Отчёт о выполнении тестового задания

Описание задачи

Требовалось реализовать модель машинного обучения для предсказания знака изменения свободной энергии связывания (ddG) на основе данных из базы SKEMPI 2.0.

Выполненная работа

1. Предобработка данных (Jupyter notebook)

Исходный файл skempi_v2.csv содержал 7085 записей. В ходе обработки были выполнены следующие шаги:

- Удалены столбцы, содержащие только пустые значения
- Отфильтрованы записи с множественными мутациями, так как я не придумал как работать с множественными мутациями и упростил себе задачу до предсказания эффекта единичной мутации
- Оставлены только записи с валидными местоположениями мутаций: COR, INT, SUP, RIM, SUR
- Исправлены форматы чисел с плавающей точкой (замена запятой на точку)
- Удалены записи с аномальными температурами (>400K)
- Рассчитаны значения ddG по формуле: ddG = -R*T*ln(K_mut/K_wt)

После очистки осталось 2365 записей.

2. Создание признаков (make_features.py)

Разработан парсер для извлечения информации из строк мутаций формата 'IA96A':

- Цепь (I)
- Исходная аминокислота (A)
- Позиция (96)
- Мутантная аминокислота (A)

Для каждой аминокислоты использовались физико-химические свойства:

- Гидрофобность (по шкале Кайта и Дулиттла)
- Размер
- Заряд
- Гибкость (зависит от ротамеров и возможностей вращения колец)

Созданы признаки:

- Свойства исходной аминокислоты (4 признака)
- Свойства мутантной аминокислоты (4 признака)
- Изменения свойств при мутации (4 признака)
- Закодированное местоположение мутации (1 признак)

Целевая переменная: бинарная (1 если ddG > 0, иначе 0).

После обработки получен датасет из 1903 записей с 13 признаками.

3. Обучение моделей (train_models.py)

Протестированы следующие алгоритмы:

- Логистическая регрессия
- Random Forest
- Gradient Boosting
- SVM

Данные разбиты на обучающую (80%) и тестовую (20%) выборки с сохранением пропорций классов.

4. Результаты

Распределение классов: 1479 отрицательных (77.7%), 424 положительных (22.3%).

Метрики моделей на тестовой выборке:

Модель	Accuracy	ROC AUC	Precision	Recall	F1
Gradient Boosting	0.787	0.666	0.700	0.082	0.147
Random Forest	0.774	0.655	0.481	0.153	0.232
Логистическая регрессия	0.777	0.632	0.000	0.000	0.000
SVM	0.777	0.559	0.000	0.000	0.000

После оптимизации гиперпараметров Random Forest (50 деревьев, max_depth=10, min_samples_leaf=4, min_samples_split=2):

- Accuracy: 0.780
- ROC AUC: 0.644
- Precision: 0.500
- Recall: 0.040

Точность предсказания по местоположениям мутаций:

- INT: 0.850
- COR: 0.847
- SUP: 0.763
- RIM: 0.676
- SUR: 0.613

5. Наблюдения

1. Все модели испытывают трудности с предсказанием положительного класса из-за его меньшей представленности в данных.
2. Gradient Boosting показал лучшую производительность по метрике ROC AUC.
3. Точность предсказания зависит от местоположения мутации, причём мутации во интерфейсных и центральных областях предсказываются лучше.
4. Логистическая регрессия и SVM не смогли обучиться предсказывать положительный класс.

6. Сохраненные файлы

- cleaned_skempi.csv - очищенные данные
- cleaned_with_ddG.csv - данные с рассчитанным ddG
- single_mutations_features_factorized.csv - признаки для обучения
- X_features_factorized.csv - матрица признаков
- y_target.csv - целевая переменная
- best_random_forest_model_factorized.joblib - обученная модель
- scaler_factorized.joblib - объект для масштабирования признаков
- location_encoder.joblib - кодировщик местоположений

- raw_data_screening.ipynb — первичный обзор и процессинг сырых данных из skempi