"Оценка моделей дозирования варфарина на нескольких наборах данных с использованием нового программного каркаса и эволюционной оптимизации" авторы Gianluca Truda и P. Marais

<https://www.sciencedirect.com/science/article/pii/S1532046420302628>

В статье представлено исследование по оценке различных моделей дозирования варфарина.

В исследовании использовались следующие наборы данных:

- IWPC (International Warfarin Pharmacogenetics Consortium) - международный консорциум по фармакогенетике варфарина.

- набор данных южноафриканских пациентов, предоставленных PathCare.

Подготовка датасета.

Набор данных IWPC был разделен на два подмножества: обучающий и валидационный, в соотношении 80/20.

Отфильтровали пациентов, у которых отсутствовали данные о росте, весе, возрасте или генотипе, а также пациентов, которые не находились на стабильной дозе варфарина. Это оставило 4798 пациентов в наборе данных.

Обучающий набор данных (X\_train) включал различные признаки, такие как рост, вес, возраст, генотип и другие параметры, в то время как целевая переменная (y\_train) представляла собой дозу варфарина, которую необходимо было предсказать.

Перед обучением входные данные были масштабированы к единичной дисперсии с использованием StandardScaler в Scikit-learn, чтобы сделать алгоритмы менее чувствительными к масштабам (размерности) данных.

В обоих наборах данных еженедельная доза (в мг) была преобразована в квадратный корень, чтобы скорректировать искаженные распределения. Соответствующее значение было установлено в качестве целевого признака во время обучения. Во время оценки прогнозы модели были возведены в квадрат и сравнивались с исходными значениями, сохраняя клиническую релевантность.

Авторы применяли различные модели машинного обучения для предсказания оптимальной дозы варфарина, включая:

- Линейную регрессию (LR)

- Линейный SVR (SVR)

- SV (Support Vector Machines)

- Ridge Regression (RR)

- Градиентный ускоряющий регрессор (BRT и GBT)

- Нейронные сети (NN)

- Ансамбли и стековую генерализацию (Stacked\_SV и Stacked\_RR)

Эффективность моделей оценивалась с использованием двух метрик:

1. PW20 (Percentage Within 20%) - процент пациентов, у которых предсказанная доза варфарина находится в пределах 20% от терапевтической дозы.

2. MAE (Mean Absolute Error) - средняя абсолютная ошибка, которая измеряет разницу между предсказанными и истинными значениями дозирования варфарина.

Основные выводы:

- Модели, разработанные с использованием генетического программирования, показали сопоставимые результаты с моделями, оптимизированными экспертами в области машинного обучения.

- Явного лидера по эффективности среди моделей не выявлено (на мой взгляд).

- Разработан и представлен в свободный доступ код на питоне Warfit-learn, который позволяет автоматизировать процесс разработки и оценки моделей дозирования варфарина. <https://github.com/gianlucatruda/warfit-learn/blob/master/README.md>

Подробная информация о моделях, использованных в исследовании, включая их реализацию и ключевые гиперпараметры:

1. Линейная регрессия (LR)

- Реализация: sklearn.linear\_model.LinearRegression

- Ключевые гиперпараметры:

- normalize: False

- fit\_intercept: True

2. Метод опорных векторов с линейным ядром (SVR)

- Реализация: sklearn.svm.LinearSVR

- Ключевые гиперпараметры:

- epsilon: 0.0

- tol: 0.0001

- C: 1.0

- loss: epsilon\_insensitive

3. Метод опорных векторов (SV)

- Реализация: sklearn.svm.SVR

- Ключевые гиперпараметры:

- kernel: linear

- cache\_size: 1000

4. Гребневая регрессия (RR)

- Реализация: sklearn.linear\_model.Ridge

- Ключевые гиперпараметры:

- alpha: 1.0

5. Ускоренные регрессионные деревья (BRT)

- Реализация: sklearn.ensemble.GradientBoostingRegressor

- Ключевые гиперпараметры:

- loss: least squares

- learning\_rate: 0.1

- n\_estimators: 100

6. Градиентный бустинг деревьев (GBT)

- Реализация: sklearn.ensemble.GradientBoostingRegressor

- Ключевые гиперпараметры:

- learning\_rate: 0.1

- loss: lad

- max\_depth: 4

7. Нейронные сети (NN)

- Реализация: sklearn.neural\_network.MLPRegressor

- Ключевые гиперпараметры:

- hidden\_layer\_sizes: (100,)

- activation: logistic

- solver: lbfgs

8. Ансамбль с методом опорных векторов (Stacked\_SV)

- Реализация: mlxtend.regressor.StackingCVRegressor

- Ключевые гиперпараметры:

- regressors: {GBT, SV, NN}

- meta\_regressor: SV

- cv: 5

9. Ансамбль с гребневой регрессией (Stacked\_RR)

- Реализация: mlxtend.regressor.StackingCVRegressor

- Ключевые гиперпараметры:

- regressors: {GBT, RR, NN}

- meta\_regressor: RR

- cv: 5