

Задание 1:

Согласно определению из википедии AutoML - процесс автоматизации сквозного процесса применения **машинного обучения** к задачам реального мира. Под капотом предполагается, что алгоритмы автоматического ML подбирают модель и вычисляют оптимальные параметры для модели для решения некоторой задачи.

Сегодня в мире существует несколько библиотек, реализующие алгоритмы AutoML. Такие библиотеки в первую очередь направлены на поиск универсального метода для решения наибольшего числа задач.

Примеры библиотек [1]:

- H2O.ai (<http://docs.h2o.ai/h2o/latest-stable/h2o-docs/downloading.html>)
- auto-sklearn (<https://automl.github.io/auto-sklearn/master/>)

Однако, в силу того, что библиотеки AutoML направлены на решение универсальным методом наибольшего числа задач, то такие библиотеки не всегда показывают лучшие результаты по сравнению с вручную подобранными алгоритмами машинного обучения.

Например, в следующей статье (<https://arxiv.org/pdf/1907.00909v1.pdf>) приводится сравнительный анализ различных AutoML библиотек и RandomForest (с настроенными параметрами) на различных датасетах. И сравнительная таблица показывает, что результат AutoML алгоритма на некоторых задачах хуже, чем стандартные настроенные подходы.

Проведите небольшое исследование AutoML библиотек самостоятельно.

1. Возьмите датасет для бинарной классификации (к примеру, можете взять датасет, где параметрами являются описание человека, а таргетом - значения $>50k$ или $\leq 50k$, которые означают годовой заработок человека <https://archive.ics.uci.edu/ml/datasets/adult>). Данный датасет приведен для примера, можете взять любой другой датасет по желанию.
2. Получите результаты запусков библиотек указанных в [1]. Можете заменить данные библиотеки на другие либо исследовать дополнительные библиотеки.
3. Предложите свою модель для датасета. Опишите, как Вы подбирали гиперпараметры для Вашей модели.
 - а. *Дополнительное задание, по желанию:* В качестве эксперимента попробуйте использовать подходы подбора гиперпараметров из лекции (например, байесовская оптимизация) вместо обычного прохода по сетке возможных параметров.
4. Сравните ROC-AUC кривые вашей модели и моделей, полученных при использовании библиотек.
5. Снабдите все инструкциями по запуску, четкими объяснениями полученных результатов, наглядными графиками, поясните мотивацию выбранной Вами модели.

Задание 2:

Требуется реализовать алгоритм Нелдера-Мида в n -мерном пространстве. В реализации алгоритма поддержать операции отражения, растяжения, сжатия.

Реализовать на питоне класс `NelderMeadSolver` с публичным методом `solve(self, f)`. Передаваемая функция `f` - это callable объект, который высчитывает значение функции в точке $f(x)$, где x - это вектор из n -мерного пространства (***VectorN()***).

Ожидается, что метод будет решать задачу поиска оптимального решения с помощью алгоритма Нелдера-Мида (http://www.scholarpedia.org/article/Nelder-Mead_algorithm)

Конструктор класса должен в произвольной форме принимать условие останова поиска решения (число итераций, точность полученного решения, т.п.) и необходимые параметры поиска, размерность пространства.

Решение должно поддерживать n -мерное пространство.

В качестве тестирования работы алгоритма привести примеры работы на функциях в двумерных и трехмерных пространствах.

(Визуализация по желанию).

Дополнительное задание (по желанию): сравнить скорость поиска решения с помощью Вами реализованного алгоритма Нелдера-Мида с реализацией алгоритма байесовской оптимизации. Можно использовать любые библиотеки, упомянутые на семинаре (Sigopt, spearmint, hyperopt). Сравнение произвести на собственных примерах по количеству вызовов функции $f(x)$ (f - является тяжеловесной функцией).