**Отчет по анализу и прогнозированию эффективности лекарственных препаратов.**

**Введение**

Настоящий отчет представляет собой результаты комплексного анализа и прогнозирования эффективности лекарственных препаратов на основе химических данных. Основная цель проекта заключалась в разработке и тестировании различных моделей машинного обучения для идентификации наиболее эффективных подходов к прогнозированию ключевых биологических показателей, способствующих созданию оптимальных лекарственных средств. Отчет подготовлен с уклоном на модели машинного обучения и обработку данных, поэтому глубокого погружения в химическую область не представлено.

В рамках проекта были решены следующие задачи:

* Регрессия: Прогнозирование значений IC50​ (полумаксимальная ингибирующая концентрация), CC50​ (полумаксимальная цитотоксическая концентрация) и SI (индекс селективности, представляющий собой отношение CC50​/IC50).
* Классификация: Определение, превышает ли значение IC50​ медианное значение представленной выборки.
* Классификация: Определение, превышает ли значение CC50​ медианное значение представленной выборки.
* Классификация: Определение, превышает ли значение SI медианное значение представленной выборки.
* Классификация: Определение, превышает ли значение SI пороговое значение 8.

В отчете описана методология обработки данных, процесс отбора признаков, а также представлен анализ и сравнение производительности различных моделей машинного обучения для каждой из поставленных задач. В завершение даны общие выводы и рекомендации для дальнейших исследований.

**2. Обзор методологии и подготовка данных**

Проект следовал стандартным этапам машинного обучения, начиная с исследовательского анализа данных (EDA) и заканчивая оценкой моделей. Особое внимание уделялось предобработке данных и тщательному отбору признаков для обеспечения высокого качества моделей.

**2.1. Исследовательский анализ данных (EDA) и предобработка**

На этапе исследовательского анализа данных (EDA) были выполнены следующие ключевые шаги:

**Изучение структуры данных**: были идентифицированы целевые переменные (IC50​, CC50​, SI) а также получены общие сведения о структуре данных - молекулярных дескрипторов, таких как дескрипторы EState, BCUT2D, VSA, топологические, зарядовые, фрагментные и другие, характеризующие физико-химические свойства молекул.

**Обработка пропущенных значений**: в датасете были обнаружены пропуски в 3 уникальных строках. Две из них (Препараты 78 и 80) имели SI=1.00, что указывает на отсутствие селективности и низкую терапевтическую ценность, поэтому они были удалены. Для Препарата 79 – индекс SI=74.626866 (очень высокое значение) делало его потенциально важным примером. Препарат был оставлен, пропущенные значения были заполнены путем импутации (KNN Imputer).

**Удаление дубликатов**: обнаруженные полностью идентичные дубликаты строк были удалены для обеспечения корректности обучения моделей и предотвращения переобучения.

**Корреляционный анализ целевых переменных с признаками**: были рассчитаны коэффициенты корреляции Пирсона между целевыми переменными (IC50​, СС50 и SI) и а также проведен корреляционный анализ на мультиколениарность в признаках.

Для IC50​ выявлены отрицательные корреляции с рядом признаков, что логично (чем выше значение признака, тем ниже IC50​, т.е. выше эффективность). К таким признакам относятся дескрипторы, связанные с распределением зарядов, липофильностью, размером, формой и сложностью молекулы, а также наличием определенных циклических структур.

CC50 взяты положительное значение корреляции, которое означает, что чем выше значение признака, тем выше CC50, а это означает меньшую токсичность.

Среди значимых признаков наибольшую корреляцию с целевой переменной показали признаки, которые соотносятся с такими признаками как плотность, форма и размер (более компактные и менее разветвлённые), умеренность зарядов и липофильности, лекарствоподобность и синтетическая доступность (легко синтезируемые, чаще всего более безопасны), влияние функциональных групп (присутствие определенных структурных элементов).

На этом этапе уже были выявлены противоречия - между Эффективностью (IC50) и Безопасностью (CC50), так как повышение концентрации одних элементов приводит к снижению других. Разработка лекарственных препаратов — это всегда поиск компромисса, поскольку свойства, которые повышают эффективность (желательно низкий IC50), часто могут негативно влиять на безопасность (желательно высокий CC50), и наоборот. В представленных данных это хорошо видно.

Для SI (индекс селективности) наблюдались очень низкие положительные корреляции со всеми признаками (максимум около 0.17). Это указывает на то, что текущий набор молекулярных дескрипторов плохо объясняет вариативность SI, что впоследствии подтвердилось в результатах регрессии и в целом построении моделей машинного обучения. Наиболее коррелирующие признаки для SI были связаны с компактностью, гибкостью, способностью образовывать водородные связи, а также показателями "лекарствоподобности" (qed).

**Анализ мультиколлинеарности среди признаков**: для выявления сильных взаимосвязей между самими признаками были рассчитаны коэффициенты корреляции как Пирсона (для линейных зависимостей), так и Спирмена (для монотонных зависимостей). Обнаружено большое количество высококоррелированных пар признаков (коэффициент корреляции ≥0.9 или ≤−0.9).

Для уменьшения избыточности и потенциального влияния на стабильность моделей, из каждой такой пары был удален один признак. Приоритет отдавался удалению признака, который не имел сильной корреляции с целевыми переменными. Сначала удаление проводилось на основе коэффициента Пирсона, затем — Спирмена.

Удаление признаков с низкой вариативностью: признаки, которые фактически являлись константами или имели крайне низкую вариативность (почти константы), были удалены из датасета, поскольку они не несут полезной информации для обучения моделей.

**Трансформация целевых переменных**: изначально распределения целевых переменных (IC50​, CC50​, SI) были скошены вправо (отличны от нормального). Для приведения их к более симметричному и близкому к нормальному распределению, было применено логарифмирование по основанию 10 (np.log10). Это помогает моделям лучше улавливать закономерности. Для обратного преобразования предсказаний в исходный масштаб необходимо использовать 10ypred (прим. в работе не использовался).​

**2.2. Отбор и трансформация признаков для моделирования**

После первичной предобработки и трансформации целевых переменных, был проведен более детальный отбор признаков, направленный на формирование оптимального набора для каждой задачи:

**Отбор признаков с помощью Lasso:**

Для каждой из трех логарифмированных целевых переменных (log\_IC50, log\_CC50​, log\_SI) был применен метод Lasso (L1-регуляризация). Lasso, помимо снижения размерности, способствует отбору наиболее значимых признаков, обнуляя коэффициенты менее важных. В результате этого этапа было отобрано 35 признаков, которые были общими для всех трех целевых переменных.

С целью отбора нелинейно зависимых признаков были отобраны признаки с помощью Random Forest. Были отобраны топ-50 наиболее важных признаков для каждой целевой переменной, и из них были выделены 39 признаков, которые были общими для всех трех целевых переменных.

**Объединение признаков:**

Отобранные признаки из Lasso (35) и Random Forest (39) были объединены. Поскольку между ними было пересечение, итоговое количество уникальных признаков составило 68. Эти 68 признаков стали основой для обучения всех последующих моделей.

**Трансформация отобранных признаков (Yeo-Johnson):**

После формирования финального набора из 68 признаков, их распределение также было проанализировано. Многие из них не имели нормального распределения, а некоторые были бинарными или дискретными. Было принято решение применить преобразование Йео-Джонсона (Yeo-Johnson transformation) ко всем 68 признакам. Этот метод позволяет привести данные к более нормальному распределению, сохраняя при этом информацию о выбросах и не требуя положительных значений, что делает его более гибким подходом в преобразовании данных для обучения моделей машинного обучения. Применение Yeo-Johnson помогает моделям лучше улавливать зависимости в данных, особенно в случае нелинейных взаимосвязей.

**Финальный датасет:**

В итоге был сформирован конечный датасет, состоящий из 68 логарифмированных с помощью Yeo-Johnson признаков и трех логарифмированных целевых переменных. На этом этапе подготовка данных была завершена, и датасет был готов к обучению моделей.

**2.3. Используемые модели и метрики**

Для решения задач регрессии и классификации были выбраны следующие алгоритмы машинного обучения:

**Регрессионные модели**:

* Линейная Регрессия
* Random Forest Регрессор
* LightGBM Регрессор
* XGBoost Регрессор
* CatBoost Регрессор
* Простая Нейронная Сеть

**Классификационные модели**:

* Логистическая Регрессия
* Random Forest Классификатор
* LightGBM Классификатор
* XGBoost Классификатор
* CatBoost Классификатор
* Простая Нейронная Сеть

**Для оценки** производительности моделей использовались следующие метрики:

**Регрессия:**

* **R2** (коэффициент детерминации): Показывает долю дисперсии целевой переменной, объясняемой моделью. Чем ближе к 1, тем лучше.
* **MSE** (среднеквадратичная ошибка): Средняя квадратичная разница между предсказанными и истинными значениями. Чем ниже, тем лучше.
* **MAE** (средняя абсолютная ошибка): Средняя абсолютная разница между предсказанными и истинными значениями. Чем ниже, тем лучше.

**Классификация:**

* **ROC AUC** (Area Under the Receiver Operating Characteristic Curve): Показывает способность модели различать классы. Чем ближе к 1, тем лучше.
* **Accuracy (точность)**: Доля правильно классифицированных образцов.
* **F1-score**: Гармоническое среднее Precision и Recall. Особенно полезен при несбалансированных классах, поскольку учитывает как ложноположительные, так и ложноотрицательные предсказания.
* **Precision (точность):** Доля истинно положительных предсказаний из всех положительных предсказаний.
* **Recall (полнота)**: Доля истинно положительных предсказаний из всех фактических положительных образцов.

**Гиперпараметры** для каждой модели были оптимизированы с использованием методов кросс-валидации (GridSearchCV).

**3. Результаты моделирования и анализ**

**3.1. Анализ результатов для регрессии logI​C50​, мМ**

|  |  |  |  |
| --- | --- | --- | --- |
| Модель | Лучший R2 (кросс-вал) | R2 на тестовой выборке | MAE на тестовой выборке |
| Линейная Регрессия | 0.2813 | 0.2753 | 0.6009 |
| Random Forest Регрессор | 0.4293 | 0.4364 | 0.5325 |
| LightGBM Регрессор | 0.4253 | 0.3758 | 0.5697 |
| XGBoost Регрессор | 0.4438 | 0.4559 | 0.5267 |
| CatBoost Регрессор | 0.4530 | 0.4266 | 0.5395 |
| Простая Нейронная Сеть | 0.3641 | 0.2775 | 0.5947 |

**Вывод**: XGBoost Regressor показал наилучшие результаты для прогнозирования logI​C50​ с самым высоким R2 на тестовой выборке (0.4559) и самым низким MAE. Это указывает на его высокую эффективность в улавливании нелинейных зависимостей. CatBoost и Random Forest также продемонстрировали хорошие, но несколько уступающие результаты. Линейная регрессия и простая нейронная сеть показали значительно худшую производительность, что подтверждает сложный, нелинейный характер взаимосвязей в данных.

**3.2. Анализ результатов для регрессии logC​C50​, мМ**

|  |  |  |  |
| --- | --- | --- | --- |
| Модель | Лучший R2 (кросс-вал) | R2 на тестовой выборке | MAE на тестовой выборке |
| Линейная Регрессия | 0.258 | 0.2913 | 0.4699 |
| Random Forest Регрессор | 0.4011 | 0.4369 | 0.4095 |
| LightGBM Регрессор | 0.3936 | 0.4228 | 0.4282 |
| XGBoost Регрессор | 0.4013 | 0.4154 | 0.4099 |
| CatBoost Регрессор | 0.4085 | 0.4434 | 0.3986 |
| Простая Нейронная Сеть | 0.2397 | 0.3346 | 0.4196 |

**Вывод**: CatBoost Regressor является лучшей моделью для прогнозирования logC​C50​, достигая наивысшего R2 на тестовой выборке (0.4434) и самого низкого MAE. Random Forest также показал конкурентные результаты. Градиентные бустинговые модели в целом значительно превзошли линейную регрессию и простую нейронную сеть.

**3.3. Анализ результатов для регрессии logS​I**

|  |  |  |  |
| --- | --- | --- | --- |
| Модель | Лучший R2 (кросс-вал) | R2 на тестовой выборке | MAE на тестовой выборке |
| Линейная Регрессия | 0.0681 | 0.0015 | 0.5716 |
| Random Forest Регрессор | 0.2040 | 0.1348 | 0.5183 |
| LightGBM Регрессор | 0.1792 | 0.0880 | 0.5227 |
| XGBoost Регрессор | 0.2142 | 0.1409 | 0.5244 |
| CatBoost Регрессор | 0.2160 | 0.1468 | 0.5176 |
| Простая Нейронная Сеть | 0.1905 | -0.0584 | 0.5545 |

**Вывод**: Результаты регрессии для logS​I оказались крайне слабыми для всех моделей. Даже лучшая модель, CatBoost Regressor, достигла R2 на тестовой выборке всего лишь 0.1468. Отрицательный R2 у нейронной сети указывает на ее полную неспособность к прогнозированию. Это подтверждает выводы корреляционного анализа из EDA: текущий набор признаков очень плохо объясняет дисперсию logS​I. MAE в 0.5176 (для CatBoost) в логарифмическом масштабе соответствует очень большому разбросу в исходном масштабе (например, истинное SI=100 может быть предсказано как 30 или 330), что делает предсказания непрактичными.

**3.4. Анализ результатов для классификации "IC50 выше медианы"**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Модель | Accuracy | F1-score | ROC AUC | Precision (Класс 0) | Recall (Класс 0) | F1-score (Класс 0) | Precision (Класс 1) | Recall (Класс 1) | F1-score (Класс 1) |
| Логистическая Регрессия | 0.6959 | 0.7122 | 0.7668 | 0.72 | 0.64 | 0.68 | 0.68 | 0.75 | 0.71 |
| Random Forest | 0.6856 | 0.6935 | 0.7881 | 0.70 | 0.66 | 0.68 | 0.68 | 0.71 | 0.69 |
| LightGBM | 0.6753 | 0.6897 | 0.7664 | 0.69 | 0.63 | 0.66 | 0.66 | 0.72 | 0.69 |
| XGBoost | 0.6856 | 0.6995 | 0.7906 | 0.70 | 0.64 | 0.67 | 0.67 | 0.73 | 0.70 |
| CatBoost | 0.7165 | 0.7264 | 0.8074 | 0.73 | 0.68 | 0.71 | 0.70 | 0.75 | 0.73 |
| Простая Нейронная Сеть | 0.7165 | 0.7236 | 0.7948 | 0.73 | 0.69 | 0.71 | 0.71 | 0.74 | 0.72 |

**Вывод:** CatBoost и Простая Нейронная Сеть показали наилучшие результаты для этой классификационной задачи, демонстрируя высокие и очень близкие значения ROC AUC (около 0.80), Accuracy (0.7165) и F1-score (около 0.72). Эти модели хорошо сбалансированы по Precision и Recall для обоих классов, что говорит об их эффективности в предсказании, превышает ли IC50​ медиану.

**3.5. Анализ результатов для классификации "CC50 выше медианы"**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Модель | Accuracy | F1-score | ROC AUC | Precision (класс 0) | Recall (класс 0) | F1-score (класс 0) | Precision (класс 1) | Recall (класс 1) | F1-score (класс 1) |
| Логистическая Регрессия | 0.7320 | 0.7400 | 0.7803 | 0.75 | 0.70 | 0.72 | 0.72 | 0.76 | 0.74 |
| Random Forest | 0.7268 | 0.7282 | 0.8101 | 0.73 | 0.72 | 0.73 | 0.72 | 0.73 | 0.73 |
| LightGBM | 0.7371 | 0.7385 | 0.8210 | 0.74 | 0.73 | 0.74 | 0.73 | 0.74 | 0.74 |
| XGBoost | 0.7423 | 0.7525 | 0.8203 | 0.76 | 0.70 | 0.73 | 0.72 | 0.78 | 0.75 |
| CatBoost | 0.7371 | 0.7463 | 0.8174 | 0.76 | 0.70 | 0.73 | 0.72 | 0.77 | 0.75 |
| Простая Нейронная Сеть | 0.7216 | 0.7273 | 0.8194 | 0.73 | 0.70 | 0.72 | 0.71 | 0.74 | 0.73 |

**Вывод:** Для этой задачи XGBoost (Accuracy: 0.7423, F1-score: 0.7525), LightGBM (ROC AUC: 0.8210) и CatBoost (F1-score: 0.7463) показали себя наиболее эффективными. Эти бустинговые модели значительно превосходят Логистическую Регрессию и демонстрируют высокую способность к предсказанию, превышает ли CC50​ медиану. Выбор между ними зависит от конкретных приоритетов: XGBoost лидирует по общей точности, а LightGBM по разделительной способности и сбалансированности метрик.

**3.6. Анализ результатов для классификации "SI выше медианы"**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Модель | Accuracy | F1-score | ROC AUC | Precision (класс 0) | Recall (класс 0) | F1-score (класс 0) | Precision (класс 1) | Recall (класс 1) | F1-score (класс 1) |
| Логистическая Регрессия | 0.6701 | 0.6484 | 0.7272 | 0.65 | 0.73 | 0.69 | 0.69 | 0.61 | 0.65 |
| Random Forest | 0.6237 | 0.6138 | 0.6771 | 0.62 | 0.65 | 0.63 | 0.63 | 0.60 | 0.61 |
| LightGBM | 0.6598 | 0.6633 | 0.6981 | 0.66 | 0.65 | 0.66 | 0.66 | 0.67 | 0.66 |
| XGBoost | 0.6649 | 0.6409 | 0.6947 | 0.65 | 0.73 | 0.69 | 0.69 | 0.60 | 0.64 |
| CatBoost | 0.6546 | 0.6257 | 0.7152 | 0.63 | 0.73 | 0.68 | 0.68 | 0.58 | 0.63 |
| Простая Нейронная Сеть | 0.6443 | 0.6387 | 0.7062 | 0.64 | 0.66 | 0.65 | 0.65 | 0.63 | 0.64 |

**Вывод:** В этой задаче способности моделей классифицировать классы оказались хуже, о чем свидетельствуют более низкие абсолютные значения метрик (ROC AUC в диапазоне 0.67−0.72). Логистическая Регрессия (ROC AUC: 0.7272, Accuracy: 0.6701) и LightGBM (F1-score: 0.6633, сбалансированные Precision/Recall) показали лучшие результаты. Логистическая Регрессия, несмотря на свою простоту, продемонстрировала неожиданно хорошую разделительную способность. Random Forest показал признаки переобучения. Все это совместно с EDA, еще раз подчеркивает проблематику в изначальных данных.

**3.7. Анализ результатов для классификации "SI > 8" (с учетом дисбаланса классов)**

Задача классификации "SI > 8" характеризуется значительным дисбалансом классов, где количество образцов, где SI не превышает 8 (класс 0), значительно больше, чем образцов, где SI превышает 8 (класс 1). Для решения этой проблемы был применен метод взвешивания классов (Class Weighting). Ниже представлены метрики производительности моделей на тестовой выборке.

|  |  |  |  |
| --- | --- | --- | --- |
| Модель | F1-score (класс 1) | ROC AUC | Recall (класс 1) |
| Логистическая Регрессия | 0.58 | 0.6991 | 0.65 |
| Random Forest | 0.60 | 0.7212 | 0.64 |
| LightGBM | 0.59 | 0.7177 | 0.62 |
| XGBoost | 0.59 | 0.7201 | 0.65 |
| CatBoost | 0.58 | 0.7007 | 0.64 |
| Простая Нейронная Сеть | 0.52 | 0.6635 | 0.58 |

**Обоснование выбора наиболее качественных решений для "SI > 8"**

С учетом сильного дисбаланса классов и важности предсказания миноритарного класса (SI > 8), приоритетными метриками для оценки моделей являются **F1-score для класса 1** и **ROC AUC**.

В данном случае **Random Forest** и **XGBoost** продемонстрировали наилучшие показатели **ROC AUC (0.7212 и 0.7201 соответственно)**. Это указывает на их лучшую способность различать классы при различных порогах.

**Логистическая Регрессия** также показала конкурентоспособный ROC AUC (0.6991) и при этом имеет F1-score для класса 1, сопоставимый с другими моделями (0.58).

Хотя **CatBoost** (ROC AUC 0.7007) и **LightGBM** (ROC AUC 0.7177) также показали достойные результаты, их F1-score для класса 1 (0.58 и 0.59 соответственно) оказались не выше, чем у Random Forest и XGBoost.

**Простая Нейронная Сеть** показала наихудшие результаты по всем метрикам, включая ROC AUC (0.6635) и F1-score для класса 1 (0.52), что делает ее наименее подходящей для данной задачи.

В целом, несмотря на использование взвешивания классов, предсказание класса "SI > 8" остается сложной задачей, что подтверждается относительно невысокими значениями F1-score для миноритарного класса у всех моделей. Это указывает на необходимость дальнейшего исследования признаков или применения более продвинутых методов для работы с несбалансированными данными.

**4. Общие выводы и рекомендации**

На основе проведенного анализа и сравнения моделей можно сделать следующие ключевые выводы и дать рекомендации:

**Прогнозируемость IC50 и CC50 (регрессия)**

Модели, особенно XGBoost для IC50 и CatBoost для CC50, способны объяснить значительную часть вариабельности этих показателей (R2 ≈ 0.44–0.45). Это указывает на то, что используемые молекулярные дескрипторы имеют предсказательную силу для токсичности и эффективности соединений.

**Слабая предсказательная способность для SI (регрессия)**

Предсказание SI оказалось самой сложной задачей с текущим набором признаков. Очень низкие значения R2 для logSI (даже у лучшей модели CatBoost R2 едва превышает 0.14) указывают на то, что модели практически не улавливают закономерности в этой целевой переменной. Это подтверждается и анализом корреляций, показавшим, что SI имеет очень низкие корреляции со всеми признаками (даже самые сильные едва достигают 0.17). Это значит, что если признаки слабо линейно коррелируют с целевой переменной, то даже самые продвинутые модели могут испытывать трудности с поиском нелинейных зависимостей, если эти зависимости не очень сильны или скрыты в шуме.

**Возможные пути улучшения для SI** (дальнейшее исследование не проводилось)

* Добавление новых, более информативных признаков, специфичных для селективности. Возможно, существуют другие типы дескрипторов (например, 3D-дескрипторы, дескрипторы взаимодействий с белками-мишенями, если известны) или новые методы генерации признаков, которые лучше отражают селективность.
* Использование других подходов к моделированию, возможно, многозадачное обучение (multi-task learning), где модели совместно обучаются предсказывать IC50, CC50 и SI одновременно, чтобы учесть их взаимосвязи.
* Детальный анализ природы SI и его связи с дескрипторами.
* Увеличение объема обучающих данных, особенно для соединений с крайними или "интересными" значениями SI.

**Классификационные задачи**

* **Классификация IC50 выше медианы:** Для задачи определения, превышает ли IC50 медиану, CatBoost и Простая Нейронная Сеть показали наилучшие и очень близкие результаты по ROC AUC, Accuracy и F1-score.
* **Классификация CC50 выше медианы**: Для этой задачи XGBoost, LightGBM и CatBoost показали себя наиболее эффективными. Выбор между ними зависит от приоритетов: XGBoost лидирует по Accuracy и F1-score, LightGBM по ROC AUC и сбалансированным Precision/Recall.
* **Классификация SI выше медианы**: Эта задача оказалась сложнее предыдущих, но Логистическая Регрессия и LightGBM показали наиболее конкурентоспособные результаты по ROC AUC на тестовой выборке.
* **Классификация SI > 8 (с дисбалансом классов)**: Эта задача оказалась самой сложной из всех классификационных, что подтверждается более низкими абсолютными метриками и выраженным дисбалансом классов. Random Forest и XGBoost показали наилучшие результаты по ROC AUC (0.7212 и 0.7201 соответственно), превзойдя остальные модели. Несмотря на использование взвешивания классов, F1-score для миноритарного класса остаётся невысоким, что указывает на сохраняющиеся сложности в точном определении таких образцов.

**Общие наблюдения по моделям**

* Преимущество бустинговых моделей: Градиентный бустинг (XGBoost, CatBoost, LightGBM) показал себя наиболее эффективным подходом для регрессии на данном датасете, а также в большинстве классификационных задач.
* Производительность нейронных сетей: В регрессионных задачах простая нейронная сеть значительно уступает моделям градиентного бустинга. Однако в задаче классификации IC50 она показала себя очень конкурентоспособной. Для дальнейшего улучшения её производительности в регрессии и для более сложных классификационных задач потребуется гораздо более глубокая настройка архитектуры, применение регуляризации и, возможно, гораздо больший объем данных.
* Количество признаков (68): Отобранные 68 признаков оказались вполне адекватными для построения моделей, сохранилась интерпретируемость признаков.

**Окончательный выбор моделей**

На данном этапе у нас есть явные лидеры для каждой задачи, которые могут быть использованы:

* Регрессия logIC50: XGBoost.
* Регрессия logCC50: CatBoost.
* Классификация IC50 выше медианы: CatBoost или Простая Нейронная Сеть.
* Классификация SI выше медианы: Логистическая Регрессия или LightGBM.
* Классификация SI > 8: Random Forest или XGBoost.

**Возможные следующие шаги**

* Изучение важности признаков: для выбранных моделей (XGBoost, CatBoost, Логистическая Регрессия, Random Forest) можно извлечь важность признаков. Это позволит увидеть, какие из 68 химических дескрипторов оказывают наибольшее влияние на предсказание каждой из целевых переменных, что будет очень ценно для химической интерпретации результатов.
* Дальнейшее улучшение: для дальнейшего улучшения потребуется выполнить анализ остатков и дополнительную оптимизацию гиперпараметров, а также произвести feature engineering (например, выделить один элемент, отвечающий за эффективность препарата – например, водород или какой-то другой атом).