

Прогнозирование временных рядов

Домашнее задание

1. Общие требования

Работа выполняется индивидуально. Необходимо сдать краткий отчет с основными графиками, таблицей сравнения и выводами, а также реплицируемый код. Данные и шаблон проекта предоставляются вместе с заданием.

2. Как будем сравнивать модели

Чтобы сравнение не зависело от одного момента истории, проверим модели в двух режимах рынка. Для спокойного периода используйте январь, март, май, июль, сентябрь и ноябрь 2017 г.; для кризисного — те же месяцы 2020 г.

В каждую из этих дат постройте прогноз Brent на 1–12 месяцев вперед, используя только информацию, доступную на тот момент. Обучающая выборка каждый раз включает всю предыдущую историю. RMSE и MAE считайте в уровнях — в целом, по горизонтам и отдельно для 2017 и 2020 гг. Если модель прогнозирует $\log$ или $\Delta \log$, сначала восстановите уровень цены. Будущие значения внешних признаков использовать нельзя.

3. Задание

1. Начните с простого прогноза.

Постройте наивный прогноз: цена на всех горизонтах равна последнему известному значению. Проведите оценку качества, посчитайте RMSE и MAE и посмотрите, как ошибка меняется с увеличением горизонта.

★ Добавьте наивный прогноз с дрейфом. Помогает ли учет среднего исторического изменения цены и на каких горизонтах?

2. Подготовьте ряд для классической модели.

До оценки SARIMAX посмотрите, что именно разумнее моделировать: уровень Brent, $\log(\text{Brent})$ или $\Delta \log(\text{Brent})$. Сравните графики и свойства рядов. Объясните, зачем классической модели могут понадобиться логарифмирование и разности — например, из-за тренда, меняющейся дисперсии или нестационарности. Для преобразованных таргетов прогноз перед расчетом метрик необходимо восстановить обратно в уровень цены.

★ Проверьте стационарность исходного и преобразованных рядов одним из стандартных тестов. Совпадает ли формальный результат с тем, что видно на графиках?

3. Постройте SARIMAX.

Выберите представление ряда из предыдущего пункта, оцените SARIMAX и проведите оценку качества. Сравните результат с наивным прогнозом: стало ли точнее и на каких горизонтах?

★ Сравните несколько спецификаций SARIMAX или добавьте STL. Насколько вывод зависит от конкретной классической модели и ее параметров?

4. Перейдите к бустингу.

Постройте CatBoost сначала только на лагах Brent, затем добавьте скользящие средние, momentum-признаки, волатильность и т.д. После этого вернитесь к вопросу преобразований: сравните прогноз уровня цены с прогнозом $\log$ или $\Delta \log$. Какой получается вывод?

★ Сравните CatBoost с LightGBM или XGBoost на одном и том же наборе признаков. Что сильнее влияет на качество: алгоритм, признаки или представление целевой переменной?

5. Попробуйте нейросетевую модель.

Постройте одну модель из `NeuralForecast` и выполните тот же набор действий, что раньше. Затем запустите эту же архитектуру на одном из готовых датасетов библиотеки. Одинаково ли хорошо архитектура работает на разных данных?

★ Сравните две архитектуры, например N-BEATS и N-HiTS или LSTM и GRU. Есть ли у более сложной модели устойчивое преимущество?

6. Расширьте набор данных.

Добавьте к лучшей ML модели внешние признаки: другие сырьевые цены, макроэкономические или финансовые показатели. Используйте только значения, доступные на момент времени, избегая лика. Сравните результат с моделью, использующей только историю Brent. Всегда ли больше данных означает лучший прогноз?

★ Добавляйте группы признаков по отдельности — сырьевые, макроэкономические и финансовые. Какие данные дают наибольший прирост и на каких горизонтах?

7. Сопоставьте результаты.

Сведите все модели в одну таблицу и постройте RMSE в зависимости от горизонта. Сначала сравните модели по всем прогнозам, затем отдельно для 2017 и 2020 гг. Меняется ли лидер между спокойным и кризисным режимами? Какие модели лучше переживают резкие изменения рынка, а какие выигрывают в основном в стабильный период?

★ Проверьте устойчивость и качество моделей при помощи статистических тестов.

8. ★ Сделайте эксперимент воспроизводимым.

Соберите проведенные эксперименты в один проект:

- разделите загрузку данных, построение признаков, обучение моделей, оценку качества и расчет метрик;
- вынесите в конфигурацию модель, преобразование целевого ряда, набор признаков и т.д.;
- зафиксируйте зависимости и `random seed`;
- подготовьте `Dockerfile` и запуск проекта одной командой;
- автоматически сохраняйте в `artifacts/` прогнозы, метрики и конфигурацию конкретного запуска;
- добавьте в `README` короткую инструкцию, как полностью воспроизвести эксперимент.