#### 1. Определение целевой переменной

Целевая переменная создана для двух задач:

* **Регрессия**: Прогнозирование средней цены (mid\_price) BTC-USD через 50 сделок вперед.
  + Параметры: type=price, price\_type=mid\_price, use\_time=False, forward\_periods=50, time\_window=3.0.
* **Классификация**: Прогнозирование направления движения цены (рост или падение).
  + Параметры: type=direction, price\_type=mid\_price, use\_time=False, forward\_periods=50, time\_window=3.0.

Средняя цена (mid\_price) выбрана как основа, так как она отражает рыночное равновесие и менее подвержена шуму по сравнению с последней ценой сделки. Горизонт прогнозирования в 50 сделок

#### 2. Инженерия признаков

Создано 26 признаков :

* **Дисбаланс книги ордеров** (BTC-USD\_book\_imbalance): Отражает давление покупателей/продавцов.
* **Возвраты** (returns\_1, returns\_5, returns\_10, returns\_20): Измеряют ценовой импульс.
* **Скользящие средние** (sma\_5, ema\_5, sma\_10, ema\_10, sma\_20, ema\_20): Улавливают тренды.
* **Волатильность** (volatility\_10, volatility\_20): Оценивает нестабильность рынка.
* **RSI** (rsi\_14): Индикатор перекупленности/перепроданности.
* **MACD** (macd): Сигнализирует о возможных разворотах тренда.
* **Другие**: Относительный спред, агрегированные объемы и лаги дисбаланса.

#### 3. Предобработка данных

Данные разделены на тренировочную (37,500 строк), валидационную (12,500 строк) и тестовую (7,500 строк) выборки. Размеры признакового пространства: 26 признаков.

* **Масштабирование**: Использован RobustScaler для устойчивости к выбросам.
* **Обработка выбросов**: Удалены строки с z-оценкой целевой переменной > 3.0.
* **Разделение**: Временной сплит (TimeSeriesSplit) для избежания look-ahead bias.

**Проверка распределения**:

* Тест Колмогорова-Смирнова показал, что большинство признаков имеют схожие распределения между выборками (p-value > 0.05), за исключением imbalance\_ma\_10 и trade\_flow\_10 (p-value ~0.0507 для Train-Test), что указывает на возможные различия в этих признаках.

#### 4. Обучение и оценка моделей

##### Регрессия (LGBMRegressor)

* **Метрики**:
  + Тренировочная выборка: RMSE = 0.000303, MAE = 0.000221.
  + Валидационная выборка: RMSE = 0.000316, MAE = 0.000238.
  + Тестовая выборка: RMSE = 0.000314, MAE = 0.000236.
* **Анализ**: Низкие значения RMSE и MAE указывают на высокую точность предсказания цены. Модель слегка переобучается (RMSE на тренировочной выборке ниже, чем на валидационной и тестовой), но разница минимальна, что говорит о хорошей обобщающей способности.

##### Классификация (LGBMClassifier, XGBClassifier, Ensemble)

* **Метрики**:
  1. **LightGBM**:
     + Тренировочная: Accuracy = 0.6606, ROC-AUC = 0.7344.
     + Валидационная: Accuracy = 0.5351, ROC-AUC = 0.4962.
     + Тестовая: Accuracy = 0.5297, Precision = 0.4453, Recall = 0.2253, F1 = 0.2992, ROC-AUC = 0.5067.
  2. **XGBoost**:
     + Тренировочная: Accuracy = 0.4350, ROC-AUC = 0.5626.
     + Валидационная: Accuracy = 0.4301, ROC-AUC = 0.4941.
     + Тестовая: Accuracy = 0.4456, Precision = 0.4456, Recall = 1.0000, F1 = 0.6165, ROC-AUC = 0.5023.
  3. **Ансамбль (LightGBM + XGBoost)**:
     + Тренировочная: Accuracy = 0.6015, ROC-AUC = 0.7277.
     + Валидационная: Accuracy = 0.4686, ROC-AUC = 0.4959.
     + Тестовая: Accuracy = 0.4831, Precision = 0.4518, Recall = 0.7499, F1 = 0.5638, ROC-AUC = 0.5060.
* **Анализ**:
  1. **LightGBM**: Показывает умеренную точность на тренировочной выборке, но на валидационной и тестовой выборках метрики близки к случайному угадыванию (ROC-AUC ~0.5). Низкий recall (0.2253) указывает на проблемы с предсказанием положительного класса (рост цены).
  2. **XGBoost**: Высокий recall (1.0) на тестовой выборке, но низкая accuracy и precision, что говорит о смещении модели в сторону предсказания положительного класса. F1-метрика (0.6165) выше, чем у LightGBM, но ROC-AUC остается низким (~0.5).
  3. **Ансамбль**: Улучшает F1-метрику (0.5638) по сравнению с LightGBM, но не превосходит XGBoost. ROC-AUC (~0.5) подтверждает, что модель не способна эффективно разделять классы на валидационной и тестовой выборках.
  4. Переобучение: Все модели показывают значительно лучшие результаты на тренировочной выборке, что указывает на переобучение.

#### 5. Выводы

* **Регрессия**: Модель успешно предсказывает цену с высокой точностью (RMSE и MAE < 0.00032). Это делает её пригодной для задач, где требуется точное прогнозирование цены.
* **Классификация**: Все три модели (LightGBM, XGBoost, ансамбль) демонстрируют низкую предсказательную способность на валидационной и тестовой выборках (ROC-AUC ~0.5). Ансамбль улучшает некоторые метрики (например, F1), но не решает проблему низкой обобщающей способности.
* **Проблемы**:
  + Я пробовала и обучать на больших данных, но результат по классификации плюс минус такой же. Скорее всего, в этих моделях нужны абсолютно другие фичи

#### 6. Рекомендации

1. **Улучшение признаков**:
   * Добавить признаки на основе глубины книги ордеров (уровни 1–5) и еще какие то другие
2. **Балансировка классов**:
   * Проверить эффективность SMOTE и попробовать другие методы (например, ADASYN).
   * Настроить веса классов в LightGBM и XGBoost более тщательно.
3. **Настройка моделей**:
   * Увеличить число итераций в RandomizedSearchCV или использовать Optuna для оптимизации гиперпараметров.
   * Добавить регуляризацию (например, L1/L2 в XGBoost) для борьбы с переобучением.
4. **Ансамбль**:
   * Попробовать взвешенное голосование в VotingClassifier, отдавая больший вес XGBoost, так как он показал лучшую F1-метрику.
   * Добавить в ансамбль другие модели, например, Random Forest или CatBoost.