Введение

Машинное обучение - один из наиболее обширных и значимых разделов искусственного интеллекта, основным назначением которого является построение обучаемых моделей и оценка их качества. Среди алгоритмов машинного обучения обычно различают обучение с учителем (supervised learning) и обучение без учителя  (unsupervised learning). Классический пример обучения с учителем - обучение многослойного перцептрона методами Back propagation и Resilient propagation. В этом же ключе можно упомянуть и метод опорных векторов (support vector machine), а также деревья принятия решений (decision trees). Среди алгоритмов обучения без учителя наиболее известным и, зачастую, упоминаемым в первую очередь является алгоритм K-средних (K-means).

Наиболее базовыми задачами машинного обучения являются задачи классификации и кластеризации данных. Задача классификации состоит в том, чтобы определить класс объектов из некоторой тестовой выборки на основе имеющихся данных о классах объектов из обучающей выборки. Задача же кластеризации состоит в организации объектов в однородные группы на основе совокупности признаков. При этом задача классификации относится к классу задач обучения с учителем, в то время как задача кластеризации относится к классу обучения без учителя.

Одним из наиболее интересных инструментов машинного обучения являются нейронные сети, которые могут решать широкий круг задач обработки и анализа данных: распознавание и классификация образов, прогнозирование, управление, автоматическая торговля, анализ естественного языка, медицинских данных и т.д. Конкурентами нейронных сетей являются классические методы анализа данных, однако нейронные сети имеют над ними ряд преимуществ. Используя способность обучаться на множестве примеров, нейронная сеть способна решать задачи, в которых неизвестны закономерности развития ситуации и зависимости между входными и выходными данными. Нейронные сети устойчивы к зашумленности входных данных и способны адаптироваться к изменениям окружающей среды. Также нейронные сети обладают потенциальным сверхвысоким быстродействием и значительной отказоустойчивостью за счет массового параллелизма обработки информации.

Однако есть у нейронных сетей и недостатки. Как следствие отсутствия формализованных алгоритмов настройки сети и ее высокой сложности, возникает необходимость в привлечении высококлассных специалистов, обладающих необходимыми знаниями. Отсутствует строгая теория по выбору архитектуры нейронной сети под специфические задачи, хотя некоторая работа в этом направлении ведется. Из обученной сети практически невозможно извлечь приобретенные знания, то есть, нейронная сеть является своего рода черным ящиком.

Для реализации нейросетевых алгоритмов используются различные языки программирования, в том числе и специализированные диалекты LISP. Для написания реально используемых систем часто используют С++ благодаря его чрезвычайно мощным возможностям параллельного программирования, в том числе и на GPU, а также высокой скорости выполнения. В то же время, для прототипирования приложений, связанных с машинным обучением, обычно используются более специализированные языки: Matlab, R, Python.

В то время, как для задачи классификации используются многослойный перцептрон, метод опорных векторов и деревья принятия решений, для задачи кластеризации используют нейронные сети Кохонена, алгоритмы K-means, Affinity propagation, Spectral clustering, Hierarchical clustering, DBScan и Gaussian mixtures.

Наиболее серьезным препятствием перед использованием алгоритмов машинного обучения в реальных условиях является выбор оптимального алгоритма для определенной задачи. Помочь в этом могут алгоритмы оценки качества обучения. Однако и они тоже подвержены проблеме оптимального выбора. Разные метрики, предназначенные для разных алгоритмов машинного обучения, оценивают алгоритмы с разных позиций. Нельзя с уверенностью сказать, какая метрика будет более точно отображать реальную эффективность алгоритма на определенном наборе данных. Поэтому целью данной работы будет являться создание инструмента, позволяющего исследовать поведение различных алгоритмов классификации и кластеризации на различных наборах данных с помощью различных метрик.

Основным языком для приложения является Python – популярный язык программирования, используемый для разработки самостоятельных программ и различного рода прикладных сценариев в самых разных областях применения. Связано это, прежде всего, с возможностью использования библиотек машинного обучения PyBrain и Scikit-learn. PyBrain – одна из лучших библиотек в мире Python для реализации и изучения разнообразных алгоритмов, связанных с нейронными сетями. Scikit-learn реализует часть алгоритмов машинного обучения, которая лежит за пределами нейросетевых алгоритмов: бинарные деревья, метод опорных векторов, алгоритмы кластеризации. Кроме того, данная библиотека снабжена набором метрик оценки качества алгоритмов машинного обучения буквально на все случаи жизни.

Для построения графического интерфейса используется библиотека Glade, генерирующая описание графического интерфейса на основе XML. Потенциально есть возможность получить кроссплатформенную реализацию графического интерфейса, так как существует порт Glade для семейства ОС Windows. В пользу Glade также говорит наличие удобного инструмента для проектирования интерфейса – Glade Interface Designer.