23, x24], цель = x25. Смещаемся на одну строчку вправо: набор признаков = [x2, x3, x4, …, x24, x25], цель = x26. И так далее, до конца временного ряда.

Нейронные сети также имеют свои преимущества и недостатки:

1. Нейронные сети, в отличие от статистических моделей, способны обрабатывать сложные и нелинейные зависимости между переменными.

2. Временной ряд необязательно должен быть стационарным. Требование, что остатки должны быть белым шумом, также опускается.

3. Способность автоматически извлекать признаки из данных, что уменьшает необходимость вручную выбирать признаки и улучшает точность прогнозирования.

4. Многослойный перцептрон может прогнозировать любое количество данных. Таким образом можно делать прогноз не по одному наблюдению за раз, а сразу на день или неделю вперед без накопления ошибки.

Несмотря на свои преимущества, нейронные сети также имеют некоторые недостатки в прогнозировании временных рядов:

1. Если для статистических моделей было достаточно всего пары десятков наблюдений, то для обучения нейронных сетей речь идет уже минимум о паре сотен данных.

2. Нейронные сети могут быть склонны к переобучению, особенно если данные содержат шум или выбросы. Эта проблема частично решается введением регуляризации.

3. Нейронные сети часто называют «черными ящиками» из-за сложностей интерпретации полученных результатов.

4. Обучение нейронной сети требует большое количество ресурсов, таких как вычислительная мощность и время.

1.2.3. Дерево решений и случайный лес

Дерево решений и случайный лес – это два известных метода машинного обучения, которые широко используются для решения задач классификации и регрессии. Оба метода позволяют строить модели, которые могут прогнозировать значения целевой переменной на основе имеющихся данных.

В 1950-х годах были представлены основные идеи моделирования человеческого поведения с помощью компьютерных систем, что послужило началом развития использования деревьев. Джон Р. Куинлен и Лео Брейман связаны с дальнейшим развитием данного метода как самообучающихся моделей для анализа данных, включая метод случайного леса.

Дерево решений – это графическая модель, которая представляет собой древовидную структуру. В каждом узле дерева происходит разбиение данных на две или более частей, в зависимости от значений одного из признаков. Этот процесс продолжается до тех пор, пока не будет достигнут критерий остановки, например, определенная глубина дерева или недостаточное количество объектов в узле. В итоге каждый лист дерева содержит предсказание целевой переменной.

Деревья решений могут быть использованы для прогнозирования временных рядов. Для этого временной ряд, как и в случае с многослойным перцептроном, должен быть разбит на признак и целевую переменную.

Для построения дерева решений можно выделить четыре этапа:

• Выбор атрибута для разбиения в узле;

• Определение критерия остановки обучения;

• Выбор метода отсечения ветвей;

• Оценка точности построенного дерева.

К преимуществам использования деревьев решений относятся:

1. они легко интерпретируются, что позволяет понимать, какие признаки наиболее важны для предсказания целевой переменной;

2. они могут работать с различными типами данных, включая категориальные и числовые;

3. могут обрабатывать отсутствующие значения;

4. не требует много параметров для подбора.

Тем не менее, дерево решений может быть не самым подходящим методом для прогнозирования потребления электроэнергии промышленным предприятием по нескольким причинам, поскольку:

1. неэффективно для обработки слишком больших объемов данных;

2. иногда не способно учитывать сложные взаимосвязи между различными факторами;

3. может быть неустойчивым к изменениям в данных, что может привести к неадекватным прогнозам;

4. Дерево решений может быть склонным к переобучению, что может привести к низкой точности прогнозирования.

Чтобы улучшить точность прогнозирования, можно использовать случайный лес. Случайный лес – это ансамбль деревьев решений, где каждое дерево строится на случайном подмножестве признаков и объектов. Каждое дерево в лесу даёт свой прогноз, а окончательный прогноз получается путем усреднения прогнозов всех деревьев. Этот способ построения ансамбля моделей называется беггинг.

Случайный лес – это ансамблевая модель машинного обучения, которая использует несколько деревьев решений для прогнозирования значения целевой переменной. Каждое дерево строится на основе случайной выборки из обучающих данных и случайного набора признаков. Затем прогнозы каждого дерева объединяются (усредняются) для получения окончательного прогноза. Случайный лес может использоваться для классификации или регрессии и обычно имеет лучшую точность, чем отдельные деревья решений. Он также может быть эффективным в работе с большими объемами данных и при наличии шумовых или несбалансированных данных.

Каждое дерево в модели «случайный лес» строится независимо от других. Сначала выбирается подвыборка обучающей выборки определенного размера, на основе которой строится дерево. Затем для каждого нового расщепления в дереве исследуется определенное количество случайных признаков, и выбирается наилучший признак и его расщепление с помощью заранее заданного критерия. Обычно дерево продолжается до тех пор, пока не будет исчерпана выборка, но существуют параметры, которые могут ограничивать высоту дерева, число объектов в листьях и число объектов в подвыборке, при которых проводится расщепление.

Чем больше деревьев в модели, тем выше качество на данных, но при этом растет скорость подбора параметров и обучения, а также риск переобучения.

Преимущества случайного леса:

1. Точность прогнозирования выше, чем у дерева решений.
2. Могут обрабатывать отсутствующие значения.
3. Могут работать с большими наборами данных.
4. Устойчивы к переобучению, чем деревья.

Недостатки случайного леса:

1. При обучении на больших наборах данных требуется много времени.
2. Могут быть сложными для интерпретации.

Деревья решений и случайный лес широко используются в различных областях, включая финансы, маркетинг, медицину и т.д. Они могут быть использованы для прогнозирования цен на акции, спроса на товары, заболеваемости и т.д. В целом, они представляют собой мощные инструменты для анализа данных и прогнозирования будущих значений целевой переменной.

Глава 2 Прогнозирование потребления электроэнергии промышленным предприятием

2.1 Анализ исходных данных

2.1.1. Предварительный анализ данных

В качестве исходных данных для прогнозирования были использованы объемы потребления электроэнергии в кВт. по часовым интервалам за январь 2017 – декабрь 2020 гг. Всего в датасете 35 063 наблюдений.

Исходная таблица состоит из трех столбцов, содержащих дату (Date), временной интервал (Time) и объем потребления электроэнергии (Usage_kWh).

	Date	Time	Usage_kWh
0	2017-01-01	00 - 01	570.685479
1	2017-01-01	01 - 02	604.642705
2	2017-01-01	02 - 03	518.732113
3	2017-01-01	03 - 04	608.188829
4	2017-01-01	04 - 05	714.140572

Таблица 1. Первые пять строк датасета

Пропусков в данных нет.

	Тип	Количество Nan	Количество уникальных	Уникальные значения
Date	datetime64[ns]	0	1461	[2017-01-01T00:00:00.000000000, 2017-01-02T00:...
Time	object	0	24	[00 - 01, 01 - 02, 02 - 03, 03 - 04, 04 - 05, ...
Usage_kWh	float64	0	35063	[570.6854791729655, 604.6427047280158, 518.732...

Таблица 2. Предварительная информация о датасете

При работе с временными рядами важную роль играет визуализация, поскольку различные виды графиков могут помочь выявить выбросы, обнаружить тренд, сезонность, а также получить первичное представление о зависимостях и закономерностях.

«Ящик с усами» является довольно популярным инструментом для поиска аномалий и представления о распределении данных. Минимальное значение потребление электроэнергии за все время – 238.525049 кВт ч, максимальное – 2302.898062 кВт ч. Большинство значений (т.е. 50% от всех

наблюдений) находятся в диапазоне от 609.685837 до 1181.648661, что указывает на относительно высокую концентрацию данных в этом интервале. Кроме того, стандартное отклонение (355.419642) показывает, что данные имеют довольно большой разброс относительно среднего значения. Распределение скорее всего является асимметричным, так как среднее значение (917.634664) находится справа от медианы (865.619898), а также потому что максимальное значение находится далеко от среднего и медианы. На ящике с усами можно заметить выбросы. Проведем дополнительный тест для выявления аномалий в данных.

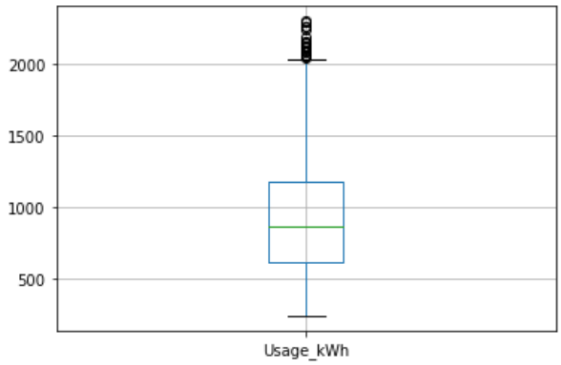

Рисунок 1. «Ящик с усами»

	Usage_kWh
count	35064.000000
mean	917.634664
std	355.419642
min	238.525049
25%	609.685837
50%	865.619898
75%	1181.648661
max	2302.898062

Таблица 3. Описательная статистика датасета

Одним из популярных тестов для временных рядов является тест Ирвина [23]. Для этого, чтобы определить, является ли наблюдение аномалией или нет, рассчитаем для каждого значения (начиная со второго) соответствующую ему $\lambda_t = \frac{|y_t - y_{t-1}|}{\sigma_y}$. Будем считать аномальными те значения, λ_t которых больше 1. Всего таких значений 5545, при этом только 315 из них также считались аномальными по «ящику с усами». Это связано с тем, что тест Ирвина

рассчитывается по отношению к предыдущему значению, и если y_t действительно принимает экстремально высокое (или низкое) значение, то y_{t+1} также будет считаться аномалией. Поэтому будем считать аномальными наблюдениями только те, которые были отмечены и «ящиком с усами», и тестом Ирвина.

Поскольку мы имеем дело с временным рядом, то замена аномальных значений на медиану, и тем более удаление такого наблюдения, не будет правильным решением. Вместо этого заменим все аномальные наблюдения на среднее арифметическое его соседей.

Гистограмма подтверждает выводы об асимметрии. Распределение данных в целом по форме похоже на нормальное.

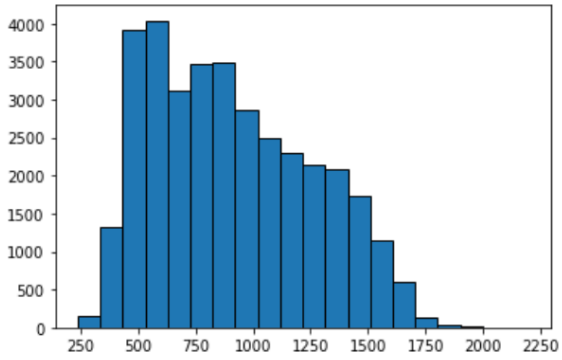

Рисунок 2. Гистограмма распределения

Посмотрим на суммарное потребление электроэнергии по годам и месяцам. Можно заметить, что в апреле потребление электроэнергии ниже, чем в начале года и идет на спад, а в августе показатель снова растет. Четко выраженного снижения или повышения потребления электроэнергии с каждым годом не замечено.

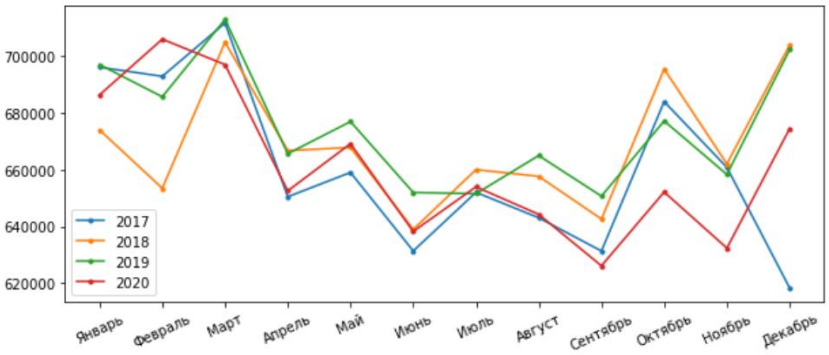

Рисунок 3. Динамика потребления по годам

Рассмотрим помесячное потребление на примере 2018 года. На оси oX поставим отметку о каждом 24-м наблюдении, чтобы видеть начало следующего дня. Вертикальные линии отсекают начало следующей недели, а горизонтальная пунктирная линия отмечает среднее значение потребление электроэнергии за указанный месяц. Таким образом мы сможем увидеть, есть ли сезонность и тренд.

На данных графиках легко заметить циклическое потребление в течение дня: сначала потребление растет до некоторой отметки, незначительно падает, поднимается, и только после этого идет на спад. Также можно отметить неравномерное потребление электроэнергии каждую неделю.

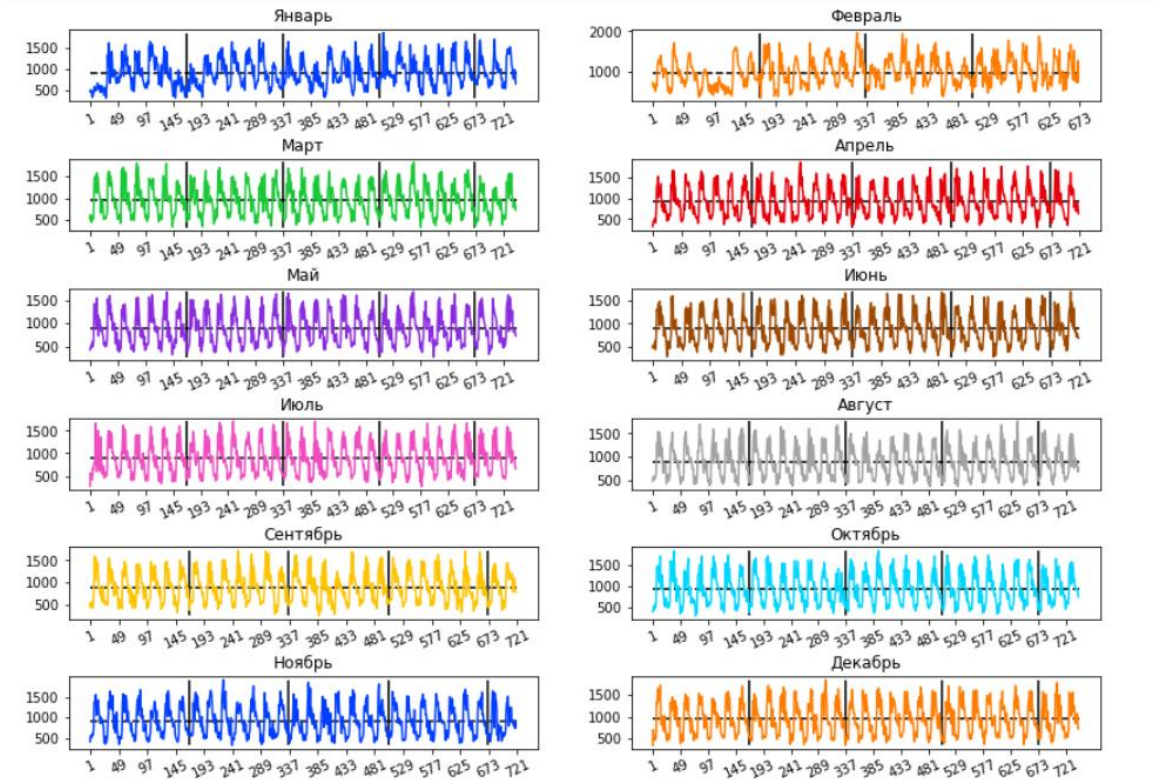

Рисунок 4. Потребление по месяцам за 2018 год

Рассмотрим внимательнее среднесуточное потребление за каждый месяц. Очевидна дневная динамика потребления: в 6 часов утра наблюдается резкий скачок; рост прекращается в 10 часов утра, в 11 потребление падает до некоторого уровня; снова рост до 13 часов, после чего потребление плавно снижается.

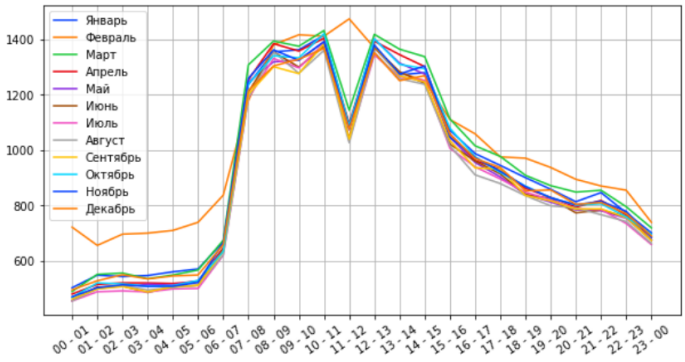

Рисунок 5. Дневная динамика потребления

После визуального анализа данных можно сделать несколько выводов. Во-первых, такая четкая дневная динамика потребления должна позволить нам делать более точные прогнозы, так как мы знаем, в каких часовых промежутках ожидать пики и впадины. Во-вторых, из-за явной сезонности простая ARIMA будет давать прогнозы, несоответствующие реальным значениям, зато сезонная ARIMA скорее всего покажет высокое качество. В-третьих, оптимальной шириной окна выглядит значение 24*7 — так многослойный перцептрон и деревья смогут лучше уловить дневную динамику.

Для выявления зависимости погодных условий на потребление электроэнергии были взяты данные о средней температуре, средней влажности и скорости ветра, а также количество ясных, облачных, пасмурных, дождливых и снежных дней за соответствующие месяца [9].

Для корреляционного анализа также добавим следующие столбцы, основанные на дате и времени наблюдения:

- weekend: 1 если этот день был выходным (включая праздники согласно производственному календарю), иначе 0;

- peak: 1 если наблюдение записано в период с 6 до 13 часов, иначе 0;

- peak_month: 1 если месяц с января по март или с сентября по декабрь, иначе 0;

- year2017, year2018, year2019: 1 если год 2017 (2018 или 2019 соответственно), иначе 0. Переменная, соответствующая 2020 году, не включена для избежания мультиколлинеарности;

- days_in_month: количество дней в месяце [19];

- month_mean: среднее потребление за все года за этот месяц [40];

- year_mean: среднее за этот год.

Для визуализации зависимости признаков построим тепловую карту – матрицу корреляции, рассчитанной для каждого признака по следующей формуле:

$$r_{xy} = \frac{\overline{xy} - \bar{x} \cdot \bar{y}}{\sigma_x \cdot \sigma_y},$$

Где x и y – попарно перебираемые признаки.

Сильной линейной зависимости между признаками и целевой переменой не обнаружено, за исключением булевого признака peak. Во второй главе при применении моделей для прогнозирования данные признаки использоваться не будут.

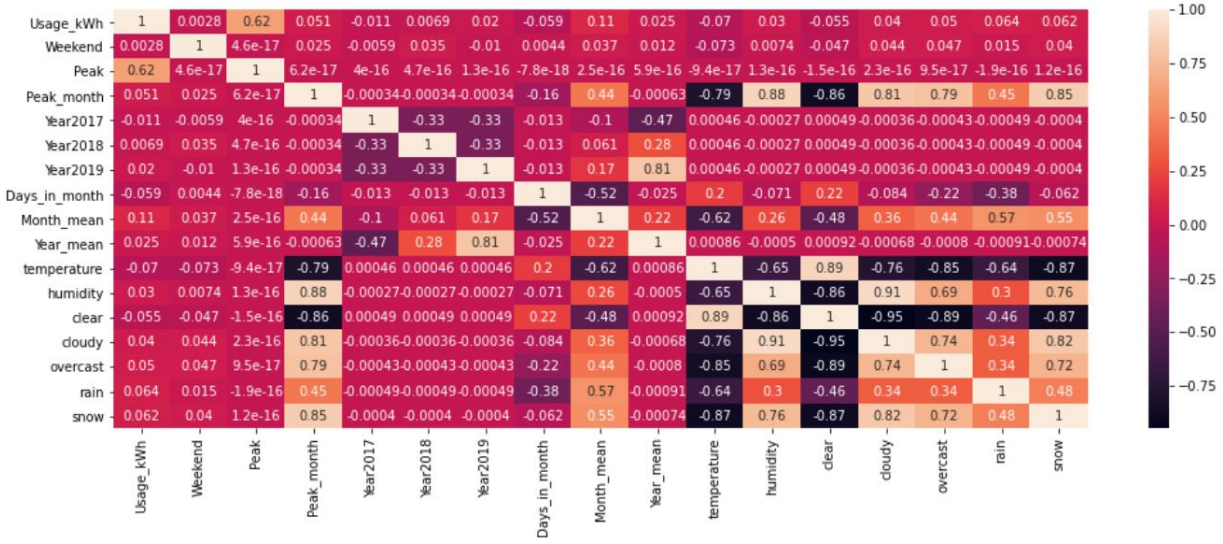

Рисунок 6. Коррелограмма

2.1.2. Базовые модели

Прежде чем переходить к сложным моделям прогнозирования, попробуем сделать прогноз «по интуиции». Наивный (или базовый) прогноз — это простейший метод прогнозирования временных рядов, при котором

прогнозируемое значение рассчитывается на основе предыдущих значений ряда. Этот метод часто используется как базовая модель для сравнения с более сложными моделями прогнозирования.

Зачастую наивный прогноз может оказаться достаточно эффективным и точным в прогнозировании простых рядов, что позволяет существенно упростить задачу прогнозирования. Если же наивный прогноз показывает недостаточную точность, то можно применить более сложные модели прогнозирования. Такой подход позволяет снизить затраты на разработку более сложных моделей и временные затраты на их обучение. Кроме того, наивный прогноз позволяет получить базовый уровень точности, на который можно ориентироваться при сравнении с более сложными моделями.

В качестве основной метрики возьмем классическую метрику регрессии – коэффициент детерминации:

$$R^2(y, \hat{y}) = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2},$$

где y_i — это фактическое значение, $\hat{y}_i$ – прогноз, $\bar{y}$ – среднее значение по всей выборке, n – количество наблюдений.

Чем ближе метрика к 1, тем большая доля дисперсии зависимой переменной объясняется моделью, т.е. тем лучше модель описывает данные. Если модель дает совсем неадекватный прогноз, то метрика примет отрицательные значения.

В качестве метрик ошибок возьмем среднюю абсолютную ошибку в процентах (MAPE), среднеквадратичную ошибку (MSE) и среднюю абсолютную ошибку (MAE):

$$\text{MAPE}(y, \hat{y}) = \frac{1}{n_{\text{samples}}} \sum_{i=0}^{n_{\text{samples}}-1} \frac{|y_i - \hat{y}_i|}{max(\epsilon, |y_i|)} \cdot 100\%,$$

$$\text{MSE}(y, \hat{y}) = \frac{1}{n_{\text{samples}}} \sum_{i=0}^{n_{\text{samples}}-1} (y_i - \hat{y}_i)^2.$$

$$\mathrm{MAE}(y, \hat{y}) = \frac{1}{n_{\text{samples}}} \sum_{i=0}^{n_{\text{samples}}-1} |y_i - \hat{y}_i|.$$

ϵ – очень маленькое положительное число, позволяющие избежать деления на 0.

При наличии в данных выбросов MSE будет значительно выше, чем должно быть. MAE позволяет избежать этой ситуации и хорошо подходит для случаев, где важно минимизировать ошибку на каждом примере. MAPE делает объекты в выборке равнозначными и штрафует модель, если был неправильно угадан порядок прогноза.

Кроме этого, будем замерять время на подбор параметров (если использовался), обучение и прогноз моделей.

Качество моделей будем проверять на последних 168 наблюдений – то есть на данных с 25 декабря 2020 года по 31 декабря 2020 года. При этом по возможности будем проверять различные способы прогнозировать значения за интересующий промежуток времени:

МП1. Прогнозировать сразу всю неделю, чтобы выявить, подходит ли модель для долгосрочного прогнозирования;

МП2. Имитировать постепенное прогнозирование, то есть сначала для прогноза на 25 декабря ориентироваться только на данные до 24 декабря включительно, для прогноза на 26 декабря – на данные до 25 декабря включительно и так далее.

К базовым моделям относятся:

1. Историческое среднее – то есть среднее арифметическое за все имеющиеся данные;

2. Среднее за последний период времени – среднее арифметическое за год, месяц, неделю или день. Мы рассматриваем почасовые ежедневные данные, поэтому в нашем случае период равен одному дню или 24 наблюдениям;

3. Последнее значение – в нашем случае последнее значение потребления со вчерашнего дня;

4. Наивный сезонный прогноз – мы берем последний наблюдаемый цикл и повторяет его в будущем. В данном случае цикл выполняется за сутки, поэтому прогнозом тестовой выборки будут являться последние 24 наблюдения обучающей выборки.

Результаты прогнозов базовых моделей представлены ниже. По графику с тестовыми данными и предсказаниями видно, что наилучшим образом показала себя последняя модель – наивный сезонный прогноз. Этот результат можно объяснить тем, что в данных ярко выражена дневная динамика, благодаря чему повтор «вчерашнего» потребления хорошо описывает «сегодняшнее» потребление.

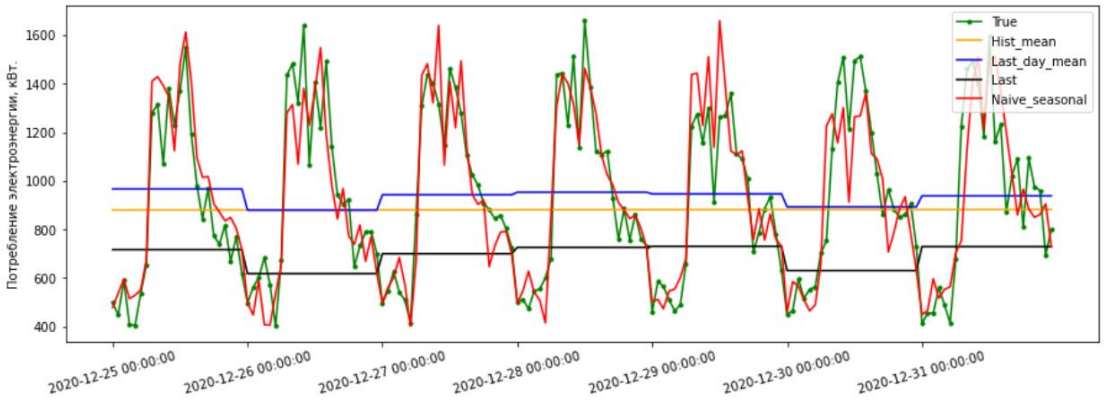

Рисунок 7. Прогноз базовых моделей

Ошибки у наивного прогноза ожидаемо низкие, MAPE всего 12,69%. Коэффициент детерминации довольно высокий, 0,81. Возьмем эти показатели за основные, и рассмотрении других моделей будем отбирать те, кто даст результаты лучше или близко к этим.

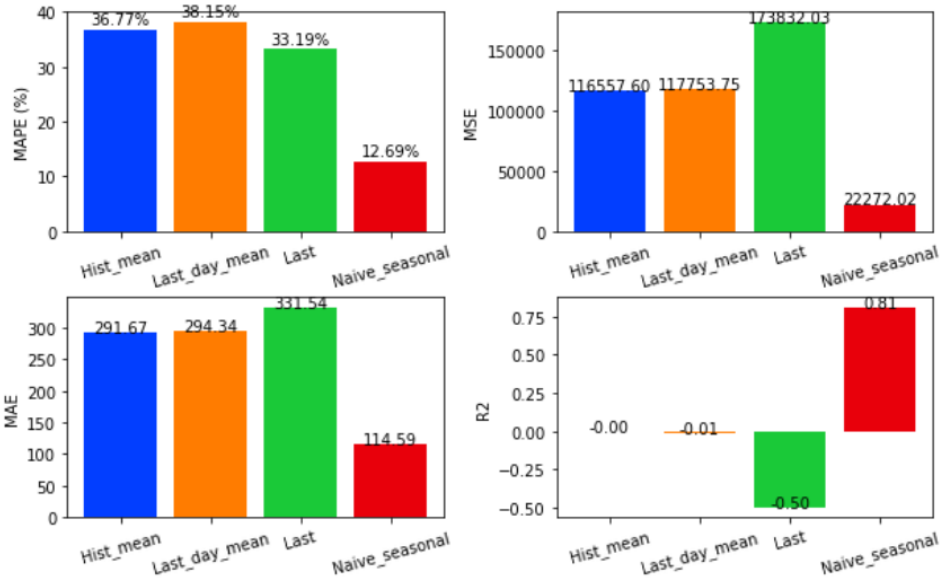

Рисунок 8. Значения метрик для базовых моделей

2.2 Применение современных методов прогнозирования потребления электроэнергии

2.2.1. Статистические модели: ARMA, ARIMA, SARIMA

В данном разделе использовались модели и функции из библиотеки statsmodels.

Стоит отметить, что статистическим моделям не нужно так много данных для построения прогноза. Более того, тридцать пять тысяч наблюдений могут потребовать нерационально большого количества времени на работу моделей. Поэтому в этом разделе будем рассматривать только последние несколько месяцев потребления электроэнергии, а именно период с 01.09.2020 по 31.12.2020, где последняя неделя декабря отложена на проверку качества.

Перед тем, как перейти к подбору оптимальных p и q, следует проверить данные на стационарность и сделать предположение о типе процесса.

Среднее и дисперсия со временем становятся постоянными, что может свидетельствовать о стационарности временного ряда.

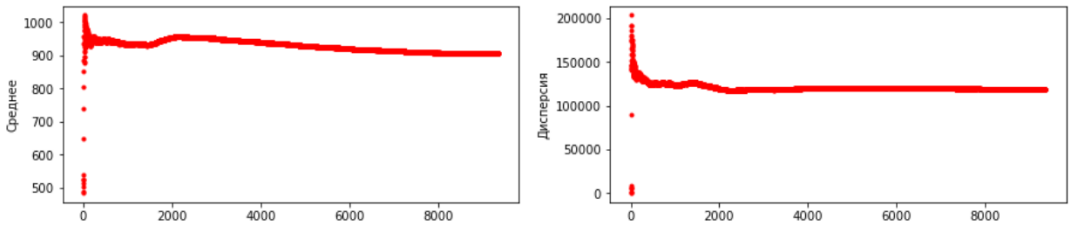

Рисунок 9. Изменение среднего и дисперсии со временем

Проверим это с помощью теста Дикки-Фуллера:

ADF: -11.6006982733305

p-value: 2.6532260249241345e-21

Critical values: {'1%': -3.4310516090751135, '5%': -2.8618500669962805, '10%': -2.5669350435801634}

Значение статистики по модулю больше критических значений, а p-value значительно меньше нуля. Значит, гипотеза о наличии единичного корня отвергается, временной ряд стационарен.

Посмотрим на графики ACF и PACF. На графике автокорреляционной функции коэффициенты убывают медленно, можно даже заметить синусоидальное поведение. Значит, наш временной ряд представляет собой не чистый процесс скользящего среднего.

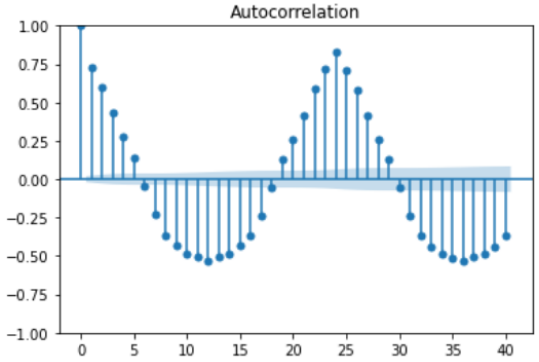

Рисунок 10. График автокорреляционной функции

На графике PACF также коэффициенты похожи на синусоиду. Значит, модель авторегрессии не полностью описывает происходящие во временном ряду процессы.

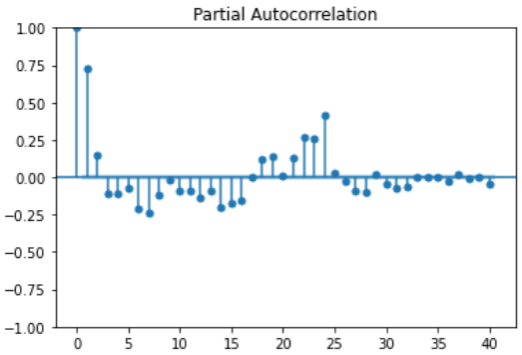

Рисунок 11. График частичной автокорреляционной функции

Следовательно, наиболее подходящей моделью будет ARMA (или ARIMA с I=0).

Оптимальным способом нахождения параметров p и q для модели ARMA является перебор различных комбинаций. В качестве наилучшей модели берется та, у которой критерий Акаике будет наименьшим. После этого остатки требуется проверить на автокорреляцию, и если они ведут себя как белый шум, то модель можно использовать дальше для прогноза. В противном случае следует попробовать другую комбинацию p и q.

Для проверки остатков воспользуемся тестом Льюса-Бокса. Нулевая гипотеза гласит, что данные распределены независимо, что означает отсутствие автокорреляции. Если p-value больше 0.05, мы не можем отвергнуть нулевую гипотезу, следовательно, остатки распределены независимо. Таким образом, автокорреляция отсутствует, остатки похожи на белый шум, и модель можно использовать для прогнозирования.

Еще одним фактором, который следует отслеживать при работе со статистическими моделями является значимость коэффициентов модели. Если p-value для коэффициента превышает 0.05, то рассматриваемый коэффициент не отличен от нуля, и данная модель не подходит для прогнозирования.

Как было указано в предыдущей главе, одно из положительных качеств статистических моделей заключается в их интерпретируемости, однако при p и q выше порядка 3 эта способность теряется. Кроме того, высокие порядки лагов могут привести к переобучению модели и снижению ее точности при

прогнозировании. Если в модель добавляются лаги порядка выше 4-го, то модель может начать аппроксимировать шум, а не реальные зависимости в данных. Также следует учитывать, что чем выше p и q, тем дольше будут вычисляться коэффициенты модели, что может усложнить процесс прогнозирования.

В связи с этим будем рассматривать различные комбинации p и q от 0 до 3 включительно. Для полученной модели будем заносить в таблицу значения критерия Акаике, результаты теста на автокорреляцию остатков и информация о том, все ли коэффициенты модели значимы. Затем останется только провести фильтрацию, убрав модели, которые не подходят хотя бы по одному критерию, и затем рассматривать оставшиеся модели.

В Python есть реализация автоматического подбора параметров для статистических моделей, однако поиск будет осуществлен вручную, чтобы одновременно с фиксацией значения критерия Акаике можно было проводить дополнительные тесты.

Прогноз будем выполнять по МП2.

Перебор параметров занял 20 секунд.

Ниже представлена таблица, отсортированная по убыванию значения критерия Акаике. Как можно заметить, из 16 моделей подходит только 7: ARMA(3,1), ARMA(0,3), ARMA(1,2), ARMA(3,0), ARMA(2,1), ARMA(2,0), ARMA(1,1).

	Order	AIC	Prob(Q)	P>|z|
10	(2, 0, 2)	37333.22	<0.05	all
13	(3, 0, 1)	37450.40	>0.05	all
14	(3, 0, 2)	37794.58	>0.05	not all
15	(3, 0, 3)	37795.93	>0.05	not all
6	(1, 0, 2)	37806.63	>0.05	all
7	(1, 0, 3)	37808.55	>0.05	not all
11	(2, 0, 3)	37809.84	>0.05	not all
12	(3, 0, 0)	37810.95	>0.05	all
9	(2, 0, 1)	37831.55	>0.05	all
8	(2, 0, 0)	37844.67	>0.05	all
5	(1, 0, 1)	37861.96	>0.05	all
4	(1, 0, 0)	37904.94	<0.05	all
3	(0, 0, 3)	37945.79	>0.05	all
2	(0, 0, 2)	38228.18	<0.05	all
1	(0, 0, 1)	38813.67	<0.05	all
0	(0, 0, 0)	39980.88	<0.05	all

Таблица 4. Результаты проверок для ARMA

Построим графики предсказаний. Для статистических моделей прогноз строится так: на основе имеющихся данных предсказывается значение на один вперед, этот прогноз добавляется к уже известным данным, и на основе увеличившийся на одно значения выборке предсказывается следующее значение, и так до нужного номера наблюдения.

Таблицы с полной информацией о значениях ошибки, коэффициента детерминации и времени предсказания можно найти в Приложении.

Как можно увидеть по графикам, прогноз ни по одной из моделей не удается повторить дневную динамику данные. Предсказания представляют собой скорее сильно сглаженные временные ряды.

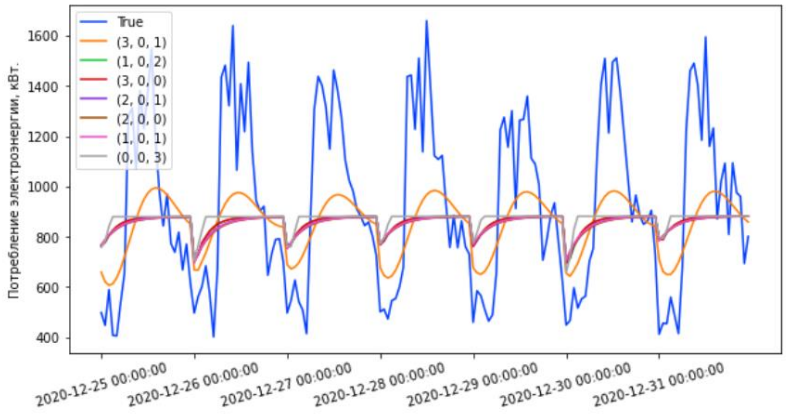

Рисунок 12. Прогноз моделей ARMA

Все коэффициенты детерминации отрицательные, MAPE около 30%, что выше, чем было получено с помощью сезонного прогноза. Коэффициент детерминации положителен, однако очень низкий.

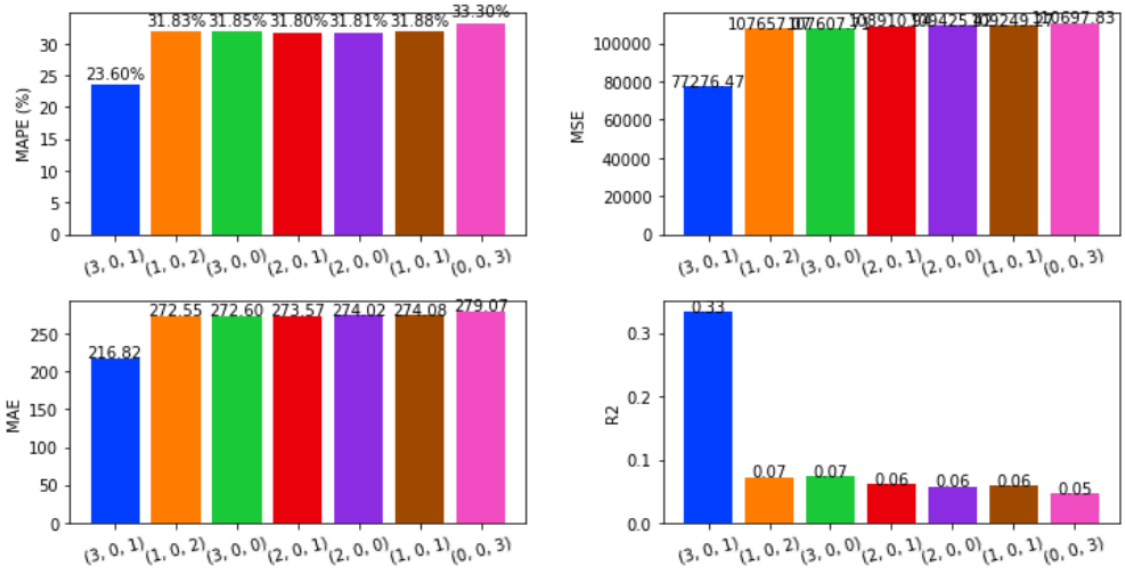

Рисунок 13. Значение метрик для моделей ARMA

Ни одна из моделей ARMA не подходит для дальнейшей работы. Усложним статистическую модел, добавив фактор сезонности. Для этого возьмем частоту в 24 наблюдения. Алгоритм отбора сезонных моделей аналогичен отбору без сезонных. Поскольку теперь требуется перебирать не два параметра, а четыре, что будет затратно по времени и может привести к переполнению оперативной памяти, разобьем подбор параметров на несколько отдельных этапов.

Сначала будем перебирать различные p и q от 0 до 3 при P=1. Это значит, что модель также будет учитывать при прогнозе t-ого наблюдения соответствующее ему наблюдение за вчерашний день.

Перебор параметров занял 3,2 минуты.

Из 16 сезонных моделей остается 11: ARIMA(2,0,3)(1,0,0)[24], ARIMA(3,0,2)(1,0,0)[24], ARIMA(3,0,0)(1,0,0)[24], ARIMA(1,0,2)(1,0,0)[24], ARIMA(1,0,1)(1,0,0)[24], ARIMA(3,0,1)(1,0,0)[24], ARIMA(2,0,0)(1,0,0)[24], ARIMA(0,0,3)(1,0,0)[24], ARIMA(0,0,2)(1,0,0)[24], ARIMA(1,0,0)(1,0,0)[24], ARIMA(0,0,1)(1,0,0)[24].

	Order	AIC	Prob(Q)	P>|z|
11	(2, 0, 3)	36140.53	>0.05	all
14	(3, 0, 2)	36161.50	>0.05	all
10	(2, 0, 2)	36196.67	<0.05	all
15	(3, 0, 3)	36319.31	<0.05	all
7	(1, 0, 3)	36450.88	>0.05	not all
12	(3, 0, 0)	36459.52	>0.05	all
6	(1, 0, 2)	36462.42	>0.05	all
9	(2, 0, 1)	36463.52	>0.05	not all
5	(1, 0, 1)	36465.32	>0.05	all
13	(3, 0, 1)	36482.87	>0.05	all
8	(2, 0, 0)	36498.74	>0.05	all
3	(0, 0, 3)	36501.62	>0.05	all
2	(0, 0, 2)	36548.00	>0.05	all
4	(1, 0, 0)	36557.88	>0.05	all
1	(0, 0, 1)	36585.76	>0.05	all
0	(0, 0, 0)	36667.01	<0.05	all

Таблица 5. Результаты проверок для SARIMA(P=1)

Предсказания выглядит намного точнее. (здесь и далее для удобства в легендах и подписях к осям будет вынесена информация только о параметрах p и q).

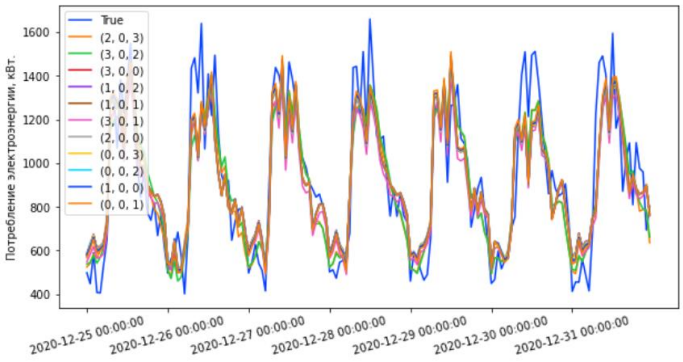

Рисунок 14. Прогноз моделей SARIMA(P=1)

Значения ошибок довольно низкие, коэффициент детерминации около 0,8. Наилучший результат у модели с параметрами p=2, q=3.

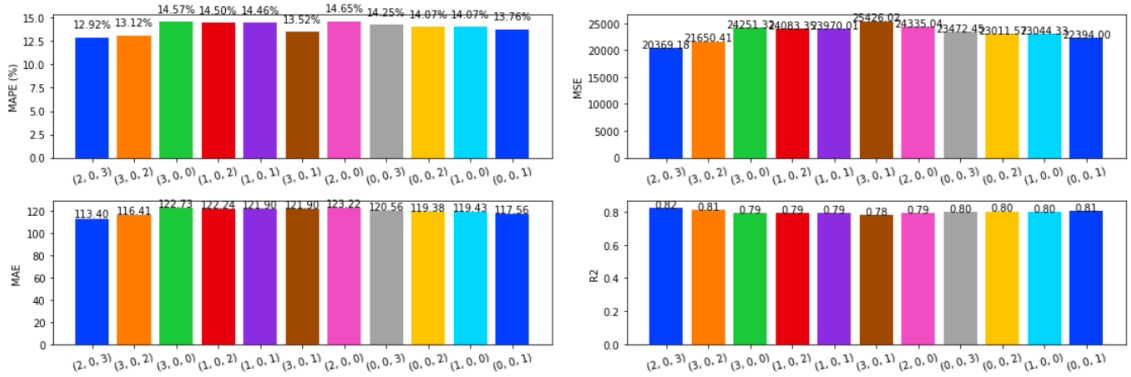

Рисунок 15. Значение метрик для SARIMA(P=1)

Переходим к следующему этапу работы с сезонными моделями. Теперь перебираем p и q при Q=1, то есть исходим из предположения, что следует учитывать не наблюдение за соответствующее наблюдение за день ранее, а его остаток.

Перебор параметров занял 2,5 минуты.

Моделей, подходящих по критериям, стало значительно меньше – 7: ARIMA(1,0,3)(0,0,1)[24], ARIMA(2,0,3)(0,0,1)[24], ARIMA(1,0,2)(0,0,1)[24], ARIMA(3,0,1)(0,0,1)[24], ARIMA(2,0,0)(0,0,1)[24], ARIMA(1,0,1)(0,0,1)[24], ARIMA(0,0,3)(0,0,1)[24].

	Order	AIC	Prob(Q)	P>|z|
10	(2, 0, 2)	36621.54	<0.05	all
15	(3, 0, 3)	36690.26	<0.05	all
7	(1, 0, 3)	37196.33	>0.05	all
11	(2, 0, 3)	37199.30	>0.05	all
6	(1, 0, 2)	37202.08	>0.05	all
13	(3, 0, 1)	37206.83	>0.05	all
14	(3, 0, 2)	37207.35	>0.05	not all
8	(2, 0, 0)	37209.25	>0.05	all
12	(3, 0, 0)	37211.08	>0.05	not all
9	(2, 0, 1)	37211.15	>0.05	not all
5	(1, 0, 1)	37228.67	>0.05	all
4	(1, 0, 0)	37323.11	<0.05	all
3	(0, 0, 3)	37330.60	>0.05	all
2	(0, 0, 2)	37500.12	<0.05	all
1	(0, 0, 1)	37822.86	<0.05	all
0	(0, 0, 0)	38486.04	<0.05	all

Таблица 6. Результаты проверок для SARIMA(Q=1)

Прогнозы напоминают результаты, полученные для несезонной ARIMA.

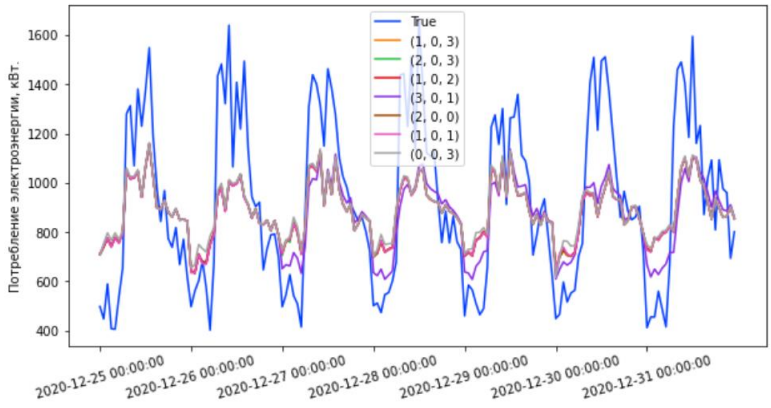

Рисунок 16. Прогноз моделей SARIMA(Q=1)

Качество моделей, ожидаемо, также невысокое.

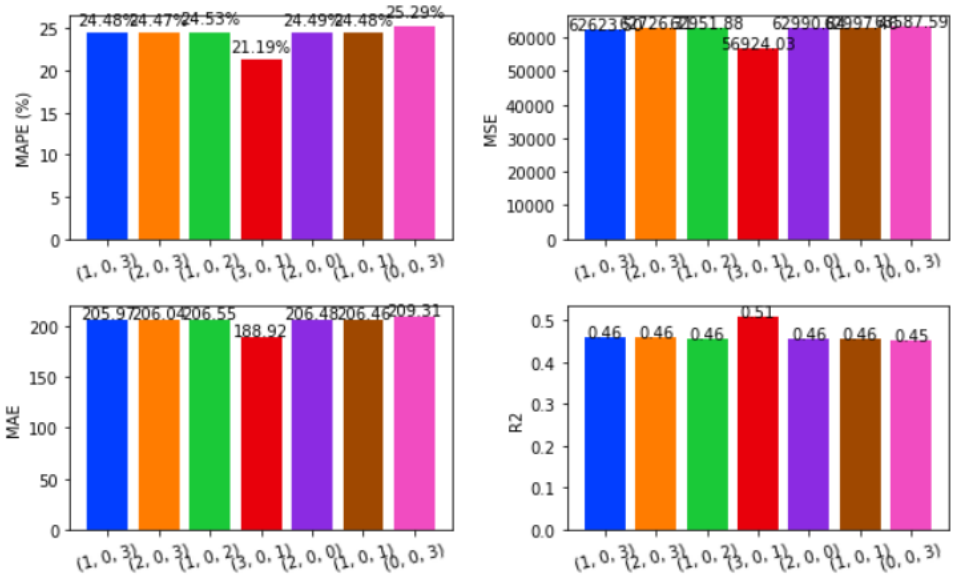

Рисунок 17. Значение метрик для SARIMA(Q=1)

Сезонные модели с Q=1 значительно проигрывают моделям, учитывающим наблюдения за вчерашний день. Не будем фиксировать ни одну из них для дальнейшего рассмотрения.

Наконец переходим к перебору p и q при P=1, Q=1.

Перебор занял уже целых 5,7 минут. В данном случае имеем не 16 моделей, а только 11, поскольку для некоторых комбинаций параметров модели было невозможно построить. Из них удовлетворяют требованиям только 5: ARIMA(0,0,3)(1,0,1)[24], ARIMA(0,0,2)(1,0,1)[24], ARIMA(1,0,3)(1,0,1)[24], ARIMA(1,0,0)(1,0,1)[24], ARIMA(3,0,1)(1,0,1)[24].

	Order	AIC	Prob(Q)	P>|z|
3	(0, 0, 3)	35144.60	>0.05	all
2	(0, 0, 2)	35169.48	>0.05	all
5	(1, 0, 1)	35174.11	<0.05	all
6	(1, 0, 3)	35179.23	>0.05	all
4	(1, 0, 0)	35186.26	>0.05	all
1	(0, 0, 1)	35194.16	<0.05	all
7	(2, 0, 0)	35195.59	<0.05	all
0	(0, 0, 0)	35233.85	<0.05	all
8	(2, 0, 2)	35431.64	<0.05	not all
9	(3, 0, 1)	35740.71	>0.05	all
10	(3, 0, 2)	35971.30	>0.05	not all

Таблица 7. Результаты проверок для SARIMA(P=1, Q=1)

Результаты прогнозирования оказались неутешительными. Одна модель дает отрицательный прогноз, другая выдает слишком большие значения.

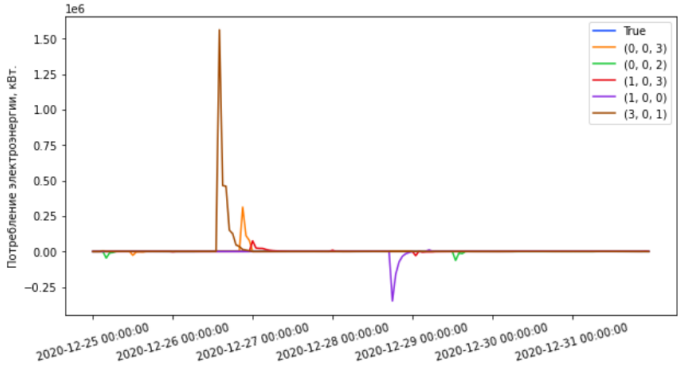

Рисунок 18. Прогноз моделей SARIMA(P=1,Q=1)

Ошибки довольно высокие. Самая минимальная ошибка MAPE = 136,87%, что в несколько раз выше, чем было получено при P=1, Q=0.

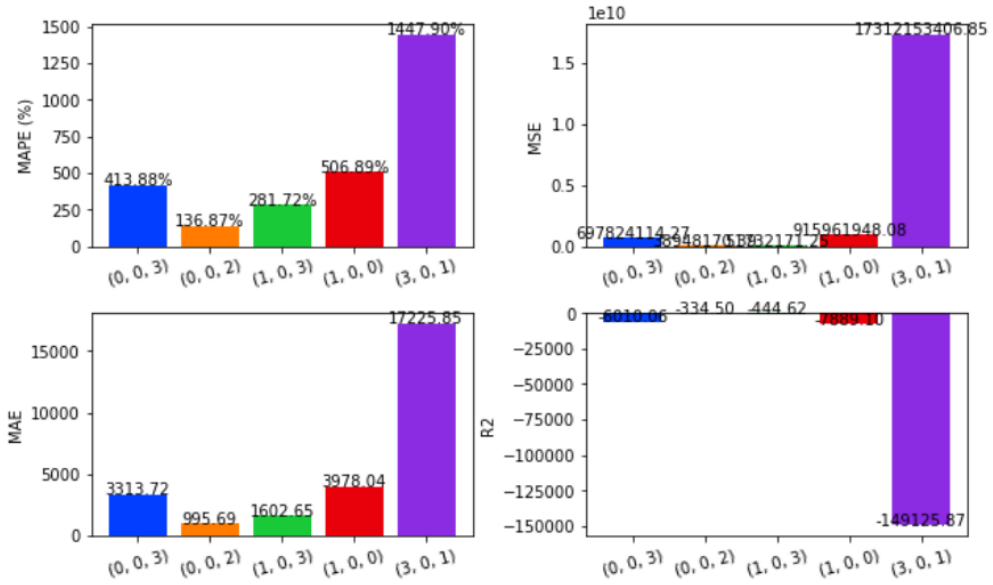

Рисунок 19. Значение метрик для SARIMA(P=1, Q=1)

Стоит отметить, что на этом возможности перебора не заканчиваются. Также можно проверить другие значения P и Q, однако это займет еще большее количество времени и много оперативной памяти. Скорее всего данных «за вчерашний» день более, чем достаточно для хорошего прогноза. Мы уже увидели, что ARIMA(2,0,3)(1,0,0)[24] дает MAPE = 12,92%, R2 = 0,82. Хотя ошибка прогноза уступает тому, что мы получили с помощью наивного сезонного прогноза (разница на 0,23%), статистическая модель лучше подходит для описания динамики потребления энергии (коэффициент детерминации больше на 0,1).

Имеет смысл остановить изучение статистических моделей на данном этапе и попробовать применить для прогноза нейронные сети.

1.2.2. Многослойный перцептрон

Использовалась реализация многослойного перцептрона из библиотеки scikit-learn.

Для тестирования качества модели будут использоваться все три способа: по одному наблюдению за раз, сутки за раз и неделя за раз. Временной ряд был разбит на признаки и целевую переменную с шириной окна 24*7.

Использовалось три подхода к прогнозированию:

МП1. По одному наблюдению за раз. При этом происходит накопление ошибки из-за того, что модель делает прогноз по прогнозу прошлых значений. Этот метод позволит лучше сравнить многослойный перцептрон и деревья со статистическими моделями. Пример входных и выходных данных:

[01.01.2017 00-01, …, 07.01.2017 23-00] -> [08.01.2017 00-01]

МП2. Одни сутки за раз. Модель не учитывает свой предыдущий прогноз и учится сразу предсказывать некоторое количество наблюдений на основе предыдущих фактических данных. Пример входных и выходных данных:

[01.01.2017 00-01, …, 07.01.2017 23-00] -> [08.01.2017 00-01, 08.01.2017 01-02,…, 08.01.2017 23-00]

МП3. Семь дней за раз. Модель также учитывает для прогноза только фактические значения, однако должна суметь предсказать сразу весь горизонт прогноза. Этот метод нужен для оценки моделей по способности строить долгосрочные прогнозы. Пример входных и выходных данных:

[01.01.2017 00-01, …, 07.01.2017 23-00] -> [08.01.2017 00-01, 08.01.2017 01-02,…, 14.01.2017 23-00]

По умолчанию параметр hidden_layer_sizes в MLPRegressor равен (100,), что означает один скрытый слой с 100 нейронами. Функция активация – ReLU, количество эпох на обучение — 200.

Таблицы с метриками и замерами времени также представлены в Приложении.

Многослойный перцептрон даже без подбора оптимальных параметров дал довольно неплохой результат – лучше, чем у наивного прогноза (MAPE=12,69%, R2=0,81) и у статистических моделей (MAPE=12,92%, R2=0,82 для ARIMA(2,0,3(1,0,0)[24]).

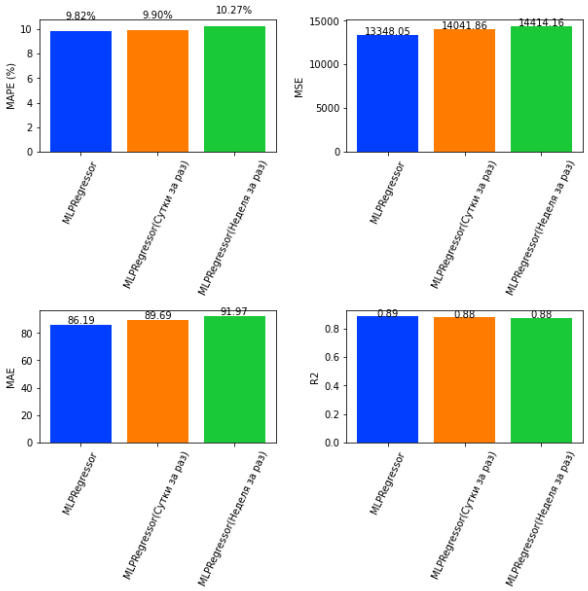

Рисунок 20. Значение метрик для MLP

Если посмотреть непосредственно на график с прогнозами, можно заметить, что перцептрон верно повторяет дневную динамику потребления, но не улавливает скачки во время пика.

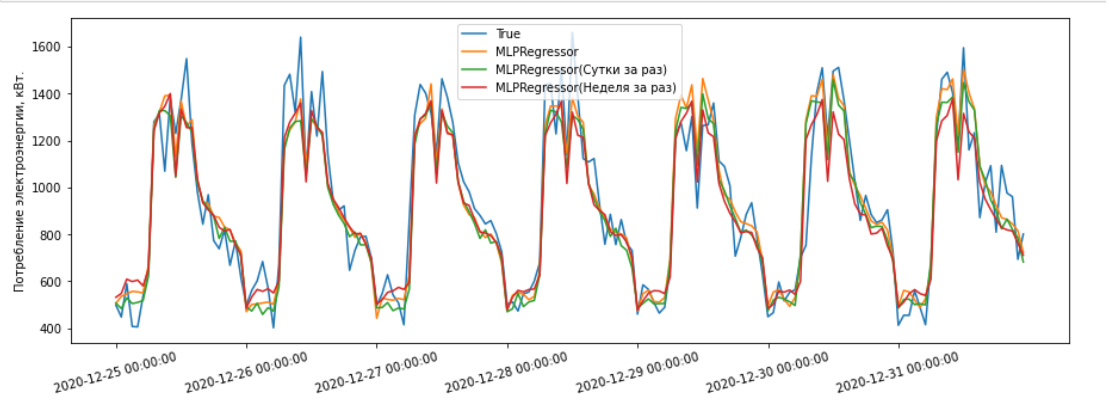

Рисунок 21. Прогноз модели MLP

Воспользуемся поиском по сетке (также из библиотеки scikit-learn) для поиска оптимальных параметров. Случай «неделя за раз» оптимизировать не будем, поскольку это требует большого объема времени и памяти.

Поиск по сетке (Grid Search) — это метод оптимизации гиперпараметров модели, который позволяет автоматически подобрать наилучшие значения гиперпараметров из заданного диапазона значений. Гиперпараметры — это параметры модели, которые не могут быть оптимизированы во время обучения, и должны быть установлены до начала обучения модели.

Найдем оптимальное количество эпох для обучения и функцию активации.

Во всех случаях поиск по сетке остановился на ReLU, максимальное количество итераций составило 400 и 300 эпох соответственно. Прогноз с наблюдением за раз от увеличения эпох обучения ухудшился, в то врем как прогноз на сутки вперед дал выигрыш на 0,03 процента. Коэффициент детерминации, напротив, уменьшился.

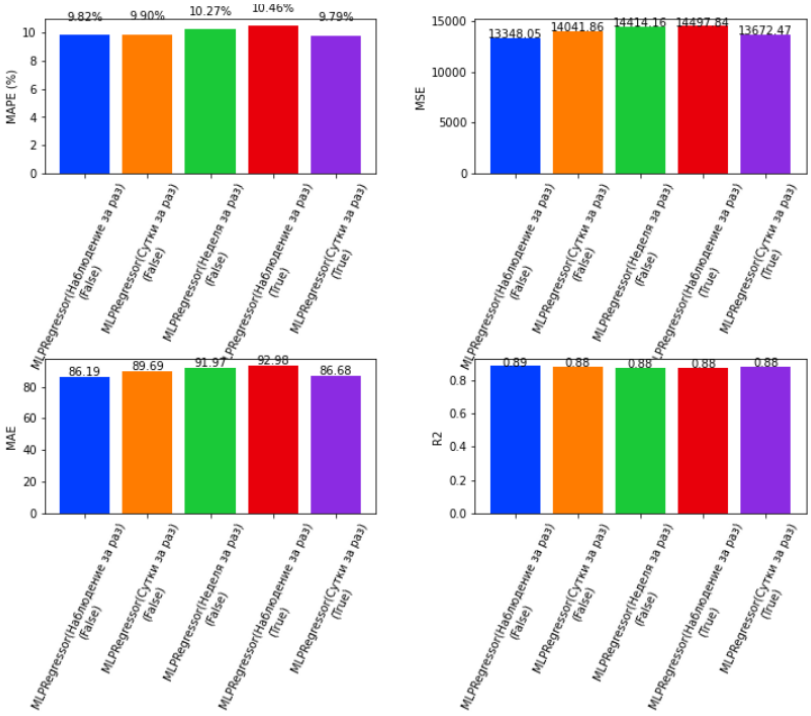

Рисунок 22. Значение метрик для MLP с учетом оптимизации

Дополнительно попробуем провести предобработку данных с целью попытаться улучшить прогноз. Для этого сначала стандартизируем данные, то есть преобразуем каждое наблюдение по следующей формуле:

$$y_t = \frac{y_t - \bar{y}}{\sigma_y},$$

где $\bar{y}$ – среднее по обучающей выборке, σ_y – стандартное отклонение по ней же. Стандартизация (которую иногда также называют z-преобразованием) приводит данные к среднему значению 0 и стандартному отклонению 1.

Кроме этого проведем нормализацию, то есть преобразуем каждое наблюдение по следующей формуле:

$$y_t = \frac{y_t - min(y)}{max(y) - min(y)},$$

Где min(y) и max(y) означают минимальное и максимальное значение по всей выборке соответственно.

Нормализация (масштабирование) используется для приведения значений произвольных признаков к единому диапазону [0, 1]. Это позволяет уменьшить влияние больших значений на результаты анализа.

Преобразования были проведены на некоторых способах прогнозирования для функции активации ReLU (первые три модели) и сигмоиды (последние две модели). Во всех случаях перцептрон продемонстрировал ухудшение качества. Возможно, это связано с нарушением структуры временного ряда.

	Model	Tuner	Params	MAPE	MSE	MAE	R2
5	MLPRegressor (стандартизация) (Наблюдение за раз)	False	{'hidden_layer_sizes': (100,), 'activation': '...	49966.434520	1.939607e+11	4.286894e+05	-1.670776e+06
6	MLPRegressor (стандартизация)(Сутки за раз)	False	{'hidden_layer_sizes': (100,), 'activation': '...	39546.597515	1.345475e+11	3.522819e+05	-1.158991e+06
7	MLPRegressor (нормализация) (Наблюдение за раз)	False	{'hidden_layer_sizes': (100,), 'activation': '...	118919.053481	1.575876e+12	1.131464e+06	-1.357459e+07
8	MLPRegressor (нормализация) (Наблюдение за раз)	False	{'hidden_layer_sizes': (100,), 'activation': '...	256225.421458	5.606313e+12	2.280564e+06	-4.829277e+07
9	MLPRegressor (стандартизация)(Сутки за раз)	False	{'hidden_layer_sizes': (100,), 'activation': '...	31722.801914	9.896559e+10	2.931940e+05	-8.524885e+05

Таблица 8. Результаты применения стандартизации и нормализации

Зафиксируем наилучшие результаты: MAPE = 9,82% и R2 = 0,89 (наблюдение за раз с параметрами по умолчанию) и MAPE = 9,9% и R2 = 0,88 (сутки за раз с параметрами по умолчанию)

Перейдем к деревьям.

1.2.3. Дерево решений и случайный лес

В данном разделе проверялись два вида моделей: деревья решений и случайный лес. Использовались модели из библиотеки scikit-learn.

Дерево решений дает хороший результат только при прогнозировании всех недели за один раз. В остальных случаях ошибка выше, чем у наивного сезонного прогноза. Тем не менее можно предположить, что дерево решений хорошо справляется с долгосрочными прогнозами, если в данных ярко выраженная сезонность.

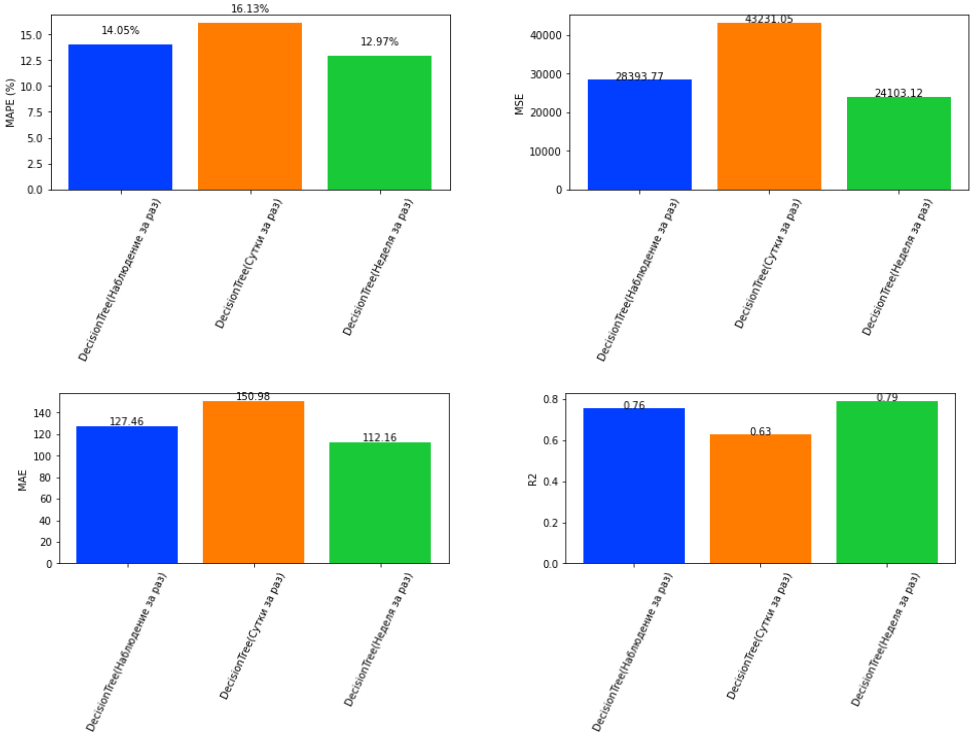

Рисунок 23. Значение метрик для дерева решений

Как можно заметить по графику, дерево решений, в отличие от многослойного перцептрона, учитывает скачки в потреблении электроэнергии, но не всегда правильно улавливает высоту пиков и впадин, из-за чего и возникает большая ошибка.

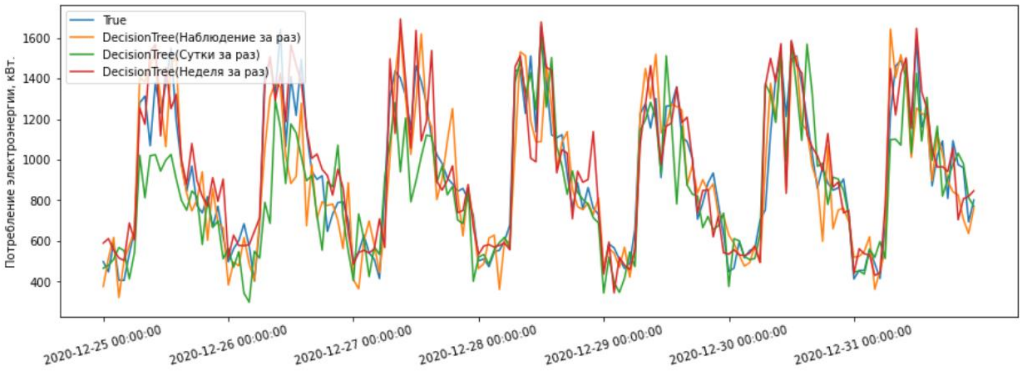

Рисунок 24. Прогноз моделей дерева решений

Воспользуемся поиском по сетке для поиска оптимальной глубины дерева. Для трех типов прогнозирования наилучшими оказались значения 7, 8 и 8. Подобранные значения позволили снизить ошибку MAPE на 3,73%, 5,58% и 3,7% соответственно.

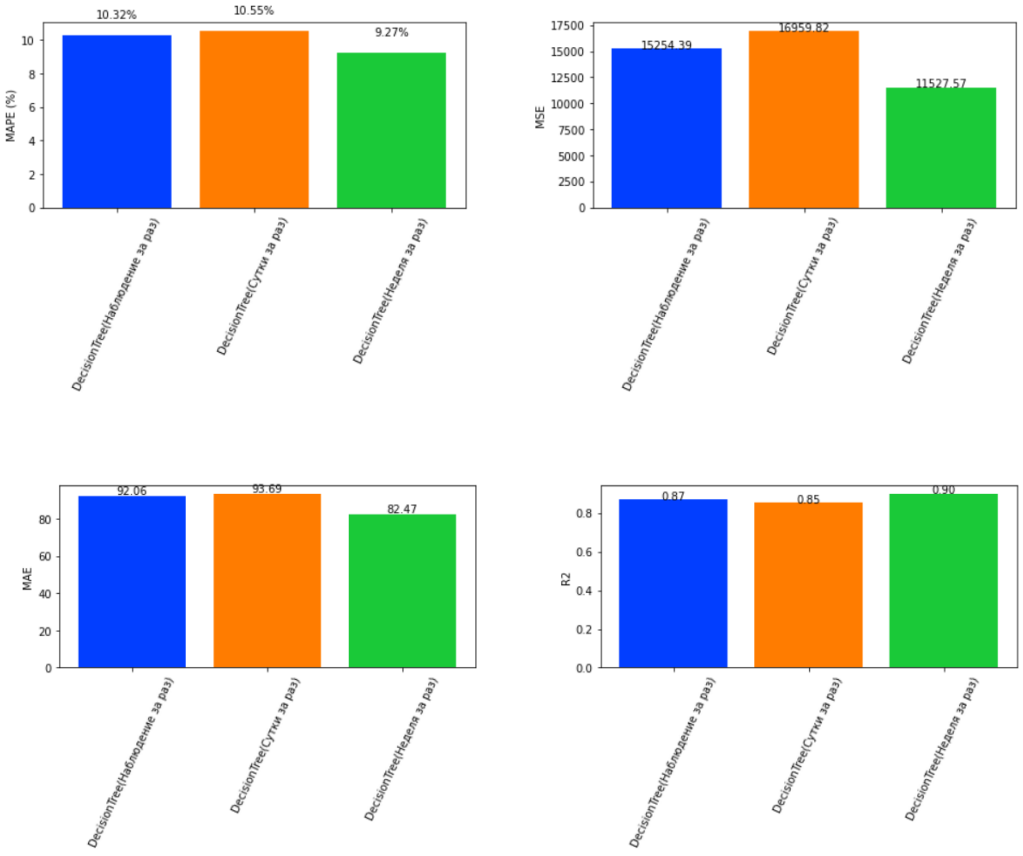

Рисунок 25. Значение метрик для дерева после поиска по сетке

Прогнозные значения при этом довольно похожи по форме на те, что мы могли видеть у многослойного перцептрона, но высота пиков и впадин больше похоже на реальные значения.

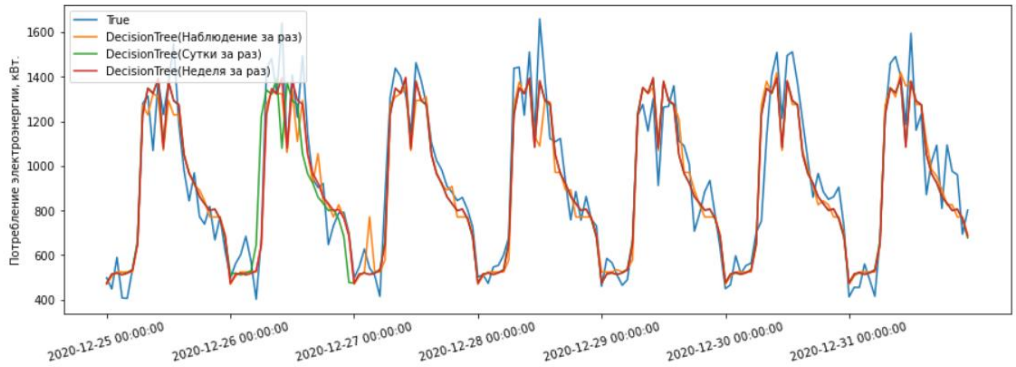

Рисунок 26. Прогноз моделей дерева после поиска по сетке

Повторим все то же самое для случайного леса. Результаты для МПЗ получить не удалось: модель больше пяти часов не могла обучиться на данных, после чего было принято решение прервать процесс как нецелесообразный для продолжения.

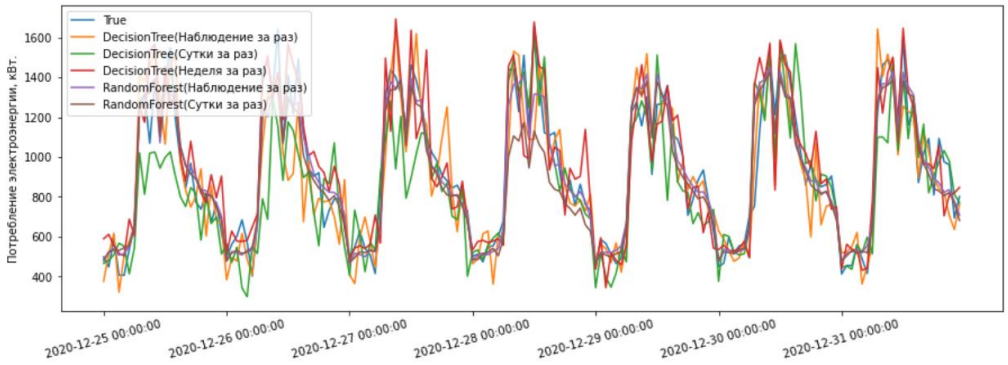

Рисунок 27. Прогноз моделей случайного леса

Даже без учета третьего способа можно заметить, что случайный лес дает более точные прогнозы, нежели дерево решений, причем предсказания с наблюдением на раз оказалась более удачной стратегией. На графике сравниваются модели без применения поиска по сетке.

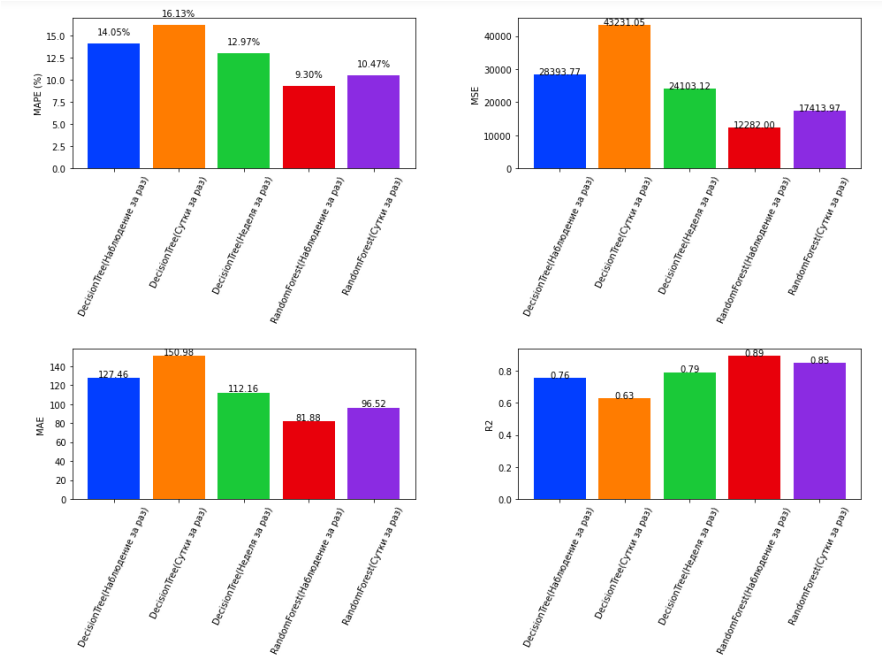

Рисунок 28. Значение метрик для дерева и леса

Для случайного леса оптимизировалось количество деревьев. Были подобраны 150 и 110 деревьев для прогноза по одному наблюдению за раз и для суток за раз. Качество модели после применения данных параметров не улучшилось.

	Model	Tuner	Params	MAPE	MSE	MAE	R2
0	DecisionTree(Наблюдение за раз)	True	{'max_depth': 7}	10.319345	15254.388766	92.061003	0.868599
1	DecisionTree(Сутки за раз)	True	{'max_depth': 8}	10.552595	16959.824443	93.688000	0.853908
2	DecisionTree(Неделя за раз)	True	{'max_depth': 8}	9.273271	11527.566307	82.471311	0.900702
3	RandomForest(Наблюдение за раз)	True	{'n_estimators': 150}	9.312250	12257.438720	82.477158	0.894414
4	RandomForest(Сутки за раз)	True	{'n_estimators': 110}	10.755710	17896.516883	99.554433	0.845839

Таблица 9. Результаты применения стандартизации и нормализации

Стандартизация и нормализация для случайного леса не проводились.

Наилучшие результаты были получены с помощью случайного леса, количество деревьев – 100, прогноз – наблюдение за раз. MAPE в этом случае составляла 9,29%, R2 = 0,89. Это лучше, чем было получено с помощью MLP и SARIMA. Однако, если рассматривать прогнозирование суток за раз (которое в дальнейшем берется за основу при добавлении признаков), то можно заметить, что многослойный перцептрон лучше справляется с задачей: 9,9% (R2 = 0,88) перцептрона против 10,47% (R2 = 0,85) леса.

Глава 3 Оценка результатов применения алгоритма для прогнозирования потребления электроэнергии

3.1 Сравнение моделей прогнозирования и повышение качества прогноза

Проведем сравнение моделей по качеству прогноза и затраченному на обучение и подбор параметров времени с целью выявить наиболее подходящую модель для дальнейшего улучшений алгоритма прогнозирования.

Базовые модели являются самыми и простыми и быстрыми для прогноза временных рядов, особенно если для них характерная ярко выраженная сезонность. Время, потраченное на все четыре модели, измеряется в долях секунды. Благодаря ярко выраженной дневной динамики потребления наивный сезонный прогноз дал довольно низкую ошибку – 12,69%, и коэффициент детерминации равный 0,81.

Среди статистических моделей наиболее релевантными оказались SARIMA с P=1, Q=0. Наилучшие показатели были получены при p=2, q=3: MAPE=12,92%, R2 = 0,82. Результаты подбора параметров позволяют сделать вывод, что при прогнозировании t-ого значения временного наиболее важную роль играют значения t-1, t-2, t-24, а также остатки до третьего лага включительно. Самый главный недостаток в статистических моделях – затраты по времени.

Каждая из возможных комбинаций p, q, P и Q (а в случае не стационарности временного ряда к перебору также добавляются параметры d и D) проверяются на соответствие определенным критериям. После чего для определения модели с наименьшей ошибки требуется построить прогноз по каждой из подходящих моделей.

В таблице приведены показатели по времени для каждой из статистических моделей из предыдущей главы.

Модель	Время на подбор параметров (мин)	Среднее время на прогноз (мин)
ARIMA	2,61	0,33
SARIMA(P=1,Q=0)	50	4,25
SARIMA(P=0,Q=1)	28,59	3,72
SARIMA(P=1,Q=1)	67,13	12,28

Таблица 10. Замер времени для статистических моделей

Очевидно, что статистические модели нерационально использовать для дальнейшего рассмотрения из-за высокой ошибки и времени, уходящего на составление прогноза.

Многослойный перцептрон позволяет подходить к задаче прогноза более гибко: прогнозировать можно как одно наблюдение, так и любое другое количество за один раз. Стратегия прогнозирования по одному наблюдению все остается наиболее подходящей для данной модели. На подбор оптимальных параметров требуется в среднем 38 минут. Обучение модели занимает в среднем 1 минуту, на предсказание уходит несколько секунд.

Стандартизация и нормализация приводят к значительному ухудшению прогноза. Подбор оптимальной функции активации и эпох на обучение не позволило уменьшить ошибку.

Чем больше наблюдений прогнозируем за раз, тем больше времени требуется на обучения модели; время на прогноз же наоборот, уменьшается.

Деревья также демонстрируют преимущество перед статистическими моделями. Ансамбль деревьев решений – случайный лес – позволяет снизить ошибку в среднем на 5%. Прогнозирования по наблюдению за раз является оптимальным вариантом для случайного леса, но деревья по отдельности при построении долгосрочного прогноза дают наилучший прогноз, если делать прогноз на весь горизонт (в нашем случае, неделю) за один раз.

Оптимизация глубины дерева позволила дать выигрыш по качеству модели, но оптимизация количества деревьев в лесу дала обратный эффект.

Стандартизация и нормализация также не подходят как инструмент для повышения точности прогноза для временных рядов с сильно выраженной дневной динамикой.

Случайный лес требует больше времени, чем отдельное дерево или многослойный перцептрон, но качество от использования данной модели действительно становится выше.

Модель	Среднее время на подбор параметров (мин)	Среднее время на обучение (мин)	Среднее время на прогноз (мин)
Дерево решений	7,45	0,9685	0,0005
Случайный лес	34,53	21,92	0,0162

Таблица 11. Замер времени для деревьев

До этого при построении прогноза использовались лишь предыдущие значения временного ряда. Попробуем улучшить качество, добавив новые признаки или произведя дополнительные преобразования. В рассмотрении будет участвовать как случайной лес, так и многослойный перцептрон.

При эксперименте с добавлением новых признаков к временному ряду будет использоваться прогнозирование сразу 24 наблюдений за раз, чтобы избежать накопления ошибки при прогнозировании по одному наблюдению за раз. Это связано с тем, что прогноз по одному наблюдению требует прогнозирования значений признаков, что может привести к накоплению ошибки в долгосрочной перспективе. Прогнозирование на сутки вперед позволяет уменьшить влияние этих ошибок и повысить точность прогноза.

Первыми были рассмотрены следующие признаки:

- 1 если наблюдение записано в период с 6 до 13 часов, иначе 0 (согласно матрице корреляции, этот признак сильнее всего коррелирует с целевой переменной). В таблице имеет индекс 0.

- три переменные зима, лето и осень: 1 если наблюдение соответствует указанному времени года, иначе 0. В таблице имеет индекс 1.

- среднее потребление за все года за этот месяц. Индекс 2.

- шесть столбцов, отвечающих за каждый отдельный день недели, кроме воскресения; 1 если наблюдение относится к этому дню, иначе 0. Индекс 8.

- для каждого дня недели был рассчитаны максимально и минимальное возможные значения. Индекс 9.

Новые признаки были рассмотрены по отдельности и комбинациями:

- индекс 3: времена года и среднее потребление за месяц;

- индекс 10: пик потребления за день и время года;

- индекс 11: пик и среднее потребление за месяц;

- индекс 12: все представленные признаки вместе.

Будем сравнивать их со значениями, полученными для моделей до применения подбора гиперпараметров: MAPE = 9,9%, R2 = 0,88 для многослойного перцептрона и MAPE = 10,47%, R2 = 0,85 для случайного леса.

Как можно заметить по таблице, введение дополнительных переменных только ухудшило качество прогноза многослойного перцептрона. Исключением является только переменная peak – ее использование позволило уменьшить ошибку MAPE на 0,24%. Время в таблице указано в секундах.

	Model	Tuner	Params	MAPE	MSE	MAE	R2	Training_time	Tuner_time	Time_prediction
0	MLPRegressor(+peak)	False	{'hidden_layer_sizes': (100,), 'activation': '...	9.655904	13115.709804	86.488395	0.887021	83.455122	None	0.000000
1	MLPRegressor(+season)	False	{'hidden_layer_sizes': (100,), 'activation': '...	10.095392	14171.505830	89.118830	0.877927	159.574660	None	0.000000
2	MLPRegressor(+month_mean)	False	{'hidden_layer_sizes': (100,), 'activation': '...	10.068140	13173.387665	87.663783	0.886524	82.506755	None	0.000000
3	MLPRegressor(+season+month_mean)	False	{'hidden_layer_sizes': (100,), 'activation': '...	11.158617	16333.361888	99.086663	0.859304	92.650582	None	0.015630
8	MLPRegressor (Weekday)(Сутки за раз)	False	{'hidden_layer_sizes': (100,), 'activation': '...	10.278218	14955.876656	91.731301	0.871170	421.062005	None	0.006003
9	MLPRegressor (Day_min_max)(Сутки за раз)	False	{'hidden_layer_sizes': (100,), 'activation': '...	10.483676	15048.417115	90.937062	0.870373	86.049972	None	0.029009
10	MLPRegressor(+peak+seasons)	False	{'hidden_layer_sizes': (100,), 'activation': '...	10.711048	14530.092321	90.349169	0.874838	283.156462	None	0.000000
11	MLPRegressor(+peak+mean_month)	False	{'hidden_layer_sizes': (100,), 'activation': '...	10.820576	16247.867018	96.889089	0.860041	119.801331	None	0.000000
12	MLPRegressor(+)	False	{'hidden_layer_sizes': (100,), 'activation': '...	10.690665	15512.546031	94.661697	0.866375	240.264622	None	0.000000

Таблица 12. Результаты MLP для дополнительных признаков

Можно заметить, что введение дополнительных признаков увеличивает время на обучение модели, хотя время на предсказание остаётся практически без изменений.

Один из возможных способов повысить качество модели прогнозирования временного ряда – добавить как признаки компоненты тренда, сезонности и остатков. Для декомпозиции использовалась библиотека statsmodels. Периоды сезонности были заданы 24 и 24*7, чтобы учесть дневную динамику и недельную. Кроме этого сезонная компонента была рассмотрена отдельно от остальных. Все три варианта оказались неподходящими для данных, ошибка выросла в несколько раз.

	Model	Tuner	Params	MAPE	MSE	MAE	R2	Training_time	Tuner_time	Time_prediction
4	MLPRegressor (декомпозиция)(Сутки за раз)	False	{'hidden_layer_sizes': (100,), 'activation': '...	37.607393	116151.637374	291.672382	-0.000530	65.058694	None	0.000000
5	MLPRegressor (декомпозиция, period=24*7(Сутки ...	False	{'hidden_layer_sizes': (100,), 'activation': '...	37.654788	115938.542695	292.877396	0.001305	103.220293	None	0.015629
6	MLPRegressor (дек.,seasonal)(Сутки за раз)	False	{'hidden_layer_sizes': (100,), 'activation': '...	114.038343	871800.659399	927.732474	-6.509690	46.297348	None	0.000000

Таблица 13. Результаты MLP для дополнительных признаков

Временные ряды часто содержат шумы и выбросы, которые могут сильно влиять на производительность модели при прогнозировании значений временного ряда. Сглаживание позволяет убрать эти шумы и выбросы, что улучшает качество данных и помогает модели лучше улавливать тренды и сезонности в данных. Существует несколько методов сглаживания временного ряда, таких как скользящее среднее, экспоненциальное сглаживание и скользящее медианное значение. В каждом из этих методов применяется свой подход к уменьшению шумов и выбросов в данных. Экспоненциальное сглаживание — это метод, при котором каждое значение временного ряда заменяется на взвешенную сумму текущего значения и предыдущих значений с использованием экспоненциального весового коэффициента. Этот метод хорошо работает для устранения шумов и выбросов в данных, и может улавливать как тренд, так и сезонность в данных.

Применение экспоненциального сглаживания не решило проблему понижения ошибки прогноза, однако можно заметить, что время обучения модели сократилось почти в два раза.

	Model	Tuner	Params	MAPE	MSE	MAE	R2	Training_time	Tuner_time	Time_prediction
7	MLPRegressor (ExponentialSmoothing)(Сутки за раз)	False	{'hidden_layer_sizes': (100,), 'activation': '...	15.3666	42633.597042	141.378529	0.632754	44.991345	None	0.004997

Таблица 14. Результаты MLP для экспоненциального сглаживания

Далее были рассмотрены данные о погоде (meteo), включающие в себя среднюю температуру, влажность, скорость ветра, а также количество ясных, облачных, пасмурных, дождливых и снежных дней за соответствующие месяца.

Метеоданные рассматривались как отдельная группа признаков и вместе со всеми предыдущими. Ошибка также выросла.

	Model	Tuner	Params	MAPE	MSE	MAE	R2	Training_time	Tuner_time	Time_prediction
13	MLPRegressor(+meteo)	False	{'hidden_layer_sizes': (100,), 'activation': '...	10.468377	14641.813807	90.425583	0.873875	533.928230	None	0.042003
14	MLPRegressor(+)	False	{'hidden_layer_sizes': (100,), 'activation': '...	10.771490	14688.769681	91.667145	0.873471	623.738128	None	0.137008

Таблица 15. Результаты MLP для данных о погоде

Поскольку преобразование данных и добавление новых признаков ничего не дало, рассмотрим разную ширину окна при разделении временного ряда. Нельзя однозначно сказать, что увеличение или уменьшение ширины окна ведет к улучшению качества – этот параметр также строит подбирать в каждом конкретном случае. Уменьшение окна до 96 наблюдений (то есть рассматривается не неделя, а четыре дня) позвонило снизить ошибку MAPE до 9,39% по сравнению с исходной моделью.

	Model	Tuner	Params	MAPE	MSE	MAE	R2	Training_time	Tuner_time	Time_prediction
15	MLPRegressor(window=24)	False	{'hidden_layer_sizes': (100,), 'activation': '...	11.542390	16082.146265	99.224752	0.861468	67.067709	None	0.000000
16	MLPRegressor(window=48)	False	{'hidden_layer_sizes': (100,), 'activation': '...	10.282400	13240.718035	90.270200	0.885944	69.770292	None	0.004210
17	MLPRegressor(window=72)	False	{'hidden_layer_sizes': (100,), 'activation': '...	10.086013	12781.992587	84.345181	0.889896	72.254937	None	0.000000
18	MLPRegressor(window=96)	False	{'hidden_layer_sizes': (100,), 'activation': '...	9.385679	11744.354642	81.245121	0.898834	78.133830	None	0.000000
19	MLPRegressor(window=120)	False	{'hidden_layer_sizes': (100,), 'activation': '...	10.216974	13164.097054	87.151256	0.886604	93.598202	None	0.003002
20	MLPRegressor(window=144)	False	{'hidden_layer_sizes': (100,), 'activation': '...	10.336029	13768.631560	88.880555	0.881397	85.999246	None	0.003001
21	MLPRegressor(window=192)	False	{'hidden_layer_sizes': (100,), 'activation': '...	11.199636	16796.790663	103.291409	0.855312	64.806209	None	0.040003
22	MLPRegressor(window=216)	False	{'hidden_layer_sizes': (100,), 'activation': '...	11.241642	14620.442440	92.066523	0.874060	53.283562	None	0.042003
23	MLPRegressor(window=336)	False	{'hidden_layer_sizes': (100,), 'activation': '...	10.894178	17237.468785	102.154317	0.851516	55.148426	None	0.055006
24	MLPRegressor(window=12)	False	{'hidden_layer_sizes': (100,), 'activation': '...	14.917751	22447.758114	120.544014	0.806635	59.338384	None	0.001001
25	MLPRegressor(window=36)	False	{'hidden_layer_sizes': (100,), 'activation': '...	11.253332	15231.652938	98.301644	0.868795	66.056443	None	0.000910

Таблица 16. Результаты MLP для разной ширины окна

Результаты этого эксперимента позволили сделать вывод, что добавление новых признаков при использовании многослойного перцептрона зачастую может вести к ухудшению прогноза. Следовательно, прежде чем строить многофакторные модели, важно попробовать вывести на минимальную ошибку за счет одного лишь временного ряда с потреблением электроэнергии, и только после этого усложнять модель.

Для случайного леса были рассмотрены некоторые из признаков, используемых для многослойного перцептрона.

Случайный лес дал улучшения при использовании всех признаков, кроме декомпозиции и экспоненциального сглаживая. Наименьшая ошибка – 9,646% — была получена при добавлении максимально и минимально возможного потребления за этот день недели. Как и в случае с многослойным перцептроном, скорость работы модели увеличивается при добавлении новых признаков.

	Model	Tuner	Params	MAPE	MSE	MAE	R2	Training_time	Tuner_time	Time_prediction
0	RandomForest(+peak)	False	{'n_estimators': 100, 'max_depth': None, 'max_...	10.486980	16817.424798	96.741089	0.855135	1293.969913	None	0.086174
1	RandomForest(+season)	False	{'n_estimators': 100, 'max_depth': None, 'max_...	9.959994	14755.975652	89.837962	0.872892	1837.463272	None	0.063428
2	RandomForest(+month_mean)	False	{'n_estimators': 100, 'max_depth': None, 'max_...	10.139437	15453.931762	92.032890	0.866880	2789.697496	None	0.134156
3	RandomForest(+season+month_mean)	False	{'n_estimators': 100, 'max_depth': None, 'max_...	10.041118	15465.749658	91.589696	0.866778	3141.574125	None	0.135363
4	RandomForestRegressor (декомпозиция) (Сутки за ...	False	{'max_depth': None, 'max_leaf_nodes': None, 'm...	37.469634	117026.514135	293.162931	-0.008066	5528.487760	None	0.483948
5	RandomForestRegressor (ExponentialSmoothing)(C...	False	{'max_depth': None, 'max_leaf_nodes': None, 'm...	15.604154	45605.921319	145.236786	0.607151	1635.611406	None	0.059867
6	RandomForestRegressor (Weekday) (Сутки за раз)	False	{'max_depth': None, 'max_leaf_nodes': None, 'm...	9.816322	14791.477838	89.741873	0.872586	1928.316648	None	0.127975
7	RandomForestRegressor (Day_min_max) (Сутки за раз)	False	{'max_depth': None, 'max_leaf_nodes': None, 'm...	9.645502	14123.313283	87.753817	0.878342	2882.832672	None	0.135008
8	RandomForestRegressor (Weekday+peak)(Сутки за ...	False	{'max_depth': None, 'max_leaf_nodes': None, 'm...	10.138485	15484.183340	92.444296	0.866619	5904.829086	None	0.431782
9	RandomForestRegressor(+meteo)	False	{'max_depth': None, 'max_leaf_nodes': None, 'm...	9.731870	13934.399122	85.828651	0.879969	5727.496444	None	0.127657
10	RandomForest(+Weekday+Day_min_max)	False	{'max_depth': None, 'max_leaf_nodes': None, 'm...	10.191279	14927.788739	91.977574	0.871412	2897.115137	None	0.149992
11	RandomForest(+Day_min_max+meteo)	False	{'max_depth': None, 'max_leaf_nodes': None, 'm...	9.836327	16128.059437	90.021425	0.861073	4372.448235	None	0.526765

Таблица 17. Результаты для леса с дополнительными признаками

Затем было исследование влияние ширины окна на качество модели. Наименьшая ошибка была получена при окне 120 наблюдений (то есть пять дней) (MAPE=9,25%). При проверке некоторых признаков с этой шириной окна оказалось, что добавление данных о времени года и дня недели ошибка уменьшается до 9,08%. Качество меньше этого без использования поиска по сетке добиться не удалось.

	Model	Tuner	Params	MAPE	MSE	MAE	R2	Training_time	Tuner_time	Time_prediction
12	RandomForest(window=24)	False	{'max_depth': None, 'max_leaf_nodes': None, 'm...	9.742900	13350.619154	87.217700	0.884998	224.068148	None	0.076532
13	RandomForest(window=48)	False	{'max_depth': None, 'max_leaf_nodes': None, 'm...	9.310687	11751.002546	82.598195	0.898777	505.055036	None	0.140013
14	RandomForest(window=72)	False	{'max_depth': None, 'max_leaf_nodes': None, 'm...	9.389635	11892.505130	83.879044	0.897558	958.401173	None	0.350011
15	RandomForest(window=96)	False	{'max_depth': None, 'max_leaf_nodes': None, 'm...	9.479737	12112.287954	83.570180	0.895665	843.985928	None	0.095303
16	RandomForest(window=120)	False	{'max_depth': None, 'max_leaf_nodes': None, 'm...	9.252344	11392.998677	82.029204	0.901861	1324.926885	None	0.442034
17	RandomForest(window=144)	False	{'max_depth': None, 'max_leaf_nodes': None, 'm...	9.648523	12359.251600	85.493467	0.893537	1838.990504	None	0.483414
18	RandomForest(window=96)	False	{'max_depth': None, 'max_leaf_nodes': None, 'm...	9.284736	11095.699854	81.699611	0.904422	1146.709918	None	0.121009
19	RandomForest (Day_min_max, window=120)	False	{'max_depth': None, 'max_leaf_nodes': None, 'm...	9.400094	11946.433542	83.370697	0.897093	1862.016893	None	0.067815
20	RandomForest (seasons, window=120)	False	{'max_depth': None, 'max_leaf_nodes': None, 'm...	9.204820	11722.651462	80.345143	0.899021	1930.621379	None	0.516465
21	RandomForest (seasons, weekday, window=120)	False	{'max_depth': None, 'max_leaf_nodes': None, 'm...	9.079022	11129.510740	78.886332	0.904130	1972.090617	None	0.083265

Таблица 18. Результаты для леса с разной шириной окна

Можно и дальше продолжить изучение влияние различных комбинаций признаков и гиперпараметров на ошибку модели случайного леса, однако попробуем другой способ.

Посмотрим на графики ошибки модели на обучающей выборке. Визуально закономерностей не видно, остатки похожи на белый шум.

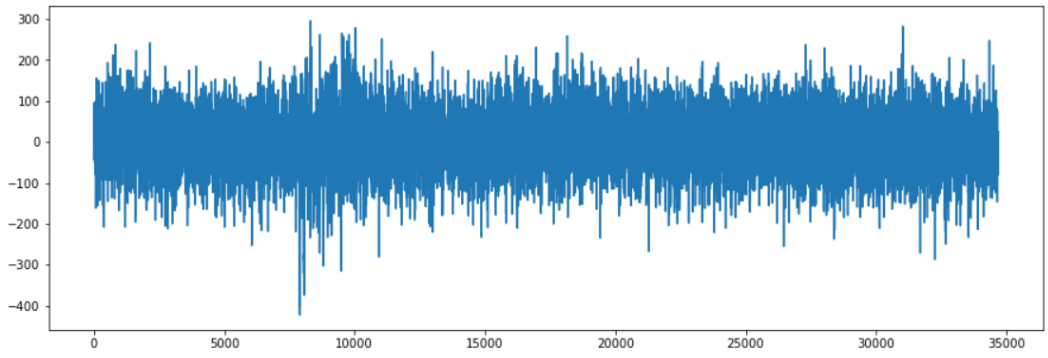

Рисунок 29. График остатков за весь период

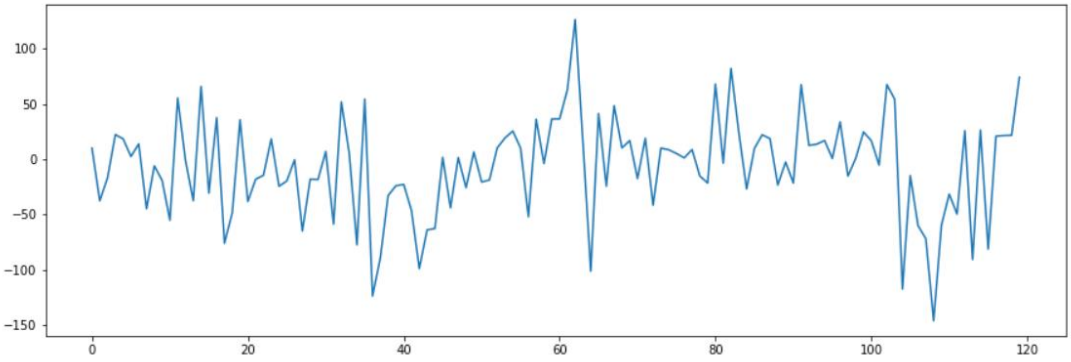

Рисунок 30. График остатков за последнюю неделю

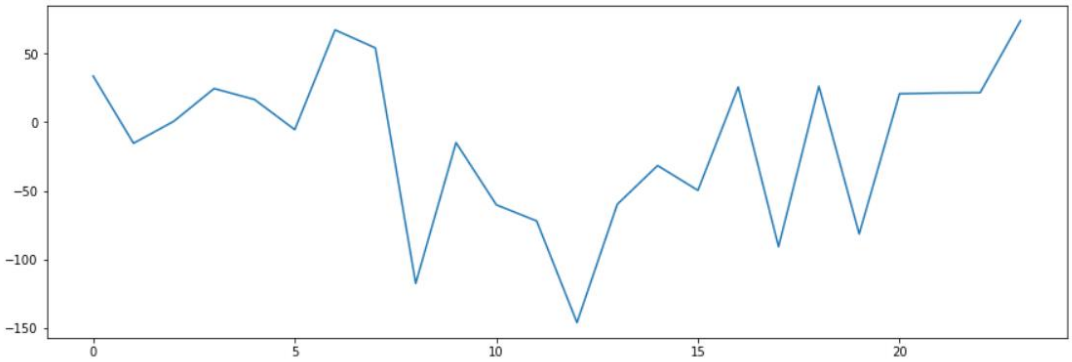

Рисунок 31. График остатков за последний день

Однако следует все равно попробовать подобрать способ улучшить модель за счет предсказания будущей ошибки прогноза: возможно, случайный лес не учитывает некую закономерность, которую можно попробовать извлечь другим способом.

(модель 0). Один из способов – использовать ошибку за «вчерашний» день. Следовательно, для корректировки прогноза 25 декабря будет использоваться разница между фактом и прогнозом за 24 декабря (из обучающей выборки). Тогда:

$$y_{adj} = y_{pred} + e_{lastday}$$

(модель 1). Будем прогнозировать будущие остатки с помощью многослойного перцептрона, который быстро обучается и более гибкий. Для обучения модели использовались остатки с обучающей выборки; ширина окна для разбиения временного ряда на части возьмем окно 120 наблюдений, прогноз – 24 наблюдения будущей ошибки.

(модель 2). Эту же модель оптимизируем с помощью поиска по сетке. Оптимальными оказалась функция активации логическая, максимальное количество эпох для обучения – 150.

(модель 3). Построим модель линейной регрессии, где признаки – это предсказания случайного леса, а целевая переменная – фактические значения. Эта модель не будет учитывать структуру временного ряда, однако линейная регрессия может уловить закономерности в ошибке прогноза.

(модель 4). Попробуем добавить в предсказание шум из нормального распределения со средним 0 и стандартным отклонением 1.

Результаты попыток скорректировать прогноз приведены в таблице ниже. К сожалению, добиться улучшения не удалось. Это может также означать, что случайный лес уже учитывает все возможные закономерности в данных, и остатки представляют собой белый шум, который невозможно спрогнозировать.

	Model	MAPE	MSE	MAE	R2
0	Previous_residues	10.611883	16271.713614	95.917775	0.859835
1	Predict_residues	10.550038	16019.065652	93.998577	0.862012
2	Predict_residues(tuner)	9.234793	12083.629798	80.636094	0.895912
3	Predict_true	9.230689	12037.021438	80.510660	0.896313
4	+noise	9.228742	12023.820495	80.469216	0.896427

Таблица 19. Результаты корректировки прогноза

Еще один способ скорректировать прогноз: ввести еще две модели, которые будут предсказывать максимальное и минимальное значение потребления на следующий день. Для этого использовался случайный лес.

На графике изображен нескорректированный прогноз потребления электроэнергии (синяя линия) и предсказанные минимальные и максимальные значения за эти дни (зеленый пунктир).

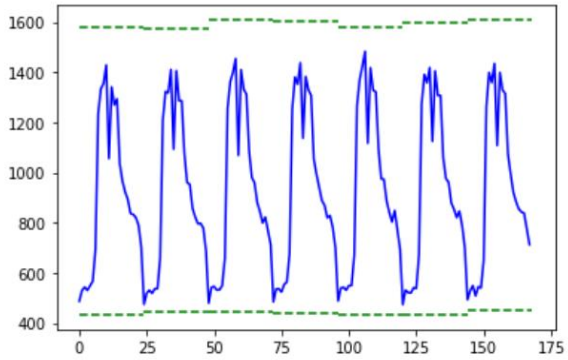

Рисунок 32. Прогноз случайного леса и прогноз min и max

Ошибка после корректирования все равно высокая – 9,55% против исходных 9,02%

	Model	MAPE	MSE	MAE	R2
5	predict min-max	9.551067	12920.249522	83.697382	0.888705

Таблица 20. Результаты корректировки с учетом предсказания min и max

Однако если посмотреть на графики фактических значений и скорректировано прогноза, можно заметить, что после «вытягивания» прогноза потребления результат стал чуть больше походить на правду. Скорее всего при правильном дальнейшем подборе модели для прогнозирования минимума и максимума удастся добиться большей точности.

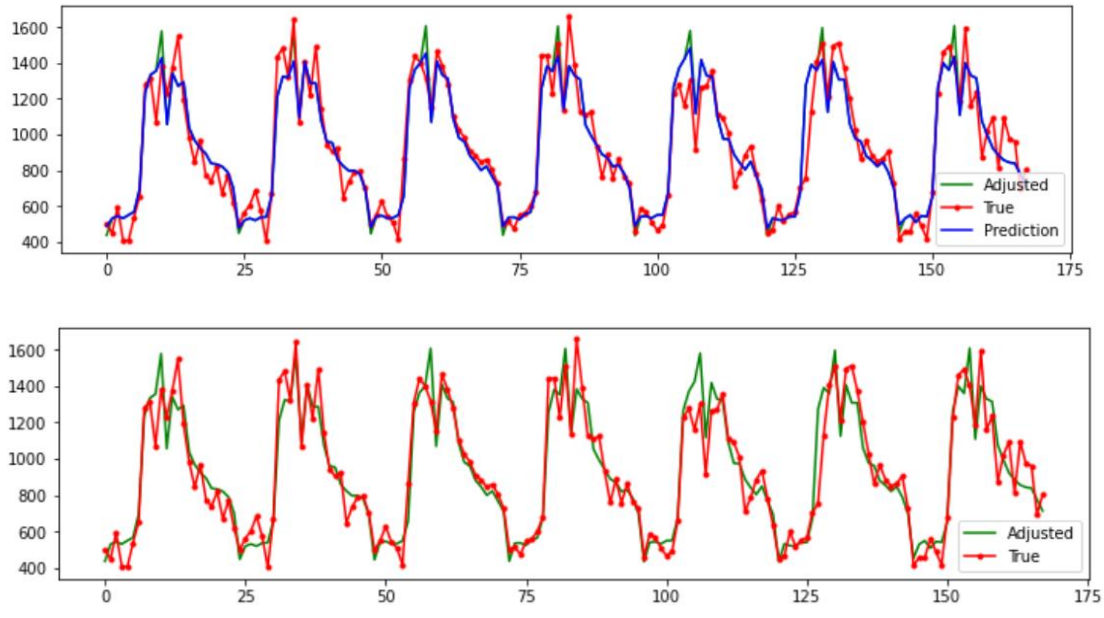

Рисунок 33. Фактические значения, прогнозные и скорректированные

В заключение раздела обратимся к одному из преимуществ случайного леса – возможности интерпретации результатов – и посмотрим на первые несколько правил, по которым происходило разделение для первого дерева. «Truncated branch of depth» здесь означает, что на данной ветке дерева обрывается ветка, потому что глубина дерева достигла предела. Обычно это происходит, когда дерево достигло определенной глубины, чтобы избежать переобучения.

```
Ввод [829]: print(export_text(model.estimators_[0], feature_names=feature_names))
|--- Usage_kWh t-21 <= 925.79
|   |--- Usage_kWh t-86 <= 855.28
|   |   |--- Usage_kWh t-58 <= 835.26
|   |   |   |--- Usage_kWh t-104 <= 825.28
|   |   |   |   |--- Usage_kWh t-77 <= 722.92
|   |   |   |   |   |--- Usage_kWh t-51 <= 830.34
|   |   |   |   |   |   |--- Usage_kWh t-120 <= 852.70
|   |   |   |   |   |   |   |--- Usage_kWh t-35 <= 1039.38
|   |   |   |   |   |   |   |   |--- Usage_kWh t-17 <= 651.38
|   |   |   |   |   |   |   |   |   |--- value: [775.60, 1333.64, 1045.68, 1614.04, 1293.16, 661.39, 1014.77, 1256.92, 1403.7
2, 1084.98, 805.86, 907.23, 870.91, 975.87, 925.36, 720.04, 657.06, 753.41, 806.55, 716.72, 669.45, 709.28, 861.05, 790.77]
|   |   |   |   |   |   |   |   |   |--- Usage_kWh t-17 >  651.38
|   |   |   |   |   |   |   |   |   |   |--- Summer t-90 <= 0.50
|   |   |   |   |   |   |   |   |   |   |   |--- Usage_kWh t-52 <= 578.14
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |   |   |--- Usage_kWh t-52 >  578.14
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |   |--- Summer t-90 >  0.50
|   |   |   |   |   |   |   |   |   |   |   |--- Usage_kWh t-61 <= 622.31
```

Рисунок 34. Начало первого дерева в лесу

Информацию, полученную из деревьев в случайном лесу, аналитик предприятия может использовать в дальнейшем для оптимизации процессов на производстве или для дальнейшего улучшения качества прогноза.

3.2 Алгоритм прогнозирования и рекомендации по дальнейшему повышению точности прогноза

Наилучший результат из проведенных экспериментов был достигнут с помощью случайного леса после добавления двух дополнительных признаков. Модель объясняет 90% изменчивости данных.

На основе полученной информации составим алгоритм подготовки модели случайного леса и дальнейшего прогнозирования потребления электроэнергии с ее помощью. Поскольку различные данные могут давать другие результаты по качеству на рассмотренных во второй главе моделях, будет приведен алгоритм, включающий несколько развилок.

1. Анализ данных и выбор первоначальных параметров.

1.1. Выявить выбросы в данных на основе критерия Ирвина и «ящика с усами». Выбросы, если такие присутствуют, заменить на среднее арифметическое их соседей.

1.2. Построить графики по годам, по месяцам, по дням недели, по дням с целью визуально выявить сезонность и цикличность. Зафиксировать количество наблюдений, при котором сохраняется сезонность и цикличность. Определить горизонт прогнозирования, учитывая данные о сезонности. Рекомендуется брать горизонт не дальше одной недели из-за риска накопления ошибки.

1.3. Проверить данные на стационарность с помощью теста Дикки-Фуллера. Если ряд не стационарен, провести взятие разностей до тех пор, пока гипотеза о наличии единичного корня не будет отвергнута.

1.4. Построить базовые модели прогнозирования: на основе исторического среднего, среднего за последний период времени; по последнему значению; наивный сезонный прогноз. Выбрать за основу ту модель, которая дает минимальную ошибку (если на данных наблюдается ярко выраженная сезонность, достаточно построить только наивный сезонный прогноз).

2. Подбор модели.

2.1. Из имеющихся данных о потреблении электроэнергии отобрать последние несколько месяцев. Провести перебор параметров p, d, q, P, D и Q (в зависимости от стационарности и наличия сезонности подбор некоторых параметров может быть опущен). Перебор p и q рекомендуется проводить до третьего лага; P и Q – до второго.

2.2. Проверить остатки моделей на нормальность, коэффициенты – на значимость. Неподходящие модели исключить из рассмотрения.

2.3. Оставшиеся модели отсортировать по возрастанию значения критерия Акаике. Проверить качество моделей на тестовой выборке. Если статистическая модель дает результаты лучше, чем базовая модель, отложить ее для дальнейшего рассмотрения. В противном случае модель не подходит

для прогноза. Если разница по качеству между моделями несущественная, стоит сделать выбор в пользу базовой из-за времени, требующего на прогноз. Прогноз строится путем добавления спрогнозированного наблюдения к имеющемуся временному ряду, после чего статистическая модель обучатся на основе обновлённого ряда; процесс повторяется, пока не будет получен прогноз нужной длины.

2.4. Вне зависимости от результатов предыдущего пункта построить модель случайного леса с параметрами по умолчанию. Как признак использовать только временной ряд потребления электроэнергии, ширину окна выбрать исходя из визуального анализа сезонности (п.1.2). Рекомендуется строить прогноз по одному наблюдению (п.2.3.), а также на сутки вперед. В этом случае накопление ошибки не происходит. Прогноз на весь горизонт вперед за раз, если он больше нескольких дней, проводить не рекомендуется из-за перегруза модели случайного леса.

2.5. Построить модель случайного леса с другой шириной окна. Выбрать пять-семь вариантов в большую и меньшую сторону от исходной. Если было достигнуто лучшее качество, зафиксировать новое значение окна.

2.6. Добавить новые признаки в модель, используя в качестве ширины окна оптимальное. Рекомендуется использовать:

- Данное о времени года;
- Данные о дне недели;
- Данные о максимальном и минимальном возможном значении потребления за день недели;
- Среднее значение потребление электроэнергии за день.

Эти признаки стоит проверять по отдельности и комбинациями. При этом модель получает как входные данные информацию о признаках за предыдущий период (равный ширине окна), а не той, которая соответствует периоду прогнозирования. Зафиксировать модель (модели) с наименьшей ошибкой.

2.7. Провести оптимизацию параметров случайного леса с отобранными признаками. Если это даст меньшую ошибку, зафиксировать подобранные гиперпараметры.

2.8. Если после пунктов 2.4-2.7 удалось получить модель с ошибкой, меньше чем у базовой и статистической моделей, в качестве основной выбирается случайный лес с подобранными параметрами. В противном случае выбирается та модель, которая имеет наименьшую ошибку.

3. Построение дополнительной модели для повышения качества модели. По возможности стоит построить модель, предсказывающую либо ошибки для основной модели, либо фактические значения на основе прогноза основной модели. Эта модель должна повысить качество прогноза, если основная модель по каким-то причинам не уловила некоторые закономерности в данных. В таком случае выполняются следующие пункты:

3.1. Выбирается стратегия и модель для прогноза, использующая как входные/выходные данные остатки от основной модели. Для нее отдельно (по возможности) проводится подбор оптимальных параметров и ширины окна.

3.2. Корректировка прогноза по основной модели происходит исходя из равенства:

$$e = y_{true} - y_{prediction}$$

3.3. Если модель из пунктов 3.1.-3.2 не дала улучшений или качество модели остается недостаточно хорошим, следует ввести одну-две модели, которая будет прогнозировать отдельный признак на следующий период времени (например, минимальное и максимально возможное значение временного ряда на этот день). Затем следует провести корректировку на основе новой информации.

Данный алгоритм позволил получить на исходных данных ошибку в 9%, что на 3% меньше, чем было получено с помощью наивного сезонного прогноза. Это является довольно хорошим показателем для посуточного прогноза, хотя при дополнительных действиях (например, использования

поиска по сетке для оптимизации всех гиперпараметров случайного леса, а не только количества деревьев) можно добиться еще меньше ошибки.

ЗАКЛЮЧЕНИЕ

Прогнозирование потребления электроэнергии является сложным и важным процессом для промышленных предприятий. Точность прогноза на ближайшее время позволяет не только избежать дополнительных затрат на энергию, но и оптимизировать производственные процессы. Для достижения этой цели необходимо использовать гибкие решения, обеспечивающие точный и интерпретируемый результат в короткие сроки.

В данной работе проводилось сравнение нескольких методов прогнозирования временных рядов с целью использовать наилучший как основу для дальнейшего составления алгоритма прогнозирования потребления электроэнергии промышленным предприятием. Для оценки качества использовались коэффициент детерминации, средняя абсолютная ошибка, замеры время на подбор параметров, обучение, прогноз модели, а также среднеквадратичная и средняя абсолютная ошибки.

Были использованы данные о потреблении электроэнергии лесопромышленным предприятием ООО «Харовсклеспром» по часовым интервалам за период с января 2017 года по декабрь 2020 года. Исходный датасет содержал 35 063 наблюдения и не имел пропущенных значений. Для анализа данных были использованы различные методы визуализации, такие как гистограммы, ящик с усами и линейные графики. Анализ помесячного потребления электроэнергии на примере 2018 года показал, что существует дневная динамика потребления, которая позволила сделать относительно точный прогноз с помощью наивного сезонного прогноза – MAPE 12,69%, R2 0,81.

Использование статистических моделей ARIMA и SARIMA оказалось нецелесообразным на представленных данных из-за низкого качества прогноза и большого количества времени, требующего на обучение модели и построение прогноза. Кроме этого, модели не могу обработать слишком большое количество данных, и почасовой временной ряд приходится

ограничить последними несколькими месяцами, что может привести к потере закономерности.

Многослойный перцептрон и случайный лес без дополнительных надстроек позволяют снизить ошибку до 9-10%. Случайный лес требует значительного времени на обучение (30-40 минут для данных за четыре года), однако прогнозы этой модели в среднем точнее, чем у перцептрона. Дальнейший анализ показал, что случайный лес позволяет снизить ошибку на 1,5% при введении дополнительных переменных, хотя требуется проводить тщательный отбор признаков. При этом новые признаки могут быть основаны исключительно на данный о потреблении, дате и времени, что может быть хорошо для предприятий, которые не могут вести учет других показателей и погоды.

Было показано, что при использовании многослойного перцептрона и случайного леса для временного ряда не требуется проводить стандартизацию и нормализацию – это приведет к значительному ухудшению прогноза.

Важную роль при работе с временным рядом играет ширина окна. Хотя этот параметр можно подобрать по визуальному анализу графика, следует провести дополнительные эксперименты. Так, хотя при анализе данных было предположено, что модели будут давать наилучший результат при окне 24*7, оказалось, что многослойный перцептрон лучше улавливает закономерности на данных за последние четыре дня, а случайный лес – за последние пять.

Кроме введения дополнительных переменных попытаться повысить качество прогноза можно путем добавления моделей, которые будут заточены на прогнозирования проблемных участков основной модели: прогнозировании на основе неучтённых закономерностях в остатках, прогнозировании минимальных и максимальных потреблений в будущих сутках.

В работе приведен алгоритм прогнозирования потребления электроэнергии, который на данных за четыре года ООО «Харовсклеспрома» позволил достичь ошибки в 9%. Получившийся прогноз на семь дней вперед

достаточно точно повторяет динамику дневного потребления, но не всегда верно предсказывает поведение на пике потребления, что приводит к потере точности.

Проведенные в ходе разработки исследования могут послужить отправной точкой для дальнейшего изучения проблемы с целью добиться еще меньшей ошибки. Благодаря своей интерпретируемости случайный лес может позволить оптимизировать процессы на производстве, что наравне с довольно высокой скорости работы модели также является преимуществом алгоритма.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

[1] Абузяров А. А., Макаров А. А. Система прогнозирования потребления электроэнергии на пищевом производстве на основе потоковых данных // Инженерный вестник Дона. - 2023. - №1

[2] Айвазян С. А. Прикладная статистика. Основы эконометрики. Том 1. - 2-е изд. - М.: Юнити-Дана, 2001. - 432 с. — Текст : непосредственный.

[3] Баев, И. С. Прогнозирование промышленного электропотребления в условиях волатильности ценовых сигналов / И. С. Баев, И. А. Соловьева, А. П. Дзюба. — Текст : непосредственный // Экономика региона. — 2012. — № 3. — С. 119-116.

[4] Временные ряды. Простые решения. — Текст : электронный // Хабр : [сайт]. — URL: https://habr.com/ru/articles/553658/ (дата обращения: 29.04.2023).

[5] Григорьева, Д. Р. Методы статистического прогнозирования экономического показателя расхода электроэнергии на предприятии / Д. Р. Григорьева, А. Г. Файзуллина. — Текст : непосредственный // Экономический анализ: теория и практика. — 2015. — № 416. — С. 43-52.

[6] Гужов, С. В. Нейронные сети как инструмент прогнозирования энергопотребления / С. В. Гужов. — Текст : электронный // ЭнергиаВита : [сайт]. — URL: https://energiavita.ru/2020/06/28/nejronnye-seti-kak-instrument-prognozirovaniya-ehnergopotrebleniya/ (дата обращения: 29.04.2023).

[7] Дауб, И. С. Исследование статистических методов прогнозирования временных рядов с трендом и сезонностью / И. С. Дауб. — Текст : непосредственный // StudNet. — 2021. — № 5. — С. .

[8] Ильиных, М. В. Применение методов искусственного интеллекта при прогнозировании энергопотребления (на примере ФГБОУ во «ПГУ им. Шолом-Алейхема») / М. В. Ильиных. — Текст : непосредственный // Вестник Приамурского государственного университета им. Шолом-Алейхема. — 2020. — № 41. — С. 29-41.

[9] История погоды: справочно-информационный портал. – URL: https://weatherarchive.ru/(дата обращения 29.04.2023). – Текст: электронный.

[10] История энергетики России. — Текст : электронный // Свободная Пресса : [сайт]. — URL: https://svpressa.ru/energy/ (дата обращения: 29.04.2023).

[11] Кирилычев, И. А. Преимущества использования искусственных нейронных сетей в прогнозировании энергопотребления и цен на электроэнергию / И. А. Кирилычев. — Текст : непосредственный // Молодой учёный. — 2022. — № 413. — С. 72-73.

[12] Магнус Я. Р., Катышев П. К., Пересецкий А. А. Эконометрика. Начальный курс. - 6-е изд. - М.: Дело, 2007. - 504 с. — Текст : непосредственный.

[13] Матренин, П. В. [Антикейс] Прогнозирование и планирование потребления электроэнергии с помощью machine learning (эксперимент) / П. В. Матренин. — Текст : электронный // Хабр : [сайт]. — URL: https://habr.com/ru/articles/577732/ (дата обращения: 29.04.2023).

[14] Машинное обучение в энергетике, или не только лишь все могут смотреть в завтрашний день. — Текст : электронный // Хабр : [сайт]. — URL: https://habr.com/ru/companies/lanit/articles/487944/ (дата обращения: 29.04.2023).

[15] Мигранов, М. М. Прогнозирование потребления электроэнергии. Практика применения / М. М. Мигранов, А. А. Устинов, А. В. Мельников. — Текст : непосредственный // Электроэнергия. Передача и распределение. — 2018. — № 47. — С. 44-53.

[16] О разработке гибридных нейросетевых моделей в задачах прогнозирования временных рядов // Я - Интеллект URL: http://i-intellect.ru/articles/2365/ (дата обращения 29.04.2023)

[17] Основные характеристики российской электроэнергетики. — Текст : электронный // Министерство Энергетики РФ : [сайт]. — URL: https://minenergo.gov.ru/node/532 (дата обращения: 29.04.2023).

[18] Прогнозирование объемов потребления электроэнергии. — Текст : электронный // StatSoft Russia : [сайт]. — URL: http://statsoft.ru/solutions/ExamplesBase/branches/detail.php?ELEMENT_ID=644 (дата обращения: 29.04.2023).

[19] Прогнозирование потребления электрической энергии промышленным предприятием с помощью методов машинного обучения / А. Д. Моргоева, И. Д. Моргоев, Р. В. Клюев, О. А. Гаврина. — Текст : непосредственный // Известия Томского политехнического университета. — 2022. — № 7. — С. 115-125.

[20] Прогнозирование потребления электроэнергии предприятиями народнохозяйственного комплекса в условиях неполноты информации / И. Д. Моргоев, А. Э. Дзгоев, Р. В. Клюев, А. Д. Моргоева. — Текст : непосредственный // Известия Кабардино-Балкарского научного центра РАН. — 2022. — № 107. — С. 9-20.

[21] Российская электроэнергетика. — Текст : электронный // Ассоциация «НП Совет рынка» : [сайт]. — URL: https://www.np-sr.ru/ru/market/cominfo/rus/index.htm (дата обращения: 29.04.2023).

[22] Сафаралиев М. Х., Матренин П. В., Дмитриев С. А., Ахьеев Д. С., Кокин С.Е. адаптивные ансамблевые модели для среднесрочного прогнозирования выработки электроэнергии гидроэлектростанциями в изолированных энергосистемах с учётом изменений температуры // Электротехнические системы и комплексы. - 2022. - №54. - С. 38-45.

[23] Татаренко, С. И. Методы и модели анализа временных рядов : метод. указания к лаб. работам / С. И. Татаренко. — 1-ое изд. — Тамбов : Изд-во Тамб. гос. техн. ун-та, 2008. — 32 с. — Текст : непосредственный.

[24] Федеральный закон от 26.03.2003 N 35-ФЗ (ред. от 21.11.2022) "Об электроэнергетике" (с изм. и доп., вступ. в силу с 01.01.2023)

[25] Хальясмаа, А. И. Анализ ошибок применения алгоритмов машинного обучения в задачах электроэнергетики / А. И. Хальясмаа, П. В. Матренин, С.

А. Ерошенко. — текст : непосредственный // электроэнергия. Передача и распределение. — 2021. — № 66. — С. 46-53.

[26] «Харовсклеспром» – кардинальное обновление. — Текст : электронный // Леспроминформ : [сайт]. — URL: https://lesprominform.ru/jarticles.html?id=3563 (дата обращения: 29.04.2023).

[27] Чеканов, И. И. Проблемы и перспективы электроэнергетики российской федерации / И. И. Чеканов. — Текст : непосредственный // Научные известия. — 2022. — № 26. — С. 235-240.

[28] Ярыгина, Е. А. Разработка методики краткосрочного прогнозирования электропотребления системы собственных нужд тэц : специальность 05.14.02 «Электрические станции и электроэнергетические системы» : диссертация на соискание ученой степени кандидата технических наук / Ярыгина Екатерина Александровна ; Самарский государственный технический университет. — Самара, 2021. — 125 с. — Текст : непосредственный.

[29] Benchmarking of Regression Algorithms and Time Series Analysis Techniques for Sales Forecasting / C. Catal, K. Ece, B. Arslan, A. Akbulut. — Текст : непосредственный // Balkan Journal of Electrical and Computer Engineering. — 2019. — № 7. — С. 20-26.

[30] Energy-saving potential prediction models for large-scale building: a state-of-the-art review / Xiu'e Yang, Shuli Liu, Yuliang Zou, Wenjie Ji, Qunli Zhang, Abdullahi Ahmed, Xiaojing Han, Yongliang Shen, Shaoliang Zhang // Renewable and Sustainable Energy Reviews. - 2022. -V. 156. - 111992.

[31] Estimation of energy consumption in machine learning / E. García-Martín, C. Faviola Rodrigues, G. Riley, H. Grahn // Journal of Parallel and Distributed Computing. - 2019. - V. 134. - P. 75-88.

[32] Forecasting of electricity consumption by industrial enterprises with a continuous nature of production / I,U Rakhmonov. — Текст : электронный // E3S Web of Conferences : [сайт]. — URL: https://www.e3s-conferences.org/articles/e3sconf/pdf/2023/21/e3sconf_rses2023_01030.pdf (дата обращения: 29.04.2023).

[33] Peixeiro M. Time Series Forecasting in Python. - 1-е изд. - M.: Manning, 2022. - 456 с. — Текст : непосредственный.

[34] Performance Comparison of Simple Regression, Random Forest and XGBoost Algorithms for Forecasting Electricity Demand // ReseacrhGate URL: https://www.researchgate.net/publication/366693600_Performance_Comparison_of_Simple_Regression_Random_Forest_and_XGBoost_Algorithms_for_Forecasting_Electricity_Demand (дата обращения: 29.04.2023).

[35] Random Forests. — Текст : электронный // Berkeley Statistics : [сайт]. — URL: https://www.stat.berkeley.edu/~breiman/RandomForests/cc_home.htm (дата обращения: 29.04.2023).

[36] Random forest Algorithm in Machine learning: An Overview. — Текст : электронный // Great Learning : [сайт]. — URL: https://www.mygreatlearning.com/blog/random-forest-algorithm/ (дата обращения: 29.04.2023).

[37] Random forests model for one day ahead load forecasting // ReseacrhGate URL: https://www.researchgate.net/publication/280555451_Random_forests_model_for_one_day_ahead_load_forecasting (дата обращения: 29.04.2023).

[38] Top 5 advantages and disadvantages of Decision Tree Algorithm. — Текст : электронный // Medium : [сайт]. — URL: https://dhirajkumarblog.medium.com/top-5-advantages-and-disadvantages-of-decision-tree-algorithm-428ebd199d9a (дата обращения: 29.04.2023).

[39] When to Use MLP, CNN, and RNN Neural Networks. — Текст : электронный // Machine Learning Mastery : [сайт]. — URL: https://machinelearningmastery.com/when-to-use-mlp-cnn-and-rnn-neural-networks/ (дата обращения: 29.04.2023).

[40] Why you should try Mean Encoding. — Текст : электронный // Towards Data Science : [сайт]. — URL: https://towardsdatascience.com/why-you-should-try-mean-encoding-17057262cd0 (дата обращения: 29.04.2023).

ПРИЛОЖЕНИЯ

Приложение 1. Результаты замеров по всем моделям

Model	MAPE	MSE	MAE	R2	Time
Hist_mean	35,67868336	118234,0062	290,8502088	-0,018467622	0,001999855
Last_day_mean	38,1466319	117753,7546	294,3363955	-0,014330735	0,001999378
Last	33,18510173	173832,0325	331,5403647	-0,49738897	0,000999689
Naive_seasonal	12,69362135	22272,01889	114,5898357	0,80814885	0,002000332
					0,000116654

Таблица 1. Базовые модели

Model	MAPE	MSE	MAE	R2	Time_prediction
ARIMA(3, 0, 1)	23,59703962	77276,4685	216,8191133	0,334340571	47,85965967
ARIMA(1, 0, 2)	31,82906268	107657,0687	272,5478302	0,072642109	12,89308596
ARIMA(3, 0, 0)	31,84904728	107607,7134	272,597944	0,073067255	7,7270751
ARIMA(2, 0, 1)	31,79695619	108910,9358	273,5660889	0,061841298	32,29109287
ARIMA(2, 0, 0)	31,81154188	109425,4241	274,0239069	0,057409497	4,604589224
ARIMA(1, 0, 1)	31,8789477	109249,2706	274,0772277	0,058926883	6,702898026
ARIMA(0, 0, 3)	33,29562636	110697,8329	279,0728207	0,046448969	24,7148459
				Подбор:	19,81971788
					2,61

Таблица 2. ARMA()

Model	MAPE	MSE	MAE	R2	Time_prediction
ARIMA(2, 0, 3)	12,91792506	20369,17742	113,4014545	0,824539924	442,0308323
ARIMA(3, 0, 2)	13,11770557	21650,41413	116,4069753	0,813503352	537,7750802
ARIMA(3, 0, 0)	14,56792941	24251,31947	122,7254214	0,791099156	273,5559802
ARIMA(1, 0, 2)	14,50294538	24083,35239	122,2421423	0,792546025	240,5669796
ARIMA(1, 0, 1)	14,45552022	23970,00633	121,8958566	0,793522387	179,8323262
ARIMA(3, 0, 1)	13,52137558	25426,01548	121,9043879	0,780980326	460,9819047
ARIMA(2, 0, 0)	14,6491772	24335,04063	123,2212008	0,790377982	175,9671001
ARIMA(0, 0, 3)	14,25270753	23472,44611	120,5627623	0,797808371	157,0202637
ARIMA(0, 0, 2)	14,07017029	23011,56717	119,3829993	0,801778381	122,7062085
ARIMA(1, 0, 0)	14,07348453	23044,32604	119,4306317	0,801496197	115,8200803
ARIMA(0, 0, 1)	13,76378564	22394,00127	117,5591176	0,807098094	97,77898288
				Подбор:	196,1462028
					50,00

Таблица 3. ARIMA()(P=1,0,Q=0)[24]

Model	MAPE	MSE	MAE	R2	Time_prediction
ARIMA(1, 0, 3)	24,47708275	62623,49535	205,9719559	0,460561268	306,7124178
ARIMA(2, 0, 3)	24,46517427	62726,3148	206,0376281	0,459675582	385,0982864
ARIMA(1, 0, 2)	24,53127612	62951,88322	206,5503366	0,457732535	186,8653941
ARIMA(3, 0, 1)	21,18940854	56924,02845	188,9247627	0,509656471	257,5581648
ARIMA(2, 0, 0)	24,48836011	62990,83763	206,4754198	0,457396982	123,8249459
ARIMA(1, 0, 1)	24,48336931	62997,45728	206,460302	0,45733996	138,4786422
ARIMA(0, 0, 3)	25,29383752	63587,59109	209,3071614	0,452256548	162,3094065
				Подбор:	154,6964066
					28,59

Таблица 4. ARIMA()(P=0,0,Q=1)[24]

Model	MAPE	MSE	MAE	R2	Time_prediction
ARIMA(0, 0, 3)	413,883293	697824114,3	3313,721386	-6010,056283	621,921351
ARIMA(0, 0, 2)	136,8730697	38948170,39	995,6906243	-334,4995042	495,1172032
ARIMA(1, 0, 3)	281,7248593	51732171,25	1602,652225	-444,620876	1191,542863
ARIMA(1, 0, 0)	506,8872191	915961948,1	3978,035455	-7889,095385	311,763639
ARIMA(3, 0, 1)	1447,897926	17312153407	17225,85478	-149125,8736	1064,425838
				Подбор:	343,0394654
					67,13

Таблица 5. ARIMA()(P=1,0,Q=1)[24]

Model	Tuner	Params	MAPE	MSE	MAE	R2	Training_time	Tuner_time	Time_prediction
MLPRegressor (Наблюдение за раз)	ЛОЖЬ	{'hidden_layer_sizes': (100,), 'activation': 'relu'}	9,823182302	13348,05242	86,1892611	0,885019889	48,9951055		0,068312883
MLPRegressor (Сутки за раз)	ЛОЖЬ	{'hidden_layer_sizes': (100,), 'activation': 'relu'}	9,895108716	14041,86014	89,68512283	0,87904343	131,0618689		0,003998518
MLPRegressor (Неделя за раз)	ЛОЖЬ	{'hidden_layer_sizes': (100,), 'activation': 'relu'}	10,27324084	14414,15707	91,97485348	0,875836464	188,5462167		0
MLPRegressor (Наблюдение за раз)	ИСТИНА	{'activation': 'relu'}	10,46379673	14497,84201	92,97661431	0,875115602	17,18419576	1979,647132	0,031261683
MLPRegressor (Сутки за раз)	ИСТИНА	{'activation': 'relu', 'max_iter': 300}	9,792794426	13672,46722	86,68338739	0,88222538	78,37569332	2591,394983	0,020330906
MLPRegressor (стандартизация) (Наблюдение за раз)	ЛОЖЬ	{'hidden_layer_sizes': (100,), 'activation': 'relu'}	49966,43452	1,93961E+11	428689,42	-1670775,964	90,5186305		0,064005136
MLPRegressor (стандартизация) (Сутки за раз)	ЛОЖЬ	{'hidden_layer_sizes': (100,), 'activation': 'relu'}	39546,59752	1,34547E+11	352281,9477	-1158990,616	34,93623185		0,004002094
MLPRegressor (нормализация) (Наблюдение за раз)	ЛОЖЬ	{'hidden_layer_sizes': (100,), 'activation': 'relu'}	118919,0535	1,57588E+12	1131463,892	-13574590,54	6,762070179		0,054005384
MLPRegressor (нормализация) (Наблюдение за раз)	ЛОЖЬ	{'hidden_layer_sizes': (100,), 'activation': 'sigmoid'}	256225,4215	5,60631E+12	2280563,971	-48292774,85	7,267928362		0,043004513
MLPRegressor (стандартизация) (Сутки за раз)	ЛОЖЬ	{'hidden_layer_sizes': (100,), 'activation': 'sigmoid'}	31722,80191	98965585879	293194,0457	-852488,4664	27,8774991		0
							631,5254402	**4571,042114**	**0,288921118**
									86,7142746

Таблица 6. Многослойный перцептрон

Model	Tuner	Params	MAPE	MSE	MAE	R2	Training_time	Tuner_time	Time_prediction
DecisionTree (Наблюдение за раз)	ЛОЖЬ	{'max_depth': None, 'max_leaf_nodes': None, 'min_samples_leaf': 1, 'min_samples_split': 2}	14,047	28393,767	127,460	0,755	20,022		0,022
DecisionTree (Сутки за раз)	ЛОЖЬ	{'max_depth': None, 'max_leaf_nodes': None, 'min_samples_leaf': 1, 'min_samples_split': 2}	16,128	43231,052	150,976	0,628	28,373		0,002
DecisionTree (Неделя за раз)	ЛОЖЬ	{'max_depth': None, 'max_leaf_nodes': None, 'min_samples_leaf': 1, 'min_samples_split': 2}	12,972	24103,124	112,162	0,792	170,406		0,001
RandomForest (Наблюдение за раз)	ЛОЖЬ	{'n_estimators': 100, 'max_depth': None, 'max_leaf_nodes': None, 'min_samples_leaf': 1, 'min_samples_split': 2}	9,299	12282,000	81,881	0,894	962,176		1,390
RandomForest (Сутки за раз)	ЛОЖЬ	{'n_estimators': 100, 'max_depth': None, 'max_leaf_nodes': None, 'min_samples_leaf': 1, 'min_samples_split': 2}	10,469	17413,969	96,522	0,850	1976,314		0,171
DecisionTree (стандартизация) (Наблюдение за раз)	ЛОЖЬ	{'max_depth': None, 'max_leaf_nodes': None, 'min_samples_leaf': 1, 'min_samples_split': 2}	50,282	161755,224	334,301	-0,393	13,456		0,027
DecisionTree (стандартизация) (Сутки за раз)	ЛОЖЬ	{'max_depth': None, 'max_leaf_nodes': None, 'min_samples_leaf': 1, 'min_samples_split': 2}	76,123	398747,332	495,601	-2,435	46,887		0,000
DecisionTree (стандартизация) (Неделя за раз)	ЛОЖЬ	{'max_depth': None, 'max_leaf_nodes': None, 'min_samples_leaf': 1, 'min_samples_split': 2}	64,760	298990,684	451,624	-1,576	202,951		0,000
DecisionTree (нормализация) (Наблюдение за раз)	ЛОЖЬ	{'max_depth': None, 'max_leaf_nodes': None, 'min_samples_leaf': 1, 'min_samples_split': 2}	53,447	170566,830	352,439	-0,469	15,797		0,025
DecisionTree(Наблюдение за раз)	ИСТИНА	{'max_depth': 7}	10,319	15254,389	92,061	0,869	5,336	447,067	0,206
DecisionTree(Сутки за раз)	ИСТИНА	{'max_depth': 8}	10,553	16959,824	93,688	0,854	19,741	639,921	0,008
RandomForest(Наблюдение за раз)	ИСТИНА	{'n_estimators': 150}	9,312	12257,439	82,477	0,894	887,436	1201,100	2,132
RandomForest(Сутки за раз)	ИСТИНА	{'n_estimators': 110}	10,756	17896,517	99,554	0,846	1435,127	2942,830	0,201
							4603,046	**5230,918**	**2,770**
									163,946

Таблица 7. Деревья решений и случайный лес