ML-модель принятия решений для торговли опционами на S&P 500

Гальперин Д. И.
Беркимбаев Р. М.

2025 г.

Опционы Call и Put

- **Call** — право купить актив по цене страйка.
- **Put** — право продать актив по цене страйка.
- Типы опционов: *American* (до экспирации) и *European* (только в дату экспирации).

Индекс S&P 500

- Составной индекс 500 крупнейших компаний США.
- Барометр состояния рынка акций и экономики.

Описание датасета

- **Записи:** 256480 (3 янв 2011 – 15 апр 2024).
- **Опционы:** недельные или с оставшимся временем до экспирации < 1 недели.
- **Поля:**
  - `Date`, `option_symbol`, `dte`, `expiration_date`
  - `call_put`, `price_strike`
  - Цены: `price_open`, `price_high`, `price_low`, `price`, `Bid`, `Ask`
  - `volume`, `openinterest`
  - `iv`, `delta`, `gamma`, `theta`, `vega`, `rho`
  - `underlying_price`

Постановка задачи

- Это задача **многоклассовой классификации** с учителем.
- Каждому опциону присваивается метка в зависимости от его доходности за всё время до экспирации:

$$\text{label} = \begin{cases} -1, & \text{если доходность } \leq -0.21, \\ +1, & \text{если доходность } \geq +0.07, \\ 0, & \text{иначе} \end{cases}$$

Скользящие статистики (5 дней)

$$\begin{aligned}
\mathrm{ret}_t &= \frac{P_t - P_{t-1}}{P_{t-1}}, \\
\mathrm{volatility}_{5d,t} &= \mathrm{std}(\mathrm{ret}_{t-4}, \ldots, \mathrm{ret}_t), \\
\mathrm{return}_{5d,t} &= \frac{P_t - P_{t-5}}{P_{t-5}}, \\
\mathrm{vol_to_ret_ratio}_t &= \frac{\mathrm{volatility}_{5d,t}}{|\mathrm{return}_{5d,t}| + \varepsilon}.
\end{aligned}$$

Динамика греков

- Формула: $g_t^{\text{chg}} = g_t - g_{t-1}$
- Где $g \in \{\delta, \gamma, \text{vega}, \theta, \rho\}$:
  - δ — чувствительность к цене базового актива
  - γ — чувствительность δ к цене актива
  - vega — чувствительность к волатильности
  - θ — временной распад
  - ρ — чувствительность к процентной ставке

Индикатор «скачка»

$$\text{jump\_flag}_t = \begin{cases} 1, & \text{если } |\text{ret}_t| > 2 \cdot \text{std}(\text{ret}_{t-19}, \ldots, \text{ret}_t), \\ 0, & \text{иначе} \end{cases}$$

ATR-подобные признаки

- **price_range**: $\mathrm{Bid}_t - \mathrm{Ask}_t$.
- **range_std**: $\mathrm{std}(\mathrm{price\_range}_{t-4}, \ldots, \mathrm{price\_range}_t)$.
- **range_mean**: $\mathrm{mean}(\mathrm{price\_range}_{t-4}, \ldots, \mathrm{price\_range}_t)$.

Режим рынка

$$\text{regime}_t = \begin{cases} +1, & \text{если } \text{return}_{5d,t} > 0.01, \\ -1, & \text{если } \text{return}_{5d,t} < -0.01, \\ 0, & \text{иначе} \end{cases}$$

Фракционная дифференциация (d = 0.5)

$$x_t^{(d)} = \sum_{k=0}^{K} w_k \, x_{t-k}, \qquad w_k = \frac{(-1)^k \prod_{j=0}^{k-1}(d-j)}{k!}, \qquad d = 0.5$$

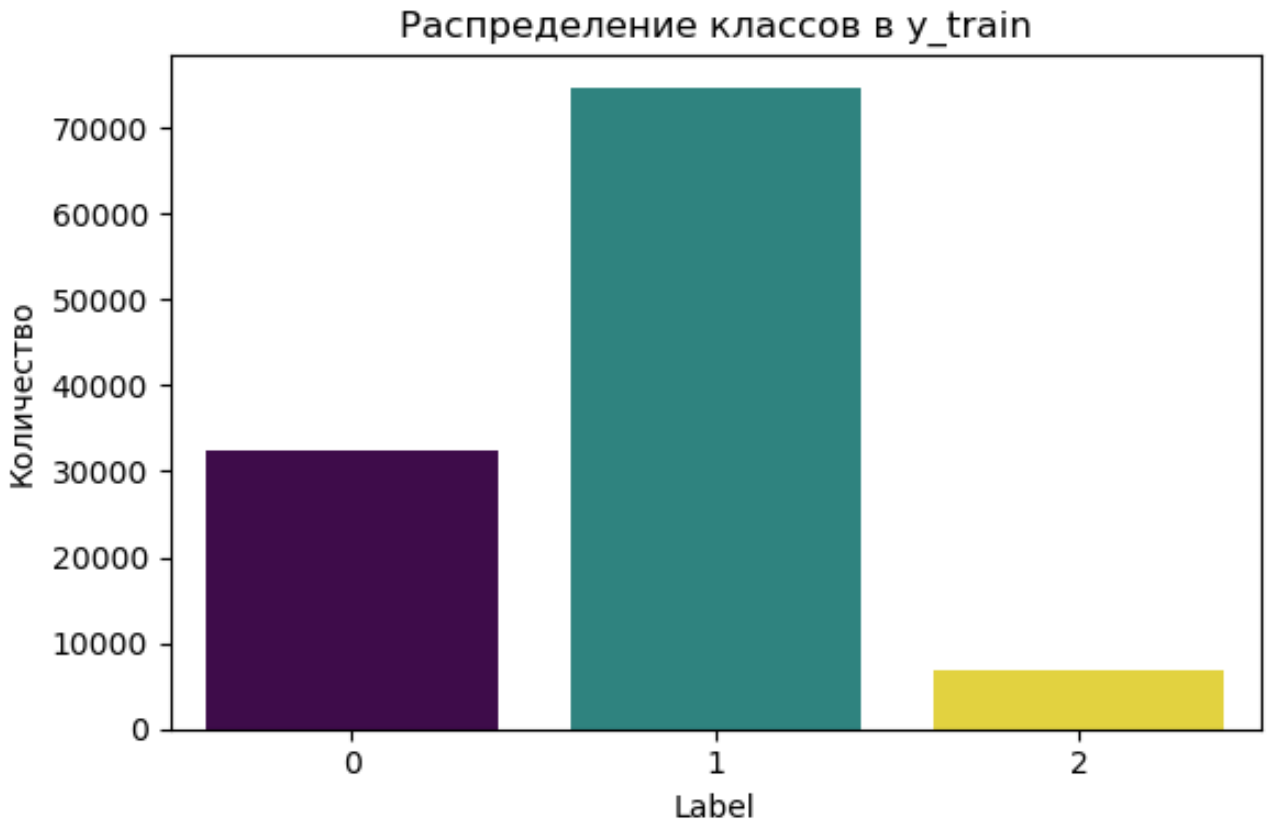
Выбор модели

Для решения задачи классификации мы выбрали модель **LightGBM** — одну из самых быстрых и эффективных реализаций градиентного бустинга.

Характеристика	**LightGBM**	**Random Forest**
Тип ансамбля	Градиентный бустинг	Бэггинг
Скорость обучения	Быстрая	Медленнее
Склонность к переобучению	Низкая	Средняя
Обработка пропусков	Есть	Нет
Категориальные признаки	Встроенная	One-hot

Разделение данных и балансировка классов

- Данные делились по дате:
  - **Train**: записи до 1 января 2021
  - **Test**: записи с 1 января 2021 и позже



Балансировка классов

- Использовали встроенную балансировку весов в LightGBM: `class_weight='balanced'`
- Классические веса рассчитывались автоматически по обратной частоте каждого класса:

$$w_i = \frac{N}{k \cdot N_i},$$

где N — общее число примеров, k — число классов, N_i — число примеров класса i.

Результаты на обучающей выборке

- Accuracy: **0.80**
- Macro F1-score: **0.69**
- Weighted F1-score: **0.81**

Класс	Precision	Recall	F1-score	Support
-1	0.75	0.58	0.65	32356
0	0.93	0.90	0.91	74649
1	0.36	0.88	0.51	6754

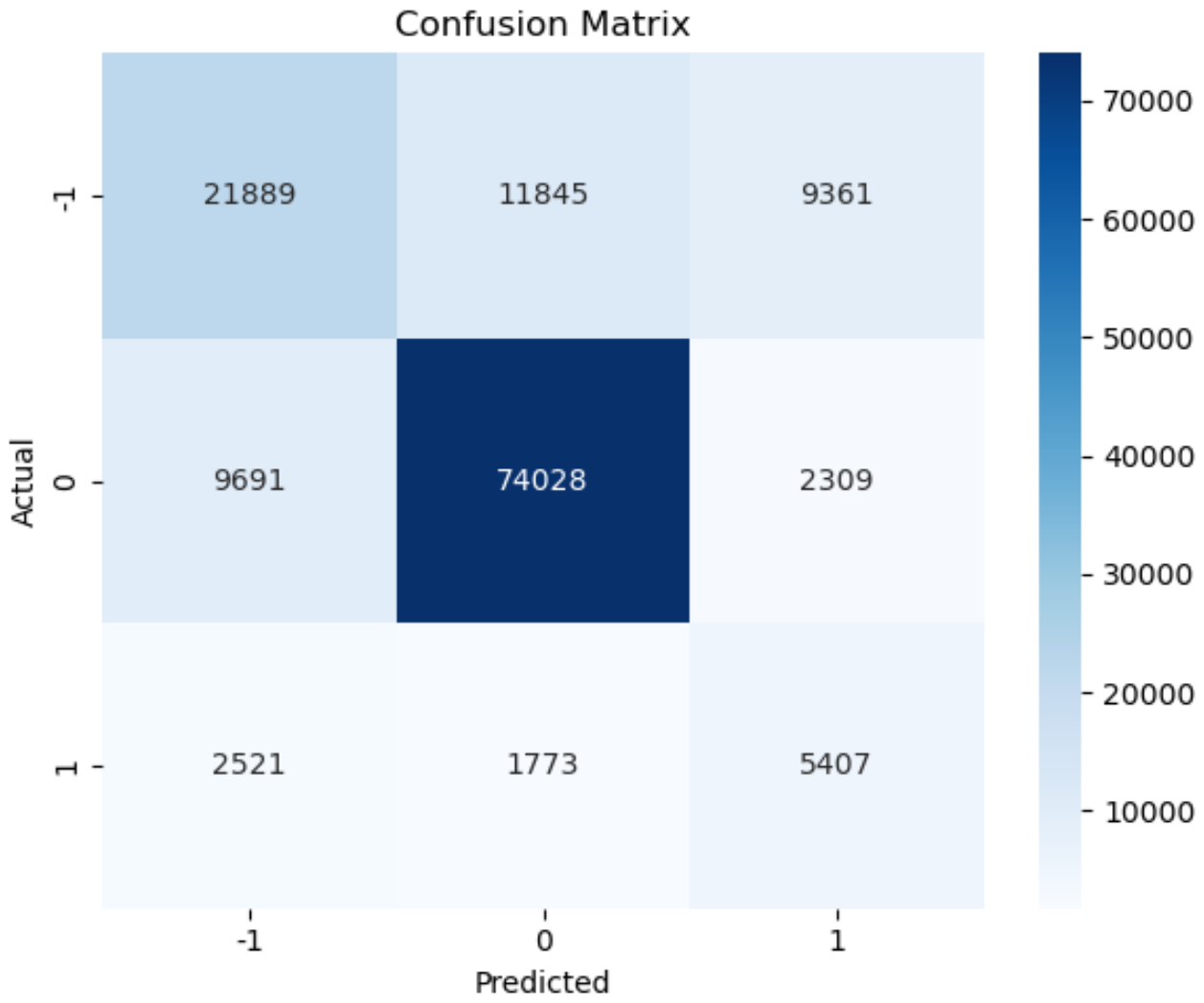
Результаты на тестовой выборке

- Accuracy: **0.73**
- Macro F1-score: **0.61**
- Weighted F1-score: **0.73**

Класс	Precision	Recall	F1-score	Support
-1	0.64	0.51	0.57	43095
0	0.84	0.86	0.85	86028
1	0.32	0.56	0.40	9701

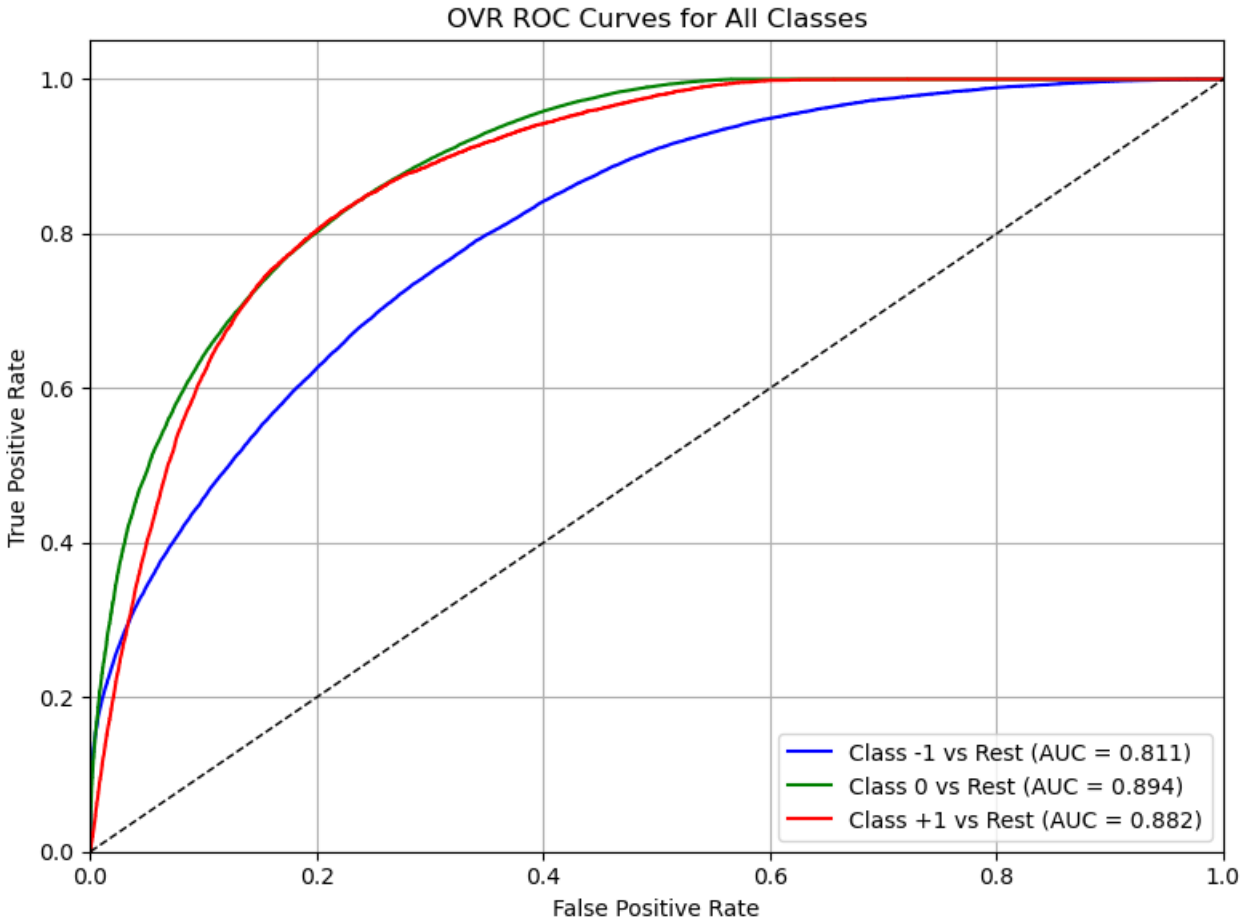
Матрица ошибок на тестовой выборке

- По оси Y — реальные метки классов.
- По оси X — предсказанные моделью метки.



ROC-кривые (One-vs-Rest)

- Для многоклассовой классификации ROC-кривые можно построить по схеме **один против всех**:
  - Для каждого класса считается, насколько хорошо модель отличает его от остальных.



Метрика ROC AUC (многоклассовый случай)

- ROC AUC — это площадь под ROC-кривой: отражает качество ранжирования модели.
- В многоклассовой задаче используется схема **One-vs-Rest (OVR)**:
  - Строятся 3 ROC-кривые: для каждого класса против всех остальных.
- Итоговое значение считается как **средневзвешенное**:

$$\text{ROC AUC}_{\text{weighted}} = \sum_{i=1}^{3} \frac{n_i}{N} \cdot \text{AUC}_i$$

где n_i — число примеров класса i, N — общее число.

- Чем ближе ROC AUC к 1, тем лучше модель различает классы.

Log Loss (логарифмическая функция потерь)

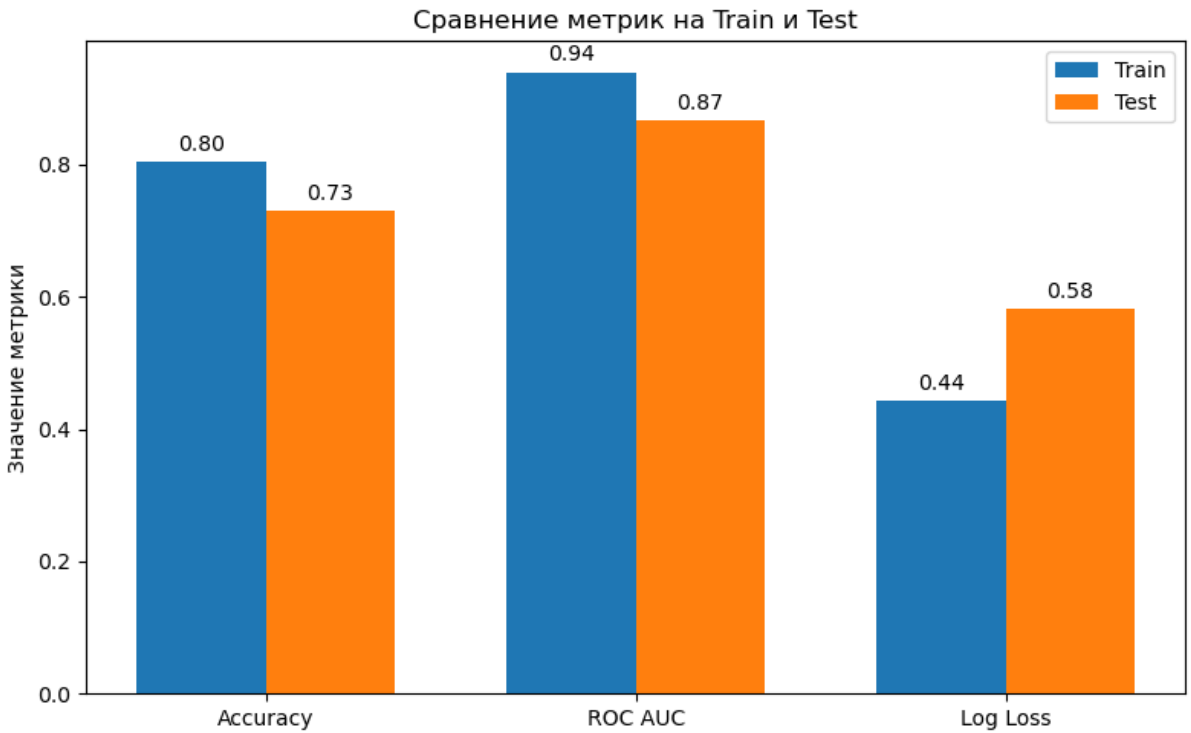
- **Log Loss** измеряет «насколько уверена» модель в своих вероятностях.
- Наказывается неуверенность и особенно уверенные ошибки.
- Чем меньше log loss, тем лучше.

$$\text{LogLoss} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{K} y_{i,j} \cdot \log(p_{i,j})$$

- N — число наблюдений, K — число классов
- $y_{i,j} = 1$, если объект i относится к классу j, иначе 0
- $p_{i,j}$ — предсказанная вероятность класса j для объекта i

Сравнение метрик на Train и Test

- Сравниваются значения ключевых метрик на обучающей и тестовой выборках:
  - **Accuracy** — доля правильных предсказаний
  - **ROC AUC** — качество ранжирования
  - **Log Loss** — штраф за ошибочные вероятности
- Наблюдается некоторое переобучение, но в разумных пределах.



SHAP: Важность признаков

- SHAP рассчитывает вклад каждого признака f_j в предсказание модели на основе теории игр:

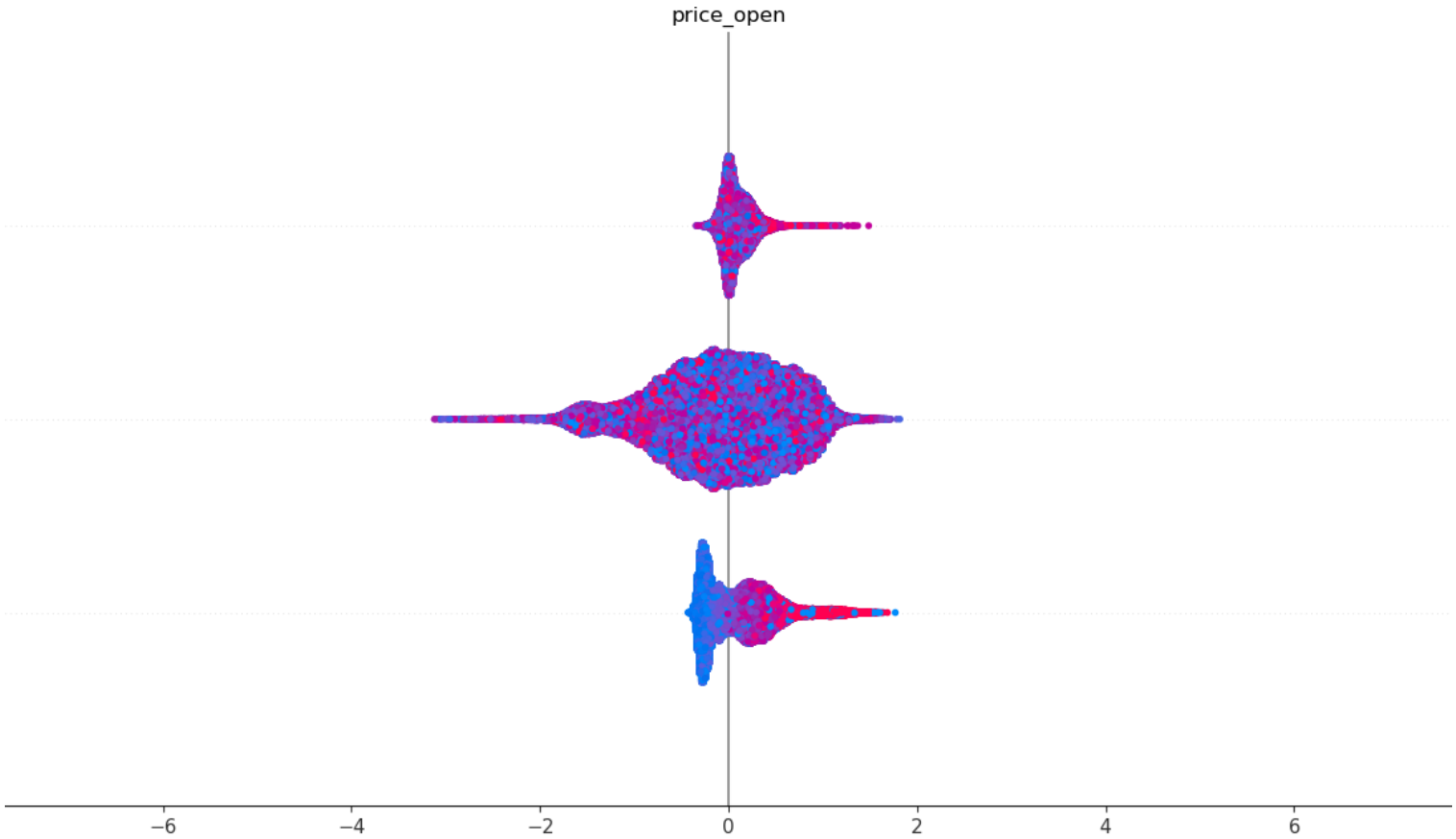
$$\phi_j = \sum_{S \subseteq F \setminus \{j\}} \frac{|S|!\,(|F| - |S| - 1)!}{|F|!} \left[f_{S \cup \{j\}}(x) - f_S(x) \right]$$

- Где:
  - ϕ_j — значение SHAP для признака f_j,
  - F — множество всех признаков,
  - S — подмножество признаков, не включающее j,
  - $f_S(x)$ — предсказание модели, построенной только на призн. из S
- Итоговая важность — среднее по абсолютным значениям SHAP:

$$\text{Importance}(f_j) = \frac{1}{n} \sum_{i=1}^{n} |\phi_j^{(i)}|$$

Важнейший признак модели: price_open

- Согласно SHAP-анализу, наиболее существенный вклад в предсказания модели даёт признак `price_open` — цена открытия опциона.



Заключение и перспективы

- Модель LightGBM демонстрирует разумное качество:
  - Accuracy на тесте: 73%
  - ROC AUC: от 0.81 до 0.89 в зависимости от класса
  - Наилучшая точность — у нейтральных опционов
- Возможные пути улучшения модели:
  - Учёт корпоративных событий: отчётностей, дивидендов, сплитов
  - Добавление новостных и юридических факторов (регуляторные риски, экономические отчёты)
  - Использование временных моделей
  - Интеграция с торговыми стратегиями, основанными на вероятностях