

Содержание

Предисловие ко второму изданию	18
Предисловие к первому изданию	21
Глава 1. Введение	23
Пример 1. Спам по электронной почте	24
Пример 2. Рак простаты	24
Пример 3. Распознавание рукописных цифр	26
Пример 4. Микрочипы экспрессии ДНК	26
Для кого предназначена книга	27
Как организована эта книга	29
Веб-сайт книги	30
Примечание для преподавателей	30
Глава 2. Обзор методов обучения с учителем	31
2.1. Введение	31
2.2. Виды переменных и терминология	31
2.3. Два простых подхода к предсказанию: методы наименьших квадратов и ближайших соседей	33
2.3.1. Линейные модели и метод наименьших квадратов	33
2.3.2. Метод ближайших соседей	36
2.3.3. От метода наименьших квадратов к методу ближайших соседей	39
2.4. Теория статистических решений	40
2.5. Локальные методы в больших измерениях	45
2.6. Статистические модели, обучение с учителем и приближение функций	50
2.6.1. Статистическая модель для совместного распределения $Pr(X, Y)$	51
2.6.2. Обучение с учителем	52
2.6.3. Аппроксимация функций	52
2.7. Структурированные модели регрессии	55
2.7.1. Сложность задачи	55
2.8. Классы ограниченных оценок	57
2.8.1. Штрафование негладкости и байесовские методы	57
2.8.2. Ядерные методы и локальная регрессия	58
2.8.3. Базисные функции и методы словарей	59
2.9. Выбор модели и компромисс между смещением и дисперсией	60
Библиографические заметки	62
Упражнения	62

Глава 3. Линейные методы регрессии	65
3.1. Введение	65
3.2. Модели линейной регрессии и наименьших квадратов	65
3.2.1. Пример: рак предстательной железы	71
3.2.2. Теорема Гаусса–Маркова	73
3.2.3. Множественная регрессия из простой одномерной регрессии	75
3.2.4. Множественные отклики	78
3.3. Выбор подмножества	79
3.3.1. Выбор наилучшего подмножества	80
3.3.2. Прямой и обратный выбор подмножества	81
3.3.3. Прямая ступенчатая регрессия	83
3.3.4. Пример данных о раке предстательной железы (продолжение)	83
3.4. Методы сжатия	86
3.4.1. Гребневая регрессия	86
3.4.2. Метод LASSO	91
3.4.3. Обсуждение: выбор наилучшего подмножества, гребневая регрессия и LASSO	92
3.4.4. Регрессия наименьших углов	96
3.5. Методы, использующие направления, определяемые по входным данным	102
3.5.1. Регрессия на главные компоненты	103
3.5.2. Метод частичных наименьших квадратов	104
3.6. Обсуждение: сравнение методов выбора и сжатия	106
3.7. Выбор и сжатие множественных откликов	108
3.8. Подробнее об алгоритме LASSO и его модификациях	110
3.8.1. Последовательная прямая ступенчатая регрессия	110
3.8.2. Кусочно-линейные алгоритмы	113
3.8.3. Селектор Данцига	113
3.8.4. Групповой метод LASSO	114
3.8.5. Другие свойства метода LASSO	115
3.8.6. Покоординатная оптимизация траекторий	117
3.9. Вычислительные вопросы	118
Библиографические заметки	118
Упражнения	118
Глава 4. Линейные методы классификации	125
4.1. Введение	125
4.2. Линейная регрессия индикаторной матрицы	126
4.3. Линейный дискриминантный анализ	130

8 СОДЕРЖАНИЕ

4.3.1. Регуляризованный дискриминантный анализ	136
4.3.2. Вычисления в методе LDA	137
4.3.3. Линейный дискриминантный анализ пониженного ранга	138
4.4. Логистическая регрессия	142
4.4.1. Обучение моделей логистической регрессии	144
4.4.2. Пример: ишемическая болезнь сердца в Южной Африке	146
4.4.3. Квадратичные аппроксимации и вывод	149
4.4.4. Регуляризованная логистическая регрессия L_1	150
4.4.5. Логистическая регрессия или LDA?	151
4.5. Разделяющие гиперплоскости	153
4.5.1. Алгоритм обучения перцептрона Розенблатта	155
4.5.2. Оптимальное разделение гиперплоскостей	156
Библиографические заметки	159
Упражнения	159
Глава 5. Разложение по базису и регуляризация	163
5.1. Введение	163
5.2. Кусочно-полиномиальные функции и сплайны	165
5.2.1. Естественные кубические сплайны	168
5.2.2. Пример: сердечно-сосудистые заболевания в Южной Африке (продолжение)	169
5.2.3. Пример: распознавание фонем	172
5.3. Фильтрация и выбор признаков	174
5.4. Сглаживание сплайнов	174
5.4.1. Степени свободы и матрицы сглаживания	176
5.5. Автоматический выбор параметров сглаживания	181
5.5.1. Фиксация числа степеней свободы	181
5.5.2. Компромисс между смещением и дисперсией	181
5.6. Непараметрическая логистическая регрессия	184
5.7. Многомерные сплайны	185
5.8. Регуляризация и гильбертовы пространства с воспроизводящим ядром	191
5.8.1. Пространства функций, генерируемых ядрами	191
5.8.2. Примеры пространств RKHS	193
Полиномиальная регрессия со штрафом	194
Гауссовы радиальные базисные функции	195
Классификаторы опорных векторов	197
5.9. Вейвлет-сглаживание	197
5.9.1. Вейвлет-базисы и вейвлет-преобразование	199

5.9.2. Адаптивная вейвлет-фильтрация	202
Библиографические заметки	204
Упражнения	205
Приложение: расчеты для сплайнов	209
В-сплайны	209
Расчеты для сглаживания сплайнов	211
Глава 6. Ядерные методы сглаживания	213
6.1. Сглаживатели с одномерным ядром	213
6.1.1. Локальная линейная регрессия	216
6.1.2. Локальная полиномиальная регрессия	219
6.2. Выбор ширины ядра	220
6.3. Локальная регрессия в пространстве \mathbb{R}^p	222
6.4. Модели структурированной локальной регрессии в пространстве \mathbb{R}^p	224
6.4.1. Структурированные ядра	225
6.4.2. Функции структурированной регрессии	225
6.5. Локальное правдоподобие и другие модели	226
6.6. Ядерная оценка плотности и классификация	230
6.6.1. Оценка плотности ядра	230
6.6.2. Классификация с помощью ядерной плотности	231
6.6.3. Наивный байесовский классификатор	233
6.7. Радиальные базисные функции и ядра	234
6.8. Модели смесей для оценки плотности и классификации	236
6.9. Вычислительные вопросы	238
Библиографические заметки	239
Упражнения	239
Глава 7. Оценивание и выбор моделей	243
7.1. Введение	243
7.2. Смещение, дисперсия и сложность модели	243
7.3. Разложение на смещение и дисперсию	247
7.3.1. Пример: компромисс смещения	249
7.4. Оптимизм, связанный с уровнем ошибок обучения	251
7.5. Оценки внутривыборочной ошибки предсказания	254
7.6. Эффективное число параметров	255
7.7. Байесовский подход и критерий BIC	257
7.8. Минимальная длина описания	259
7.9. Размерность Вапника–Червоненкиса	261
7.9.1. Пример (продолжение)	264

10 СОДЕРЖАНИЕ

7.10. Перекрестная проверка	265
7.10.1. К-блочная перекрестная проверка	266
7.10.2. Неправильный и правильный способ перекрестной проверки	269
7.10.3. Перекрестная проверка действительно работает?	271
7.11. Методы бутстрэпа	274
7.11.1. Пример (продолжение)	277
7.12. Условная или ожидаемая ошибка тестирования?	279
Библиографические заметки	280
Упражнения	282
Глава 8. Вывод моделей и усреднение	285
8.1. Введение	285
8.2. Методы бутстрэп и максимального правдоподобия	285
8.2.1. Пример сглаживания	285
8.2.2. Вывод методом максимального правдоподобия	288
8.2.3. Сравнение методов бутстрэп и максимального правдоподобия	291
8.3. Байесовские методы	291
8.4. Связь между бутстрэп-методом и байесовским выводом	294
8.5. EM-алгоритм	296
8.5.1. Модель двухкомпонентной смеси	296
8.5.2. EM-алгоритм в целом	300
8.5.3. EM-алгоритм как процедура совместной максимизации	301
8.6. Метод Монте-Карло по схеме марковских цепей для выбора из апостериорного распределения	302
8.7. Баггинг	306
8.7.1. Пример: деревья с искусственными данными	308
8.8. Усреднение и стекинг моделей	312
8.9. Стохастический поиск: бампинг	315
Библиографические заметки	317
Упражнения	317
Глава 9. Аддитивные модели, деревья и связанные с ними методы	319
9.1. Обобщенные аддитивные модели	319
9.1.1. Аппроксимация аддитивных моделей	321
9.1.2. Пример: аддитивная логистическая регрессия	323
9.1.3. Резюме	328
9.2. Древовидные методы	329
9.2.1. История вопроса	329

9.2.2. Деревья регрессии	331
9.2.3. Деревья классификации	333
9.2.4. Другие проблемы	334
Недостающие значения предиктора	335
9.2.5. Пример спама (продолжение)	337
9.3. PRIM: поиск пиков	342
9.3.1. Пример спама (продолжение)	345
9.4. MARS: многомерные адаптивные регрессионные сплайны	346
9.4.1. Пример о спама (продолжение)	350
9.4.2. Пример (смоделированные данные)	351
9.4.3. Другие задачи	352
Смешанные входные переменные	353
9.5. Иерархическое смешение мнений экспертов	354
9.6. Отсутствующие данные	357
9.7. Вычислительные вопросы	358
Библиографические заметки	359
Упражнения	359
Глава 10. Бустинг и аддитивные деревья	363
10.1. Методы бустинга	363
10.1.1. Краткое содержание этой главы	366
10.2. Аппроксимация аддитивной модели с помощью бустинга	367
10.3. Прямое ступенчатое аддитивное моделирование	368
10.4. Экспоненциальные потери и AdaBoost	369
10.5. Почему используются экспоненциальные функции потерь	371
10.6. Функции потерь и робастность	372
Надежные функции потерь для классификации	372
10.7. Стандартные процедуры для интеллектуального анализа данных	376
10.8. Пример: данные о спама	379
10.9. Бустинг деревьев	382
10.10. Численная оптимизация с помощью градиентного бустинга	384
10.10.2. Градиентный бустинг	385
10.10.3. Реализация градиентного бустинга	387
10.11. Правильный размер деревьев для бустинга	388
10.12. Регуляризация	391
10.12.1. Сжатие	391
10.12.2. Подвыборка	393
10.13. Интерпретация	393
10.13.1. Относительная важность предикторов	393

12 СОДЕРЖАНИЕ

10.13.2. Графики частичной зависимости	395
10.14. Примеры	397
10.14.1. Рынок жилья в Калифорнии	397
10.14.2. Новозеландская рыба	401
10.14.3. Демографические данные	406
Библиографические заметки	408
Упражнения	410
Глава 11. Нейронные сети	415
11.1. Введение	415
11.2. Метод регрессионного поиска проекции	415
11.3. Нейронные сети	418
11.4. Обучение нейронных сетей	421
11.5. Некоторые проблемы обучения нейронных сетей	423
11.5.1. Начальные значения	423
11.5.2. Переобучение	424
11.5.3. Масштабирование входных данных	426
11.5.4. Количество скрытых элементов и слоев	426
11.5.5. Несколько минимумов	427
11.6. Пример: искусственные данные	427
11.7. Пример: данные о почтовом индексе	430
11.8. Обсуждение	434
11.9. Байесовские нейронные сети и конкурс NIPS 2003	435
11.9.1. Байесовский подход, бустинг и баггинг	436
11.9.2. Сравнение эффективности	438
11.10. Вычислительные вопросы	441
Библиографические заметки	441
Упражнения	442
Глава 12. Метод опорных векторов и гибкие дискриминанты	445
12.1. Введение	445
12.2. Классификатор опорных векторов	445
12.2.1. Вычисление классификатора опорных векторов	447
12.2.2. Пример о смеси (продолжение)	449
12.3. Машины и ядра опорных векторов	449
12.3.1. Вычисление SVM для классификации	451
12.3.2. SVM как метод штрафа	452
12.3.3. Оценка функций и воспроизводящие ядра	456
12.3.4. SVM и проклятие размерности	457

12.3.5. Алгоритм поиска пути для классификатора SVM	460
12.3.6. Метод опорных векторов для регрессии	462
12.3.7. Регрессия и ядра	464
12.3.8. Обсуждение	465
12.4. Обобщающий линейный дискриминантный анализ	466
12.5. Гибкий дискриминантный анализ	468
12.5.1. Вычисление оценок FDA	472
12.6. Дискриминантный анализ со штрафами	474
12.7. Смешанный дискриминантный анализ	475
12.7.1. Пример: данные о форме волны	479
Вычислительные вопросы	483
Библиографические заметки	483
Епражнения	483
Глава 13. Методы прототипов и ближайших соседей	487
13.1. Введение	487
13.2. Методы прототипов	487
13.2.1. Кластеризация с помощью метода k -средних	488
13.2.2. Квантование обучающих векторов	490
13.2.3. Смеси нормальных распределений	490
13.3. Классификаторы по методу ближайших соседей	491
13.3.1. Пример: сравнительное исследование	496
13.3.2. Пример: метод k ближайших соседей и классификация сцен на изображениях	497
13.3.3. Инвариантные метрики и касательное расстояние	499
13.4. Адаптивные методы ближайшего соседа	503
13.4.1. Пример	505
13.4.2. Сокращение глобальной размерности для ближайших соседей	506
13.5. Вычислительные вопросы	508
Библиографические заметки	509
Упражнения	509
Глава 14. Обучение без учителя	513
14.1. Введение	513
14.2. Ассоциативные правила	515
14.2.1. Анализ рыночной корзины	516
14.2.2. Алгоритм Apriori	517
14.2.3. Пример: анализ рыночной корзины	520

14 СОДЕРЖАНИЕ

14.2.4. Сведение задачи обучения без учителя к задаче обучения с учителем	523
14.2.5. Обобщенные ассоциативные правила	525
14.2.6. Выбор метода обучения с учителем	527
14.2.7. Пример: анализ рыночной корзины (продолжение)	528
14.3. Кластерный анализ	530
14.3.1. Матрицы близости	531
14.3.2. Различия, основанные на атрибутах	532
14.3.3. Различие между объектами	533
14.3.4. Алгоритмы кластеризации	535
14.3.5. Комбинаторные алгоритмы	536
14.3.6. Алгоритм k -средних	538
14.3.7. Смеси нормальных распределений как мягкий вариант алгоритма кластеризации k -средних	539
14.3.8. Пример: данные микрочипа опухоли человека	540
14.3.9. Квантование вектора	542
14.3.10. Метод K медоидов	544
14.3.11. Практические вопросы	547
14.3.12. Иерархическая кластеризация	548
14.4. Самоорганизующиеся карты	557
14.5. Главные компоненты, кривые и поверхности	563
14.5.1. Главные компоненты	563
14.5.2. Главные кривые и поверхности	570
14.5.3. Спектральная кластеризация	573
14.5.4. Ядерные главные компоненты	575
14.5.5. Разреженные главные компоненты	579
14.6. Неотрицательная матричная факторизация	581
14.6.1. Анализ архетипов	584
14.7. Метод независимых компонентов и разведочный поиск наилучшей проекции	586
14.7.1. Скрытые переменные и факторный анализ	587
14.7.2. Анализ независимых компонентов	589
14.7.3. Разведочный поиск наилучшей проекции	594
14.7.4. Прямой подход к методу ICA	596
14.8. Многомерное шкалирование	600
14.9. Нелинейное уменьшение размерности и локальное многомерное шкалирование	602
14.10. Алгоритм Google PageRank	606
Библиографические заметки	608
Упражнения	609

Глава 15. Случайные леса	617
15.1. Введение	617
15.2. Определение случайных лесов	617
15.3. Детали случайных лесов	621
15.3.1. Выборки, не вошедшие в набор	622
15.3.2. Значимость переменной	623
15.3.3. Диаграмма близости	623
15.3.4. Случайные леса и переобучение	625
15.4. Анализ случайных лесов	627
15.4.1. Дисперсия и эффект декорреляции	627
15.4.2. Смещение	629
15.4.3. Адаптивный метод ближайших соседей	631
Библиографические заметки	632
Упражнения	632
Глава 16. Ансамблевые методы обучения	635
16.1. Введение	635
16.2. Бустинг и пути регуляризации	636
16.2.1. Регрессия со штрафом	637
16.2.2. Ставка на разреженность	640
16.2.3. Пути регуляризации, переоснащение и поля	643
16.3. Обучение ансамблей	647
16.3.1. Обучение хорошего ансамбля	648
16.3.2. Ансамбли правил	651
Библиографические заметки	654
Упражнения	654
Глава 17. Неориентированные графовые модели	655
17.1. Введение	655
17.2. Марковские графы и их свойства	656
17.3. Неориентированные графовые модели для непрерывных переменных	660
17.3.1. Оценка параметров при известной структуре графа	661
17.3.2. Оценка структуры графа	664
17.4. Неориентированные графовые модели для дискретных переменных	668
17.4.1. Оценка параметров при известной структуре графа	669
17.4.2. Скрытые узлы	671
17.4.3. Оценка структуры графа	672

17.4.4. Ограниченные машины Больцмана	673
Библиографические заметки	676
Упражнения	676
Глава 18. Задачи высокой размерности: $p \gg N$	681
18.1. Когда p намного больше, чем N	681
18.2. Диагональный линейный дискриминантный анализ и классификация по ближайшему сжатому центроиду	683
18.3. Линейные классификаторы с квадратичной регуляризацией	686
18.3.1. Регуляризованный дискриминантный анализ	688
18.3.2. Логистическая регрессия с квадратичной регуляризацией	689
18.3.3. Классификатор опорных векторов	690
18.3.4. Выбор признаков	691
18.3.5. Вычислительные приемы при $p \gg N$	691
18.4. Линейные классификаторы с L_1 -регуляризацией	693
18.4.1. Применение метода для масс-спектрометрии белков	697
18.4.2. Метод склеенного LASSO для функциональных данных	699
18.5. Классификация, когда признаки недоступны	701
18.5.1. Пример: строковые ядра и классификация белков	701
18.5.2. Классификация и другие модели с использованием ядер скалярных произведений и попарных расстояний	703
18.5.3. Пример: классификация аннотаций	705
18.6. Регрессия в пространствах большой размерности: метод главных компонентов с учителем	707
18.6.1. Связь с моделированием латентных переменных	712
18.6.2. Связь с методом частичных наименьших квадратов	713
18.6.3. Предобуславливание для выбора признаков	715
18.7. Оценка признаков и проблема множественного тестирования	717
18.7.1. Доля ошибочных отклонений	720
18.7.2. Асимметричные точки отсечения и процедура SAM	723
18.7.3. Байесовская интерпретация FDR	725
18.8. Библиографические заметки	726
Упражнения	727
Библиография	733
Предметный указатель	755

Случайные леса

15.1. Введение

В разделе 8.7 мы рассмотрели *баггинг*, или *бутстрэп-агрегацию* — метод уменьшения дисперсии оценочной функции предсказания. Оказалось, что баггинг особенно хорошо подходит для процедур с высокой дисперсией и низким смещением, таких как деревья. При решении задач регрессии мы многократно аппроксимируем одно и то же дерево регрессии по обучающим бутстрэп-версиям и усредняем результат. При решении задач классификации за прогнозируемый класс голосует *комитет* деревьев.

В главе 10 бустинг был первоначально предложен также в качестве метода комитетов, хотя в отличие от баггинга комитет *слабых учеников* со временем эволюционирует, и голоса членов имеют разные веса. Оказалось, что бустинг превосходит баггинг при решении большинства задач и стал предпочтительным выбором.

Случайные леса (random forests) (Breiman, 2001) — существенная модификация баггинга, которая создает большую коллекцию *декоррелированных* деревьев, а затем усредняет их. При решении многих задач точность случайных лесов сопоставима с точностью бустинга, но случайные леса проще обучать и настраивать. Как следствие, случайные леса стали популярными и реализованы в различных пакетах.

15.2. Определение случайных лесов

Основная идея баггинга (см. раздел 8.7) состоит в том, чтобы усреднить большое количество зашумленных, но приблизительно несмещенных моделей и тем самым уменьшить дисперсию. Деревья являются идеальными кандидатами для баггинга, так как могут отражать сложные структуры взаимодействия, скрытые в данных и, если их деревья достаточно глубокие, имеют относительно небольшое смещение. Поскольку общеизвестно, что деревья имеют сильный шум, их усреднение приносит большую пользу. Более того, поскольку все деревья, генерируемые при баггинге, имеют одинаковое распределение, математическое ожидание среднего значения B таких деревьев совпадает с математическим ожиданием каждого из них. Это означает, что смещение деревьев при баггинге такое же, как и у отдельных деревьев (при бутстрэпе), и единственная надежда на улучшение заключается в уменьшении дисперсии. Этим баггинг отличается от бустинга, в котором деревья выращиваются адаптивным способом, чтобы устранить смещение, и, следовательно, не являются одинаково распределенными.

Алгоритм 15.1. Случайный лес для регрессии или классификации

1. Для $b = 1$ до B :
 - а). Извлечем бутстрэп-выборку Z^* размера N из обучающих данных.
 - б). Вырастим дерево случайного леса T_b по бутстрэп-данным, рекурсивно повторяя следующие шаги для каждого конечного узла дерева, пока не будет достигнут минимальный размер узла n_{min} .
 - i. Случайным образом выберем m переменных из p переменных.
 - ii. Выберем лучшую переменную или точку разделения среди m переменных.
 - iii. Разделяем узел на два дочерних узла.
2. Возвращаем ансамбль деревьев $\{T_b\}_1^B$.
 Регрессия: $\hat{f}_{rf}^B(x) = \frac{1}{B} \sum_{b=1}^B T_b(x)$.
3. Классификация: пусть \hat{C}_b — прогнозируемый класс b -го дерева случайного леса. Тогда $\hat{C}_{rf}^B = \text{большинство голосов } \{\hat{C}_b(x)\}_1^B$.

Среднее значение B одинаково распределенных случайных величин, каждая из которых имеет дисперсию σ^2 , имеет дисперсию $\frac{1}{B}\sigma^2$. Если переменные одинаково распределены (но не обязательно независимы) с положительной попарной корреляцией ρ , то дисперсия среднего значения (см. упражнение 15.1) равна

$$\rho\sigma^2 + \frac{1-\rho}{B}\sigma^2. \quad (15.1)$$

При увеличении B второе слагаемое исчезает, но первое остается, и, следовательно, величина корреляции пар деревьев при баггинге ограничивает преимущества усреднения. Идея метода случайных лесов (см. алгоритм 15.1) состоит в том, чтобы снизить уменьшение дисперсии баггинга за счет уменьшения корреляции между деревьями, не слишком сильно увеличивая дисперсию. Это достигается в процессе построения деревьев путем случайного выбора входных переменных.

В частности, при построении дерева на бутстрэп-множестве данных необходимо выполнить следующую операцию.

Перед каждым разделением выберите $t \leq p$ случайных входных переменных в качестве кандидатов на расщепление.

Обычно значения t равны \sqrt{p} или единице.

После построения таких деревьев $\{T(x; \Theta_b)\}_1^B$ предиктор случайного леса (для регрессии) примет вид

$$\hat{f}_{rf}^B(x) = \frac{1}{B} \sum_{b=1}^B T(x; \Theta_b). \quad (15.2)$$

Как и в разделе 10.9, Θ_b характеризует b -е дерево случайного леса в терминах расщепляющих переменных, точек отсечения в каждом узле и значений терминальных узлов. Интуитивно понятно, что уменьшение m уменьшит корреляцию между любой парой деревьев в ансамбле и, следовательно, (15.1) уменьшит дисперсию среднего значения.

Не все оценки можно улучшить путем возмущения данных, как в данном случае. Оказывается, сильно нелинейные средства оценивания, такие как деревья, приносят наибольшую пользу. Для бутстрэп-деревьев величина ρ обычно невелика (около 0,05 или ниже; см. рис. 15.9), тогда как σ^2 не намного больше, чем дисперсия для исходного дерева. С другой стороны, баггинг не меняет *линейные* оценки, такие как выборочное среднее (а значит, и дисперсию); попарная корреляция между средними значениями после бутстрэпа составляет около 50% (см. упражнение 15.4).

Случайные леса популярны. Лео Брайман (Leo Breiman)¹, сотрудник Адель Катлер (Adele Cutler), поддерживает сайт, посвященный методу случайного леса². Его программное обеспечение находится в свободном доступе, и к 2002 г. было зарегистрировано более 3000 загрузок. Кроме того, существует пакет **randomForest** в языке R, поддерживаемый Энди Лиу (Andy Liaw), доступный на сайте CRAN.

Авторы делают громкие заявления об успехе случайных лесов: “наиболее точные”, “наиболее интерпретируемые” и т.п. По нашему опыту, случайные леса работают замечательно и требуют совсем небольшой настройки. На тестовых данных о спаме уровень ошибочной классификации случайного леса снижается до 4,88%, что вполне сопоставимо со всеми другими методами и не намного хуже, чем градиентный бустинг, уровень ошибок которого составил 4,5%. Уровень ошибок баггинга составил 5,4%, что значительно хуже, чем у любого другого метода (с использованием теста Мак-Немара, описанного в упражнении 10.6), поэтому в этом примере дополнительная рандомизация оказалась полезной.

На рис. 15.1 показана прогрессия ошибок тестирования на 2500 деревьях для трех методов. В данном случае есть некоторые свидетельства того, что градиентный бустинг начал переобучаться, хотя 10-блочная перекрестная проверка выбрала все 2500 деревьев.

На рис. 15.2 показаны результаты моделирования³, сравнивающего случайные леса с градиентным бустингом при решении задачи о вложенных сферах (см. уравнение (10.2) в главе 10). Здесь бустинг намного превосходит случайные леса. Обратите

¹ Лео Брейман умер в июле 2005 года.

² <http://www.math.usu.edu/~adele/forests/>

³ Детали: случайные леса были обучены с использованием пакета **randomForest 4.5-11** языка R и 500 деревьев. Модели градиентного бустинга были обучены с использованием пакета **gbm 1.5** языка R с параметром сжатия, равным 0,05, и 2000 деревьев.

внимание на то, что чем меньше m , тем лучше, хотя одна из причин может заключаться в том, что истинная граница решения является аддитивной.

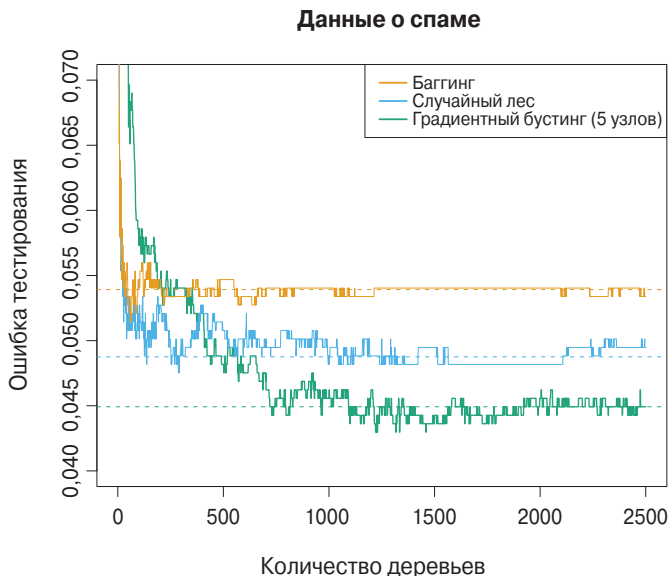


Рис. 15.1. Результаты применения баггинга, случайного леса и градиентного бустинга к данным о спаме. Для бустинга были использованы пятиузловые деревья, а количество деревьев было выбрано путем 10-блочной перекрестной проверки (2500 деревьев). Каждый шаг на рисунке соответствует изменению в одной неправильной классификации (в тестовом множестве из 1536 элементов)

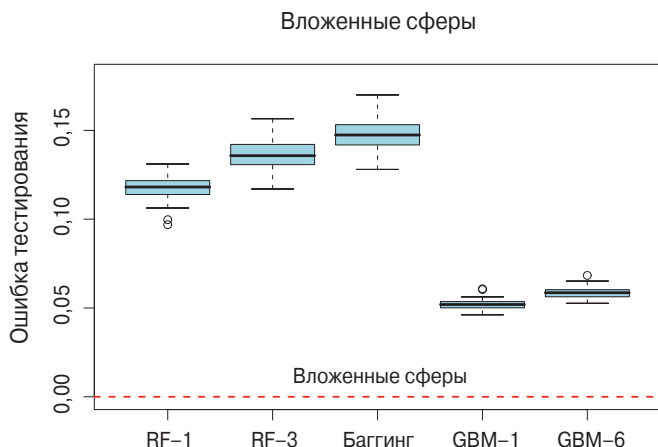


Рис. 15.2. Результаты 50 симуляций по модели вложенных сфер в пространстве \mathbb{R}^{10} . Граница байесовского решения — это поверхность сферы (аддитивная). Метка “RF-3” относится к случайному лесу с $m = 3$, а “GBM-6” — к модели с градиентным бустингом с порядком взаимодействия, равным шести; аналогично для “RF-1” и “GBM-1”. Обучающие множества содержали 2000 элементов, а тестовые множества — 10 000

На рис. 15.3 случайные леса сравниваются с бустингом (со сжатием) при решении регрессии с использованием данных о жилье в Калифорнии (см. раздел 10.14.1 главы 10). Здесь проявляются две сильные особенности.

- Случайные леса стабилизируются примерно на 200 деревьях, в то время как на 1000 деревьев бустинг продолжает улучшать точность. Бустинг замедляется из-за сжатия, а также из-за того, что деревья имеют намного меньшую глубину.
- Бустинг в данном случае превосходит случайные леса. При 1000 членах более слабая модель бустинга (GBM с глубиной, равной четырем) имеет меньшую ошибку, чем более сильный случайный лес (RF $m = 6$); р-значение критерия Уилкоксона об абсолютной разности между математическими ожиданиями равно 0,007. При больших m случайные леса работают не лучше.

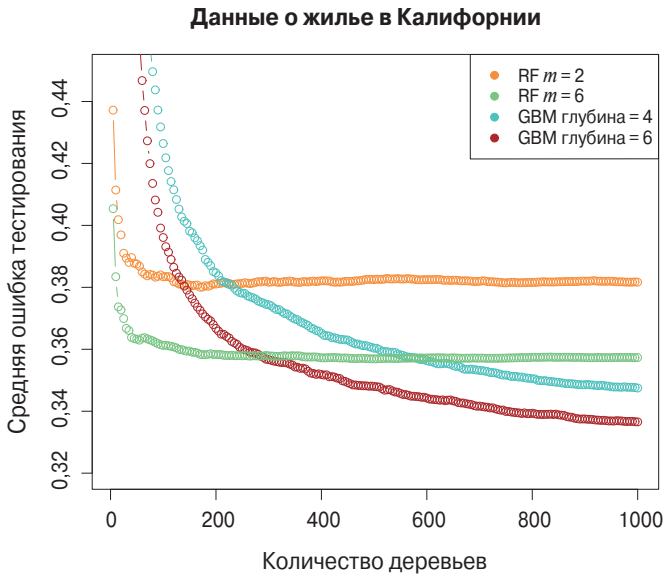


Рис. 15.3. Сравнение случайных лесов с градиентным бустингом на данных о жилье в Калифорнии. Кривые представляют среднюю абсолютную ошибку на тестовых данных как функцию от количества деревьев в моделях. Показаны два случайных леса с $m = 2$ и $m = 6$. Две модели градиентного бустинга используют параметр сжатия $\nu = 0,05$ из (10,41). Их глубины взаимодействия равны 4 и 6. Модели бустинга лучше, чем случайные леса

15.3. Детали случайных лесов

Мы не стали подчеркивать различие между случайными лесами при классификации и регрессии. При использовании для классификации случайный лес получает голос класса по каждому дереву, а затем классифицирует его по большинству голосов (аналогичное обсуждение см. в разделе 8.7 главы 8, посвященном баггингу). При использовании для регрессии прогнозы для каждого дерева в целевой точке x просто усредняются, как в (15.2). Кроме того, авторы дают следующие рекомендации.

- Для классификации значение m по умолчанию равно $\lfloor \sqrt{p} \rfloor$, а минимальный размер узла равен единице.
- Для регрессии значение m по умолчанию равно $\lfloor p/3 \rfloor$, а минимальный размер узла равен пяти.

На практике наилучшие значения этих параметров зависят от задачи, и их следует рассматривать как параметры настройки. На рис. 15.3 алгоритм намного лучше работает при $m = 6$, чем при значении по умолчанию, равном $\lfloor 8/3 \rfloor = 2$.

15.3.1. Выборки, не вошедшие в набор

Важной особенностью случайных лесов является использование *выборок, не вошедших в набор* (Out-Of-Bag — OOB).

Для каждого наблюдения $z_i = (x_i, y_i)$ создайте его предиктор случайного леса, усредняя только те деревья, которые соответствуют бутстрэп-выборкам, в которых наблюдение z_i не появлялось.

Оценка ошибки OOB почти идентична оценке, полученной с помощью N -блочной перекрестной проверки (см. упражнение 15.2). Следовательно, в отличие от многих других нелинейных оценок, случайные леса могут быть обучены одновременно с выполнением перекрестной проверки. Как только ошибка OOB стабилизируется, обучение можно прекратить.

На рис. 15.4 показан уровень ошибок классификации OOB для данных о спаме по сравнению с ошибкой тестирования. Хотя здесь усреднено 2500 деревьев, на графике видно, что около 200 уже будет достаточно.

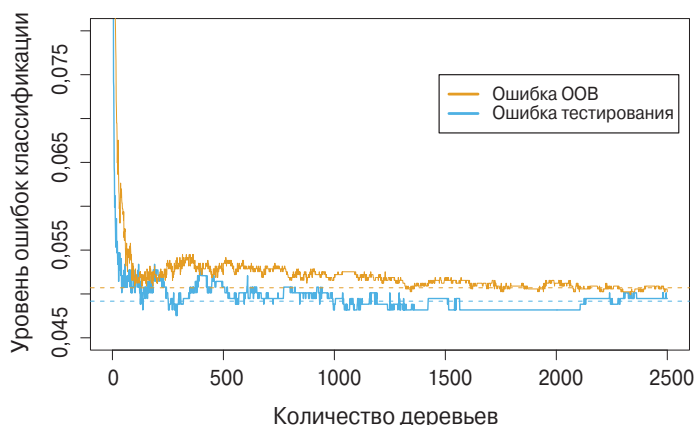


Рис. 15.4. Сравнение ошибки OOB, вычисленной на обучающих данных о спаме, с ошибкой тестирования, вычисленной на тестовом множестве

15.3.2. Значимость переменной

Так же как и для моделей с градиентным бустингом (см. раздел 10.13), для случайных лесов можно строить графики значимости переменных. При каждом разделении в каждом дереве улучшение критерия разделения является показателем значимости, который приписывается переменной разделения и накапливается по всем деревьям, входящим в лес, отдельно для каждой переменной. На левом графике на рис. 15.5 показаны уровни значимости переменных, рассчитанные таким образом по данным о спаме (сравните его с соответствующим рис. 10.6, построенным для градиентного бустинга). Бустинг полностью игнорирует некоторые переменные, а случайный лес — нет. Выбор подходящей переменной разделения увеличивает вероятность того, что любая отдельная переменная будет включена в случайный лес, в то время как при бустинге такой селекции не происходит.

Случайные леса также используют выборки ООВ, чтобы построить другую меру значимости переменных и тем самым измерить прогностическую силу каждой переменной. После того как b -е дерево построено, через него пропускают выборки ООВ и записывают точность прогноза. Затем значения j -й переменной в выборках ООВ случайным образом переставляются, и снова вычисляется точность. Уменьшение точности в результате этой перестановки усредняется по всем деревьям и используется как мера значимости переменной j в случайном лесу. Они выражены в процентах от максимума на правом графике на рис. 15.5. Хотя ранжирование в этих двух методах похоже, значимость переменных на правом графике выглядит более равномерной. Рандомизация эффективно аннулирует влияние переменной, аналогично обнулению коэффициента в линейной модели (см. упражнение 15.7). Она не измеряет влияние отсутствия переменной на прогноз, потому что, если модель была обучена без этой переменной, в качестве ее суррогатов могли использоваться другие переменные.

15.3.3. Диаграмма близости

Одним из рекламируемых результатов случайного леса является *диаграмма близости* (proximity plot). На рис. 15.6 показана диаграмма близости для смешанных данных, описанных в разделе 2.3.3 главы 2. При построении случайного леса для обучающих данных накапливается матрица близости $N \times N$. Для каждого дерева близость любой пары наблюдений ООВ, совместно использующих конечный узел, увеличивается на единицу. Эта матрица близости затем представляется в двух измерениях с использованием многомерного шкалирования (см. раздел 14.8). Идея состоит в том, что, хотя данные могут быть многомерными, включая смешанные переменные и т.д., диаграмма близости дает представление о том, какие наблюдения фактически близки с точки зрения классификатора на основе случайного леса.

Диаграммы близости для случайных лесов часто выглядят очень похожими, независимо от данных, что ставит под сомнение их полезность. Они, как правило, имеют форму звезды, по одному лучу на класс, которые тем четче выделены, чем лучше классификация.

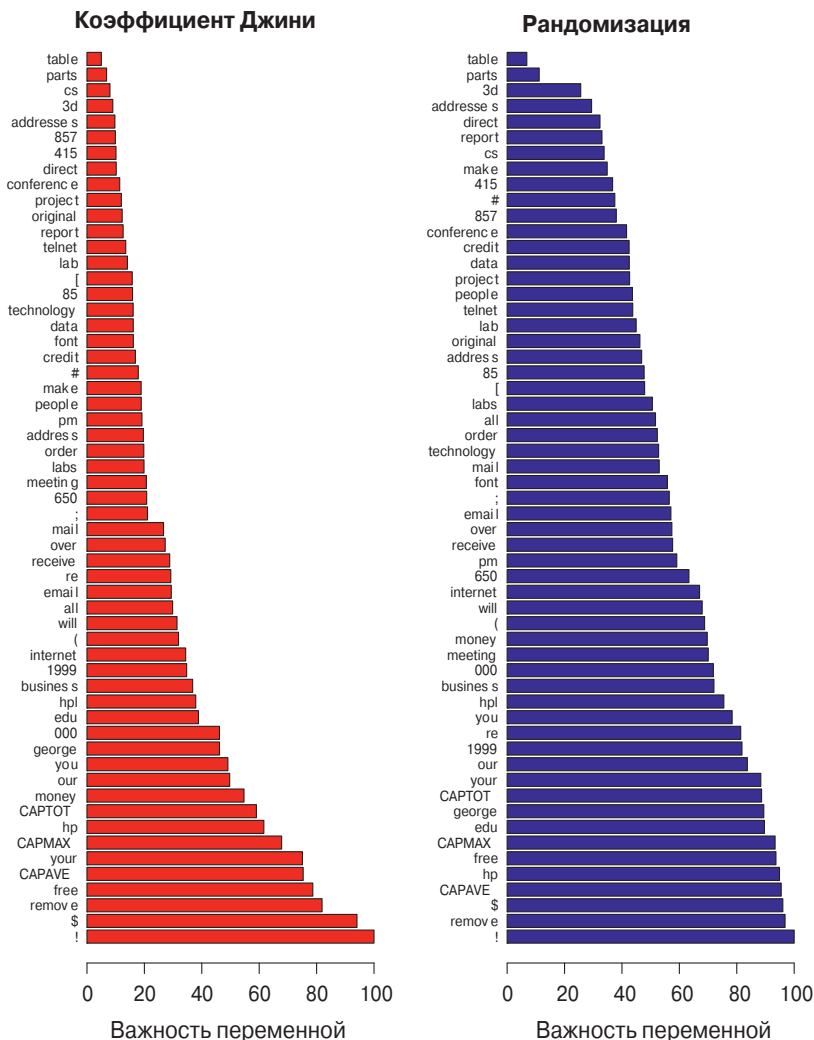


Рис. 15.5. Графики значимости переменных для классификации случайного леса, выращенного по данным о спаме. Левый график основывается на важности индекса расщепления Джини, как при градиентном бустинге. Рейтинги сопоставимы с рейтингами, полученными путем градиентного бустинга (см. рис. 10.6). На правом графике для вычисления значимости переменных использована случайная рандомизация, и значимость распределена более равномерно

Так как смешанные данные являются двумерными, мы можем отобразить точки на диаграмме близости в исходные координаты и лучше понять, что они собой представляют. Похоже, что точки в “чистых” областях отображаются по классам на лучи звезды, а точки, расположенные ближе к границам решения, отображаются ближе к центру. Это не удивительно, если изучить структуру матриц близости. Соседние точки в “чистых” областях часто оказываются в одном и том же сегменте, поскольку,

если конечный узел является “чистым”, он больше не разделяется алгоритмом построения деревьев случайного леса. С другой стороны, пары точек, которые близки, но принадлежат разным классам, иногда совместно используют терминальный узел, хотя и не всегда.

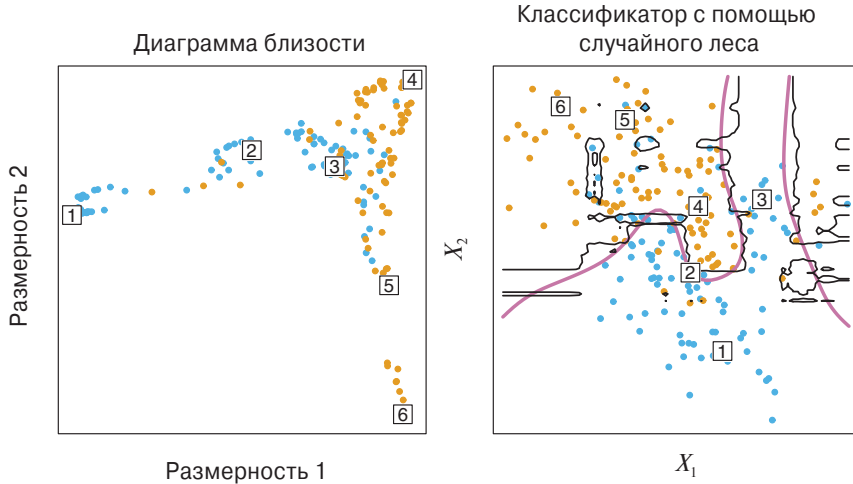


Рис. 15.6. Диаграмма близости для классификатора на основе случайного леса, построенного по смешанным данным (слева). Граница решения и обучающие данные для случайного леса, построенного по смешанным данным (справа). На каждой диаграмме были определены шесть точек

15.3.4. Случайные леса и переобучение

Если количество переменных велико, но доля релевантных переменных мала, то случайные леса, вероятно, будут плохо работать при малых m . При каждом разделении вероятность того, что будут выбраны соответствующие переменные, может быть небольшой. На рис. 15.7 показаны результаты моделирования, подтверждающего это утверждение. Подробности приведены в подписи к рис. 15.7 и упражнению 15.3. Сверху каждой пары мы видим гипергеометрическую вероятность того, что соответствующая переменная будет выбрана при любом разбиении дерева случайного леса (в этом моделировании все соответствующие переменные равны значимости). Как только эта вероятность становится небольшой, разрыв между бустингом и случайными лесами увеличивается. Когда количество релевантных переменных увеличивается, точность случайных лесов оказывается удивительно устойчивой к увеличению количества шумовых переменных. Например, с шестью релевантными и 100 шумовыми переменными вероятность выбора релевантной переменной при любом разделении равна 0,46, с учетом того, что $m = \sqrt{(6 + 100)} \approx 10$. Согласно рис. 15.7, это не влияет на точность метода случайного леса по сравнению с бустингом. Эта устойчивость в значительной степени обусловлена относительной нечувствительностью

стоимости ошибочной классификации к смещению и дисперсии оценок вероятности в каждом дереве. Мы рассмотрим случайные леса в контексте регрессии в следующем разделе.

Еще одно утверждение состоит в том, что случайные леса не могут переобучаться. Конечно, верно, что увеличение B не приводит к переобучению последовательности случайных лесов. Как и в случае с баггингом, оценка случайных лесов (15.2) аппроксимирует математическое ожидание

$$\hat{f}_{\text{rf}}(x) = E_{\Theta} T(x; \Theta) = \lim_{B \rightarrow \infty} \hat{f}(x)_{\text{rf}}^B \tag{15.3}$$

с помощью среднего значения по B реализаций Θ . Распределение Θ здесь зависит от обучающих данных. Однако *этот предел может переобучаться по данным*; среднее количество полностью построенных деревьев может привести к слишком богатой модели и ненужным отклонениям. Segal (2004) демонстрирует небольшой прирост точности, контролируя глубину отдельных деревьев, построенных в случайных лесах. Наш опыт показывает, что использование полноценных деревьев редко заслуживает внимания и приводит к уменьшению количества параметров настройки всего на единицу.

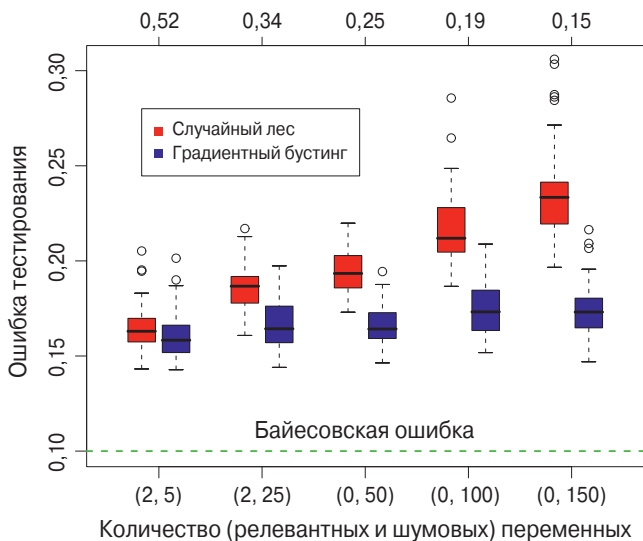


Рис. 15.7. Сравнение случайных лесов и градиентного бустинга на задачах с увеличивающимся количеством шумовых переменных. В каждом случае истинная граница решения зависит от двух переменных, и в задачу включается все больше шумовых переменных. Случайные леса используют значение по умолчанию $m = \sqrt{p}$. Сверху каждой пары указана вероятность того, что одна из релевантных переменных будет выбрана при любом разделении. Результаты основаны на 50 симуляциях для каждой пары, с обучающей выборкой, содержащей 300 элементов, и тестовой выборкой, содержащей 500 элементов. См. упражнение 15.3

На рис. 15.8 показан скромный эффект контроля глубины на простом примере регрессии. Классификаторы менее чувствительны к дисперсии, и этот эффект переобучения редко наблюдается при классификации с помощью случайных лесов.

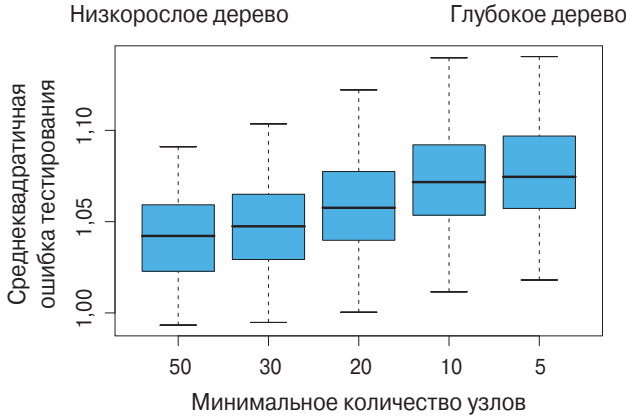


Рис. 15.8. Влияние размера дерева на ошибку регрессии с помощью случайного леса. В этом примере истинная поверхность была аддитивной по двум из 12 переменных, плюс аддитивная единичная дисперсия нормально распределенного шума. Глубина дерева здесь контролируется минимальным размером узла; чем меньше минимальный размер узла, тем глубже деревья

15.4. Анализ случайных лесов



В этом разделе анализируются механизмы дополнительной рандомизации, используемой в случайных лесах. В этом обсуждении мы сконцентрируемся на регрессии с квадратичной функцией потерь, поскольку анализ смещения и дисперсии для бинарной функции потерь намного сложнее (см. раздел 7.3.1). Кроме того, даже в случае классификации мы можем рассматривать среднее значение по случайному лесу как оценку апостериорных вероятностей класса, для которых смещение и дисперсия являются подходящими дескрипторами.

15.4.1. Дисперсия и эффект декорреляции

Предельная форма ($B \rightarrow \infty$) оценки регрессии по методу случайного леса имеет вид

$$\hat{f}_{rf}(x) = E_{\Theta|\mathbf{Z}} T(x; \Theta(\mathbf{Z})), \tag{15.4}$$

где мы подчеркнули зависимость от обучающих данных \mathbf{Z} . Здесь мы рассмотрим оценку в одной целевой точке x . Из (15.1) следует, что

$$\text{Var} \hat{f}_{rf}(x) = \rho(x) \sigma^2(x). \tag{15.5}$$

Здесь

- $\rho(x)$ — выборочная корреляция между любой парой деревьев, используемой при усреднении:

$$\rho(x) = \text{corr}[T(x; \Theta_1(\mathbf{Z})), T(x; \Theta_2(\mathbf{Z}))], \quad (15.6)$$

- где $\Theta_1(\mathbf{Z})$ и $\Theta_2(\mathbf{Z})$ — случайно извлеченная пара деревьев случайного леса, построенных по случайно выбранной переменной \mathbf{Z} ;
- $\sigma^2(x)$ — выборочная дисперсия любого произвольно построенного дерева

$$\sigma^2(x) = \text{Var} T(x; \Theta(\mathbf{Z})). \quad (15.7)$$

Величину $\rho(x)$ легко спутать со средней корреляцией между обученными деревьями в заданном ансамбле случайных лесов. Иначе говоря, обученные деревья можно интерпретировать как вектор из N элементов и вычислить среднюю попарную корреляцию между этими векторами, обусловленную данными. Это *неправильно*; эта условная корреляция не имеет прямого отношения к процессу усреднения, и зависимость $\rho(x)$ от x свидетельствует об этом отличии. Скорее $\rho(x)$ — это теоретическая корреляция между парой деревьев случайного леса, оцененных в точке x , индуцированная многократным извлечением обучающей выборки \mathbf{Z} из генеральной совокупности с последующим извлечением пары деревьев случайного леса. На статистическом жаргоне эта величина называется *выборочной корреляцией* (sampling distribution) переменных \mathbf{Z} и Θ .

Точнее говоря, переменная, усредненная по вычислениям (15.6) и (15.7), является

- зависимой от \mathbf{Z} : из-за бутстрэп-выборки и выборки признаков при каждом разделении
- и результатом изменчивости выборки самой переменной \mathbf{Z} .

Фактически условная ковариация пары деревьев, обученных в точке x , равна нулю, поскольку бутстрэп-выборка и выборка признаков являются независимыми и одинаково распределенными (см. упражнение 15.5).

Следующие примеры основаны на имитационной модели:

$$Y = \frac{1}{\sqrt{50}} \sum_{j=1}^{50} X_j + \varepsilon, \quad (15.8)$$

где все X_j и ε являются независимыми и одинаково нормально распределенными. Мы используем 500 обучающих выборок размером 100 и один набор тестовых точек размером 600. Поскольку деревья регрессии нелинейны относительно \mathbf{Z} , шаблоны, которые приведены ниже, будут несколько отличаться в зависимости от структуры модели.

На рис. 15.9 показано, как корреляция (15.6) между парами деревьев уменьшается с уменьшением m : пары предсказаний деревьев в точке x для разных обучающих наборов \mathbf{Z} , вероятно, будут менее похожими, если они не используют одинаковые переменные расщепления.

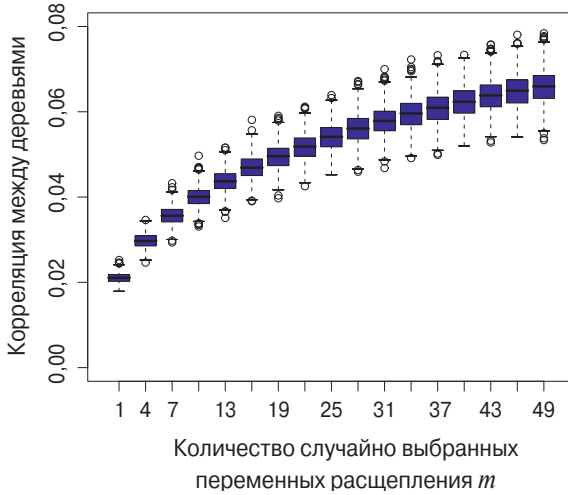


Рис. 15.9. Значения корреляции между парами деревьев, построенными алгоритмом регрессии с помощью случайного леса, в зависимости от m . Квадратные диаграммы представляют корреляции в 600 случайно выбранных точках прогнозирования x

На левой панели рис. 15.10 мы рассматриваем дисперсии предикторов одного дерева, $\text{Var}T(x; \Theta(\mathbf{Z}))$ (усредненные по 600 точкам прогнозирования x , взятым случайным образом из нашей имитационной модели). Это общая дисперсия, которая может быть разложена на две части с использованием стандартных аргументов условной дисперсии (см. упражнение 15.5):

$$\text{Var}_{\Theta|\mathbf{Z}}T(x; \Theta(\mathbf{Z})) = \text{Var}_{\mathbf{Z}}E_{\Theta|\mathbf{Z}}T(x; \Theta(\mathbf{Z})) + E_{\mathbf{Z}}\text{Var}_{\Theta|\mathbf{Z}}T(x; \Theta(\mathbf{Z})),$$

Общая дисперсия = $\text{Var}_{\mathbf{Z}}\hat{f}_{\text{rf}}(x)$ + *внутренняя дисперсия \mathbf{Z}* (15.9)

Второе слагаемое — это внутренняя дисперсия переменной \mathbf{Z} (within- \mathbf{Z} variance), представляющая собой результат рандомизации, который увеличивается с уменьшением m . Первое слагаемое фактически является выборочной дисперсией ансамбля случайных лесов (показан на правой панели), которая уменьшается с уменьшением m . Дисперсия отдельных деревьев не изменяется заметно в большей части диапазона m , поэтому в свете (15.5) дисперсия ансамбля значительно ниже, чем эта дисперсия дерева.

15.4.2. Смещение

Как и в случае с баггингом, смещение случайного леса такое же, как у любого отдельного дерева $T(x; \Theta(\mathbf{Z}))$:

$$\begin{aligned} \text{Bias}(x) &= \mu(x) - E_{\mathbf{Z}}\hat{f}_{\text{rf}}(x) = \\ &= \mu(x) - E_{\mathbf{Z}}E_{\Theta|\mathbf{Z}}T(x; \Theta(\mathbf{Z})). \end{aligned} \tag{15.10}$$

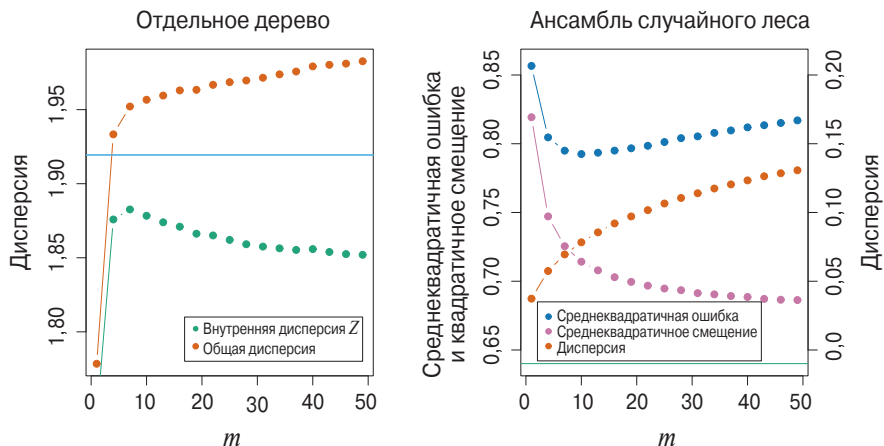


Рис. 15.10. Результаты моделирования. На левой панели показана средняя дисперсия одного дерева случайного леса как функция от m . Внутренняя дисперсия Z — это средний вклад внутри выборки в дисперсию, полученную в результате бутстрэпа и выборки разделяющей переменной (15.9). Общая дисперсия включает в себя изменчивость выборки Z . Горизонтальная линия представляет собой среднюю дисперсию одного полностью построенного дерева (без бутстрэпа). На правой панели показаны среднеквадратичная ошибка, квадратичное смещение и дисперсия ансамбля как функция от m . Обратите внимание на то, что ось отклонения находится справа (тот же масштаб, другой уровень). Горизонтальная линия представляет собой среднеквадратическое смещение полностью построенного дерева

Она также, как правило, больше (в абсолютном выражении), чем смещение необрезанного дерева, выращенного до Z , поскольку рандомизация и уменьшенное выборочное пространство накладывают свои ограничения. Следовательно, улучшения в прогнозировании, полученные с помощью баггинга или случайных лесов, являются *исключительно результатом уменьшения дисперсии*.

Любое обсуждение смещения зависит от неизвестной истинной функции. На рис. 15.1, *справа*, показан квадрат смещения для модели аддитивной модели (оценен по 500 реализациям). Хотя для разных моделей форма и скорость кривых смещения могут отличаться, общая тенденция заключается в том, что с уменьшением m смещение увеличивается. На рисунке показана среднеквадратичная ошибка, и мы видим классический компромисс между смещением и дисперсией при выборе m . Для всех m квадрат смещения случайного леса больше, чем для одного дерева (горизонтальная линия).

Эти закономерности предполагают сходство с гребневой регрессией (см. раздел 3.4.1). Гребневая регрессия полезна (в линейных моделях), когда имеется большое количество переменных с коэффициентами одинакового размера; она сжимает их коэффициенты к нулю, а коэффициенты сильно коррелированных переменных — друг к другу. Хотя размер обучающей выборки может не позволить включить в модель все переменные, эта регуляризация стабилизирует модель и позволяет всем переменным иметь свое влияние (хотя и уменьшенное). Случайные леса с небольшим m выполня-

ют аналогичное усреднение. Каждая из соответствующих переменных получает свою очередь для первичного разделения, а усреднение по ансамблю уменьшает вклад любой отдельной переменной. Так как этот пример моделирования (15.8) основан на линейной модели по всем переменным, гребневая регрессия достигает более низкой среднеквадратичной ошибки (около 0,45 с $df(\lambda_{\text{opt}}) \approx 29$).

15.4.3. Адаптивный метод ближайших соседей

Классификатор на основе случайного леса имеет много общего с классификатором по методу k ближайших соседей (см. раздел 13.3); фактически, его взвешенная версия. Поскольку каждое дерево строится до максимального размера, для конкретного Θ^* величина $T(x; \Theta^*(Z))$ является значением отклика для одной из обучающих выборок⁴. Алгоритм построения деревьев находит оптимальный путь к этому наблюдению, выбирая наиболее информативные предикторы из имеющихся в его распоряжении. Процесс усреднения присваивает веса этим обучающим ответам, которые в конечном итоге голосуют за прогноз. Следовательно, посредством механизма голосования в случайном лесу этим наблюдениям, близким к целевой точке, присваиваются веса (эквивалентное ядро), которые объединяются для формирования решения о классификации.

Рис. 15.11 демонстрирует сходство между границей решения по методу трех ближайших соседей и методом случайного леса на смешанных данных.

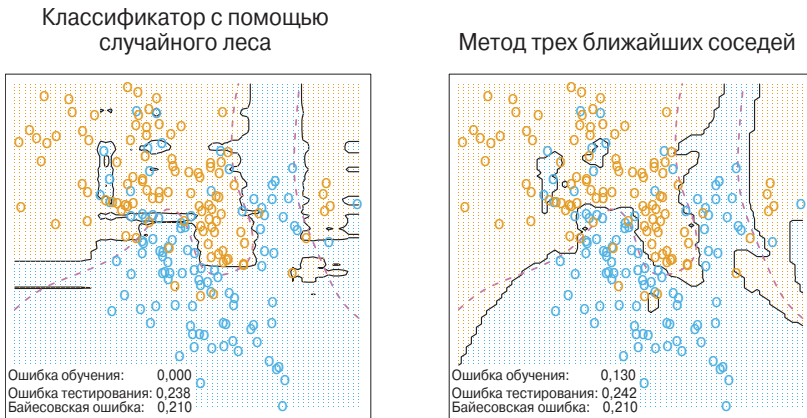


Рис. 15.11. Результаты применения метода случайного леса и метода 3NN к данным о смеси. Ориентация отдельных деревьев в случайном лесу вдоль осей приводит к областям решений с границами, ориентированными примерно по осям

⁴ Мы игнорируем тот факт, что чистые узлы не разделяются дальше, и, следовательно, в терминальном узле может быть более одного наблюдения.

Библиографические заметки

Случайные леса, описанные здесь, были введены в работе Breiman (2001), хотя многие идеи ранее встречались в литературе в разных формах. В частности, Но (1995) ввел термин “случайный лес” и использовал консенсус деревьев, выращенных в случайных подпространствах признаков. Идея использования стохастического возмущения и усреднения для избежания переобучения была предложена в работах Kleinberg (1990), а затем Kleinberg (1996). Amit and Geman (1997) использовали рандомизированные деревья, построенные на признаках изображений для задач классификации изображений. Breiman (1996a) изобрел баггинг — предшественника его версии случайных лесов. Dietterich (2000b) также предложил улучшение баггинга с помощью дополнительной рандомизации. Его подход состоял в том, чтобы ранжировать 20 лучших вариантов разделения в каждом узле, а затем выбирать разделение из списка случайным образом. Он показал с помощью моделирования и реальных примеров, что эта дополнительная рандомизация улучшает точность по сравнению с точностью баггинга. Friedman and Hall (2007) показали, что отбор проб (без замены) является эффективной альтернативой баггингу. Они показали, что построение и усреднение деревьев на выборках размера $N/2$ приблизительно эквивалентно (с точки зрения смещения/дисперсии) баггингу, в то время как использование меньших долей N еще больше уменьшает дисперсию (благодаря декорреляции).

Существует несколько бесплатных программных реализаций случайных лесов. В этой главе мы использовали пакет **randomForest** в языке R, поддерживаемый Энди Лиу (Andy Liaw), который доступен на веб-сайте CRAN. Он позволяет как выбирать переменную разделения, так и субдискретизацию. Адель Катлер поддерживает сайт случайных лесов <http://www.math.usu.edu/~adele/forests/>, где (по состоянию на август 2008 года) свободно распространяются программы, написанные Лео Брейманом и Адель Катлер. Их код и название “случайные леса” исключительно лицензированы компанией Salford Systems для коммерческого использования. Архив по машинному обучению **Weka** <http://www.cs.waikato.ac.nz/ml/weka/> в Университете Вайкато, Новая Зеландия, предлагает бесплатную Java-реализацию случайных лесов.

Упражнения

- 15.1. Выведите формулу для дисперсии (15.1). Она не работает, если ρ меньше нуля. Опишите проблему, которая возникает в этом случае.
- 15.2. Покажите, что по мере того, как количество бутстрэп-выборок B становится большим, оценка ошибки ООВ для случайного леса приближается к оценке ошибки N -блочной перекрестной проверки и что в пределе тождество является точным.
- 15.3. Рассмотрим имитационную модель, использованную на рис. 15.7 (Mease and Wuyner, 2008). Бинарные наблюдения генерируются с вероятностями

$$\Pr(Y = 1 | X) = q + (1 - 2q) I \left[\sum_{j=1}^J X_j > J/2 \right], \quad (15.11)$$

где $X \sim U[0, 1]^p$, $0 \leq q \leq 1/2$, а $J \leq p$ — некоторое заранее заданное (четное) число. Опишите эту поверхность вероятности и вычислите байесовскую ошибку.

- 15.4.** Пусть x_i , $i = 1, \dots, N$ являются независимыми и одинаково распределенными с параметрами (μ, σ^2) . Пусть \bar{x}_1^* и \bar{x}_2^* — две бутстрэп-реализации выборочного среднего. Покажите, что корреляция выборки $\text{corr}(\bar{x}_1^*, \bar{x}_2^*) = \frac{n}{2n-1} \approx 50\%$. Попутно выведите $\text{var}(\bar{x}_1^*)$ и дисперсию среднего значения при баггинге \bar{x}_{bag} . Здесь \bar{x} — линейная статистика; баггинг не уменьшает дисперсию для линейной статистики.

- 15.5.** Покажите, что выборочная корреляция между парой случайных деревьев в точке x определяется как

$$\rho(x) = \frac{\text{Var}_{\mathbf{Z}} \left[\mathbb{E}_{\Theta | \mathbf{Z}} T(x; \Theta(\mathbf{Z})) \right]}{\text{Var}_{\mathbf{Z}} \left[\mathbb{E}_{\Theta | \mathbf{Z}} T(x; \Theta(\mathbf{Z})) \right] + \mathbb{E}_{\mathbf{Z}} \text{Var}_{\Theta | \mathbf{Z}} \left[T(x; \Theta(\mathbf{Z})) \right]}. \quad (15.12)$$

Числитель равен $\text{Var}_{\mathbf{Z}} \left[\hat{f}_{\text{rf}}(x) \right]$, а второе слагаемое в знаменателе является математическим ожиданием условной дисперсии из-за рандомизации в случайных лесах.

- 15.6.** Обучите ряд классификаторов на основе случайного леса для данных о спаме, чтобы изучить их чувствительность к параметру m . Отобразите как ошибку ООВ, так и ошибку тестирования с соответствующим выбранным диапазоном значений для m .
- 15.7.** Предположим, мы обучаем модель линейной регрессии по N наблюдениям с откликом y_i и предикторами x_{i1}, \dots, x_{ip} . Предположим, что все переменные стандартизованы, т.е. имеют нулевое математическое ожидание и единичное стандартное отклонение. Пусть RSS — среднеквадратичная невязка обучающих данных и $\hat{\beta}$ — оцениваемый коэффициент. Обозначим через RSS_j^* среднеквадратичную невязку на обучающих данных, использующих ту же самую оценку $\hat{\beta}$, но с N значениями для j -й переменной, случайным образом переставленными до вычисления прогнозов. Покажите, что

$$\mathbb{E}_P \left[RSS_j^* - RSS \right] = 2\hat{\beta}_j^2, \quad (15.13)$$

где \mathbb{E}_P — математическое ожидание относительно распределения перестановок. Докажите, что это приблизительно верно, если оценки выполняются с использованием независимого тестового множества.