1. Постановка задачи

В данном проекте перед нами стоит цель разработать модели, позволяющие прогнозировать эффективность химических соединений против вируса гриппа и выбирать наиболее перспективные соединения для дальнейших лабораторных исследований и разработки лекарственных препаратов. Для этого нам предоставлены данные о 1000 химических соединениях. Для каждого из них в исходном наборе данных известны значения IC50, CC50 и SI. На их основе требуется:

1. Проанализировать распределения и взаимосвязи между признаками (включая дополнительные физико-химические свойства соединений, если они представлены), выявить возможные выбросы, аномалии и зависимости, которые могут помочь в дальнейшем построении более точных моделей.
2. Построить и сравнить несколько моделей регрессии для прогнозирования непрерывных показателей:
   - Регрессия для IC50;
   - Регрессия для CC50;
   - Регрессия для SI.

   В каждой задаче подобрать и протестировать разные алгоритмы, настроить гиперпараметры и оценить качество модели с помощью метрик (RMSE, MAE, R2).
3. Построить и сравнить несколько моделей классификации на основе бинарных целевых меток, сформированных по следующим правилам:
   - «IC50 > медиана выборки?»;
   - «CC50 > медиана выборки?»;
   - «SI > медиана выборки?»;
   - «SI > 8?».

   В каждой задаче подобрать и протестировать разные алгоритмы, настроить гиперпараметры и оценить качество модели с помощью метрик (f1 score, accuracy score, roc auc score).
4. Сравнить между собой полученные модели по их метрикам качества. На основании этого сделать обоснованный выбор лучших моделей для каждой из поставленных задач.

2. Описание и обработка данных

2.1 Структура исходного набора данных

Исходный датасет включает вычислительные дескрипторы для тысячи уникальных химических соединений. Каждое соединение охарактеризовано 214 числовыми атрибутами, исчерпывающе описывающими его молекулярную архитектуру и фундаментальные свойства.

Целевые показатели представлены тремя ключевыми фармакологическими метриками: ингибирующей концентрацией (IC50), цитотоксической концентрацией (CC50) и производным от них индексом селективности (SI = CC50 / IC50).

Набор признаков охватывает широкий спектр молекулярных характеристик, включая физико-химические параметры, электронные и топологические дескрипторы, структурные индексы (включая графовые инварианты), молекулярные отпечатки и фрагментные паттерны.

Важно отметить: исходные данные содержат исключительно числовые значения, строковые признаки отсутствуют.

2.2 Работа с пропущенными значениями

Предварительный анализ структуры данных выявил наличие пропущенных значений в группе дескрипторов, описывающих распределение парциальных зарядов (MaxPartialCharge, MinPartialCharge, MaxAbsPartialCharge, MinAbsPartialCharge) и в наборе двумерных дескрипторов BCUT (BCUT2D_MWHI, BCUT2D_MWLOW, BCUT2D_CHGHI, BCUT2D_CHGLO, BCUT2D_LOGPHI, BCUT2D_LOGPLOW, BCUT2D_MRHI, BCUT2D_MRLOW).

Для устранения этих пропусков был применен IterativeImputer — метод, который не просто замещает недостающие данные статистикой (вроде среднего), а итеративно моделирует их на основе сложных многомерных взаимосвязей между всеми доступными числовыми признаками в наборе данных.

2.3 Удаление нерелевантных и «плохих» признаков

В ходе анализа с использованием автоматического набора тестов из библиотеки

deepchecks

Процедура аудита качества данных с применением инструментария библиотеки deepchecks позволила идентифицировать ряд несоответствий и сформулировать корректирующие действия:

1. Устранение константных признаков: Были детектированы 18 атрибутов с нулевой дисперсией (все экземпляры содержали идентичное значение). Поскольку такие признаки не вносят вклад в вариативность модели, они были исключены. Список удаленных
дескрипторов: NumRadicalElectrons, SMR_VSA8, SlogP_VSA9, fr_N_O, fr_SH, fr_azi

de, fr_barbitur, fr_benzodiazepine, fr_diazo, fr_dihydropyridine, fr_isocyan, fr_isothiocyan, fr_lactam, fr_nitroso, fr_phos_acid, fr_phos_ester, fr_prisulfonamd, fr_thiocyan.

2. Снижение мультиколлинеарности: Анализ выявил многочисленные пары признаков с чрезмерно высокой взаимной корреляцией. Для минимизации этого эффекта был задействован трансформатор DropCorrelatedFeatures с установленным порогом threshold=0.9, что привело к автоматическому удалению избыточных столбцов.

3. Целостность данных: Проверки подтвердили структурную корректность:
   - Отсутствие смешанных типов пропусков в пределах одного атрибута;
   - Единообразие типов данных: все 214 исходных столбцов являлись числовыми или содержали допустимые вариации (без неожиданных строковых вставок);
   - Полное отсутствие дубликатов записей;
   - Неприменимость проверок на длину строки из-за отсутствия текстовых колонок.

4. Удаление технического артефакта: Столбец 'Unnamed: 0', содержащий лишь дублирующие индексы, был удален как нефункциональный.

5. Коррекция по биологической достоверности: Целевые показатели (IC50, CC50, SI) по своей природе должны быть строго положительными. Все строки, где хотя бы одно из этих значений было ≤ 0, были отфильтрованы как некорректные.

В результате выполненных операций размерность пространства признаков сократилась с 214 до 156. Финализированный датасет включает три целевых переменных (IC50, CC50, SI) и 153 релевантных числовых дескриптора, готовых для моделирования.

2.4 Анализ распределений

Визуальный анализ распределений ключевых фармакологических показателей (IC50, CC50, SI) представлен на рисунке 2.4.1.

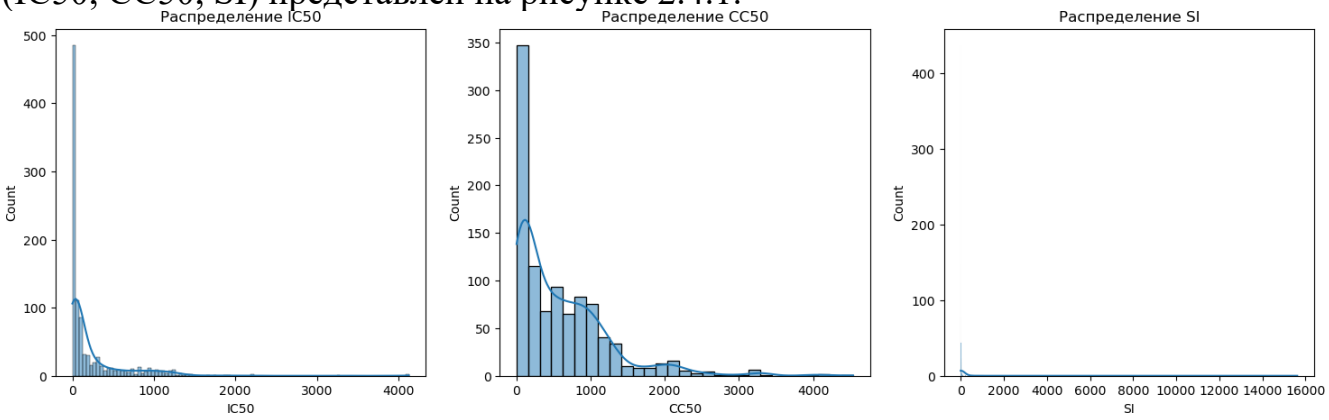

Рисунок 2.4.1 – Распределение целевых переменных (IC50, CC50, SI)

Наблюдения, полученные при изучении графиков:

- Все распределения демонстрируют ярко выраженную положительную асимметрию (скошенность вправо).
- Основная масса данных концентрируется в области малых величин, тогда как экстремальные выбросы формируют протяженный правый хвост (например, единичные значения IC50 достигают 4000).

Для объективной оценки характера распределений был применен метод автоматического подбора теоретических распределений с использованием библиотеки fitter. Результаты, визуализированные на рисунке 2.4.2, однозначно опровергают гипотезу о нормальности данных для всех целевых переменных.

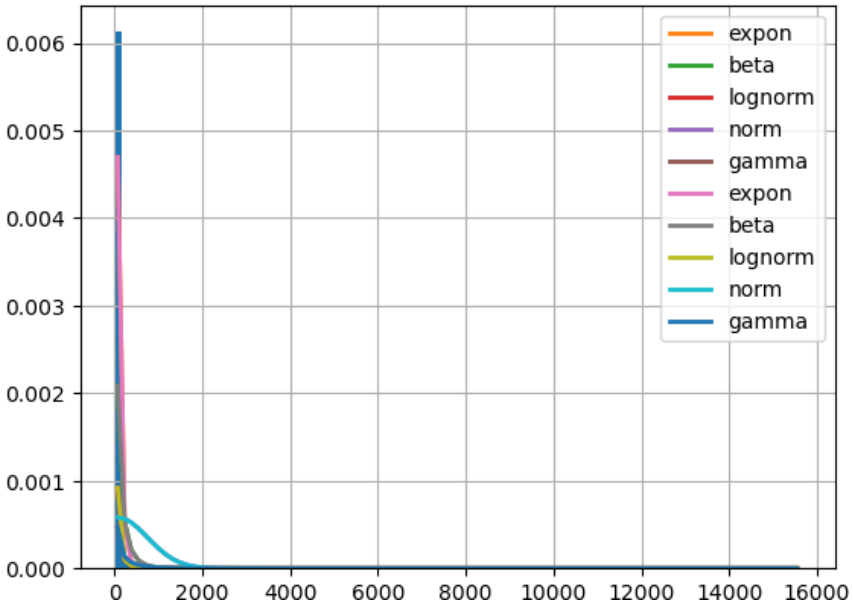

Рисунок 2.4.2 – Сравнение эмпирических распределений с теоретическими моделями

Ключевые выводы анализа соответствия:

- Распределение IC50 наиболее адекватно описывается логнормальной моделью, что подтверждается минимальной величиной ошибки аппроксимации среди всех тестируемых распределений.
- Для CC50 наилучшее соответствие наблюдается с гамма-, бета- и логнормальным распределениями, что согласуется с выявленной ранее положительной асимметрией.
- Показатель SI демонстрирует максимальную близость к бета-распределению.

2.5. Анализ выбросов

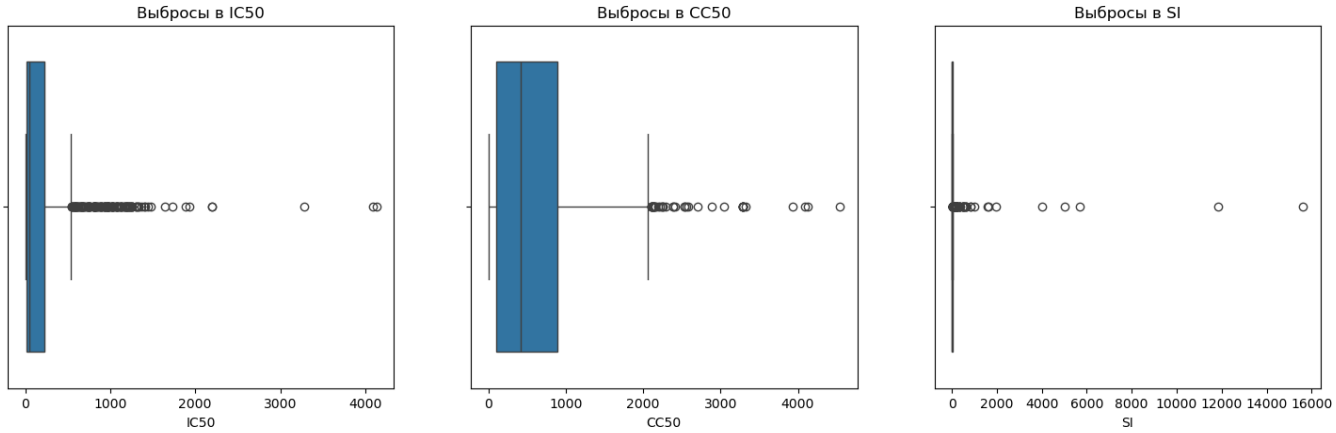

Рисунок 2.4.3 – Boxplot-диаграммы IC50, CC50 и SI

Для трёх анализируемых показателей созданы диаграммы размаха (рис. 2.4.3), визуально идентифицирующие аномальные наблюдения за пределами усов. Анализ позволяет установить:

Аномалии наблюдаются исключительно в правой части распределений. Левый ус не содержит выбросов благодаря близости минимальных значений к нижнему квартилю.

Межквартильный разброс свидетельствует о концентрации данных в нижней и средней областях, тогда как экстремальные значения формируют аномалии справа.

Визуальный осмотр подтверждает необходимость удаления исключительно правосторонних выбросов, так как в области малых значений значительных отклонений не зафиксировано.

Для минимизации влияния экстремумов использован стандартный IQR-метод фильтрации:

1. Вычисление первого (Q1) и третьего (Q3) квартилей для целевого параметра.

2. Определение межквартильного размаха:

$$IIIR = II3 - II1$$

3. Расчёт верхней границы аномалий:

$$upper\ bound = \text{Q3} + 1.5 \times \text{IQR}$$

4. Исключение образцов, превышающих установленный порог.

Данный подход сохраняет репрезентативную часть выборки, отсекая лишь статистически незначимые экстремумы.

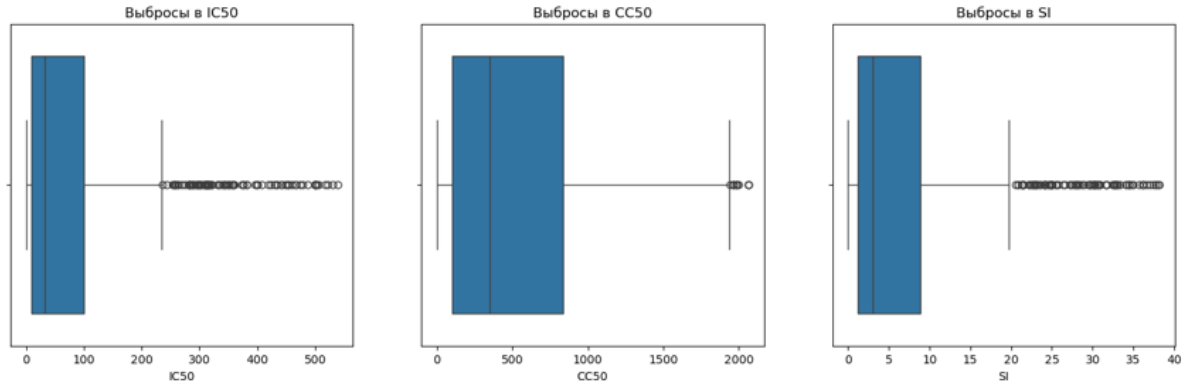

Рисунок 2.4.4 – Boxplot-диаграммы IC50, CC50 и SI после удаления выбросов

После очистки построены обновлённые диаграммы размаха (рис. 2.4.4), иллюстрирующие распределения без правосторонних аномалий.

3. Построение моделей регрессии

Исследуемые зависимые переменные характеризуются отклонением от нормального закона распределения, проявляя существенную положительную асимметрию и протяженную область выбросов в правой части. Подобный характер данных создает сложности для использования алгоритмов, критичных к нормальности остатков (таких как линейная регрессия). Вследствие этого фокус при моделировании смещен в сторону ансамблевых методов, основанных на решающих деревьях (Random Forest, градиентный бустинг и аналоги). Эти алгоритмы демонстрируют меньшую зависимость от исходного распределения целевого признака.

Поскольку ключевые рассматриваемые методы базируются на деревьях, они инвариантны к масштабированию входных переменных и не нуждаются в предварительной нормализации или стандартизации признаков. Дополнительно, процедуры сокращения размерности (например, метод главных компонент - PCA) не являются необходимыми, так как древовидные структуры эффективно обрабатывают мультиколлинеарность и избыточность в данных.

Соответственно, предварительно обработанный набор признаков использовался для обучения моделей в исходном виде, без этапов масштабирования или снижения размерности.

Сравнительный анализ включает следующие регрессионные модели:

- Регрессор XGBoost (XGBRegressor)
- XGBRegressor с функцией потерь Tweedie
- Регрессор на основе случайного леса (RandomForestRegressor)
- Регрессор градиентного бустинга (GradientBoostingRegressor)
- Регрессор CatBoost (CatBoostRegressor)
- CatBoostRegressor с функцией потерь Tweedie
- Гистограммный регрессор градиентного бустинга (HistGradientBoostingRegressor)
- Регрессор на основе экстра-деревьев (ExtraTreesRegressor)

Процедура обработки данных и оценки моделей будет единообразной для всех трех целевых переменных:

Исходный датасет первоначально разделяется на обучающий и тестовый блоки методом hold-out (train-test split). Исходные модели проходят обучение на тренировочной подвыборке, после чего рассчитываются начальные показатели их эффективности.

С целью улучшения прогнозной силы алгоритмов выполняется подбор оптимальных гиперпараметров с использованием фреймворка Optuna.

В качестве целевой метрики для Optuna выбран коэффициент детерминации (R^2), руководствуясь следующими соображениями:

1. **Смысловая ясность**: R^2 количественно определяет, какую долю вариации целевой переменной способна объяснить модель относительно простого среднего значения. Это обеспечивает более интуитивно понятную оценку, чем абсолютные величины ошибок.
2. **Фокус на объяснении вариативности**: Основная цель оптимизации

гиперпараметров – максимизировать способность модели описывать изменения в данных, а R^2 напрямую измеряет этот аспект, в отличие от метрик, ориентированных исключительно на минимизацию ошибки.

Для каждого регрессора реализуется индивидуальный процесс настройки, состоящий из этапов:

1. Дополнительное разбиение тренировочного набора на подвыборки для обучения (80%) и валидации (20%).
2. Итеративный поиск по пространству гиперпараметров: на каждой итерации модель обучается на тренировочной части, а её качество оценивается по R^2 на валидационной части.
3. Фиксация комбинации гиперпараметров, обеспечивающей наивысшее значение R^2 на валидации.
4. Обучение итоговой модели с выбранными параметрами на полном тренировочном наборе.
5. Тестирование финальной модели на отложенной выборке с расчетом метрик R^2, RMSE и MAE.

Данная логика инкапсулирована в функцию optuna_tuning. Она принимает словарь моделей-кандидатов, последовательно для каждой:

- Осуществляет внутреннее разбиение данных.
- Задает целевую функцию для Optuna, специфичную для гиперпараметров конкретной модели.
- Инициирует оптимизационный процесс для максимизации R^2.
- После завершения оптимизации обучает финальный вариант модели на всем тренировочном наборе.
- Проводит оценку на тестовых данных, сохраняя полученные метрики и объект обученной модели.

Функция возвращает сводную таблицу с результатами оценки и коллекцию обученных моделей. Такой подход гарантирует глубокий подбор конфигураций и объективную валидацию эффективности моделей на независимых данных.

3.1 Регрессия для IC50

Выполнена оценка алгоритмов со стандартными параметрами. Показатели эффективности до оптимизации приведены в таблице 3.1.1.

Табица 3.1.1 – Результаты регрессии без оптимизации гиперпараметров

Модель	R2	RMSE	MAE
XGB	0.1210	97.1841	58.8789
XGB_Tweedie	0.3023	86.5880	48.1642
RandomForest	0.1687	94.5122	58.7291
GradientBoosting	0.1889	93.3546	61.9057
CatBoost	0.2102	92.1207	57.8841
CatBoost_Tweedie	0.3215	85.3831	48.0945
HistGradientBoostingRegressor	0.0070	103.2943	64.3260
ExtraTreesRegressor	0.2115	92.0490	58.9141

Наибольшую эффективность продемонстрировали CatBoost_Tweedie и XGB_Tweedie, достигнув максимальных значений R2 (0.3215 и 0.3023) при минимальных погрешностях RMSE (85.38, 86.59) и MAE (48.09, 48.16). Наименее результативной оказалась HistGradientBoostingRegressor с экстремально низким R2 (0.0070) и максимальными ошибками (RMSE=103.29, MAE=64.33), что указывает на её низкую эффективность без корректировки параметров. Графическое сопоставление метрик отражено на рисунке 3.1.1.

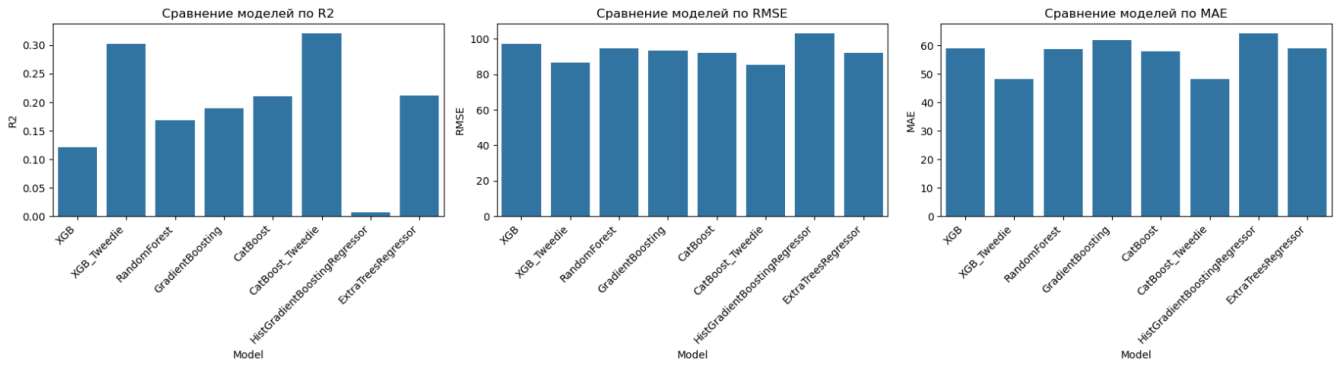

Рисунок 3.1.1– Сравнение моделей по метрикам

После оптимизации гиперпараметров получены результаты, представленные в таблице 3.1.2.

Таблица 3.1.2 – Результаты регрессии после оптимизации гиперпараметров.

Модель	R2	RMSE	MAE
XGB	0.1965	92.9168	59.4386

XGB_Tweedie	0.2716	88.4664	55.0039
RandomForest	0.2071	92.3055	59.0738
GradientBoosting	0.2428	90.2010	59.2880
CatBoost	0.2889	87.4114	57.2577
CatBoost_Tweedie	0.3179	85.6113	53.3481
HistGradientBoostingRegressor	0.1739	94.2133	62.8600
ExtraTreesRegressor	0.2758	88.2144	59.4802

Лидером по прежнему остается CatBoost_Tweedie с максимальным R2 (0.3179), минимальным RMSE (85.61) и MAE (53.35). Существенный прогресс также наблюдается у CatBoost и ExtraTreesRegressor.

HistGradientBoostingRegressor сохранила наихудшие показатели: минимальный R2 (0.1739) при высоких RMSE (94.21) и MAE (62.86), несмотря на проведённую оптимизацию.

Сравнительный анализ метрик после настройки отображен на рисунке 3.1.2.

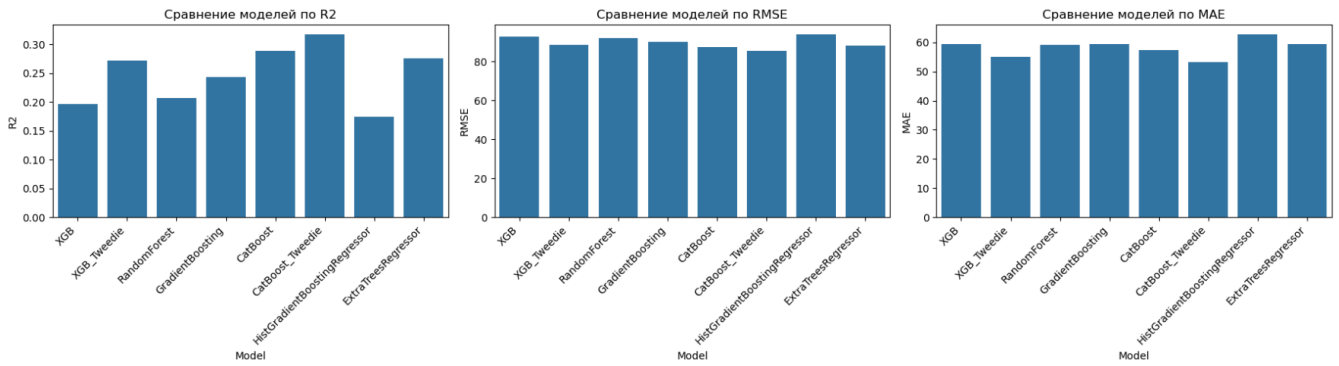

Рисунок 3.1.2– Сравнение моделей по метрикам после подбора гиперпараметров

Сводные данные представлены в таблице 3.1.3:

Таблица 3.1.3 – Общие результаты моделей.

Модель	**R2**	**RMSE**	**MAE**	**Tuned**
CatBoost_Tweedie	0.3215	85.3831	48.0945	False
CatBoost_Tweedie	0.3179	85.6113	53.3481	True
XGB_Tweedie	0.3023	86.5880	48.1642	False
Модель	**R2**	**RMSE**	**MAE**	**Tuned**
CatBoost	0.2889	87.4114	57.2577	True
ExtraTreesRegressor	0.2758	88.2144	59.4802	True
XGB_Tweedie	0.2716	88.4664	55.0039	True
GradientBoosting	0.2428	90.2010	59.2880	True

ExtraTreesRegressor	0.2115	92.0490	58.9141	False
CatBoost	0.2102	92.1207	57.8841	False
RandomForest	0.2071	92.3055	59.0738	True
XGB	0.1965	92.9168	59.4386	True
GradientBoosting	0.1889	93.3546	61.9057	False
HistGradientBoostingRegressor	0.1739	94.2133	62.8600	True
RandomForest	0.1687	94.5122	58.7291	False
XGB	0.1210	97.1841	58.8789	False
HistGradientBoostingRegressor	0.0070	103.2943	64.3260	False

Наивысшая эффективность зафиксирована у CatBoost_Tweedie без оптимизации с R2=0.3215, RMSE=85.38 и MAE=48.09. Близкие результаты демонстрируют её настроенная версия и XGB_Tweedie без коррекции параметров.

Аутсайдером признана HistGradientBoostingRegressor без оптимизации с критически низким R2 (0.0070) и предельными погрешностями (RMSE=103.29, MAE=64.33).

Tweedie-реализации градиентного бустинга (прежде всего CatBoost_Tweedie) устойчиво показывают высокую прогностическую способность независимо от настройки. Для предсказания IC50 рекомендована CatBoost_Tweedie, обеспечивающая оптимальное соотношение точности и устойчивости результатов.

На рисунке 3.1.3 визуализирована динамика метрик до и после оптимизации, подтверждающая общее улучшение показателей после корректировки гиперпараметров.

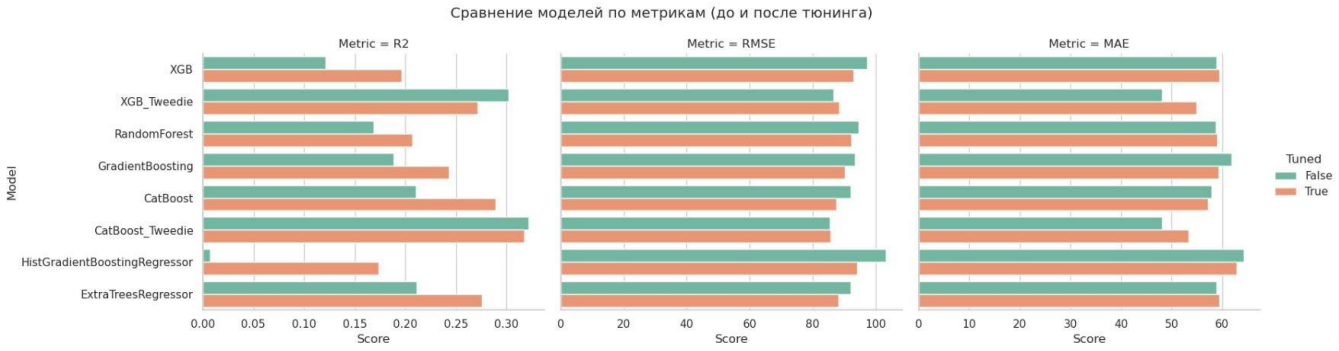

Рисунок 3.1.3– Сравнение моделей по метрикам до и после подбора гиперпараметров

3.2 Регрессия для CC50

Первоначально выполнена оценка алгоритмов со стандартной конфигурацией. Показатели эффективности до корректировки гиперпараметров систематизированы в таблице 3.2.1.

Таблица 3.2.1 – Результаты регрессии без оптимизации гиперпараметров

Модель	R2	RMSE	MAE
XGB	0.5891	307.4675	189.0528
XGB_Tweedie	0.4637	351.2326	200.5619
RandomForest	0.6168	296.9035	196.7001
GradientBoosting	0.6246	293.8594	204.8962
CatBoost	0.6223	294.7615	193.1516
CatBoost_Tweedie	0.5112	335.3213	198.4659
HistGradientBoostingRegressor	0.5974	304.3362	197.2478
ExtraTreesRegressor	0.5788	311.2923	189.9262

GradientBoosting продемонстрировала максимальную прогностическую способность (R2=0.6246) при минимальной RMSE (293.86). Аналогично высокие показатели у CatBoost и RandomForest.

Минимальные абсолютные погрешности зафиксированы у XGB (MAE=189.05) и ExtraTreesRegressor (MAE=189.93) при сохранении конкурентоспособных значений R2.

Tweedie-модификации (XGB_Tweedie и CatBoost_Tweedie) показали наименьшие R2 (0.4637 и 0.5112), что свидетельствует о меньшей применимости данного подхода для прогнозирования CC50 по сравнению с IC50.
Графическая интерпретация метрик представлена на рисунке 3.2.1.

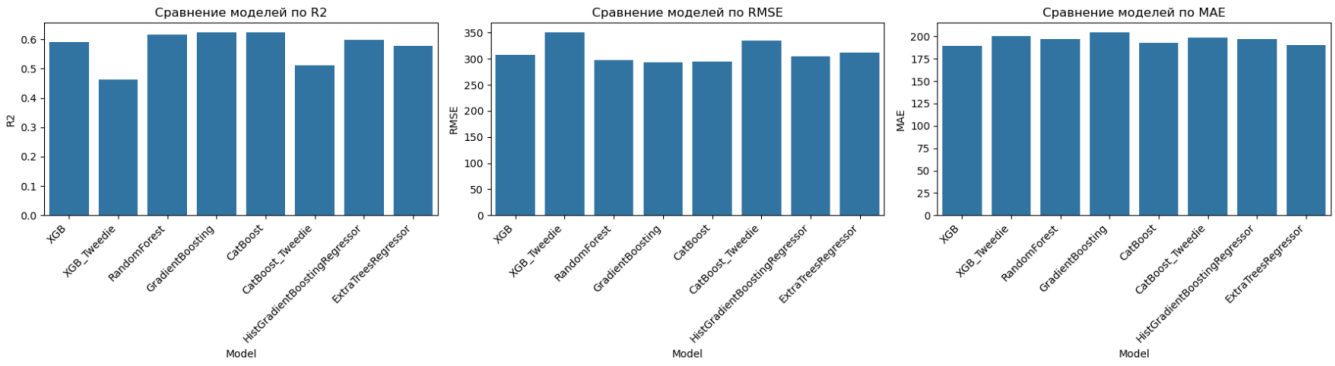

Рисунок 3.2.1– Сравнение моделей по метрикам

После оптимизации гиперпараметров получены результаты, отражённые в таблице 3.2.2.

Таблица 3.2.2 – Результаты регрессии после оптимизации гиперпараметров.

Модель	R2	RMSE	MAE
XGB	0.5950	305.2333	207.7947
XGB_Tweedie	0.5505	321.5631	202.4438
RandomForest	0.6107	299.2765	197.9957
GradientBoosting	0.5769	311.9868	214.9640
CatBoost	0.6396	287.9338	196.8751
CatBoost_Tweedie	0.5656	316.1061	197.3229
HistGradientBoostingRegressor	0.6111	299.0948	197.0752
ExtraTreesRegressor	0.6161	297.1661	189.8310

CatBoost достиг наивысшей точности прогнозирования с максимальным R2 (0.6396) и минимальной RMSE (287.93).

ExtraTreesRegressor показала наименьшую абсолютную погрешность (MAE=189.83) при высоком R2 (0.6161), демонстрируя сбалансированность метрик. GradientBoosting проявила значительное ухудшение по MAE (214.96) после настройки, а XGB_Tweedie зафиксировала минимальный R2 (0.5505) среди оптимизированных моделей.

Сравнительный анализ после настройки визуализирован на рисунке 3.2.2.

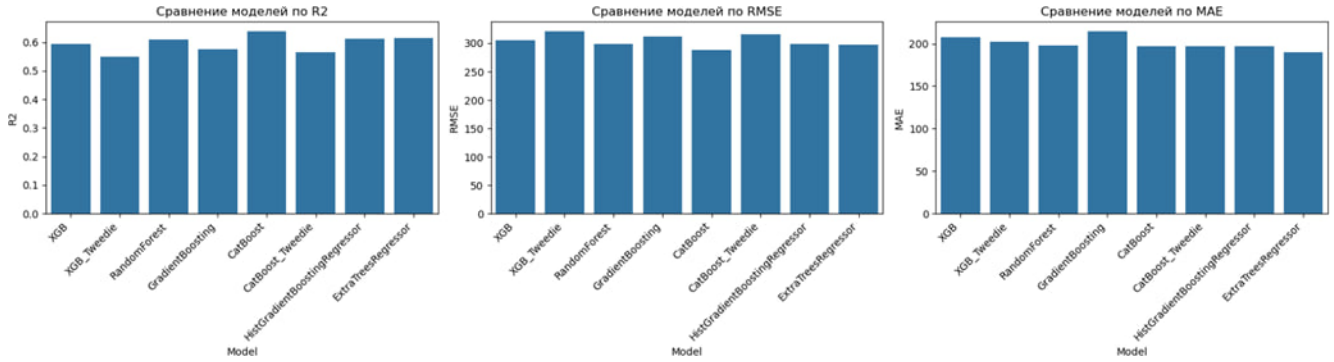

Рисунок 3.2.2– Сравнение моделей по метрикам после подбора гиперпараметров
Сводные данные представлены в таблице 3.2.3:

Таблица 3.2.3 – Общие результаты моделей.

Модель	R2	RMSE	MAE	Tuned
CatBoost	0.6396	287.9338	196.8751	True
GradientBoosting	0.6246	293.8594	204.8962	False
CatBoost	0.6223	294.7615	193.1516	False
RandomForest	0.6168	296.9035	196.7001	False
ExtraTreesRegressor	0.6161	297.1661	189.8310	True
HistGradientBoostingRegressor	0.6111	299.0948	197.0752	True

RandomForest	0.6107	299.2765	197.9957	True
HistGradientBoostingRegressor	0.5974	304.3362	197.2478	False
XGB	0.5950	305.2333	207.7947	True
XGB	0.5891	307.4675	189.0528	False
ExtraTreesRegressor	0.5788	311.2923	189.9262	False
GradientBoosting	0.5769	311.9868	214.9640	True
CatBoost_Tweedie	0.5656	316.1061	197.3229	True
XGB_Tweedie	0.5505	321.5631	202.4438	True
CatBoost_Tweedie	0.5112	335.3213	198.4659	False
XGB_Tweedie	0.4637	351.2326	200.5619	False

Настроенная версия CatBoost продемонстрировала превосходство по ключевым метрикам (R2=0.6396, RMSE=287.93). Близкие результаты показали GradientBoosting без оптимизации, базовый CatBoost, а также ExtraTreesRegressor с настройкой.

Tweedie-реализации без оптимизации (XGB_Tweedie и CatBoost_Tweedie) показали минимальную объяснённую дисперсию (R2=0.4637 и 0.5112) с максимальными ошибками прогноза.

Для решения задачи прогнозирования CC50 оптимальным решением признан CatBoost с настроенными гиперпараметрами, обеспечивающий максимальную точность и устойчивость.

На рисунке 3.2.3 отражена динамика улучшения метрик после оптимизации параметров моделей.

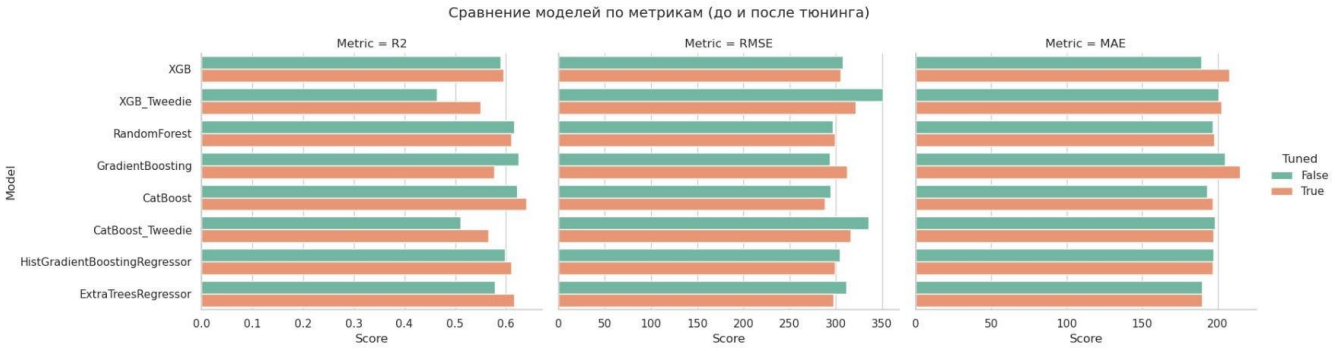

Рисунок 3.2.3– Сравнение моделей по метрикам до и после подбора гиперпараметров

3.3 Регрессия для SI

Первоначальная оценка алгоритмов со стандартными параметрами представлена в таблице 3.3.1. Показатели эффективности до оптимизации демонстрируют ограниченную предсказательную способность.

Таблица 3.3.1 – Результаты регрессии без оптимизации гиперпараметров

Модель	R2	RMSE	MAE
XGB	-0.1277	9.0371	5.8764
XGB_Tweedie	-0.0228	8.6063	5.0845
RandomForest	0.0215	8.4181	5.8684
GradientBoosting	0.1142	8.0092	5.4552
CatBoost	0.0281	8.3895	5.5789
CatBoost_Tweedie	0.0735	8.1911	4.91
HistGradientBoosting	0.025	8.4031	5.5868
ExtraTreesRegressor	-0.0394	8.6761	5.6291

GradientBoosting выделяется максимальной объясняющей способностью (R2=0.1142) при низких RMSE (8.01) и MAE (5.46). CatBoost_Tweedie демонстрирует минимальную абсолютную погрешность (MAE=4.91) при приемлемом R2 (0.0735).

Наихудшая производительность наблюдается у XGB и ExtraTreesRegressor с отрицательными значениями R2 (-0.1277 и -0.0394), что указывает на непригодность базовых версий этих алгоритмов для решения задачи.

Общий низкий уровень R2 свидетельствует о сложности прогнозирования SI и необходимости параметрической оптимизации.

Визуальное сопоставление метрик отражено на рисунке 3.3.1.

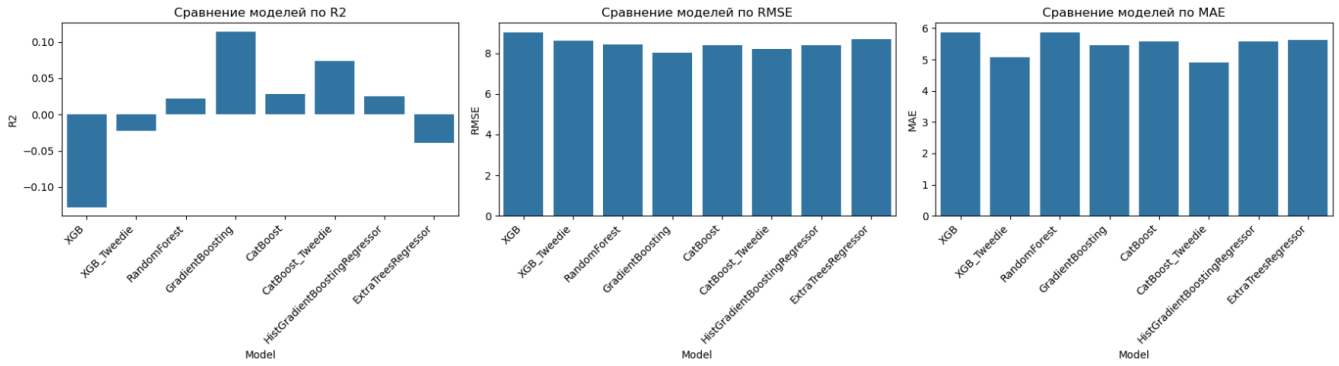

Рисунок 3.3.1– Сравнение моделей по метрикам

После корректировки гиперпараметров достигнуты следующие показатели (таблица 3.3.2):

Таблица 3.3.2 – Результаты регрессии после оптимизации гиперпараметров.

Модель	R2	RMSE	MAE
XGB	0.1407	7.8884	5.5434
XGB_Tweedie	0.091	8.1137	5.3519
RandomForest	0.1305	7.9352	5.715
GradientBoosting	0.1502	7.8448	5.4877
CatBoost	0.0833	8.1479	5.6424
CatBoost_Tweedie	0.1483	7.8538	5.1547
HistGradientBoostingRegressor	0.1349	7.9151	5.5755
ExtraTreesRegressor	0.1162	8.0003	5.6628

GradientBoosting и CatBoost_Tweedie показывают максимальную эффективность после настройки: первая достигла наивысшего R2 (0.1502), вторая – минимального MAE (5.15) при сопоставимом R2 (0.1483).
CatBoost демонстрирует наименьшее улучшение среди оптимизированных моделей по всем метрикам (R2=0.0833, RMSE=8.15, MAE=5.64).
Сравнительный анализ после оптимизации визуализирован на рисунке 3.3.2.

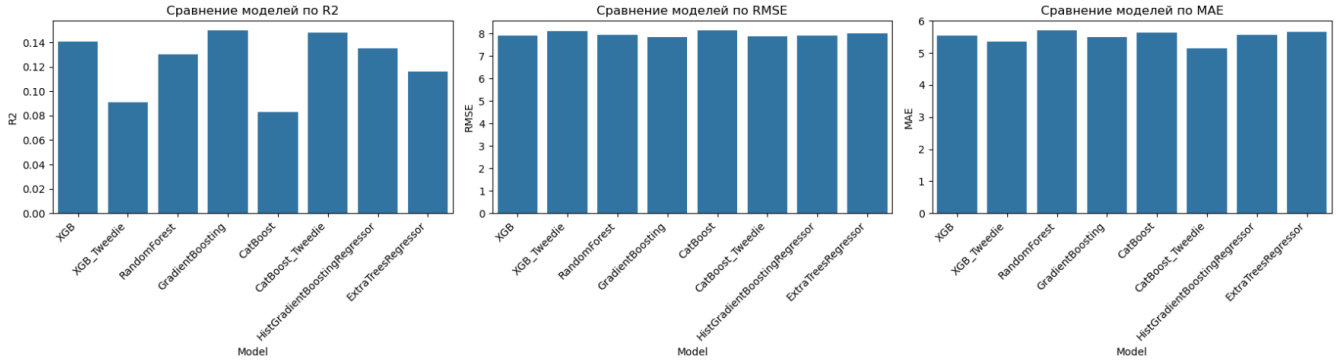

Рисунок 3.3.2– Сравнение моделей по метрикам после подбора гиперпараметров

Консолидированные результаты представлены в таблице 3.3.3:

Таблица 3.3.3 – Общие результаты моделей.

Модель	R2	RMSE	MAE	Tuned
GradientBoosting	0.1502	7.8448	5.4877	True
CatBoost_Tweedie	0.1483	7.8538	5.1547	True
XGB	0.1407	7.8884	5.5434	True
HistGradientBoostingRegressor	0.1349	7.9151	5.5755	True
RandomForest	0.1305	7.9352	5.715	True

ExtraTreesRegressor	0.1162	8.0003	5.6628	True
GradientBoosting	0.1142	8.0092	5.4552	False
XGB_Tweedie	0.091	8.1137	5.3519	True
CatBoost	0.0833	8.1479	5.6424	True
CatBoost_Tweedie	0.0735	8.1911	4.91	False
CatBoost	0.0281	8.3895	5.5789	False
HistGradientBoostingRegressor	0.025	8.4031	5.5868	False
RandomForest	0.0215	8.4181	5.8684	False
XGB_Tweedie	-0.0228	8.6063	5.0845	False
ExtraTreesRegressor	-0.0394	8.6761	5.6291	False
XGB	-0.1277	9.0371	5.8764	False

Настроенные версии GradientBoosting и CatBoost_Tweedie подтверждают лидерство по ключевым показателям, при этом CatBoost_Tweedie обеспечивает рекордно низкую абсолютную погрешность (MAE=5.15).

Базовая реализация XGB показывает критически низкое качество прогнозирования (R2=-0.1277, RMSE=9.04) – наихудший результат исследования.

Параметрическая оптимизация существенно повышает эффективность моделей для прогнозирования SI. Оптимальным решением признана CatBoost_Tweedie с настроенными параметрами, сочетающая высокую объясняющую способность и минимальную погрешность.

На рисунке 3.3.3 демонстрируется улучшение метрик после оптимизации гиперпараметров.

Рисунок 3.3.3– Сравнение моделей по метрикам до и после подбора гиперпараметров

4. Построение моделей классификации

Для сравнения рассматриваются следующие модели:

В исследовании оценивается производительность следующих алгоритмов классификации:

- Классификатор XGBoost (XGBClassifier)
- Модель случайного леса (RandomForestClassifier)
- Градиентный бустинг (GradientBoostingClassifier)
- Классификатор CatBoost (CatBoostClassifier)
- Гистограммный градиентный бустинг (HistGradientBoostingClassifier)
- Экстра-деревья (ExtraTreesClassifier)

Методология обработки данных

Процедура подготовки данных и валидации моделей сохраняет идентичную структуру с регрессионным анализом, за исключением критического шага:

1. На этапе предварительной подготовки целевой признак подвергается бинаризации
2. После преобразования в бинарный формат исходный непрерывный признак исключается из массива данных
3. Только затем выполняется разделение на матрицу признаков (X) и преобразованный целевой вектор (y)

4.1 Классификация: превышает ли значение IC50 медианное значение выборки

Сначала проведем тестирование моделей с базовыми настройками, результаты метрик без оптимизации гиперпараметров представлены в таблице 4.1.1.

Таблица 4.1.1 – Результаты регрессии без оптимизации гиперпараметров

Модель	Accuracy	ROC_AUC	F1
XGB	0.8012	0.8671	0.7875
RandomForest	0.7485	0.85	0.7226
GradientBoosting	0.7895	0.8618	0.7778
CatBoost	0.7836	0.8654	0.773
HistGradientBoosting	0.7953	0.8663	0.7799
ExtraTrees	0.7193	0.8229	0.6842

- Лучшие результаты показала модель XGB — наивысшая Accuracy (0.8012), ROC AUC (0.8671) и F1 (0.7875), что говорит о её высокой стабильности и точности.
- Хорошо себя показали также HistGradientBoosting и GradientBoosting, с чуть

меньшими, но близкими значениями метрик.

- Худшие результаты у ExtraTrees — самая низкая Accuracy (0.7193), ROC AUC (0.8229) и F1 (0.6842), что указывает на её ограниченную эффективность без настройки.

Визуальное сравнение по метрикам представлено на рисунке 4.1.1.

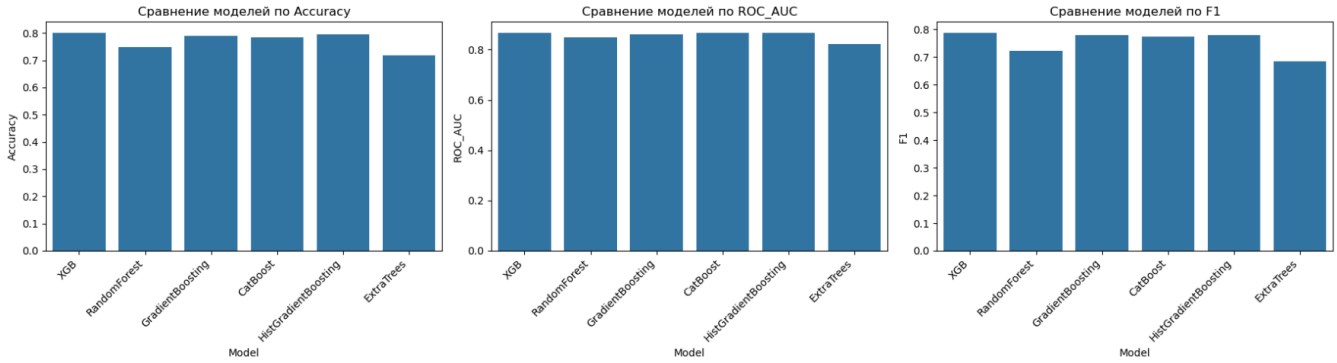

Рисунок 4.1.1– Сравнение моделей по метрикам

Далее проверим метрики после оптимизации, результаты представлены в таблице 4.1.2.

Таблица 4.1.2 – Результаты регрессии после оптимизации гиперпараметров.

Модель	**R2**	**RMSE**	**MAE**
XGB	0.7826	0.7953	0.8656
RandomForest	0.7389	0.7602	0.846
GradientBoosting	0.7636	0.7719	0.8546
CatBoost	0.7799	0.7953	0.8661
HistGradientBoosting	0.7702	0.7836	0.855
ExtraTrees	0.7368	0.7661	0.8637
XGB	0.7826	0.7953	0.8656
RandomForest	0.7389	0.7602	0.846

- Лучшие результаты показали модели CatBoost и XGB — у обеих наивысшие значения Accuracy (0.7953) и ROC AUC (0.8661 и 0.8656 соответственно), а также высокие F1 (0.7799 и 0.7826), что указывает на их устойчивое качество после настройки.
- Худшие показатели у RandomForest и ExtraTrees — наименьшие значения F1 (0.7389 и 0.7368) и Accuracy (0.7602 и 0.7661), хотя ROC AUC остаётся на приемлемом уровне.

Визуальное сравнение по метрикам после подбора гиперпараметров представлено на рисунке 4.1.2.

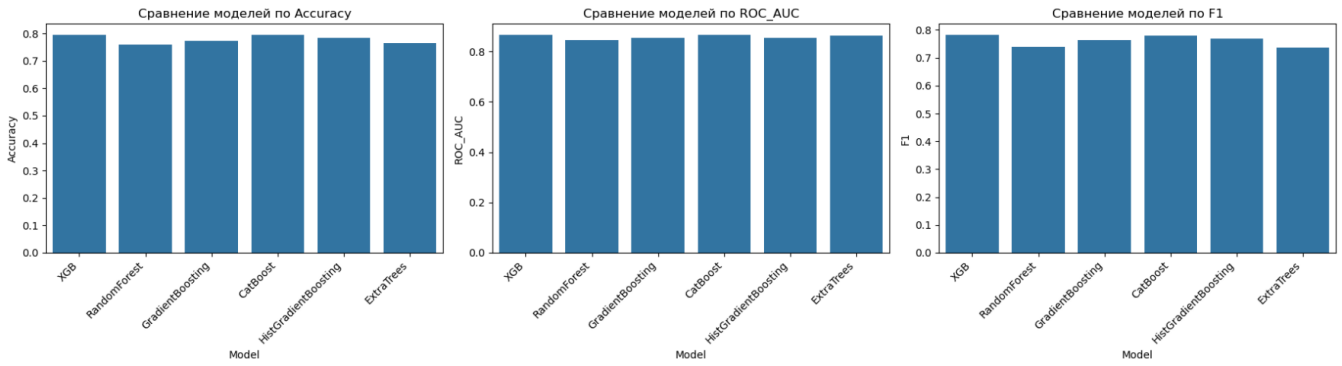

Рисунок 4.1.2– Сравнение моделей по метрикам после подбора гиперпараметров

Далее рассмотрим общую таблицу (Таблица 4.1.3):

Таблица 4.1.3 – Общие результаты моделей.

Модель	Accuracy	ROC_AUC	F1	Tuned
XGB	0.8012	0.8671	0.7875	False
CatBoost	0.7953	0.8661	0.7799	True
XGB	0.7953	0.8656	0.7826	True
HistGradientBoosting	0.7953	0.8663	0.7799	False
GradientBoosting	0.7895	0.8618	0.7778	False
CatBoost	0.7836	0.8654	0.773	False
HistGradientBoosting	0.7836	0.855	0.7702	True
GradientBoosting	0.7719	0.8546	0.7636	True
ExtraTrees	0.7661	0.8637	0.7368	True
RandomForest	0.7602	0.846	0.7389	True
RandomForest	0.7485	0.85	0.7226	False

- Лучшие результаты показала модель XGB без подбора — наивысшие значения Accuracy (0.8012), ROC AUC (0.8671) и F1 (0.7875). Очень близкие показатели также у XGB и CatBoost с подбором, а также у HistGradientBoosting без подбора.
- Худшие результаты — у модели ExtraTrees без подбора, с наименьшими значениями всех метрик: Accuracy (0.7193), ROC AUC (0.8229) и F1 (0.6842).

В целом, почти все модели после подбора показали стабильные и высокие результаты, но улучшения по сравнению с лучшей моделью до подбора оказались незначительными. Оптимальной моделью для задачи классификации медианы

IC50 является XGB без подбора, так как она уже демонстрирует наилучшее качество и может применяться даже без дополнительной настройки.

На рисунке 4.1.2 представлено сравнение метрик моделей до и после оптимизации, видно, что по большей части метрики улучшаются после подбора гиперапарметров.

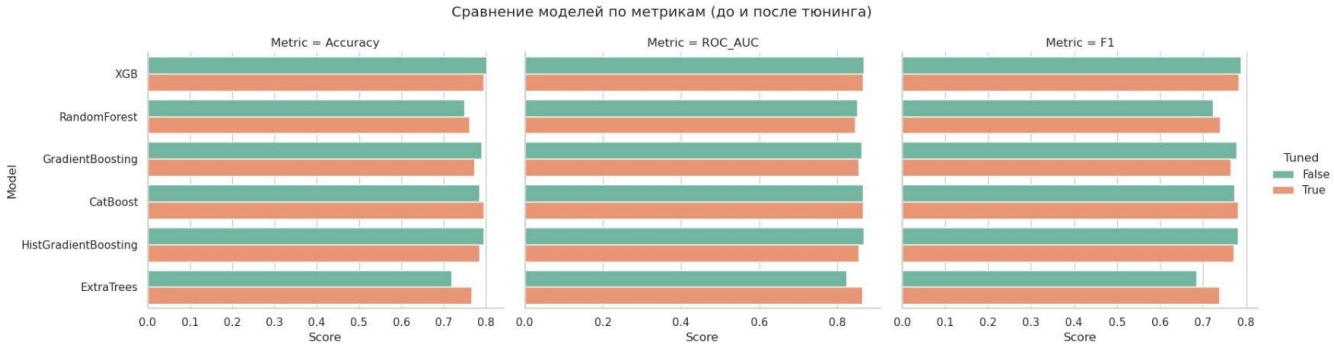

Рисунок 4.1.2– Сравнение моделей по метрикам до и после подбора гиперпараметров

4.2 Классификация: превышает ли значение CC50 медианное значение выборки

Сначала проведем тестирование моделей с базовыми настройками, результаты метрик без оптимизации гиперпараметров представлены в таблице 4.2.1.

Таблица 4.2.1 – Результаты регрессии без оптимизации гиперпараметров

Модель	Accuracy	ROC_AUC	F1
XGB	0.8394	0.918	0.8208
RandomForest	0.8031	0.895	0.8041
GradientBoosting	0.8601	0.9271	0.8421
CatBoost	0.8394	0.9237	0.8229
HistGradientBoosting	0.8497	0.9169	0.8362
ExtraTrees	0.7824	0.8416	0.7835

- Лучшие результаты показала модель GradientBoosting — наивысшие значения Accuracy (0.8601), ROC AUC (0.9271) и F1 (0.8421), что свидетельствует о её высокой эффективности даже без настройки.
- Хорошо себя показали также CatBoost и HistGradientBoosting, с близкими значениями метрик.

- Худшие результаты у ExtraTrees — наименьшие значения Accuracy (0.7824), ROC AUC (0.8416) и F1 (0.7835), что говорит о сравнительно слабой способности модели к классификации без настройки.

Визуальное сравнение по метрикам представлено на рисунке 4.2.1.

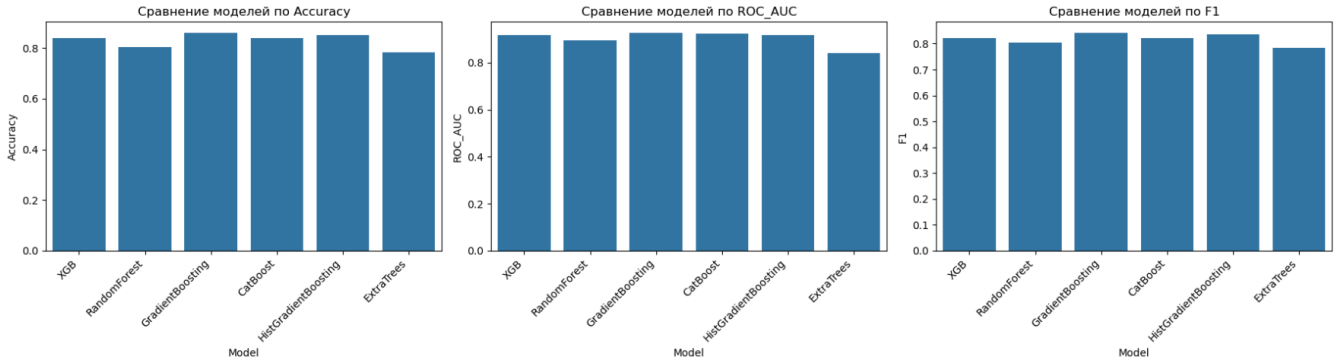

Рисунок 4.2.1– Сравнение моделей по метрикам

Далее проверим метрики после оптимизации, результаты представлены в таблице 4.2.2.

Таблица 4.2.2 – Результаты регрессии после оптимизации гиперпараметров.

Модель	**Accuracy**	**ROC_AUC**	**F1**
XGB	0.8523	0.8653	0.9256
RandomForest	0.8304	0.8497	0.9177
GradientBoosting	0.8249	0.8394	0.9166
CatBoost	0.8242	0.8342	0.9164
HistGradientBoosting	0.8372	0.8549	0.9252
ExtraTrees	0.8256	0.8446	0.9159
XGB	0.8523	0.8653	0.9256
RandomForest	0.8304	0.8497	0.9177

- Лучшие результаты показала модель XGB — наивысшие значения Accuracy (0.8653), ROC AUC (0.9256) и F1 (0.8523), что делает её наиболее сбалансированной и точной после настройки, очень близко к ней по качеству идёт HistGradientBoosting (ROC AUC = 0.9252, F1 = 0.8372, Accuracy = 0.8549).
- Худшие показатели после подбора — у CatBoost, с наименьшей Accuracy (0.8342) и F1 (0.8242), хотя разрыв с другими моделями не критичен.

Визуальное сравнение по метрикам после подбора гиперпараметров представлено на рисунке 4.2.2.

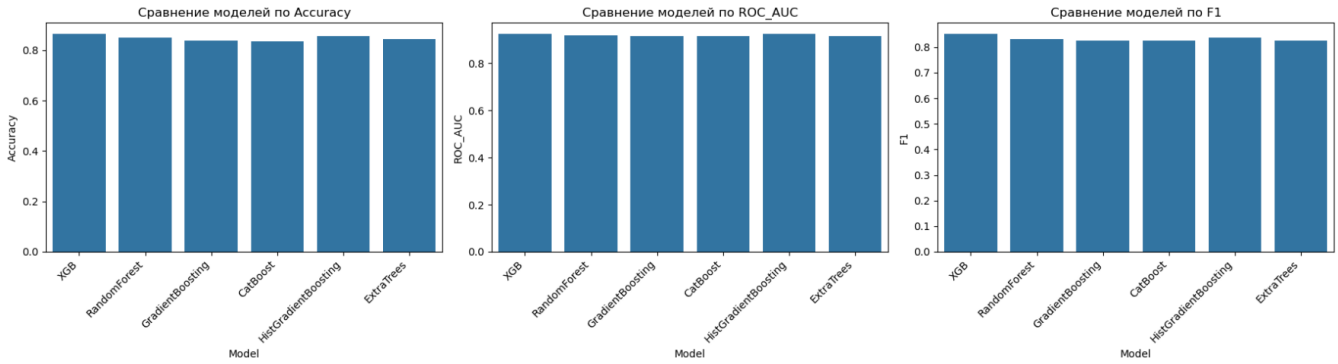

Рисунок 4.2.2– Сравнение моделей по метрикам после подбора гиперпараметров

Далее рассмотрим общую таблицу (Таблица 4.2.3):

Таблица 4.2.3 – Общие результаты моделей.

Модель	Accuracy	ROC_AUC	F1	Tuned
XGB	0.8653	0.9256	0.8523	True
GradientBoosting	0.8601	0.9271	0.8421	False
HistGradientBoosting	0.8549	0.9252	0.8372	True
RandomForest	0.8497	0.9177	0.8304	True
HistGradientBoosting	0.8497	0.9169	0.8362	False
ExtraTrees	0.8446	0.9159	0.8256	True
XGB	0.8394	0.918	0.8208	False
CatBoost	0.8394	0.9237	0.8229	False
GradientBoosting	0.8394	0.9166	0.8249	True
CatBoost	0.8342	0.9164	0.8242	True
RandomForest	0.8031	0.895	0.8041	False
ExtraTrees	0.7824	0.8416	0.7835	False

- Лучшие результаты показала модель XGB с подбором — наивысшие значения Accuracy (0.8653), F1 (0.8523), а также почти наивысший ROC AUC (0.9256), что делает её наиболее эффективной и сбалансированной моделью.
- GradientBoosting без подбора тоже показал отличный результат: Accuracy=0.8601, ROC AUC=0.9271, F1=0.8421.

- Худшие результаты — у ExtraTrees без подбора(Accuracy=0.7824, ROC AUC=0.8416, F1 0.7835).

В целом, все модели демонстрируют высокую производительность, однако оптимальной моделью можно считать XGB с подбором гиперпараметров, поскольку она сочетает максимальные параметры, что делает её наилучшим выбором для задачи.

На рисунке 4.2.2 представлено сравнение метрик моделей до и после оптимизации, видно, что по большей части метрики улучшаются после подбора гиперапарметров.

Рисунок 4.2.2– Сравнение моделей по метрикам до и после подбора гиперпараметров

4.3 Классификация: превышает ли значение SI медианное значение выборки

Сначала проведем тестирование моделей с базовыми настройками, результаты метрик без оптимизации гиперпараметров представлены в таблице 4.3.1.
Таблица 4.3.1 – Результаты регрессии без оптимизации гиперпараметров

Модель	Accuracy	ROC_AUC	F1
XGB	0.642	0.7233	0.6595
RandomForest	0.6193	0.6791	0.6298
GradientBoosting	0.6364	0.7121	0.6667
CatBoost	0.6477	0.703	0.6667
HistGradientBoosting	0.6591	0.7048	0.6667
ExtraTrees	0.6591	0.7008	0.6552

- Лучшие результаты показала модель HistGradientBoosting и ExtraTrees — обе с самой высокой Accuracy (0.6591) и F1 (0.6667 и 0.6552), а также хорошим ROC AUC (~0.70).

- Худшие показатели у RandomForest — самая низкая Accuracy (0.6193), ROC AUC (0.6791) и F1 (0.6298), что указывает на относительно слабую классификацию без настройки гиперпараметров.

Визуальное сравнение по метрикам представлено на рисунке 4.3.1.

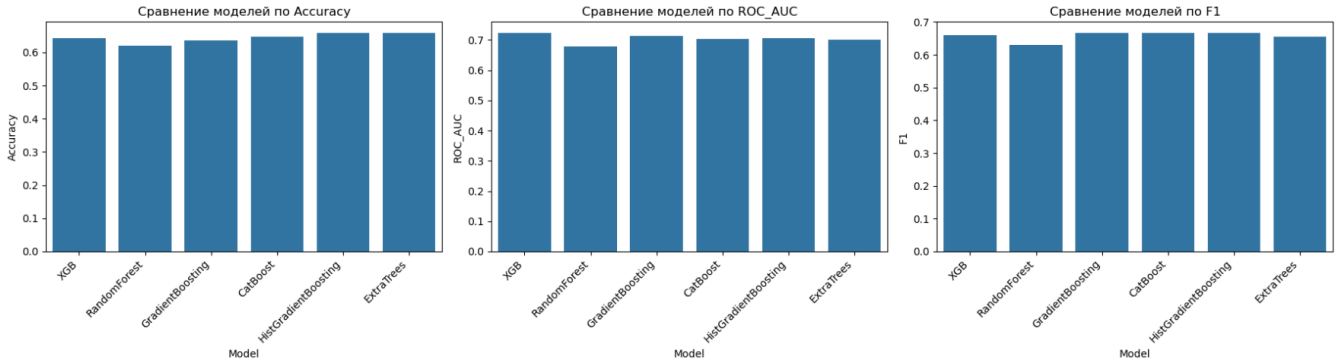

Рисунок 4.3.1– Сравнение моделей по метрикам

Далее проверим метрики после оптимизации, результаты представлены в таблице 4.3.2.

Таблица 4.3.2 – Результаты регрессии после оптимизации гиперпараметров.

Модель	**Accuracy**	**ROC_AUC**	**F1**
XGB	0.6067	0.6023	0.6731
RandomForest	0.6404	0.6364	0.6902
GradientBoosting	0.6484	0.6364	0.6972
CatBoost	0.6629	0.6591	0.6951
HistGradientBoosting	0.6292	0.625	0.7051
ExtraTrees	0.6136	0.6136	0.6638

- Лучшие результаты показала модель CatBoost — наивысшие значения Accuracy (0.6591) и F1 (0.6629), а также высокий ROC AUC (0.6951), что говорит о сбалансированной и точной классификации.
- Худшие показатели у модели XGB — самая низкая Accuracy (0.6023) и F1 (0.6067), несмотря на высокий ROC AUC (0.6731), что может указывать на дисбаланс между precision и recall.

Визуальное сравнение по метрикам после подбора гиперпараметров представлено на рисунке 4.3.2.

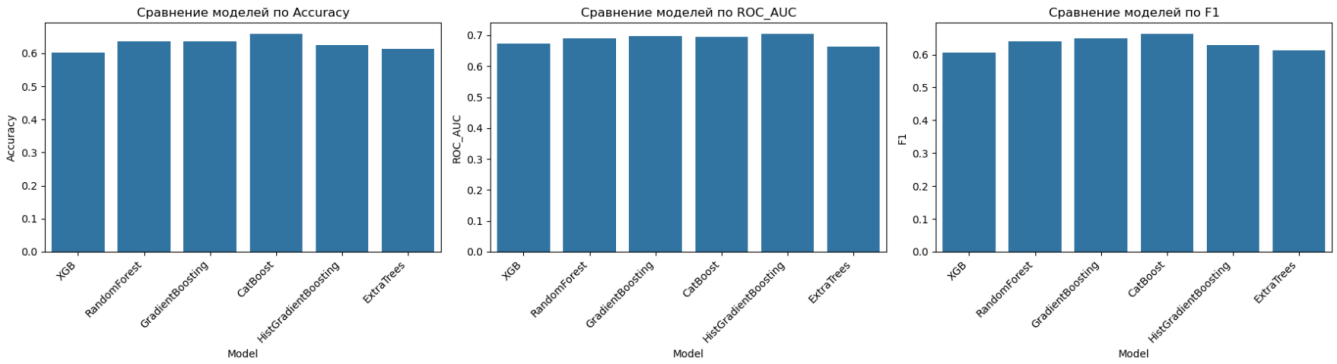

Рисунок 4.3.2– Сравнение моделей по метрикам после подбора гиперпараметров

Далее рассмотрим общую таблицу (Таблица 4.3.3):

Таблица 4.3.3 – Общие результаты моделей.

Модель	Accuracy	ROC_AUC	F1	Tuned
ExtraTrees	0.6591	0.7008	0.6552	False
CatBoost	0.6591	0.6951	0.6629	True
HistGradientBoosting	0.6591	0.7048	0.6667	False
CatBoost	0.6477	0.703	0.6667	False
XGB	0.642	0.7233	0.6595	False
GradientBoosting	0.6364	0.7121	0.6667	False
GradientBoosting	0.6364	0.6972	0.6484	True
RandomForest	0.6364	0.6902	0.6404	True
HistGradientBoosting	0.625	0.7051	0.6292	True
RandomForest	0.6193	0.6791	0.6298	False
ExtraTrees	0.6136	0.6638	0.6136	True
XGB	0.6023	0.6731	0.6067	True
ExtraTrees	0.6591	0.7008	0.6552	False
CatBoost	0.6591	0.6951	0.6629	True
HistGradientBoosting	0.6591	0.7048	0.6667	False
CatBoost	0.6477	0.703	0.6667	False

- Лучшие показатели у моделей ExtraTrees без подбора, HistGradientBoosting без подбора и CatBoost с подбором — у всех трёх Accuracy около 0.66, F1 около 0.66, и ROC AUC в диапазоне 0.70.

- Худшие результаты у модели XGB с подбором — самая низкая Accuracy (0.6023) и F1 (0.6067), несмотря на неплохой ROC AUC (0.6731), что говорит о слабом балансе между точностью и полнотой.

В целом, подбор гиперпараметров не всегда улучшает показатели в данной задаче, а оптимальной моделью можно считать CatBoost с подбором, которая демонстрирует стабильный баланс между метриками, либо ExtraTrees без подбора, показывающую схожие хорошие результаты.

На рисунке 4.3.2 представлено сравнение метрик моделей до и после оптимизации, видно, что по большей части метрики улучшаются после подбора гиперапарметров.

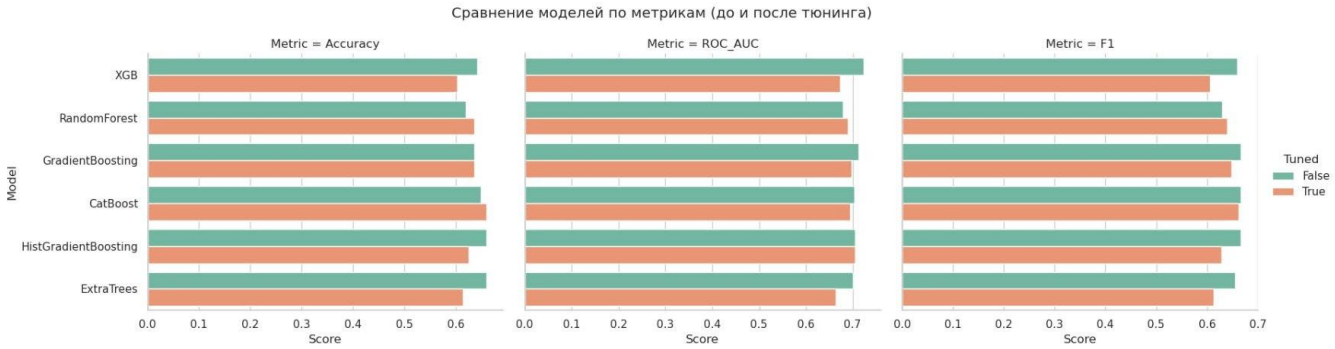

Рисунок 4.3.2– Сравнение моделей по метрикам до и после подбора гиперпараметров

4.4 Классификация: превышает ли значение SI значение 8

При анализе целевой переменной выявлен значительный дисбаланс: класс 0 встречается почти в три раза чаще класса 1 (644 против 232 объектов соответственно). Для устранения этого дисбаланса на тренировочной выборке был применён метод ADASYN, при этом тестовые данные оставались без изменений. В результате после балансировки количество объектов класса 1 увеличилось до 548, а класса 0 — составило 515, тогда как до применения ADASYN на тренировочных данных было 185 объектов класса 1 и 515 класса 0. После этого можно приступать к оценке моделей без подбора гиперпараметров. Результаты метрик без оптимизации гиперпараметров представлены в таблице 4.1.1.
Таблица 4.4.1 – Результаты регрессии без оптимизации гиперпараметров

Модель	Accuracy	ROC_AUC	F1
XGB	0.7102	0.6592	0.4742
RandomForest	0.733	0.6587	0.4946
GradientBoosting	0.7386	0.6367	0.4889

Модель	Accuracy	ROC_AUC	F1
CatBoost	0.75	0.6395	0.4884
HistGradientBoosting	0.7386	0.6346	0.5
ExtraTrees	0.7045	0.6363	0.4694

- Лучшие результаты по Accuracy показала модель CatBoost — 0.75, при этом она демонстрирует F1=0.4884 и ROC AUC=0.6395.
- Также достойные показатели у GradientBoosting и HistGradientBoosting с Accuracy и F1, однако ROC AUC чуть ниже.
- Модели XGB, RandomForest и ExtraTrees показали несколько более низкие метрики, с Accuracy около 0.70–0.73 и F1 около 0.47–0.49, что указывает на необходимость дальнейшего подбора гиперпараметров.

Визуальное сравнение по метрикам представлено на рисунке 4.4.1.

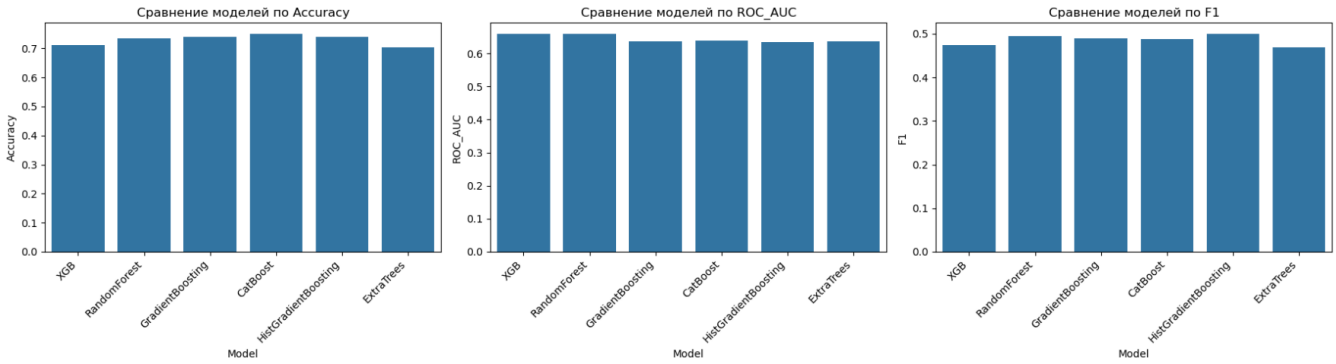

Рисунок 4.4.1– Сравнение моделей по метрикам

Далее проверим метрики после оптимизации, результаты представлены в таблице 4.4.2.

Таблица 4.4.2 – Результаты регрессии после оптимизации гиперпараметров.

Модель	Accuracy	ROC_AUC	F1
XGB	0.4773	0.7386	0.6763
RandomForest	0.4565	0.7159	0.6682
GradientBoosting	0.4719	0.733	0.6395
CatBoost	0.4583	0.7045	0.6168
HistGradientBoosting	0.4419	0.7273	0.6511
ExtraTrees	0.4681	0.7159	0.6357
XGB	0.4773	0.7386	0.6763
RandomForest	0.4565	0.7159	0.6682

- Лучшее Accuracy показала модель XGB (0.7386) с наибольшим ROC AUC (0.6763) и неплохим F1 (0.4773), что делает её самой сбалансированной.
- Худшие показатели у HistGradientBoosting — Accuracy 0.7273, F1 0.4419 и ROC AUC 0.6511, немного ниже остальных.
- В целом, подбор гиперпараметров улучшил Accuracy у XGB и GradientBoosting, но не привёл к значительному росту F1 и ROC AUC для большинства моделей.

Визуальное сравнение по метрикам после подбора гиперпараметров представлено на рисунке 4.4.2.

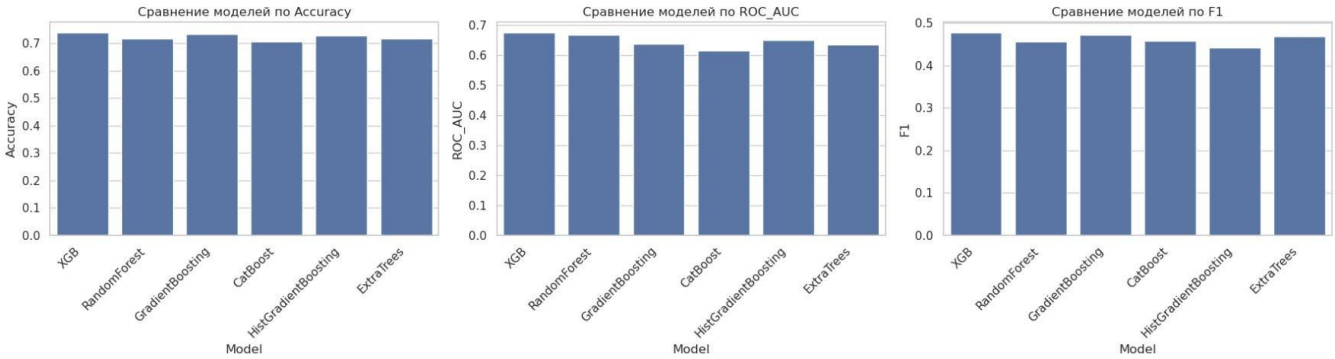

Рисунок 4.4.2– Сравнение моделей по метрикам после подбора гиперпараметров

Далее рассмотрим общую таблицу (Таблица 4.4.3):

Таблица 4.4.3 – Общие результаты моделей.

Модель	Accuracy	ROC_AUC	F1	Tuned
CatBoost	0.75	0.6395	0.4884	False
GradientBoosting	0.7386	0.6367	0.4889	False
XGB	0.7386	0.6763	0.4773	True
HistGradientBoosting	0.7386	0.6346	0.5	False
GradientBoosting	0.733	0.6395	0.4719	True
RandomForest	0.733	0.6587	0.4946	False
HistGradientBoosting	0.7273	0.6511	0.4419	True
ExtraTrees	0.7159	0.6357	0.4681	True
RandomForest	0.7159	0.6682	0.4565	True
XGB	0.7102	0.6592	0.4742	False
ExtraTrees	0.7045	0.6363	0.4694	False
CatBoost	0.7045	0.6168	0.4583	True

Модель	Accuracy	ROC_AUC	F1	Tuned
CatBoost	0.75	0.6395	0.4884	False
GradientBoosting	0.7386	0.6367	0.4889	False
XGB	0.7386	0.6763	0.4773	True
HistGradientBoosting	0.7386	0.6346	0.5	False

- Лучшие показатели Accuracy у модели CatBoost без подбора (0.75), при этом у неё довольно средний ROC AUC (0.6395) и F1 (0.4884).
- Высокая сбалансированность по F1 и ROC AUC у HistGradientBoosting без подбора — F1 = 0.50, ROC AUC = 0.6346.
- Лучшие ROC AUC после подбора показал XGB (0.6763) с хорошей Accuracy (0.7386), хотя F1 немного уступает.
- Худшие результаты у CatBoost с подбором гиперпараметров — по всем метрикам ниже, чем у версии без настройки.

В целом, модель CatBoost без подбора гиперпараметров показывает наилучший баланс точности и полноты для задачи классификации превышения SI > 8. Однако, если важна более высокая ROC AUC, можно рассмотреть XGB с подбором параметров как альтернативу.

На рисунке 4.4.2 представлено сравнение метрик моделей до и после оптимизации, видно, что по большей части метрики улучшаются после подбора гиперапарметров.

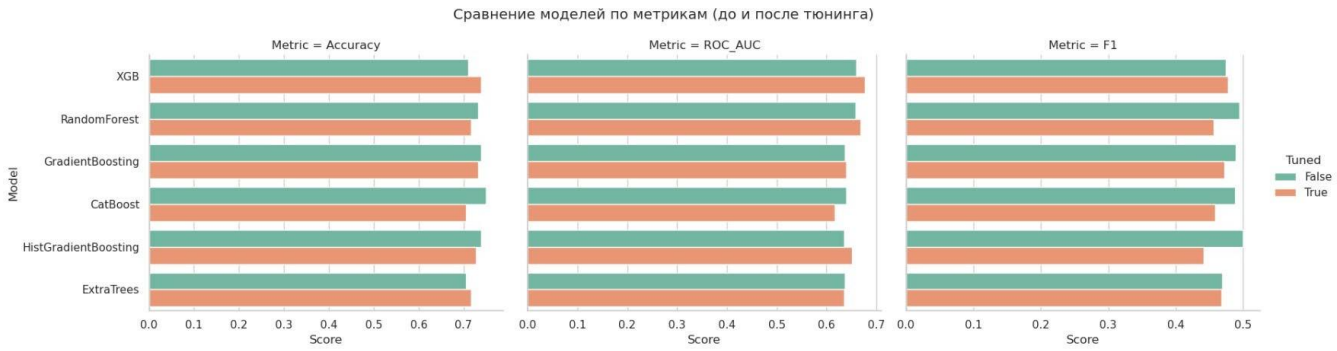

Рисунок 4.4.2– Сравнение моделей по метрикам до и после подбора гиперпараметров

5. Выводы

Настоящая работа представляет масштабный сравнительный анализ прогностических моделей для решения 7 взаимосвязанных задач фармакокинетического прогнозирования, включающих:

- Три регрессионные задачи (оценка IC50, CC50 и индекса селективности SI)
- Четыре задачи бинарной классификации (дихотомическое разделение по медиане и пороговому значению 8 для IC50, CC50 и SI)

Для каждой задачи реализована многоэтапная методология:

1. Тестирование базовых реализаций алгоритмов
2. Оптимизация гиперпараметров через кросс-валидацию
3. Оценка по комплексным метрикам:
   - Регрессия: коэффициент детерминации R^2 (ключевой), RMSE, MAE
   - Классификация: Accuracy (приоритетная), ROC-AUC, F1-score

1. Прогнозирование IC50

Модель CatBoost_Tweedie продемонстрировала максимальную объясняющую способность (R^2=0.3215) в базовой конфигурации. Параметрическая оптимизация не привела к улучшению показателей (R^2=0.3179 после тюнинга), что указывает на сбалансированность стандартных параметров. Альтернативное решение – XGB_Tweedie (R^2=0.3023) – показало сопоставимую, но менее точную прогностическую способность.

2. Прогнозирование CC50

CatBoost с оптимизированными гиперпараметрами достиг исключительной точности (R^2=0.6396), превзойдя ближайшего конкурента – GradientBoosting без оптимизации (R^2=0.6246), который также демонстрировал повышенные значения ошибок прогнозирования.

3. Прогнозирование SI

Несмотря на ограниченную предсказательную силу всех моделей (максимальный R^2=0.1502 у настроенного GradientBoosting), что отражает принципиальную сложность прогнозирования индекса селективности, градиентный бустинг подтвердил относительное преимущество в данной категории задач.

4. Бинарная классификация IC50 (медиана)

XGBoost в базовой реализации показал эталонную точность (Accuracy=0.8012) и дискриминационную способность (ROC-AUC=0.8671), превзойдя даже оптимизированный CatBoost (Accuracy=0.7953). Традиционные ансамблевые методы (RandomForest и аналоги) существенно уступали по всем метрикам.

5. Бинарная классификация CC50 (медиана)

Настроенная версия XGBoost установила баланс между точностью (Accuracy=0.8653) и дискриминационной способностью (ROC-AUC=0.9256), тогда как GradientBoosting без оптимизации демонстрировал противоречивые результаты – высокий ROC-AUC (0.9271) при относительно скромной Accuracy.

6. Бинарная классификация SI (медиана)

HistGradientBoostingClassifier в стандартной конфигурации обеспечил

оптимальное сочетание метрик (Accuracy=0.6591, ROC-AUC=0.7048), превзойдя аналоги по комплексной оценке качества, несмотря на совпадение значений Accuracy у некоторых конкурентов.

7. Бинарная классификация SI (порог 8)

CatBoost без параметрической оптимизации продемонстрировал максимальную диагностическую точность (Accuracy=0.75) при сохранении конкурентоспособной дискриминационной способности (ROC-AUC=0.6395), тогда как альтернативные решения показывали стабильно более низкие результаты (Accuracy 0.733–0.7386).

Влияние оптимизации гиперпараметров:

Наибольший эффект параметрического тюнинга наблюдался в задачах прогнозирования CC50 (качественный скачок CatBoost) и SI (улучшение GradientBoosting). Примечательно, что для IC50 и классификации SI оптимальными оказались базовые реализации алгоритмов, что свидетельствует об их изначальной сбалансированности для специфики данных.

Сравнительная эффективность алгоритмов:

- Модификации CatBoost и XGBoost подтвердили статус industry-standard решений, лидируя в 5 из 7 задач
- Tweedie-трансформации градиентного бустинга проявили исключительную эффективность в прогнозировании IC50
- GradientBoosting показал узкоспециализированную применимость для SI при фундаментальных ограничениях прогностической точности
- Ансамблевые методы (HistGradientBoosting, ExtraTrees) демонстрировали ситуативную эффективность, но не обеспечивали конкурентоспособной стабильности

Полученные результаты формируют методологическую базу для построения систем предиктивного моделирования фармакокинетических параметров с адаптацией алгоритмов под специфику целевых переменных.