Лекция 2

Библиотеки и подходы машинного обучения

Библиотека Scikit-Learn (http : / / s ciki t-learn . org/) очень проста в использовании, при этом она эффективно реализует многие алгоритмы машинного обучения, что делает ее великолепной отправной точкой для изучения машинного обучения. **scikit-learn** требует наличия еще двух пакетов Python – NumPy и SciPy. Для построения графиков и интерактивной работы необходимо также установить matplotlib, IPython и Jupyter Notebook

TensorFlow (http : / / tensorflow . org/) является более сложной библиотекой для распределенных численных расчетов с применением графов потоков данных. Это позволяет эффективно обучать и запускать

очень большие нейронные сети, потенциально распределяя вычисления между тысячами серверов с множеством графических процессоров. Библиотека TensorFlow была создана в Google и поддерживает

много крупномасштабных приложений машинного обучения. В ноябре 2015 года она стала продуктом с открытым кодом.

https://github.com/LittleBrainz/ageron--handson-ml

Anaconda

Дистрибутив Python, предназначенный для крупномасштабной обработки данных, прогнозной аналитики и научных вычислений. Anaconda уже включает NumPy, SciPy, matplotlib, pandas, IPython, Jupyter Notebook и scikit-learn.

Enthought Canopy дистрибутив Python для научных вычислений. Он уже

содержит NumPy, SciPy, matplotlib, pandas и IPython, но бесплатная версия не включает scikit-learn.

Свободный дистрибутив Python для научных вычислений, специально предназначенный для Windows

Jupyter Notebook представляет собой интерактивную среду для запуска программного кода в браузере. Это отличный инструмент для разведочного анализа данных и широко используется специалистами для анализа данных. Jupyter Notebook поддерживает множество языков программирования, нам нужна лишь поддержка Python. Jupyter Notebook позволяет легко интегрировать программный код, текст и изображения.

NumPy – это один из основных пакетов для научных вычислений в Python. Он содержит функциональные возможности для работы с многомерными массивами, высокоуровневыми математическими функциями (операции линейной алгебры, преобразование Фурье, генератор псевдослучайных чисел).

SciPy – это набор функций для научных вычислений в Python. предлагает продвинутые процедуры линейной алгебры, математическую оптимизацию функций, обработку сигналов, специальные математические функции и статистические функции. scikit-learn использует набор функций SciPy для реализации своих алгоритмов. наиболее важной частью SciPy является пакет scipy.sparse: с помощью него получаем разреженные матрицы (sparse matrices), которые представляют собой еще один формат данных, который используется в scikit-learn. Разреженные матрицы используются всякий раз, когда нам нужно сохранить 2D массив, который содержит в основном нули.

matplotlib – это основная библиотека для построения научных графиков в Python. Она включает функции для создания высококачественных визуализаций типа линейных диаграмм, гистограмм, диаграмм разброса и т.д.

pandas – библиотека Python для обработки и анализа данных, построена на основе структуры данных, называемой DataFrame и смоделированной по принципу датафреймов среды статистического программирования R

Системы машинного обучения (подходы к машинному обучению).

- С учителем и без учителя
- Частичное - semisupervised learning
- С подкреплением - Reinforcement Learning
- Динамическое обучение – алгоритмы способны к обучению на лету (или пакетные)

Подход к задачам обучения — это концепция, парадигма, точка зрения на процесс обучения, приводящая к набору базовых предположений, гипотез, эвристик, на основе которых строится модель, функционал качества и методы его оптимизации. Разделение методов «по подходам» условно.

Обучение с учителем

Обучающие данные **содержат** метки классов (label)

Типичная задача – **классификация данных.** Задача классификации (classification) отличается тем, что множество допустимых ответов конечно (*метки классов* (class label)). Класс — это множество всех объектов с данным значением метки.

Регрессия – прогнозирование целевого числового значения. Задача регрессии (regression) отличается тем, что допустимым ответом является действительное число или числовой вектор.

Пример: прогноз стоимости продажи автомобиля по набору признаков. Признаки называют predictor или предикторами.

Известные алгоритмы машинного обучения с учителем.

- k ближайших соседей (k-nearest neighbors)
- линейная регрессия (linear regression)
- логистическая регрессия (logistic regression)
- метод опорных векторов (Support Vector Machine - SVM)

- деревья принятия решений (decision tree) и случайные леса (random forest)
- нейронные сети (neural network).

Некоторые архитектуры нейронных сетей обучаются без учителя.

- Примеры:
- Автокодировщики (autoencoder)
- ограниченные машины Больцмана (restricted Boltzmann machine)
- Глубокие сети доверия (deep belief network)
- Самоорганизующиеся карты Кохонена.

Обучение без учителя

Обучающие данные не содержат метки классов (label). Каждый прецедент представляет собой пару «объект, ответ». Требуется найти функциональную зависимость ответов от описаний объектов и построить алгоритм, принимающий на входе описание объекта и выдающий на выходе ответ. Функционал качества обычно определяется как средняя ошибка ответов, выданных алгоритмом, по всем объектам выборки.

Типичная задача – кластеризация данных.

Примеры использования машинного обучения без учителя:

- иерархический кластерный анализ (Hierarchical Cluster Analysis - HCA)
- максимизация ожиданий (expectation maximization)
- Визуализация и понижение размерности - цель выделение основных признаков (feature extraction)
- анализ главных компонентов (Principal Component Analysis - PCA)
- ядерный анализ главных компонентов (kernel PCA)
- локальное линейное вложение (Locally-Linear Embedding - LLE)
- стохастическое вложение соседей с t-распределением (t-distributed Stochastic Neighbor Embedding - t-SNE)
- Обучение ассоциативным правилам (association rule learning)

Частичное обучение

Большинство алгоритмов машинного обучения – это комбинации (ансамбли) алгоритмов обучения без учителя и с учителем.

- Пример: глубокие сети доверия (DBN) основаны на компонентах обучения без учителя, называемых ограниченными машинами Больцмана (RBM). Машины RBM обучаются последовательно без учителя, после чего целая система точно настраивается с применением приемов обучения с учителем.

Обучение с подкреплением. Обучающая система называется **агентом,** может наблюдать за средой, выбирать и выполнять действия, выдавая в ответ награды или штрафы. Система должна самостоятельно выбрать лучшую стратегию, называемую **политикой.** Политика определяет, какое действие агент обязан выбирать, когда он находится в заданной ситуации

Пример: роботы реализуют алгоритмы обучения с подкреплением

Пакетное обучение

При пакетном обучении система неспособна обучаться постепенно: она должна учиться с применением всех доступных данных. В общем случае процесс будет требовать много времени и вычислительных ресурсов, поэтому обычно он проходит автономно – **автономное обучение (offline learning).**

Сначала система обучается, а затем помещается в производственную среду и функционирует без дальнейшего обучения.

При динамическом обучении система обучается постепенно за счет последовательного предоставления ей образцов данных либо по отдельности, либо небольшими группами, называемыми мини-пакетами.

Каждый шаг обучения является быстрым и недорогим, так что система может узнавать о новых данных «на лету» по мере их поступления Динамическое обучение отлично подходит для систем, которые получают данные в виде непрерывного потока (например, видеоинформация).

Различают два типа обучения.

Обучение по прецедентам*, или *индуктивное обучение, основано на выявлении общих закономерностей по частным эмпирическим данным.

Дедуктивное обучение предполагает формализацию знаний экспертов и их перенос в компьютер в виде базы знаний. Дедуктивное обучение принято относить к области экспертных систем, поэтому термины *машинное обучение* и *обучение по прецедентам* часто считают синонимами.