Домашнее задание: Анализ временных рядов

Стационарность и стратегии прогнозирования

Курс: Data Science

1 Общая информация

Важно

- **Срок сдачи:** 2 недели с момента выдачи
- **Обе гипотезы обязательны** для выполнения
- Работа выполняется **индивидуально**

2 Цель работы

Исследовать влияние стационарности временного ряда и стратегии прогнозирования на качество предсказаний финансовых данных. Вы проверите две ключевые гипотезы:

1. **Гипотеза 1:** Приведение ряда к стационарному виду улучшает качество прогнозов
2. **Гипотеза 2:** Прогнозирование с периодическим переобучением превосходит прямое прогнозирование на h шагов

3 Формат сдачи

3.1 GitHub репозиторий

Создайте **отдельный публичный репозиторий** для данного домашнего задания.

3.2 Отчет (обязательно)

Отчет должен содержать:

1. **Введение:** описание выбранного актива и его характеристик
2. **Exploratory Data Analysis:**
   - Визуализация исходного ряда
   - Декомпозиция (тренд, сезонность, остатки)
   - ACF/PACF графики
   - Результаты тестов стационарности (ADF, KPSS)
3. **Гипотеза 1:** методология, эксперименты, результаты

4. **Гипотеза 2:** методология, эксперименты, результаты
5. **Сравнительный анализ:** таблицы метрик, графики
6. **Выводы:** статистическая значимость результатов, практические рекомендации

Формат: Jupyter Notebook (.ipynb) + экспортированный PDF

4 Выбор данных

4.1 Рекомендуемые источники

Где взять данные

Опция 1: Библиотека yfinance (рекомендуется)
Загрузите данные по акции за 5 лет. Используйте столбец `Close` для анализа.
Опция 2: Kaggle Datasets

- Stock Market Dataset (исторические данные S&P 500)
- Cryptocurrency Historical Prices

Опция 3: pandas-datareader
Импортируйте данные из Yahoo Finance через pandas-datareader.

4.2 Требования к данным

- **Период:** минимум 3 года (желательно 5 лет)
- **Частота:** дневные данные (daily close price)
- **Тип актива:** акции крупной компании или популярная криптовалюта
- **Отсутствие пропусков:** заполнить методом forward fill

Рекомендуемые активы:

- Акции: AAPL, MSFT, GOOGL, TSLA, AMZN
- Криптовалюты: BTC-USD, ETH-USD
- Индексы: ^GSPC (S&P 500), ^DJI (Dow Jones)

5 Гипотеза 1: Влияние стационарности

Гипотеза 1

Утверждение: Модели, обученные на стационарном ряде, дают более точные прогнозы, чем модели на исходном нестационарном ряде.
Что проверяем: Сравниваем качество прогнозов на двух версиях данных:

- Baseline: исходный нестационарный ряд
- Transformed: приведенный к стационарности ряд (с последующим восстановлением)

5.1 Методология

5.1.1 Этап 1: Проверка стационарности исходного ряда

1. **Визуальный анализ:**
   - Постройте график цен во времени
   - Проведите декомпозицию (аддитивную или мультипликативную)
   - Постройте ACF/PACF графики
2. **Статистические тесты:**
   - Augmented Dickey-Fuller (ADF) тест
   - KPSS тест
   - Интерпретация: совместное использование обоих тестов

5.1.2 Этап 2: Приведение к стационарности

Последовательность преобразований:

1. **Логарифмирование** (стабилизация дисперсии):
$$y_t^{(\log)} = \log(y_t)$$
2. **Первая разность** (удаление тренда):
$$y_t^{(\text{diff})} = y_t^{(\log)} - y_{t-1}^{(\log)} = \log(y_t/y_{t-1})$$
Это соответствует **логарифмической доходности** (log-returns)!
3. **Проверка стационарности преобразованного ряда:**
   - Повторно примените ADF и KPSS тесты
   - Убедитесь, что p-value ADF < 0.05 и p-value KPSS > 0.05

Важно: работа с NaN

После взятия разности первое наблюдение будет NaN. Удалите его перед дальнейшим анализом.

5.1.3 Этап 3: Обучение моделей

Разделение данных:

- Train: первые 80% данных
- Test: последние 20% данных

Модели для сравнения (все 3 обязательны):

1. **Simple Exponential Smoothing (SES):**
   - Для нестационарного: применяйте напрямую к ценам
   - Для стационарного: применяйте к логарифмическим доходностям
   - Формула: $\hat{y}_{t+1|t} = \alpha y_t + (1-\alpha)\hat{y}_{t|t-1}$
2. **Метод Хольта (Holt's Linear Trend):**
   - Для нестационарного: `trend='add'`, `seasonal=None`
   - Для стационарного: возможно использование без тренда
   - Учитывает линейный тренд в данных
3. **Holt-Winters:**
   - Для нестационарного: `trend='add'`, `seasonal=None`
   - Для стационарного: настройте параметры сглаживания
   - Используйте аддитивную или мультипликативную модель в зависимости от данных

Naive Forecast как baseline

Дополнительно реализуйте Naive Forecast в качестве простейшего baseline:

$$\hat{y}_{t+1} = y_t$$

Это поможет оценить, насколько более сложные модели превосходят тривиальное решение.

5.1.4 Этап 4: Прогнозирование и восстановление

Для нестационарного ряда:

- Прогноз напрямую в исходных единицах (цены)

Для стационарного ряда (критически важно!):

1. **Прогноз в пространстве логарифмических доходностей:**

$$\hat{r}_{t+1}, \hat{r}_{t+2}, \ldots, \hat{r}_{t+h}$$

где $r_t = \log(y_t/y_{t-1})$

2. **Восстановление логарифмических цен:**

$$\log(\hat{y}_{t+1}) = \log(y_t) + \hat{r}_{t+1}$$

$$\log(\hat{y}_{t+2}) = \log(\hat{y}_{t+1}) + \hat{r}_{t+2}$$

$$\vdots$$

3. **Восстановление исходных цен:**

$$\hat{y}_{t+h} = \exp(\log(\hat{y}_{t+h}))$$

ОБЯЗАТЕЛЬНО: восстановление ряда

Сравнение моделей должно происходить в **исходных единицах** (цены в рублях/долларах), а не в логарифмических доходностях!
Используйте функцию кумулятивной суммы для восстановления логарифмических цен, затем примените экспоненту.

5.1.5 Этап 5: Оценка качества

Метрики (вычисляйте на тестовой выборке):

1. **MAE** (Mean Absolute Error):

$$\text{MAE} = \frac{1}{n}\sum_{t=1}^{n}|y_t - \hat{y}_t|$$

2. **RMSE** (Root Mean Squared Error):

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{t=1}^{n}(y_t - \hat{y}_t)^2}$$

3. **MAPE** (Mean Absolute Percentage Error):

$$\text{MAPE} = \frac{100\%}{n}\sum_{t=1}^{n}\left|\frac{y_t - \hat{y}_t}{y_t}\right|$$

4. **Directional Accuracy** (специфично для финансов):

$$\text{DA} = \frac{1}{n}\sum_{t=1}^{n}\mathbb{1}[\text{sign}(y_t - y_{t-1}) = \text{sign}(\hat{y}_t - y_{t-1})]$$

Процент правильно предсказанного направления движения цены.

5.2 Ожидаемые результаты Гипотезы 1

1. **Таблица сравнения:**

Модель	**Стационарность**	**MAE**	**RMSE**	**MAPE**
Naive	Нет	...	...	...
SES	Нет	...	...	...
SES	Да (преобразованный)	...	...	...
Holt	Нет	...	...	...
Holt	Да (преобразованный)	...	...	...
Holt-Winters	Нет	...	...	...
Holt-Winters	Да (преобразованный)	...	...	...

2. **Графики:**

   - Сравнение прогнозов на тестовой выборке (наложенные линии)
   - График ошибок во времени
   - Диаграмма сравнения метрик (bar plot)

3. **Статистический тест:**

   - Diebold-Mariano тест для сравнения точности прогнозов
   - Вывод о статистической значимости различий

6 Гипотеза 2: Стратегии прогнозирования

Гипотеза 2

Утверждение: Прогнозирование с периодическим переобучением модели превосходит прямое прогнозирование без обновления.
Что проверяем: Для фиксированного горизонта прогнозирования (весь тестовый период) сравниваем прямой прогноз vs прогнозы с переобучением каждые $k \in \{1, 2, 5, 10\}$ дней.

6.1 Две стратегии прогнозирования

Общая постановка:

- Тестовая выборка: последние 20% данных (например, 365 дней для 5-летнего ряда)
- Задача: спрогнозировать весь тестовый период
- Вопрос: как часто нужно переобучать модель?

6.1.1 Стратегия A: Прямое прогнозирование (Direct Forecast)

Описание: Обучаем модель один раз на train, делаем прогноз сразу на весь тестовый период.

Алгоритм:

1. Обучить модель на всех train данных
2. Сделать прогноз на весь test: $\hat{y}_{T+1}, \hat{y}_{T+2}, \ldots, \hat{y}_{T+N_{\text{test}}}$
3. Оценить качество на всём тестовом периоде

Преимущества:

- Очень быстро (одно обучение)
- Простая реализация
- Не требует новых данных в процессе прогнозирования

Недостатки:

- Модель не обновляется с новыми данными
- Ошибки накапливаются со временем
- Не учитывает новую информацию

6.1.2 Стратегия B: Rolling Forecast с периодом переобучения k

Описание: Делаем прогноз на k шагов, затем добавляем истинные значения за эти k дней в train и переобучаем модель.

Алгоритм для периода переобучения k:

1. Обучить модель на текущих train данных
2. Сделать прогноз на следующие k дней: $\hat{y}_{t+1}, \ldots, \hat{y}_{t+k}$

3. Получить истинные значения $y_{t+1}, \ldots, y_{t+k}$
4. Добавить эти k значений в train
5. Переобучить модель
6. Повторить до конца тестового периода

Особые случаи:

- $k = 1$: переобучаем модель **каждый день** (максимальная адаптация)
- $k = 2$: переобучаем **каждые 2 дня**
- $k = 5$: переобучаем **каждые 5 дней** (раз в неделю)
- $k = 10$: переобучаем **каждые 10 дней**

Преимущества:

- Модель адаптируется к новым данным
- Уменьшается накопление ошибок
- Более реалистично для практики

Недостатки:

- Медленно (много переобучений)
- Высокие вычислительные затраты
- Требует доступа к истинным значениям для переобучения

6.2 План экспериментов

Периоды переобучения: $k \in \{1, 2, 5, 10\}$ дней
Для каждого периода k:

1. **Стратегия A (Direct):**
   - Обучить модель один раз
   - Прогноз на весь тестовый период (365 дней)
   - Вычислить метрики и время выполнения
2. **Стратегия B (Rolling с периодом k):**
   - Прогноз на k дней
   - Добавить истинные k значений в train
   - Переобучить модель
   - Повторить до конца тестового периода
   - Вычислить метрики и время выполнения
3. **Сравнение:**
   - MAE, RMSE, MAPE, DA на всём тестовом периоде

- Время выполнения
- График ошибок во времени

Итого: 1 (Direct) + 4 (Rolling с $k \in \{1, 2, 5, 10\}$) = 5 экспериментов

Важно: корректное сравнение

Все стратегии прогнозируют один и тот же тестовый период!
Пример: Если test = 365 дней:

- **Direct**: 1 обучение, прогноз на 365 дней
- **Rolling** $k = 1$: 365 обучений, каждый раз прогноз на 1 день
- **Rolling** $k = 2$: ∼183 обучения, каждый раз прогноз на 2 дня
- **Rolling** $k = 5$: ∼73 обучения, каждый раз прогноз на 5 дней
- **Rolling** $k = 10$: ∼37 обучений, каждый раз прогноз на 10 дней

Все прогнозы оцениваются на одних и тех же 365 днях!

6.3 Дополнительный анализ

1. **Зависимость качества от частоты переобучения:**
   - Постройте график MAE vs период переобучения k (включая Direct)
   - При каком k качество приближается к Direct?
   - Есть ли точка убывающей отдачи (diminishing returns)?
2. **Вычислительная сложность:**
   - Измерьте время выполнения каждой стратегии
   - Постройте график: время vs период переобучения k
   - Вычислите соотношение: улучшение качества / увеличение времени
3. **Анализ ошибок во времени:**
   - Постройте график абсолютной ошибки по дням для всех стратегий
   - Где Direct ошибается сильнее всего?
   - Помогает ли переобучение в моменты резких изменений рынка?
4. **Практические рекомендации:**
   - Для какого периода переобучение оправдано с точки зрения trade-off?
   - Какой период k оптимален для практического применения?
   - Зависит ли оптимальный k от выбора модели (SES vs Holt vs HW)?

6.4 Ожидаемые результаты Гипотезы 2

1. **Таблица сравнения:**

Стратегия	**Период k**	**MAE**	**RMSE**	**DA (%)**	**Время (с)**
Direct	—	...	...	...	...
Rolling	$k = 1$	...	...	...	...
Rolling	$k = 2$	...	...	...	...
Rolling	$k = 5$	...	...	...	...
Rolling	$k = 10$	...	...	...	...

Примечание: DA = Directional Accuracy, процент правильно предсказанного направления

2. **Графики:**
   - Прогнозы всех стратегий на одном графике (тестовый период)
   - MAE vs период переобучения (линейный график с Direct как горизонтальной линией)
   - Время выполнения vs период переобучения (bar plot)
   - Абсолютная ошибка по дням для каждой стратегии (time series plot)
   - Trade-off график: улучшение MAE vs увеличение времени
3. **Статистический анализ:**
   - При каком k различия с Direct становятся значимыми?
   - Оптимальный период переобучения с точки зрения соотношения качество/время
   - Есть ли статистически значимая разница между $k = 1$ и $k = 2$?
4. **Дополнительно (для каждой модели):**
   - Постройте отдельные таблицы для SES, Holt и Holt-Winters
   - Зависит ли оптимальный k от сложности модели?

7 Интеграция гипотез (бонус)

Дополнительное исследование (необязательно, +20% к оценке)

Объедините обе гипотезы: для каждого периода переобучения k проверьте влияние стационарности

Стационарность	**Стратегия**	**Период** k
Нет	Direct	—
Да	Direct	—
Нет	Rolling	$k = 1$
Да	Rolling	$k = 1$
Нет	Rolling	$k = 2$
Да	Rolling	$k = 2$
Нет	Rolling	$k = 5$
Да	Rolling	$k = 5$
Нет	Rolling	$k = 10$
Да	Rolling	$k = 10$

Вопросы для исследования:

- Какая комбинация даёт лучший результат?
- Есть ли синергия между приведением к стационарности и частым переобучением?
- При каком k эффект стационарности становится незначимым?
- Зависят ли результаты от выбора модели (SES vs Holt vs Holt-Winters)?
- Стоит ли комбинировать оба подхода или достаточно одного?

Дополнительные графики:

- Heatmap: MAE для всех комбинаций (стационарность $\times$ период k)
- График: улучшение от стационарности в зависимости от k

8 Рекомендуемые библиотеки

8.1 Основные библиотеки

Создайте файл `requirements.txt` со следующими зависимостями:

- `numpy` — численные вычисления
- `pandas` — работа с данными
- `matplotlib, seaborn` — визуализация
- `scipy` — статистические функции
- `statsmodels` — модели временных рядов (SES, Holt, Holt-Winters)
- `yfinance` — загрузка финансовых данных
- `scikit-learn` — дополнительные метрики
- `tqdm` — прогресс-бары для циклов

8.2 Полезные ссылки

- **Statsmodels documentation:**
  `https://www.statsmodels.org/stable/tsa.html`
- **yfinance:**
  `https://pypi.org/project/yfinance/`
- **Rob Hyndman "Forecasting: Principles and Practice":**
  `https://otexts.com/fpp3/`

9 Чеклист перед сдачей

Проверьте перед отправкой

- ☐ Репозиторий публичный и содержит все файлы
- ☑ README.md с описанием проекта и инструкцией запуска
- ☑ requirements.txt со всеми зависимостями
- ☑ Jupyter Notebook с полным анализом
- ☑ Экспортированный PDF отчета
- ☑ Код запускается без ошибок
- ☑ Все графики с подписями и легендами
- ☑ Проведены тесты стационарности (ADF + KPSS)
- ☑ Реализовано восстановление ряда из преобразованного
- ☑ Проверены все 4 горизонта для Гипотезы 2
- ☑ Таблицы с метриками качества для обеих гипотез
- ☑ Реализованы все 3 модели: SES, Holt, Holt-Winters
- ☑ Статистические выводы с интерпретацией результатов