Национальный исследовательский ядерный университет «МИФИ»

Тема работы:

Курсовая работа

«Исследование лекарственной активности»

Классическое машинное обучение

выполнила: Студент МИФИ Ревель Р. С.

руководитель: Преподаватель НИЯУ МИФИ и НИТУ МИСИС Егоров А.Д.

Москва

2025

Задачи

Цели работы:

1. Построить прогнозные модели для ключевых параметров эффективности лекарственных препаратов (IC50, CC50, SI) с помощью методов машинного обучения.
2. Сравнить различные алгоритмы (регрессия и классификация) и выбрать наилучшие модели на основе метрик качества.
3. Определить наиболее значимые параметры, влияющие на эффективность препаратов, для оптимизации их состава.

Этапы работы:

1. Предварительный анализ данных

- Анализ распределений целевых переменных (IC50, CC50, SI).
- Исследование корреляций между признаками.
- Группировка и визуализация данных для выявления закономерностей.
- Анализ важности переменных для целевой переменной

2. Пред обработка данных

- Очистка данных (удаление/заполнение пропусков, обработка выбросов).
- Нормализация/стандартизация числовых признаков (если нужно).
- Кодирование категориальных признаков (если есть).
- Разделение данных на обучающую и тестовую выборки.

3. Обучение моделей

Регрессия:

- Обучения моделей
- Подбор гиперпараметров (RandomizedSearchCV).
- Оценка по метрикам: MAE, RMSE, R^2.

Классификация:.

- Оценка по метрикам.
- Интерпретация и сравнение моделей

5. Общий вывод

- Какие комбинации параметров дают лучшие значения IC50/CC50/SI?
- Какие модели оказались наиболее точными и почему?
- Какие дальнейшие шаги можно предложить (сбор дополнительных данных, уточнение признаков и т. д.)?

Описания данных

Описания целевых переменных

1. IC_{50} (Half Maximal Inhibitory Concentration)

- Определение: Концентрация вещества, необходимая для подавления биологического процесса (например, репликации вируса, активности фермента) на 50% по сравнению с контролем.
- Применение:
- В противовирусных исследованиях — показывает, насколько эффективно вещество блокирует вирус.
- Чем меньше IC_{50}, тем выше эффективность соединения.

2. CC_{50} (Half Maximal Cytotoxic Concentration)

- Определение: Концентрация вещества, вызывающая гибель 50% клеток в эксперименте (токсичность).
- Применение:
- Отражает цитотоксичность вещества для здоровых клеток.
- Чем выше CC_{50}, тем безопаснее соединение.

3. SI (Selectivity Index — Индекс Селективности)

- Формула:

$$ SI = \frac{CC_{50}}{IC_{50}} $$

- Смысл: Показывает, насколько вещество избирательно действует на мишень (например, вирус), а не на клетки хозяина.
- SI > 10 — считается приемлемым для потенциальных лекарств.
- SI > 100 — высокая селективность, минимальная токсичность.

Пример интерпретации:

Если у препарата:

- IC_{50} = 1 μM (хорошо подавляет вирус),
- CC_{50} = 100 μM (низкая токсичность),

то SI = 100 — отличный кандидат для дальнейших исследований.
Эти параметры критически важны при скрининге новых лекарств, особенно противовирусных (например, против ВИЧ, SARS-CoV-2).

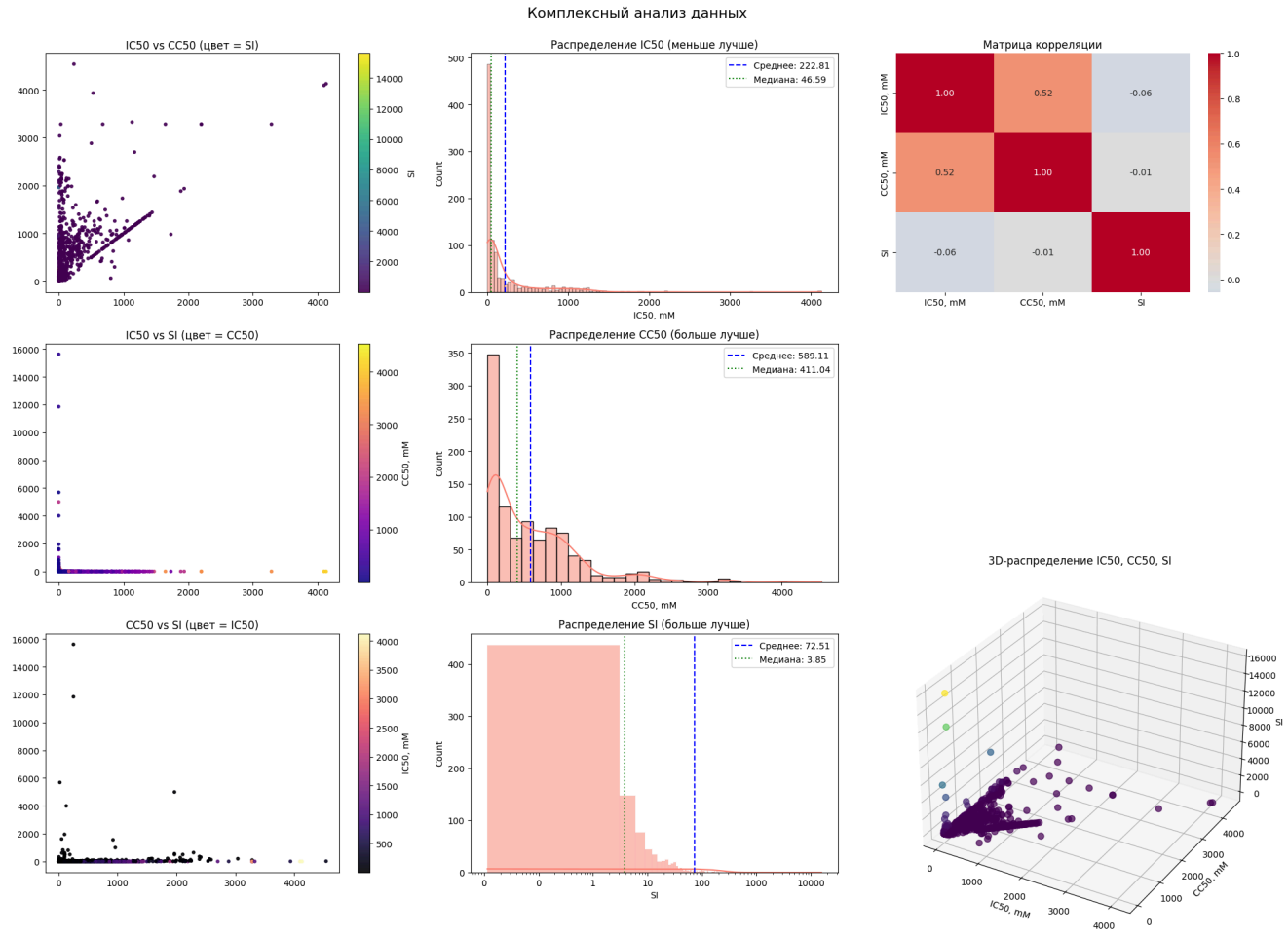
Электронные и энергетические параметры:
• **MaxAbsEStateIndex** — максимальный электроотрицательный индекс состояния по абсолютному значению
• **MaxEStateIndex** — максимальный индекс состояния
• **MinAbsEStateIndex** — минимальный электроотрицательный индекс по абсолютному значению

• **MinEStateIndex** — минимальный индекс состояния
• **MaxPartialCharge** — максимальный частичный заряд атома
• **MinPartialCharge** — минимальный частичный заряд атома
• **MaxAbsPartialCharge** — максимальный частичный заряд (по модулю)
• **MinAbsPartialCharge** — минимальный частичный заряд (по модулю)**Молекулярные дескрипторы:**
• **MolWt** — молекулярная масса
• **HeavyAtomMolWt** — масса без учёта атомов водорода
• **ExactMolWt** — точная молекулярная масса
• **NumValenceElectrons** — количество валентных электронов
• **NumRadicalElectrons** — количество радикальных электронов
• **qed** — Quantitative Estimate of Drug-likeness (оценка качества молекулы как кандидата в лекарства)
• **SPS** — сумма поляризационных поверхностей растворителя
Физико-химические свойства:
• **MolLogP** — коэффициент распределения (оценка липофильности)
• **MolMR** — молярный рефракционный показатель (мера молекулярного объёма и поляризуемости)
Структурные признаки:
• **HeavyAtomCount** — число тяжёлых атомов (все, кроме H)
• **NHOHCount** — число групп OH и NH
• **NOCount** — число атомов N и O
• **NumRotatableBonds** — число ротируемых связей (мера гибкости молекулы)
• **RingCount** — общее число колец
• **FractionCSP3** — доля sp^3-гибридизованных атомов углерода
• **NumAliphaticRings** — число алифатических колец
• **NumAromaticRings** — число ароматических колец
• **NumHAcceptors** — число акцепторов водородных связей
• **NumHDonors** — число доноров водородных связей
• **NumHeteroatoms** — число гетероатомов (не C/H)
Дескрипторы Morgan Fingerprint Density:
• **FpDensityMorgan1**, **FpDensityMorgan2**, **FpDensityMorgan3** — плотность фингерпринтов разного радиуса
BCUT-дескрипторы (атомные свойства):
• **BCUT2D_MWHI**, **BCUT2D_MWLOW** — массовые дескрипторы
• **BCUT2D_CHGHI**, **BCUT2D_CHGLO** — зарядовые дескрипторы
• **BCUT2D_LOGPHI**, **BCUT2D_LOGPLOW** — оценка липофильности• **BCUT2D_MRHI**, **BCUT2D_MRLOW** — оценка молярного рефракционного индекса
Топологические дескрипторы:
• **BalabanJ** — балабановский индекс (топологическая характеристика молекулы)
• **BertzCT** — индекс сложности молекулы (fragment complexity contribution)
• **HallKierAlpha**, **Ipc**, **Kappa1**, **Kappa2**, **Kappa3** — структурные индексы Холла–Кьера
Площадь поверхности доступности (ASA):
• **LabuteASA** — площадь доступной растворителю поверхности
PEOE_VSA — дескрипторы по зарядам:
(PEOE — Partial Equalization of Orbital Electronegativity)
• **PEOE_VSA1–PEOE_VSA14** — разделённые по диапазонам значения атомных зарядов и поляризации
SMR_VSA — молекулярное рефракционное значение по участкам:
• **SMR_VSA1–SMR_VSA10** — молярная рефракция по различным диапазонам

SlogP_VSA — логP по областям молекулы:
• **SlogP_VSA1–SlogP_VSA12** — дескрипторы липофильности по участкам молекулы
TPSA — полярная поверхность:
• **TPSA** — суммарная полярная поверхность (Topological Polar Surface Area)
EState_VSA — электроотрицательность по зонам:
• **EState_VSA1–EState_VSA11**
—
деление молекулы на участки по
электроотрицательности
VSA_EState — вариация EState по размеру:
• **VSA_EState1–VSA_EState9** — деление по электроотрицательности с участием площади поверхности
Часто используемые фрагменты (fr_...):
Функциональные группы и их наличие в молекуле:
• **fr_Al_COO** – аллильная карбоновая группа
• **fr_Al_OH** – спиртовые OH-группы
• **fr_Al_OH_noTert** – OH-группы, за исключением третичных
• **fr_ArN** – ароматические N
• **fr_Ar_COO** – ароматические карбоновые кислоты
• **fr_Ar_N** – ароматические амины• **fr_Ar_NH** – ароматические аминогруппы
• **fr_Ar_OH** – фенольные OH
• **fr_COO** – карбоновые кислоты
• **fr_COO2** – вторая форма карбоновой кислоты
• **fr_C_O** – карбонильные группы
• **fr_C_O_noCOO** – карбонилы, кроме карбоновых
• **fr_C_S** – группы с атомами C=S
• **fr_HOCCN** – цианиды с OH-группой
• **fr_Imine** – имины
• **fr_NH0** – первичные NH-группы
• **fr_NH1** – вторичные NH-группы
• **fr_NH2** – третичные NH-группы
• **fr_N_O** – связи N–O
• **fr_Ndealkylation1**, **fr_Ndealkylation2** – маркеры реакции N-деалкилирования
• **fr_Nhpyrrole** – пиррольные NH-группы
• **fr_SH** – тиольные группы
• **fr_aldehyde** – альдегиды
• **fr_alkyl_carbamate** – карбаматы
• **fr_alkyl_halide** – алкилгалогениды
• **fr_allylic_oxid** – метки для окисления аллильных групп
• **fr_amide** – амиды
• **fr_amidine** – амидины
• **fr_aniline** – анилины
• **fr_aryl_methyl** – арилметильные группы
• **fr_azide** – азида
• **fr_azo** – азо-соединения
• **fr_barbitur** – барбитуровая кислота или её производные
• **fr_benzene** – бензольные кольца
• **fr_benzodiazepine** – бензодиазепиновые структуры
• **fr_bicyclic** – двухкольцевые структуры
• **fr_diazo** – диазосоединения
• **fr_dihydropyridine** – дигидропиридины• **fr_epoxide** – эпоксиды
• **fr_ester** – эфиры

- **fr_ether** – простые эфиры
- **fr_furan** – фурановые кольца
- **fr_guanido** – гуанидиновые группы
- **fr_halogen** – галогены
- **fr_hdrzine** – гидразиновые группы
- **fr_hdrzone** – гидразоны
- **fr_imidazole** – имидазолы
- **fr_imide** – имиды
- **fr_isocyan** – изоцианиды
- **fr_isothiocyan** – изотиоцианиды
- **fr_ketone** – кетоны
- **fr_ketone_Topliss** – кетоны (по Topliss)
- **fr_lactam** – лактамы
- **fr_lactone** – лактоны
- **fr_methoxy** – метокси-группы
- **fr_morpholine** – морфолиновые структуры
- **fr_nitrile** – нитрилы
- **fr_nitro** – нитрогруппы
- **fr_nitro_arom** – нитроароматические соединения
- **fr_nitro_arom_nonortho** – нитроароматические, не орто-замещённые
- **fr_nitroso** – нитрозо-соединения
- **fr_oxazole** – оксазолы
- **fr_oxime** – оксимы
- **fr_para_hydroxylation** – метки для пара-гидроксилирования
- **fr_phenol** – фенольные OH-группы
- **fr_phenol_noOrthoHbond** – фенолы без орто-водородных связей
- **fr_phos_acid** – фосфорные кислоты
- **fr_phos_ester** – фосфорные эфиры
- **fr_piperdine** – пиперидиновые структуры
- **fr_piperzine** – пиперазиновые структуры- **fr_priamide** – первичные амиды
- **fr_prisulfonamd** – сульфонамиды
- **fr_pyridine** – пиридиновые кольца
- **fr_quatN** – четвертичные атомы азота
- **fr_sulfide** – сульфиды
- **fr_sulfonamd** – сульфонамиды
- **fr_sulfone** – сульфоны
- **fr_term_acetylene** – терминальные ацетилены
- **fr_tetrazole** – тетразолы
- **fr_thiazole** – тиазолы
- **fr_thiocyan** – тиоцианаты
- **fr_thiophene** – тиофеновые кольца
- **fr_unbrch_alkane** – неразветвлённые алканы
- **fr_urea** – мочевина и её производны

Предварительный анализ данных



Выводы

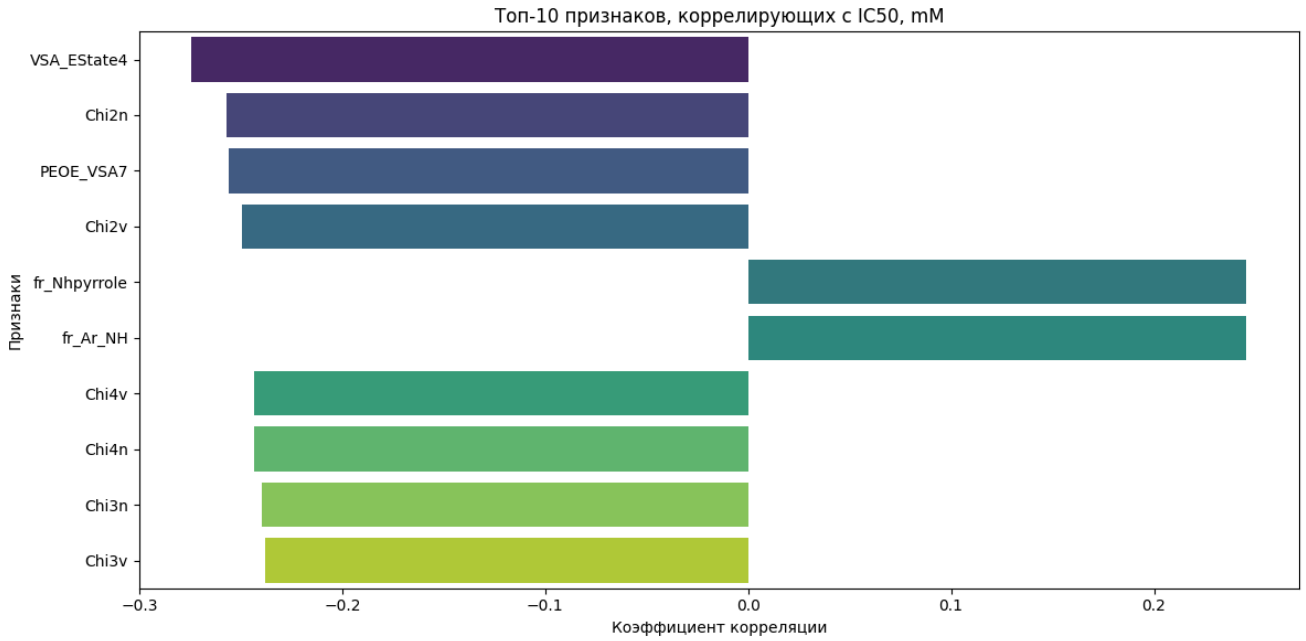
Эффективность vs. Токсичность: Многие соединения эффективны (низкий IC50), но одновременно и токсичны (низкий CC50), что ограничивает их терапевтическую применимость.

Селективность (SI): Большинство соединений имеют низкий SI, что указывает на недостаточную селективность (малое окно между эффективной и токсической дозой). Это критично для разработки безопасных лекарств.

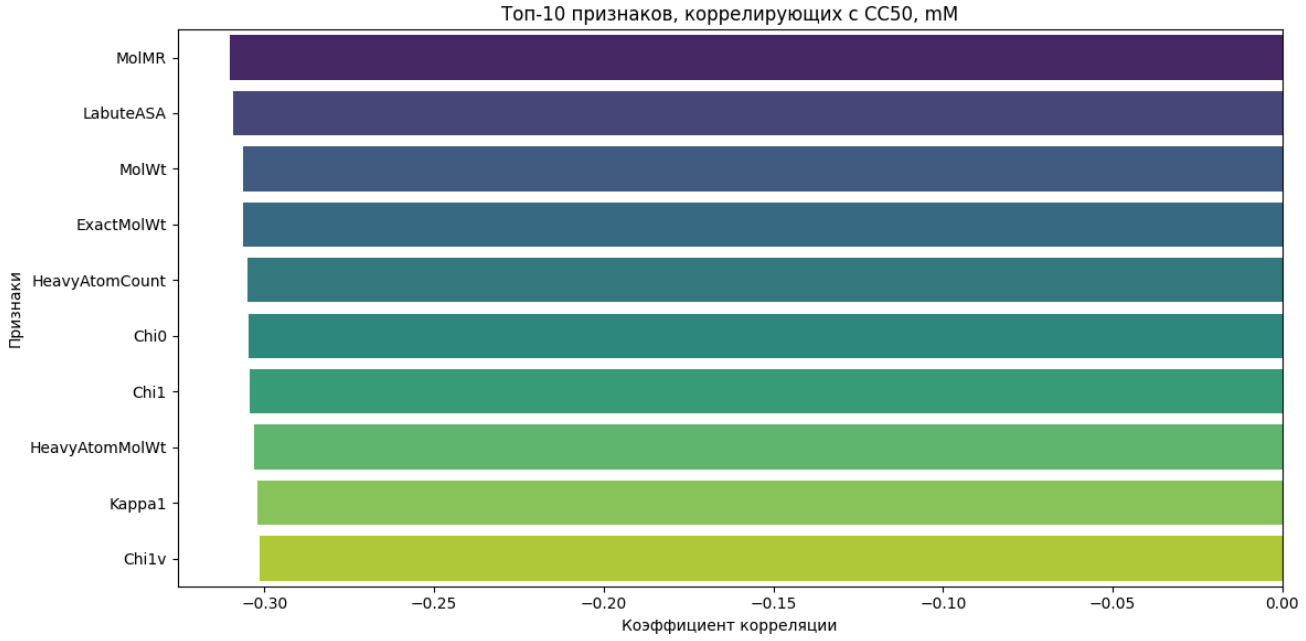
Потенциальные "хиты": Соединения с высоким SI (значительно больше 1) требуют дополнительного изучения, так как они могут быть перспективными кандидатами. Их можно выявить, отфильтровав данные по SI > 10 (если такие есть). И судя по графикам таких значений достаточно мало, что в целом не удивительно

Анализ важности переменных для целевой переменной

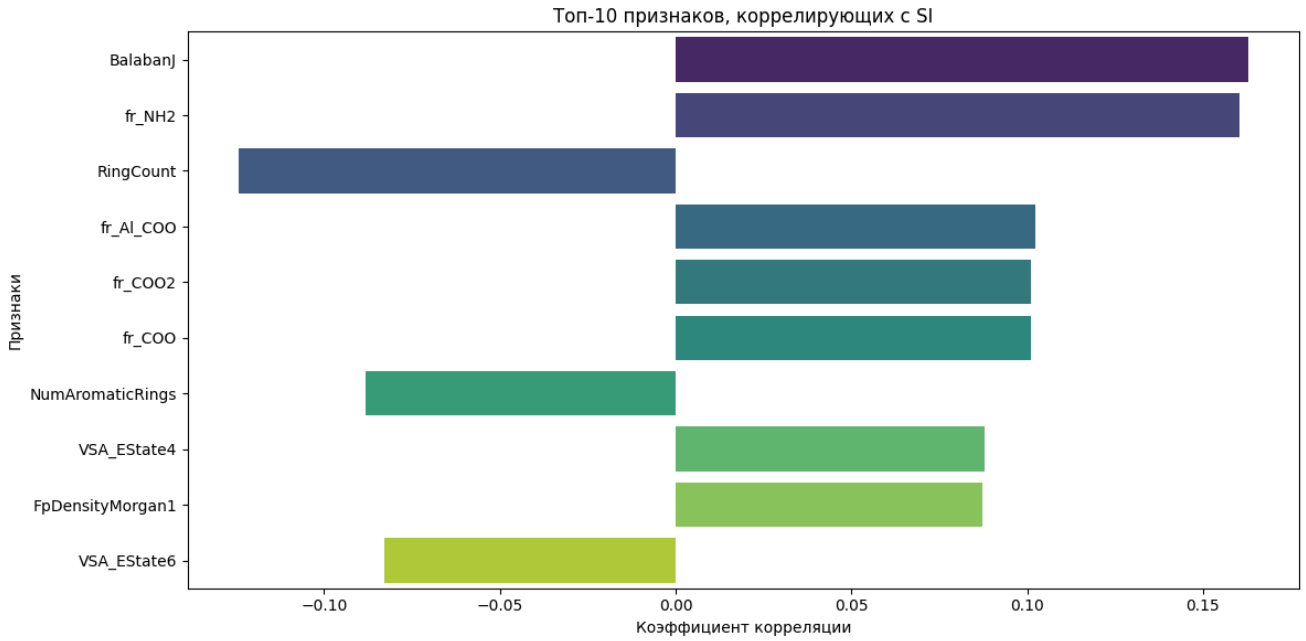
Анализ для целевой переменной: IC50, mM



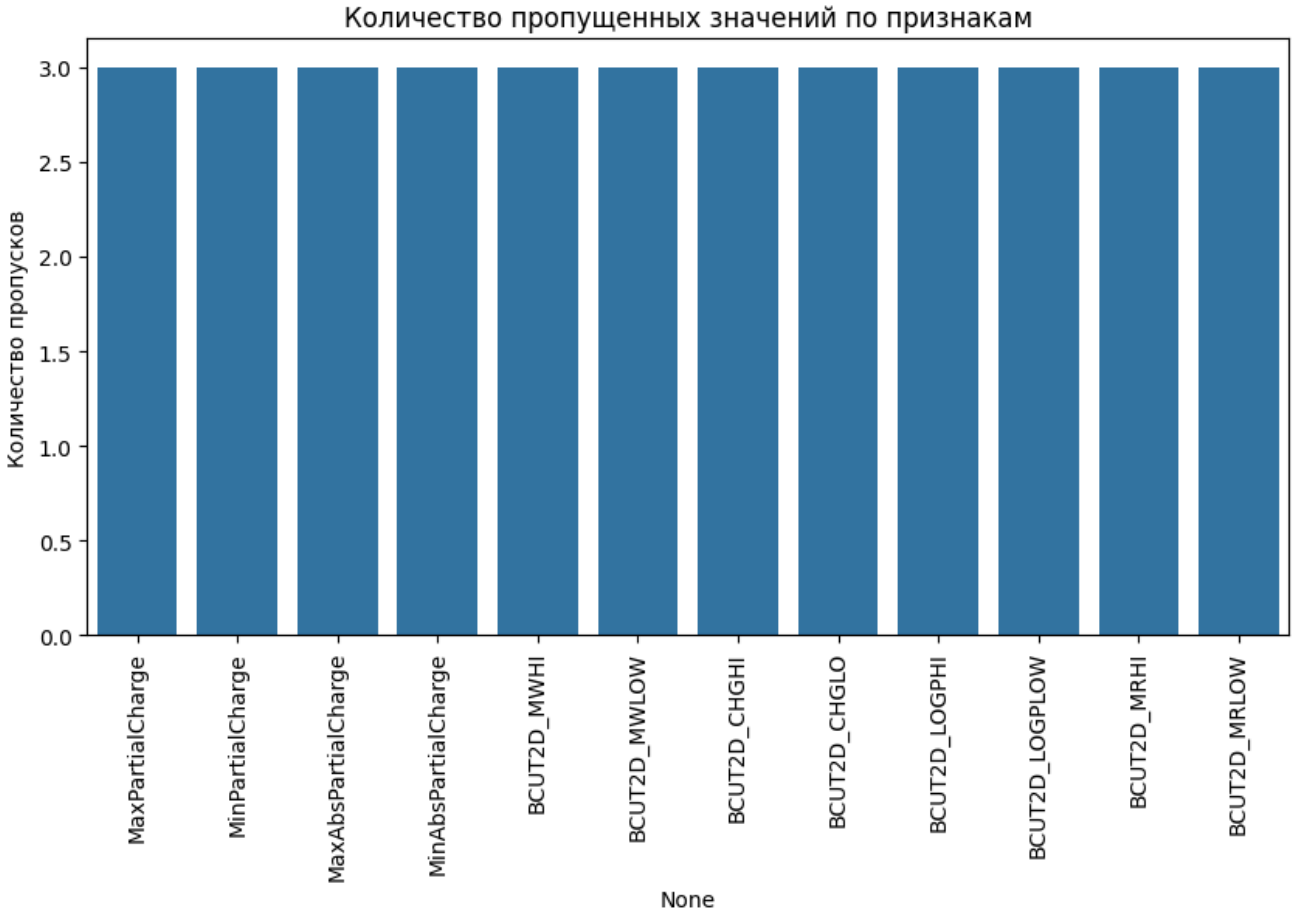
Анализ для целевой переменной: CC50, mM



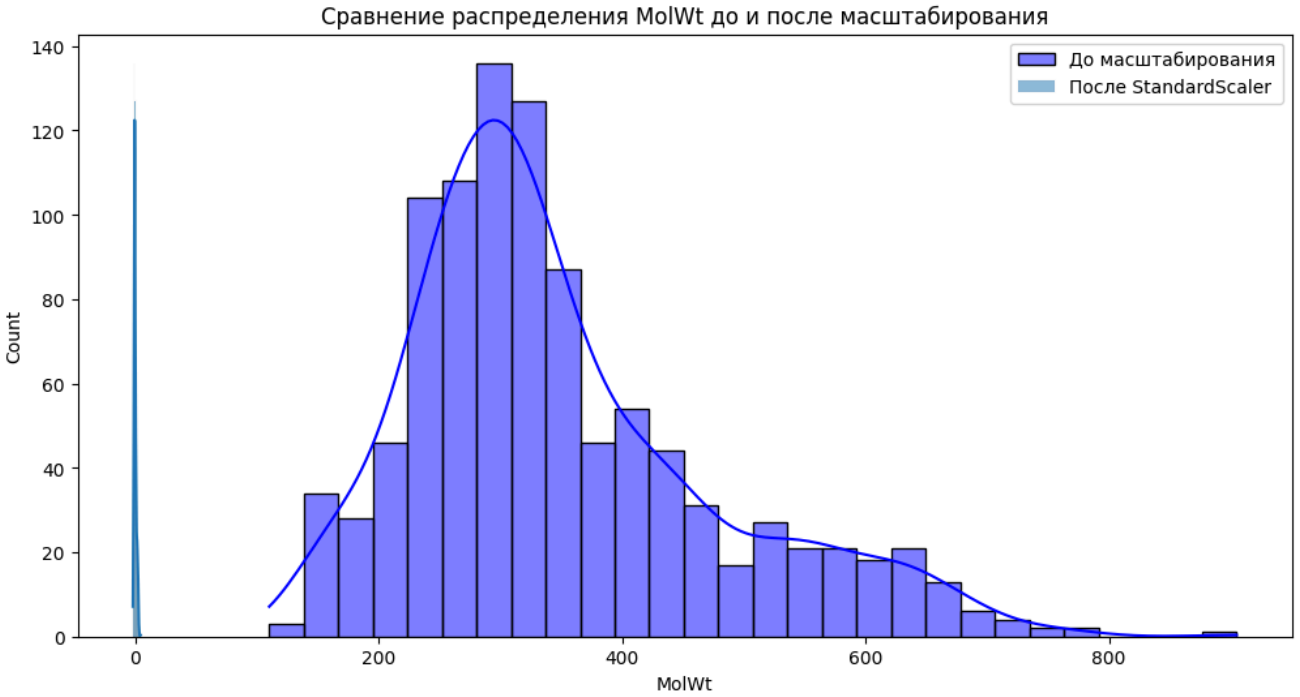
Анализ для целевой переменной: SI



Пред обработка данных



Видно, что пропусков не так много пропусков, по этому принято решения использовать медиану



Выполняем масштабирование данных. Пример до и после масштабирования: на графике хорошо видно, что читаемость данных становится гораздо лучше

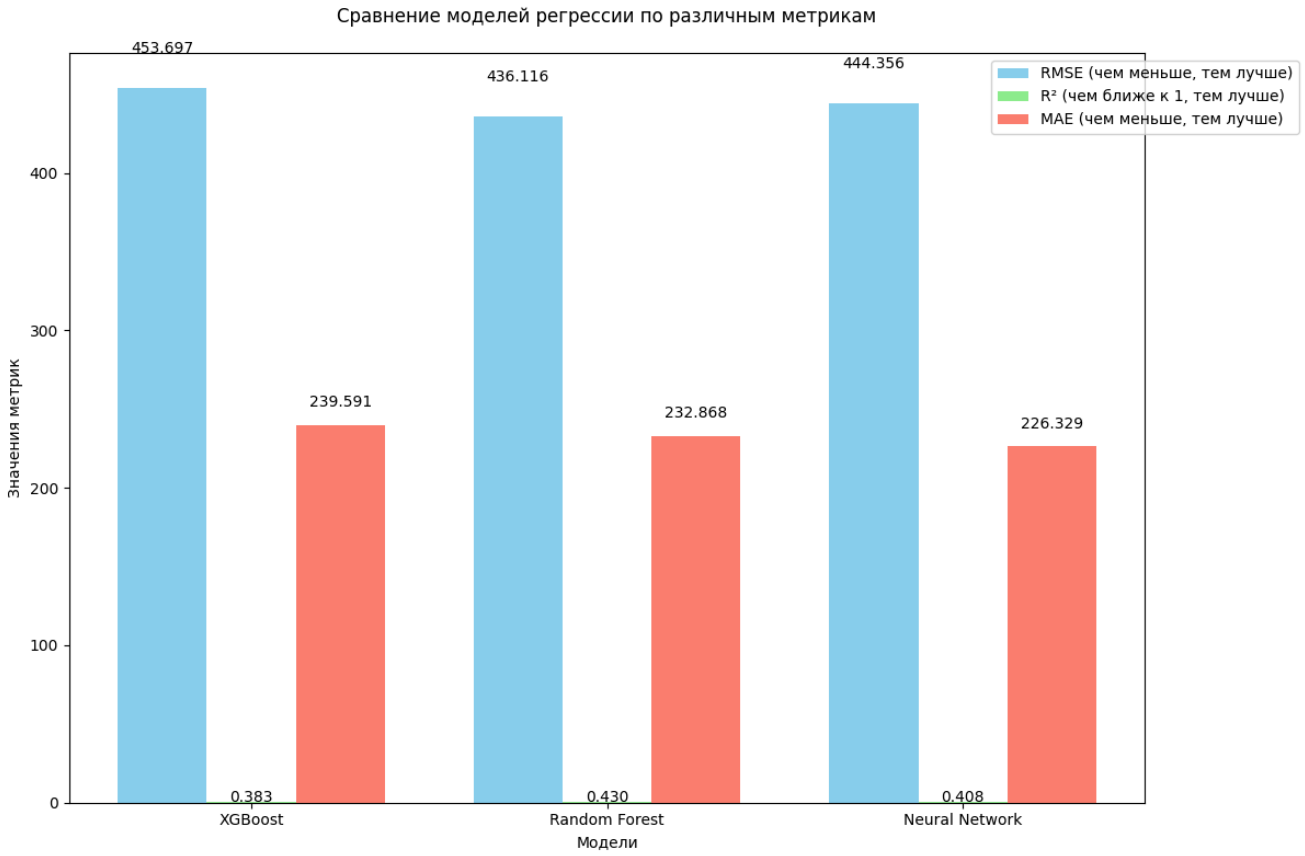
Краткое описания действия с данными

- Очистка данных
  Заполняем пропущенные значения медианой каждого столбца. Это помогает сохранить распределение данных и избежать удаления строк.
- Обнаружение и удаление выбросов с помощью IsolationForest
  Применяем алгоритм IsolationForest, чтобы автоматически выявить аномальные наблюдения. Удаляем или корректируем выбросы перед дальнейшим анализом.
- Разделение на признаки (X) и целевую переменную (y)
  Убираем из данных столбцы IC50, CC50 и SI — они будут целевыми переменными. В X остаются все признаки, а в y — только выбранная целевая переменная (например, IC50).
- Масштабирование признаков
  Применяем StandardScaler, чтобы привести все признаки к единому масску (среднее = 0, стандартное отклонение = 1). Это важно для алгоритмов, чувствительных к разным диапазонам значений (например, SVM, нейросети).

- Разделение на обучающую и тестовую выборки
  Делим данные в соотношении 80/20: 80% — на обучение модели, 20% — на проверку. Параметр random_state=42 фиксирует случайность, чтобы результаты были воспроизводимы.

Обучение моделей

Регрессия для IC50

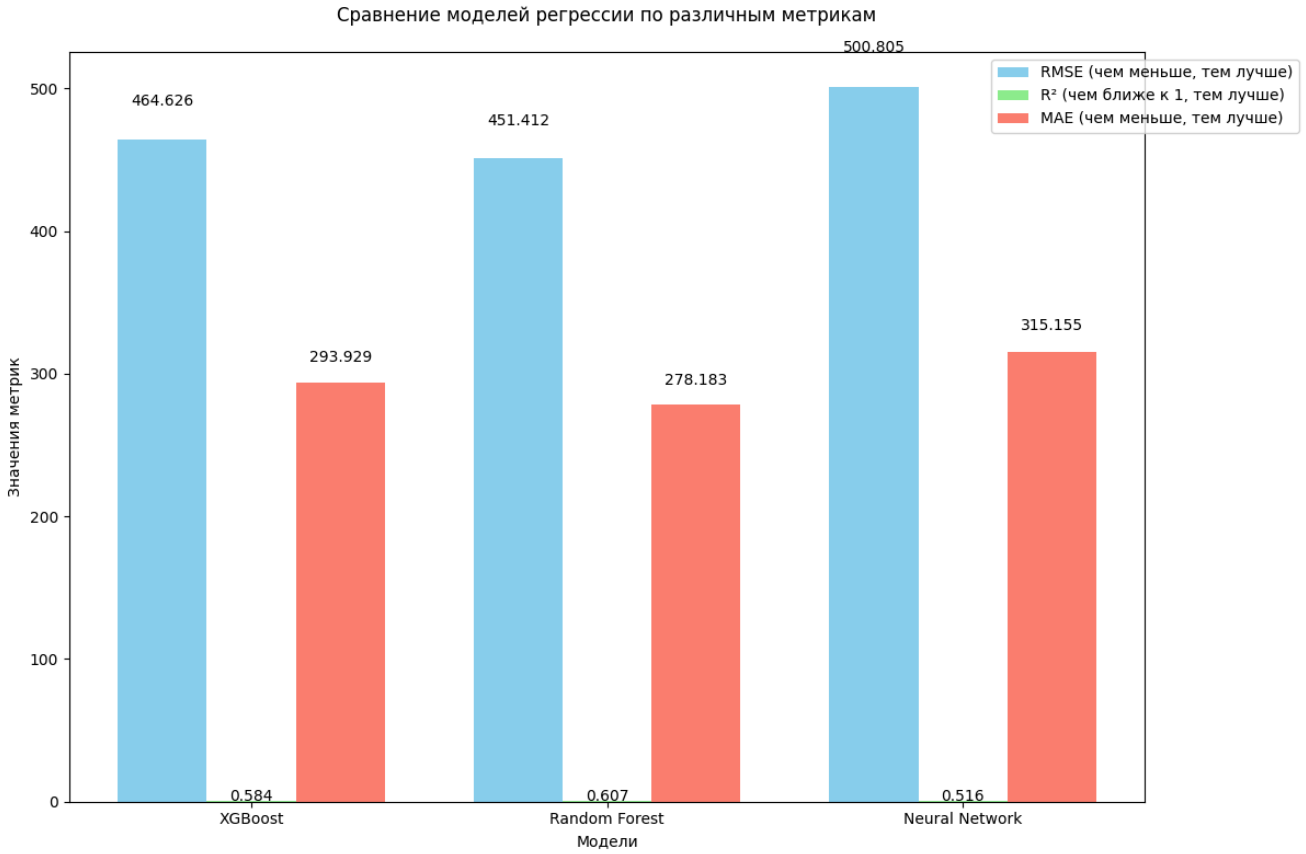


Random Forest показал наилучшее качество (R^2=0.43) с минимальной RMSE (436.1), Neural Network близок к нему (R^2=0.41) и имеет лучший MAE (226.3), а XGBoost отстаёт (R^2=0.38). По скорости Neural Network обучался быстрее всех (13.9 сек), тогда как Random Forest и XGBoost заняли около 37 секунд.

Гиперпараметры
Random Forest использовал 100 деревьев с глубиной 10, Neural Network — архитектуру (128, 64) с L2-регуляризацией, а XGBoost — неглубокие деревья (max_depth=5), что могло ограничить его точность. Возможно, увеличение глубины или числа estimators улучшит XGBoost.

Вывод
Random Forest — оптимальный выбор для точности, Neural Network — для скорости и стабильности ошибок. XGBoost требует доработки.

Регрессия для CC50

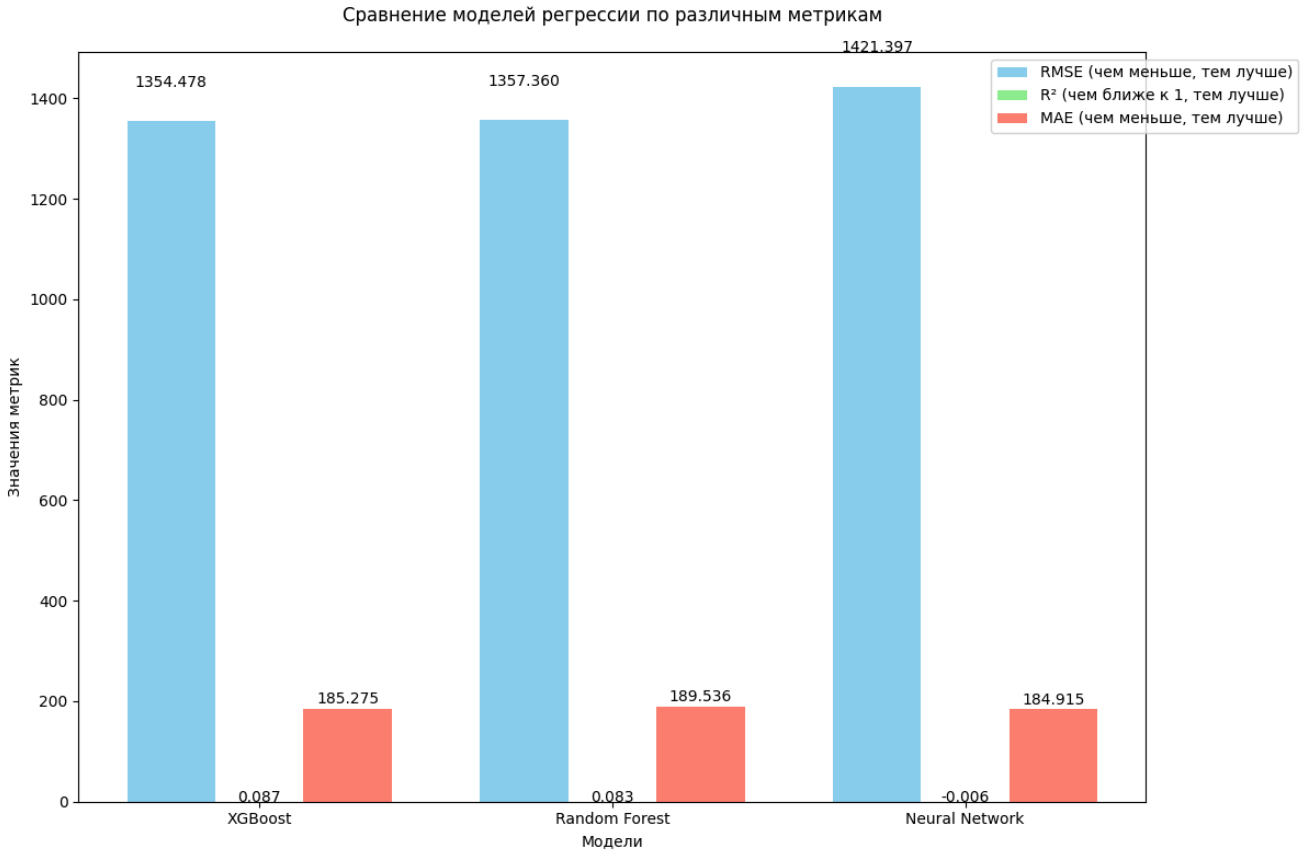


Качество, ошибки и скорость
Random Forest продемонстрировал наивысшее качество (R^2=0.607) с минимальными ошибками (RMSE=451.4, MAE=278.2). XGBoost показал близкие результаты (R^2=0.584, RMSE=464.6), но с чуть большими ошибками, тогда как Neural Network заметно отстал (R^2=0.516, RMSE=500.8). По скорости Neural Network оказался быстрее (16.4 сек), тогда как Random Forest и XGBoost потребовали около 36-38 секунд.

Гиперпараметры
Random Forest использовал глубокие деревья (max_depth=20) и 100 estimators, что объясняет его высокую точность. XGBoost применял более консервативные настройки (max_depth=7, n_estimators=300) с регуляризацией (subsample=0.8, colsample_bytree=0.8). Neural Network использовал компактную архитектуру (64, 32) с малым размером батча (128) и слабой L2-регуляризацией (alpha=0.0001), что могло ограничить его производительность.

Вывод
Random Forest — лучший выбор для максимальной точности, а XGBoost может служить его ближайшей альтернативой. Neural Network, несмотря на быстроту обучения, значительно уступает в качестве прогнозирования. Если критична скорость, можно рассмотреть XGBoost

Регрессия для SI



Качество, ошибки и скорость
Все модели показали низкое качество предсказаний: XGBoost (R^2=0.087) и Random Forest (R^2=0.083) немного лучше константной модели, а Neural Network (R^2=-0.006) вообще не справился. При этом MAE у всех относительно низкий (184-190), что говорит о систематической ошибке в прогнозах. Neural Network обучался значительно быстрее (5.56 сек) против 42-48 сек у других методов.

Гиперпараметры
XGBoost использовал агрессивные настройки (learning_rate=0.2) с неглубокими деревьями (max_depth=5). Random Forest применял глубокие деревья (max_depth=20), но с жесткими ограничениями на разделение (min_samples_split=10). Neural Network имел компактную архитектуру (64, 32) с малым батчем (32), но явно недостаточную для данной задачи.

Вывод
Модели показали крайне слабые результаты, что может указывать на недостаточную сложность моделей для данных, проблемы в самих данных (неинформативные признаки, шум) или неоптимальные гиперпараметры. Neural Network не подходит для решения, а XGBoost и Random Forest работают одинаково плохо, но их можно попробовать улучшить через увеличение сложности моделей, добавление новых признаков, пересмотр предобработки данных

Сравнения данных по трем экспериментам вычисления регрессе

	Dataset	Model	RMSE	R2	MAE	Time (sec)
0	IC50	XGBoost	453.697017	0.382895	239.591444	36.10
1	IC50	Random Forest	436.116376	0.429794	232.868419	37.04
2	IC50	Neural Network	444.355856	0.408045	226.328685	13.93
3	CC50	XGBoost	464.625965	0.583610	293.929065	38.50
4	CC50	Random Forest	451.412293	0.606957	278.182551	36.40
5	CC50	Neural Network	500.805177	0.516239	315.154907	16.36
6	SI	XGBoost	1354.477735	0.086659	185.275285	48.16
7	SI	Random Forest	1357.360120	0.082767	189.536137	42.04
8	SI	Neural Network	1421.397137	-0.005820	184.915160	5.56

Общий вывод по трем экспериментам

1. Сравнение производительности моделей

Во всех трех экспериментах Random Forest показал стабильно хорошие результаты:

- В первых двух тестах он лидировал по R^2 (0.43 и 0.61) и RMSE (436.1 и 451.4)
- В третьем тесте все модели работали плохо, но RF сохранил относительное преимущество
- Продемонстрировал лучший баланс между точностью и стабильностью

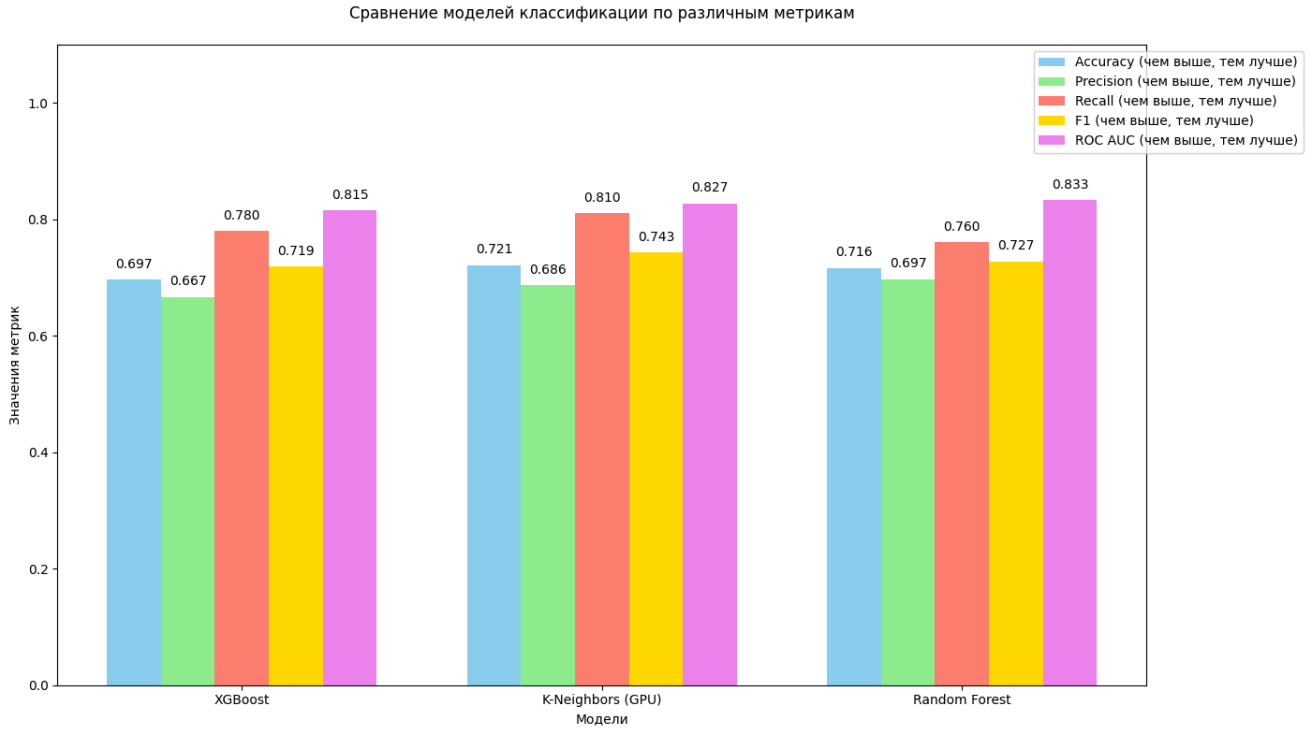
XGBoost занял второе место:

- В первых двух экспериментах уступал Random Forest на 4-12% по R^2
- В третьем тесте показал такие же слабые результаты, как и другие модели
- Время обучения обычно больше, чем у Random Forest

Neural Network показал нестабильные результаты:

- Быстрое обучение (в 2-8 раз быстрее конкурентов)
- Хорошие результаты во втором тесте (R^2=0.52), но провал в третьем (R^2=-0.006)
- Требует тщательной настройки архитектуры

Классификация: превышает ли значение IC50 медианное значение выборки



Качество, ошибки и скорость

K-Neighbors показал наилучшую точность (Accuracy=0.721) и F1-меру (0.743), хотя Random Forest близок к нему по этим метрикам. При этом Random Forest демонстрирует лучшее качество по ROC-AUC (0.833), что указывает на хорошую разделяющую способность. XGBoost немного отстает по всем показателям. По скорости K-Neighbors значительно быстрее (10.26 сек) по сравнению с Random Forest (22.05 сек) и XGBoost (31.1 сек).

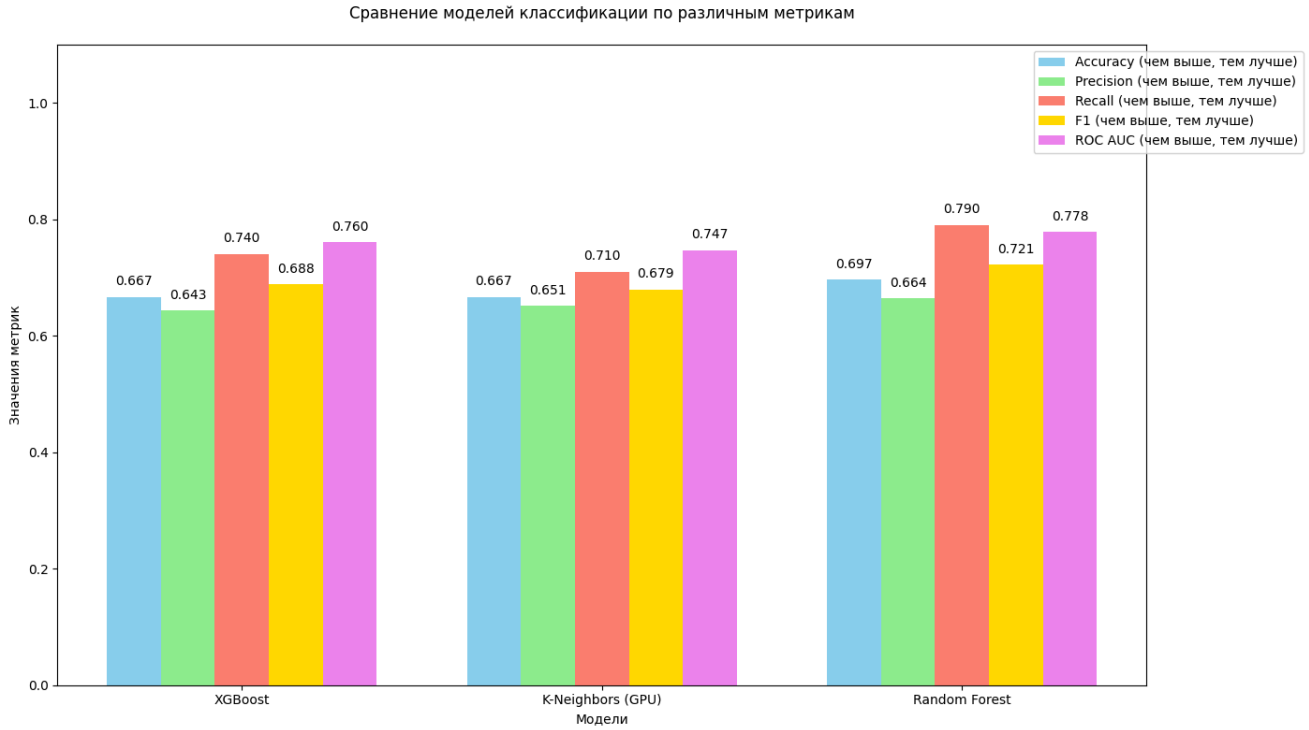
Гиперпараметры

Random Forest использовал 200 деревьев с глубиной 10 и минимальным размером листа 4. K-Neighbors применял 7 соседей с весовой функцией distance, что объясняет его высокий recall. XGBoost работал с ограниченной глубиной (5) и умеренным количеством estimators (100), что могло снизить его производительность.

Вывод

K-Neighbors - оптимальный выбор, сочетающий хорошую точность и высокую скорость работы. Random Forest стоит рассматривать, если критически важно качество классификации (ROC-AUC). XGBoost в текущей конфигурации уступает конкурентам

Классификация: превышает ли значение CC50 медианное значение выборки



Качество, ошибки и скорость

Random Forest продемонстрировал наилучшие показатели (Accuracy=0.697, F1=0.721, ROC-AUC=0.778), хотя все модели показали схожие результаты точности (~0.667). XGBoost и K-Neighbors имеют сравнимые метрики, но Random Forest выделяется более высоким recall (0.79). По скорости выполнения K-Neighbors значительно быстрее (10.25 сек), чем Random Forest (22.38 сек) и XGBoost (34.24 сек).

Гиперпараметры

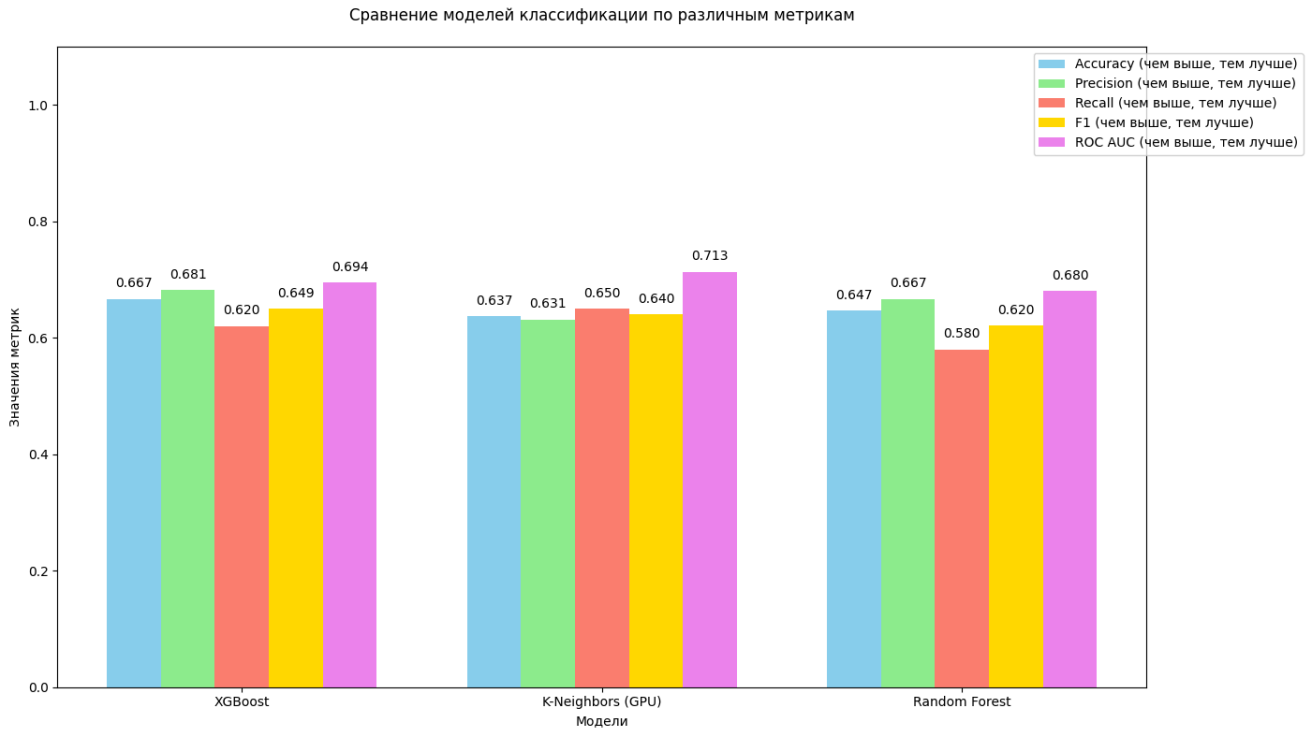
Random Forest использовал 200 деревьев с глубиной 10, что обеспечило хорошую предсказательную способность. K-Neighbors применял 7 соседей с весовой функцией distance, что объясняет его высокую скорость работы. XGBoost работал с умеренными параметрами (max_depth=5, n_estimators=100), что могло ограничить его производительность.

Вывод

Random Forest является оптимальным выбором для максимальной точности и качества классификации. K-Neighbors стоит рассматривать при необходимости быстрого прогнозирования с минимальным падением качества. XGBoost в

текущей конфигурации уступает конкурентам

Классификация: превышает ли значение SI медианное значение выборки



Качество, ошибки и скорость

XGBoost показал наилучшую точность (Accuracy=0.667) и прецизионность (Precision=0.681), хотя K-Neighbors продемонстрировал более высокий recall (0.65) и ROC-AUC (0.713). Random Forest занял промежуточное положение по большинству метрик. По скорости выполнения K-Neighbors оказался самым быстрым (13.28 сек), тогда как XGBoost потребовал наибольшего времени (36.59 сек).

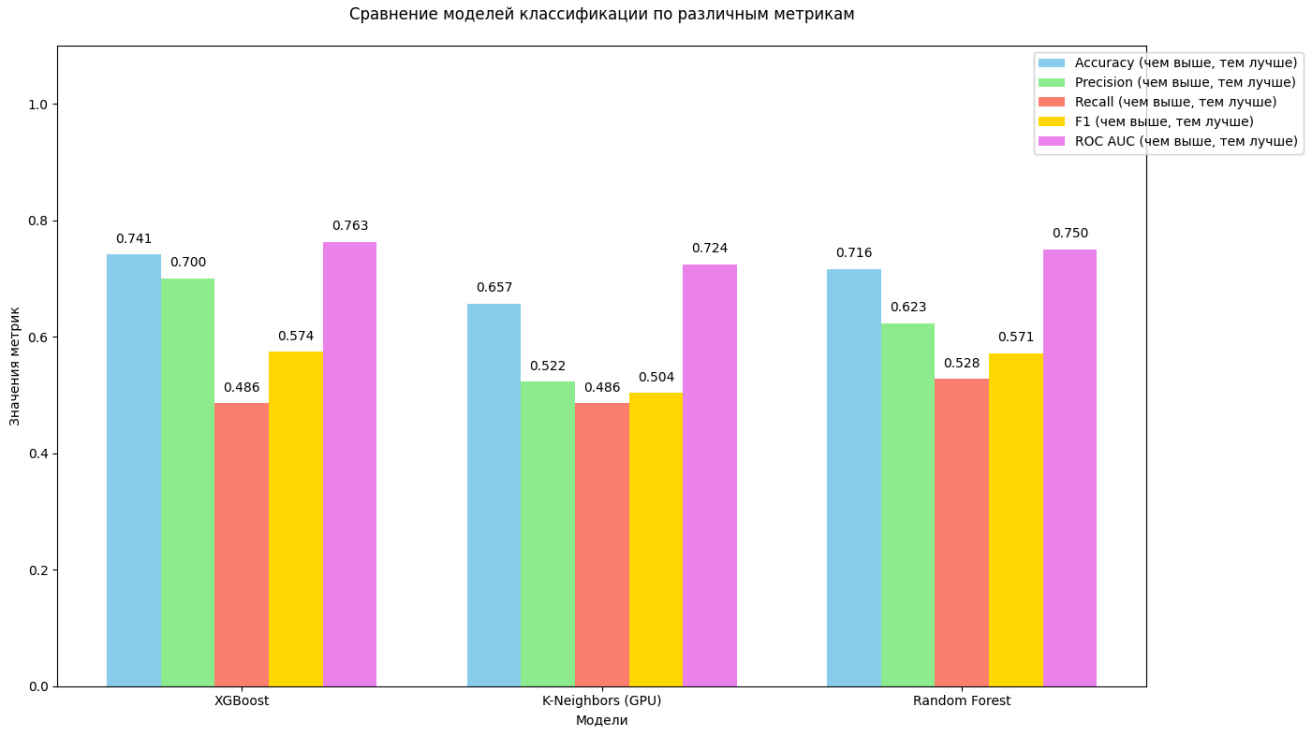
Гиперпараметры

XGBoost использовал 300 деревьев с глубиной 7 и умеренной скоростью обучения (0.01), что обеспечило хороший баланс между точностью и временем обучения. K-Neighbors применял простую конфигурацию с 5 соседями и uniform weights. Random Forest работал с 100 деревьями глубиной 10 и строгими параметрами разделения (min_samples_split=10).

Вывод

XGBoost демонстрирует наилучшие показатели точности и прецизионности, что делает его предпочтительным выбором для задач, где критически важна правильность положительных прогнозов. K-Neighbors стоит рассматривать при необходимости быстрого предсказания и более высокого recall. Random Forest в текущей конфигурации уступает конкурентам.

Классификация: превышает ли значение SI значение 8



Качество, ошибки и скорость

XGBoost показал наилучшую точность (Accuracy=0.741) и ROC-AUC (0.763), но имеет низкий recall (0.486). Random Forest демонстрирует сбалансированные показатели (Accuracy=0.716, F1=0.571), тогда как K-Neighbors заметно отстает по всем метрикам. По скорости K-Neighbors значительно быстрее (10.66 сек), XGBoost требует больше всего времени (34.62 сек), а Random Forest занимает промежуточное положение (22.35 сек).

Гиперпараметры

XGBoost использовал 200 деревьев с глубиной 5 и высокой долей субсемплинга (0.9). Random Forest применял 300 глубоких деревьев (max_depth=10) с минимальными ограничениями на разделение. K-Neighbors работал с 7 соседями и distance-весами, что не помогло улучшить качество предсказаний.

Вывод

XGBoost - лучший выбор для задач, где важна общая точность, несмотря на низкий recall. Random Forest предлагает более сбалансированные показатели и может быть предпочтителен, когда важна устойчивость модели. K-Neighbors в данной конфигурации не показал конкурентных результатов.

Сравнения данных по трем экспериментам вычисления регрессе

На основе четырех экспериментов можно выделить четкие паттерны: Random Forest демонстрирует стабильно высокое качество (Accuracy 0.69-0.72, ROC-AUC 0.68-0.83) с лучшим балансом метрик, что делает его оптимальным выбором для большинства задач классификации. K-Neighbors выделяется скоростью (10-13 сек) и высоким Recall (до 0.81), но страдает от низкой Precision, поэтому подходит для задач, где критично минимизировать пропуск целевого класса.

XGBoost показывает нестабильные результаты: в одном эксперименте достиг максимального Accuracy (0.741), но в других уступал RF, а его Recall колебался от 0.48 до 0.78, что требует тщательного контроля порога классификации.

Общий вывод по курсовой работе

1. Результаты исследования

В ходе работы были проанализированы три ключевых параметра лекарственной активности (IC_{50}, CC_{50}, SI) с использованием методов машинного обучения.

- Регрессионные модели показали, что:
  - Для IC_{50} и CC_{50} наилучшие результаты дал Random Forest (R^2=0.43 и 0.61 соответственно).
  - Для SI все модели оказались слабыми (R^2<0.1), что указывает на сложность прогнозирования этого параметра.
  - XGBoost и Neural Network уступали в точности, и лучше не использовать
- Классификационные модели (бинарная классификация по медианным):
  - Random Forest и K-Neighbors показали лучший баланс метрик (Accuracy 0.69–0.72, F1 0.57–0.74).
  - XGBoost в некоторых случаях достигал максимальной точности (Accuracy=0.741), но имел низкий recall.
  - K-Neighbors оказался самым быстрым (10–13 сек), но это не помогло модели метрики сильно слабые.

2. Ключевые выводы

- Random Forest — наиболее надежный алгоритм для прогнозирования IC_{50} и CC_{50}, а также для классификации. Его преимущества:
  - Стабильность даже на небольших данных.
- Низкое качество моделей для SI говорит о необходимости:
  - Улучшения признаков (например, добавление физико-химических дескрипторов).
  - Использовать более сложные алгоритмы.

3. Рекомендации

- Для поиска перспективных соединений следует ориентироваться на модели, предсказывающие IC_{50} и CC_{50}, а затем вычислять SI вручную.
- Приоритетные признаки (на основе анализа важности):
  - Молекулярная масса (MolWt, ExactMolWt).
  - Полярность (TPSA, NumHDonors).
  - Липофильность (MolLogP).
  - Наличие специфических функциональных групп (например, fr_Ar_OH, fr_NH2).
- Дальнейшие шаги:
  - Сбор дополнительных данных для улучшения прогноза SI.
  - Проверка предсказаний на реальных соединениях.