

Глава 1. Введение

- 1.1. Постановка задач распознавания и обучения распознавателя
 - 1.1.1. Входы и выходы
 - 1.1.2. Классификация и регрессия
 - 1.1.3. Пример: распознавание методом ближайших соседей
 - 1.1.4. Формальная постановка задачи обучения распознавателя: минимизация эмпирического риска
 - 1.1.5. Пример: распознающие деревья
 - 1.1.6. Способность распознавателя к обобщению и регуляризация
 - 1.1.7. Подбор параметров регуляризации
- 1.2. Обучение распознавателей и вероятностные модели
 - 1.2.1. Байесовский классификатор и байесовская регрессия
 - 1.2.2. Пример: асимптотика ошибок метода ближайшего соседа
 - 1.2.3. Классификация моделей и методов обучения
 - 1.2.4. Обучение порождающих и дискриминантных моделей
 - 1.2.5. Пример: наивный байесовский метод
 - 1.2.6. Обучение дискриминантных моделей (продолжение)
 - 1.2.7. Пример: регрессия методом наименьших квадратов
 - 1.2.8. Пример: применение регрессии для классификации с оценкой вероятностей классов
- 1.3. Другие задачи статистического обучения
 - 1.3.1. Обучение с учителем и без
 - 1.3.2. Оценка плотности и обнаружение выбросов
 - 1.3.3. Кластеризация
 - 1.3.4. Векторное квантование и понижение размерности

Глава 2. Линейные распознаватели: обзор

- 2.1. Линейная регрессия
 - 2.1.1. Минимизация квадратичной ошибки
 - 2.1.2. Минимизация квадратичной ошибки с регуляризацией
 - 2.1.3. Минимизация других ошибок
- 2.2. Линейная классификация
 - 2.2.1. Линейный дискриминантный анализ (дискриминант Фишера)
 - 2.2.2. Логистическая регрессия (regression)
 - 2.2.3. Перцептрон Розенблатта
 - 2.2.4. Классификаторы с разделяющей полосой (с зазором, classifiers)
- 2.3. Пространства признаков для линейных распознавателей
 - 2.3.1. Базисные функции
 - 2.3.2. Ядра (kernels)
 - 2.3.3. Слабые распознаватели

Глава 3. Нейронные сети

- 3.1. Естественные и искусственные нейронные сети
- 3.2. Многослойные перцептроны (MLP)
 - 3.2.1. Вычислительные возможности перцептрона и теорема Колмогорова
 - 3.2.2. Конструирование перцептронов (пример)
 - 3.2.3. Обучение многослойного перцептрона: метод обратного распространения ошибки (error - propagation)

- 3.2.4. Обучение нейронных сетей: применение градиентного спуска и стохастических методов
- 3.3. RBF-сети
 - 3.3.1. Обучение RBF-сетей: метод максимизации ожидания (maximization)

Глава 4. Линейные распознаватели и ядра: некоторые подробности

- 4.1. Ядра
 - 4.1.1. Свойства ядер Мерсера и теорема о реализации
 - 4.1.2. Построение ядер Мерсера
 - 4.1.3. Ядра и линейная делимость
 - 4.1.4. Сверточные ядра Мерсера
 - 4.1.5. Условно-неотрицательно определенные ядра
- 4.2. Метод опорных векторов (SVM, SVC, SVR)
 - 4.2.1. Двухклассовая классификация
 - 4.2.2. Опорные и другие векторы
 - 4.2.3. Проблемы обучения SVC
 - 4.2.4. Регрессия
 - 4.2.5. Регрессия и классификация
 - 4.2.6. Многоклассовая классификация
 - 4.2.7. Сведение многоклассовой классификации к последовательности двухклассовых

Глава 5. Линейные комбинации распознавателей

- 5.1. Общие идеи: голосование распознавателей и распознаватели над распознавателями
 - 5.1.1. Голосование независимо обучаемых распознавателей
 - 5.1.2. Предыстория и история бустинга
 - 5.1.3. "Сила слабости"
- 5.2. Градиентный спуск в пространстве распознавателей
 - 5.2.1. Регрессия
 - 5.2.2. Логистическая регрессия
 - 5.2.3. Двухклассовая классификация; AdaBoost как минимизация экспоненциальной ошибки
 - 5.2.4. Многоклассовая классификация
- 5.3. Оптимизация голосования распознавателей
- 5.4. Введение случайностей в обучение

Глава 6. Предварительное заключение

- 6.1. Сравнительный обзор рассмотренных методов распознавания
- 6.2. Не рассмотренные постановки задач распознавания и методы обучения распознавателей

Приложения. Пропущенные данные и метод максимизации ожидания

- A.1. Пропущенные данные
- A.2. Распознавание и обучение с пропущенными данными
 - A.2.1. Пример: наивное байесовское обучение с пропущенными данными
 - A.2.2. Порождающие модели и пропущенные данные
- A.3. Метод максимизации ожидания
 - A.3.1. Обозначения
 - A.3.2. Алгоритм EM для обучения порождающей модели
 - A.3.3. Примеры обучения с помощью алгоритма EM
 - A.3.4. Сходимость алгоритма EM
 - A.3.5. Обобщения алгоритма EM

Литература

Введение

Этот раздел является чем-то вроде аннотированного и иллюстрированного, но не претендующего на полноту, предметного указателя по методам статистического обучения. Некоторые важные понятия только декларируются, а часть из них разъясняется в последующих разделах. Неполнота усугубляется тем, что в этой еще не стабилизировавшейся области деятельности одни и те же понятия часто называют разными словами и по-разному переводят на другие языки. Наряду с русскими терминами, иногда неканоническими, приводятся их англоязычные прототипы.

Типичная задача статистического обучения (*statistical learning*, или *machine learning*, или *pattern recognition*, традиционный неправильный перевод последнего варианта -- *распознавание образов*) в самом общем виде выглядит так. Есть некоторое количество объектов, с какими-то *наблюдаемыми свойствами* и какими-то *ненаблюдаемыми, но известными*. Необходимо построить алгоритм, правильно вычисляющий эти *ненаблюдаемые* свойства по этим *наблюдаемым*, причем не только для заранее предъявленных объектов, но и для любых других. Или хотя бы алгоритм, *ошибающийся* не очень часто и не очень сильно. Причем нужно не однократно построить такой алгоритм, а создать алгоритм более высокого уровня (мета-алгоритм, метод, технологию, ...), строящий вычисляющий алгоритм по любому предъявленному набору объектов, чтобы этот алгоритм часто получался *приемлемым*. Саму задачу будем называть *распознаванием (recognition)*, решающий ее алгоритм -- *распознавателем (recognizer, learner)*, а построение этого алгоритма -- *обучением (learning, training, fitting)* распознавателя.

Для формализации постановки задачи нужно придать смысл всем выделенным словам предыдущего абзаца. Но сперва приведем неформальные примеры таких задач.

Узнавание образцов. Имеется некоторое количество картинок, на каждой из которых нарисована кошка (или треугольник, или жираф, или самолет, или конкретный человек), и, возможно, некоторое количество картинок, на каждой из которых она (он) отсутствует. Построить алгоритм, определяющий наличие кошки (треугольника и т.д.) на картинке.

Распознавание рукописных букв. Имеется некоторое количество картинок, на каждой из которых нарисована буква, и известно, какая именно. Построить алгоритм, узнающий нарисованные буквы.

Распознавание рукописного текста. Имеется некоторое количество отсканированных страниц рукописей и параллельных текстовых файлов с правильно прочитанным содержанием рукописей. Построить алгоритм, читающий рукописный текст с листа.

Распознавание голосовых команд. Имеется некоторое количество звуковых файлов, содержащих записи произнесения (*utterance*) голосовых команд из конечного ассортимента (например, слов "да" и "нет" или цифр для голосового набора номера), и знание, в каком файле какая команда. Построить алгоритм, понимающий голосовые команды.

Распознавание речи. Имеется некоторое количество звуковых файлов, содержащих записи естественной речи на каком-то языке, и текстовых файлов -- их расшифровок. Построить алгоритм, распознающий речь и записывающий ее в виде текста.

Медицинская диагностика. Имеется некоторое количество историй болезни и приложенных к ним результатов обследований больных. Построить алгоритм, по результатам обследований нового больного ставящий ему диагноз, назначающий лечение и/или прогнозирующий результаты лечения. Или, что более реально, подсказывающий врачу наиболее правдоподобные диагнозы.

Геологическая диагностика. Про некоторое количество разработанных нефтяных месторождений известны данные их предварительной геологической разведки (например, сейсмограммы) и результаты их эксплуатации. Построить алгоритм, предсказывающий эксплуатационные характеристики, в первую очередь мощность, разведанных, но еще не вскрытых месторождений.

Экономическое прогнозирование. Имеются данные о еженедельных объемах продаж нескольких тысяч видов товаров в нескольких сотнях магазинов за несколько лет. Построить алгоритм, предсказывающий спрос на ближайший месяц.

Кроме распознавания теория статистического обучения занимается и другими задачами, например задачей *кластеризации* (синоним: кластерный анализ, *cluster analysis*). Есть некоторое количество *объектов* с какими-то *наблюдаемыми свойствами*. Надо построить алгоритм, разбивающий объекты на группы, называемые *кластерами* (*cluster*), и алгоритм, определяющий, какому кластеру принадлежит объект, так чтобы объекты внутри каждого кластера были *похожи друг на друга*, а объекты из разных кластеров -- *непохожи*, причем чтобы это было как правило (т.е. статистически) верно не только для заранее предъявленных объектов, но и для любых других, предъявляемых впоследствии. Кластеризация и ее аналоги часто возникают в качестве вспомогательных задач в распознавании, сжатии данных и др.

В дальнейшем обсуждается в основном обучение обучающихся алгоритмов, причем достаточно общих, не использующих специфики задачи. Подробности того, что предшествует обучению, как можно его проводить, что за ним следует и как все это интерпретировать с теоретико-вероятностной точки зрения, хорошо описаны в книгах [12,13]. Теоретические оценки ошибок распознавания, гарантирующие, что обучение вообще возможно, подробно рассмотрены в книге [80]. Основы теории и классические методы распознавания изложены в книге [83]. Практически в любой толстой книге, содержащей в названии слова "pattern recognition", "machine learning" (например, [13]) или "statistical learning" (например, [46] или [79]), описаны разные распознающие системы, способы их обучения, примеры применения и оценки качества работы, хотя одни и те же вещи в разных книгах могут называться по-разному.

Об авторе



Александр Борисович МЕРКОВ (род. в 1957 г.)

Окончил механико-математический факультет МГУ им.М.В.Ломоносова в 1979 г. и поступил на работу во Всесоюзный научно-исследовательский институт системных исследований (ныне -- Институт системного анализа РАН), где работает по настоящее время. Занимался системным программированием, качественной теорией уравнений в частных производных, разработкой систем управления базами данных (СУБД), маломерной топологией. С 2000 г. -- заведующий учебной лабораторией распознавания в Московском центре непрерывного математического образования.