

УДК 004.048

ББК 32.972

М97

Кэвин П. Мэрфи

М97 Вероятностное машинное обучение. Дополнительные темы: предсказание, порождение, обнаружение, действие / пер. с англ. А. А. Слинкина. – М.: ДМК Пресс, 2024. – 766 с.: ил.

ISBN 978-5-93700-317-1

Дополняя ранее изданную книгу «Вероятностное машинное обучение. Введение», этот классический труд знакомит читателя с деталями самых актуальных теорий и методов машинного обучения (МО).

В «Дополнительных темах» излагаются различные вопросы машинного обучения на более глубоком уровне. Рассмотрено обучение и тестирование при различных распределениях, порождение многомерных выходов, таких как изображения, текст и графы.

В третьей книге дан общий обзор четырех основных видов моделей: предсказания (например, классификация и регрессия), порождения (например, изображений или текста), обнаружения («осмысленной структуры» в данных) и управления (принятия оптимальных решений).

Издание предназначено специалистам в области МО и искусственного интеллекта, а также будет полезно студентам профильных специальностей. Предполагается, что читатель знаком с МО и другими математическими дисциплинами (теорией вероятностей, статистикой, линейной алгеброй).

УДК 004.048

ББК 32.972

Все права защищены. Любая часть этой книги не может быть воспроизведена в какой бы то ни было форме и какими бы то ни было средствами без письменного разрешения владельцев авторских прав.

ISBN (анг.) 978-0-26204-843-9

ISBN (рус.) 978-5-93700-317-1

© 2023 Kevin P. Murphy

© Оформление, издание, перевод, ДМК Пресс, 2024

Оглавление

ЧАСТЬ III. ПРЕДСКАЗАНИЕ 25

Глава 14. Предсказательные модели: общий обзор..... 27

14.1. Введение	27
14.1.1. Типы моделей.....	27
14.1.2. Обучение модели с помощью ERM, MLE и MAP.....	28
14.1.3. Обучение модели байесовскими методами, методами вариационного вывода и обобщенными байесовскими методами	29
14.2. Вычисление предсказательных моделей	30
14.2.1. Собственные скоринговые правила	30
14.2.2. Калибровка	31
14.2.2.1. Ожидаемая ошибка калибровки	31
14.2.2.2. Улучшение калибровки.....	32
14.2.2.3. Масштабирование Платта	33
14.2.2.4. Непараметрические (гистограммные) методы.....	33
14.2.2.5. Температурное масштабирование.....	33
14.2.2.6. Сглаживание меток	35
14.2.2.7. Байесовские методы.....	35
14.2.3. За пределами вычисления маргинальных вероятностей	35
14.2.3.1. Доказательство утверждения	38
14.3. Конформное предсказание	39
14.3.1. Конформализация классификации.....	41
14.3.2. Конформализация регрессии	41
14.3.2.1. Конформализация квантильной регрессии	41
14.3.2.2. Конформализация предсказанных дисперсий	42

Глава 15. Обобщенные линейные модели 44

15.1. Введение	44
15.1.1. Некоторые популярные GLM.....	44
15.1.1.1. Линейная регрессия	45
15.1.1.2. Биномиальная регрессия.....	45
15.1.1.3. Регрессия Пуассона	46
15.1.1.4. Регрессия Пуассона с преобладанием нулей.....	47
15.1.2. GLM с неканоническими функциями связи.....	47
15.1.3. Оценка максимального правдоподобия	48
15.1.4. Байесовский вывод	49
15.2. Линейная регрессия	50
15.2.1. Обыкновенный метод наименьших квадратов	50
15.2.2. Сопряженные априорные распределения.....	50
15.2.2.1. Дисперсия шума известна	50

15.2.2.2. Дисперсия шума неизвестна	51
15.2.2.3. Апостериорное предсказательное распределение	53
15.2.3. Неинформативные априорные распределения.....	53
15.2.3.1. Априорное распределение Джеффриса.....	53
15.2.3.2. Связь с частотной статистикой	54
15.2.3.3. Априорное g -распределение Целльнера	54
15.2.4. Информативные априорные распределения.....	55
15.2.5. Импульсно-плоское априорное распределение	57
15.2.6. Априорное распределение Лапласа (байесовский lasso)	58
15.2.7. Подковообразное априорное распределение.....	60
15.2.8. Автоматическое определение релевантности	61
15.2.8.1. ARD для линейных моделей	61
15.2.8.2. Почему ARD дает разреженное решение?	62
15.2.8.3. Алгоритмы для ARD	63
15.2.8.4. Машины векторов релевантности	63
15.2.9. Многомерная линейная регрессия	64
15.3. Логистическая регрессия.....	66
15.3.1. Бинарная логистическая регрессия	66
15.3.2. Мультиномиальная логистическая регрессия	67
15.3.3. Несбалансированность классов и длинные хвосты.....	67
15.3.4. Априорные распределения параметров.....	68
15.3.5. Аппроксимация Лапласа апостериорного распределения	70
15.3.6. Аппроксимация апостериорного предсказательного распределения	71
15.3.7. МСМС-вывод.....	73
15.3.8. Другие приближенные методы вывода	75
15.3.9. Пример: правда ли, что женщинам труднее поступить в Беркли? ...	75
15.4. Пробит-регрессия.....	78
15.4.1. Интерпретация с латентными величинами.....	79
15.4.2. Оценка максимального правдоподобия	80
15.4.2.1. MLE с применением СГС.....	80
15.4.2.2. MLE с применением EM-алгоритма.....	80
15.4.3. Байесовский вывод	81
15.4.4. Порядковая пробит-регрессия	82
15.4.5. Мультиномиальные пробит-модели	82
15.5. Многоуровневые (иерархические) GLM	82
15.5.1. Обобщенные линейные смешанные модели (GLMM)	84
15.5.2. Пример: регрессия радона	84
15.5.2.1. Вывод апостериорного распределения	85
15.5.2.2. Нецентрированная параметризация.....	86

Глава 16. Глубокие нейронные сети..... 88

16.1. Введение	88
16.2. Построение блоков, составляющих дифференцируемые контуры	89
16.2.1. Линейные слои	89
16.2.2. Нелинейности.....	89
16.2.3. Сверточные слои	90

16.2.4. Остаточные (прямые) связи	92
16.2.5. Нормировочные слои.....	92
16.2.6. Слои прореживания	93
16.2.7. Слои внимания	94
16.2.8. Рекуррентные слои	96
16.2.9. Мультипликативные слои	97
16.2.10. Неявные слои.....	97
16.3. Канонические примеры нейронных сетей	98
16.3.1. Многослойные перцептроны (МСП).....	98
16.3.2. Сверточные нейронные сети (СНС).....	99
16.3.3. Автокодировщики.....	100
16.3.4. Рекуррентные нейронные сети (РНС)	101
16.3.5. Трансформеры.....	102
16.3.6. Графовые нейронные сети (GNN)	103
Глава 17. Байесовские нейронные сети.....	104
17.1. Введение	104
17.2. Априорные распределения для БНС	104
17.2.1. Гауссовы априорные распределения	105
17.2.2. Априорные распределения, поощряющие разреженность	107
17.2.3. Обучение априорного распределения	107
17.2.4. Априорные распределения в пространстве функций	108
17.2.5. Архитектурные априорные распределения	108
17.3. Апостериорные распределения для БНС	109
17.3.1. Прореживание Монте-Карло	109
17.3.2. Аппроксимация Лапласа.....	110
17.3.3. Вариационный вывод	111
17.3.4. Распространение математического ожидания	112
17.3.5. Методы последнего слоя	112
17.3.6. Метод SNGP.....	113
17.3.7. MCMC-методы.....	114
17.3.8. Методы, основанные на траектории СГС.....	114
17.3.9. Глубокие ансамбли	116
17.3.9.1. MultiSWAG	117
17.3.9.2. Глубокие ансамбли со случайными априорными распределениями	117
17.3.9.3. Глубокие ансамбли как приближенный байесовский вывод.....	118
17.3.9.4. Глубокие и классические ансамбли.....	118
17.3.9.5. Глубокие ансамбли, смеси экспертов и стекинг.....	119
17.3.9.6. Пакетный ансамбль	119
17.3.10. Аппроксимация апостериорного предсказательного распределения	120
17.3.10.1. Линеаризованная аппроксимация	120
17.3.10.2. Аппроксимация на основе моста Лапласа.....	121
17.3.10.3. Дистилляция	123
17.3.11. Закаленные и холодные апостериорные распределения	123

17.4. Обобщение в байесовском глубоком обучении	125
17.4.1. Острые и плоские минимумы	125
17.4.2. Связность мод и ландшафт функции потерь.....	126
17.4.3. Эффективная размерность модели	127
17.4.4. Пространство гипотез ГНС.....	128
17.4.5. РАС-байесовское обучение	129
17.4.6. Обобщение БНС при выходе за рамки распределения.....	130
17.4.6.1. ВМА может давать плохие результаты для априорных распределений по умолчанию.....	130
17.4.6.2. БНС могут быть чрезмерно уверены на входах, не принадлежащих распределению.....	132
17.4.7. Выбор модели для БНС	132
17.5. Онлайнный вывод	133
17.5.1. Последовательная аппроксимация Лапласа для ГНС	134
17.5.2. Обобщенная фильтрация Калмана для ГНС	134
17.5.2.1. Пример	135
17.5.2.2. Задание членов дисперсии	135
17.5.2.3. Уменьшение вычислительной сложности	135
17.5.3. Фильтрация с предполагаемой плотностью для ГНС	136
17.5.4. Онлайнный вариационный вывод для ГНС.....	138
17.6. Иерархические байесовские нейронные сети.....	139
17.6.1. Пример: классификация двух лун	140

Глава 18. Гауссовские процессы 143

18.1. Введение	143
18.1.1. ГП: что такое и почему?.....	143
18.2. Ядра Мерсера	145
18.2.1. Стационарные ядра.....	146
18.2.1.1. Квадратичное экспоненциальное (RBF) ядро	146
18.2.1.2. ARD-ядро	147
18.2.1.3. Ядра Матерна	148
18.2.1.4. Периодические ядра	149
18.2.1.5. Рациональное квадратичное ядро.....	150
18.2.1.6. Ядра и спектральные плотности.....	150
18.2.2. Нестационарные ядра.....	151
18.2.2.1. Полиномиальные ядра.....	151
18.2.2.2. Ядро Гиббса.....	151
18.2.2.3. Другие нестационарные ядра	152
18.2.3. Ядра для не векторных (структурных) входов	152
18.2.4. Создание новых ядер на основе существующих.....	152
18.2.5. Теорема Мерсера	154
18.2.6. Аппроксимация ядер случайными признаками.....	155
18.3. ГП с гауссовым правдоподобием	156
18.3.1. Предсказания с незашумленными наблюдениями	156
18.3.2. Предсказания с зашумленными наблюдениями	157
18.3.3. Пространство весов и пространство функций	158
18.3.4. Полупараметрические ГП.....	159
18.3.5. Предельное правдоподобие	160

18.3.6. Вычислительные и численные трудности.....	160
18.3.7. Ядерная гребневая регрессия	161
18.3.7.1. Гильбертовы пространства с воспроизводящими ядрами....	161
18.3.7.2. Сложность функции в ГПВЯ.....	163
18.3.7.3. Теорема о представителе	163
18.3.7.4. Сравнение KRR с GPR	164
18.4. ГП с негауссовыми правдоподобиями.....	165
18.4.1. Бинарная классификация.....	165
18.4.2. Многоклассовая классификация.....	167
18.4.3. Гауссовские процессы для регрессии Пуассона (процесс Кокса).....	167
18.4.4. Другие правдоподобия	168
18.5. Масштабирование ГП-вывода на большие наборы данных	169
18.5.1. Подмножество данных.....	169
18.5.1.1. Метод информативных векторов	169
18.5.1.2. Обсуждение	170
18.5.2. Аппроксимация Нистрёма	170
18.5.3. Методы со вспомогательными точками.....	171
18.5.3.1. SOR/DIC.....	172
18.5.3.2. DTC	173
18.5.3.3. FITC	173
18.5.3.4. Обучение вспомогательных точек.....	174
18.5.4. Разреженные вариационные методы.....	175
18.5.4.1. Гауссово правдоподобие.....	177
18.5.4.2. Негауссово правдоподобие.....	178
18.5.4.3. Мини-пакетный SVI.....	178
18.5.5. Использование распараллеливания и структуры путем умножения ядерных матриц	179
18.5.5.1. Использование метода сопряженных градиентов и метода Ланцоша	179
18.5.5.2. Ядра с компактным носителем	180
18.5.5.3. KISS.....	181
18.5.5.4. Тензорные методы	181
18.5.6. Преобразование ГП в SSM	182
18.6. Обучение ядра	182
18.6.1. Эмпирический байесовский подход к параметрам ядра.....	183
18.6.1.1. Пример.....	184
18.6.2. Байесовский вывод параметров ядра.....	185
18.6.3. Обучение с несколькими ядрами для аддитивных ядер.....	187
18.6.4. Автоматический поиск композиционных ядер.....	189
18.6.5. Обучение спектрального смесового ядра.....	191
18.6.6. Глубокое обучение ядра.....	193
18.7. Гауссовские процессы и глубокие нейронные сети	195
18.7.1. Ядра, выведенные из бесконечно широких ГНС (NN-GP)	195
18.7.2. Нейронное касательное ядро (NTK)	197
18.7.3. Глубокие ГП.....	198
18.8. Гауссовские процессы как инструмент прогнозирования временных рядов.....	198
18.8.1. Пример: Мауна-Лоа	199

Глава 19. За пределами предположения**о независимости и одинаковом распределении..... 201**

19.1. Введение	201
19.2. Дрейф распределения.....	201
19.2.1. Мотивирующие примеры.....	202
19.2.2. Каузальный взгляд на дрейф распределения	203
19.2.3. Четыре основных типа дрейфа распределения	204
19.2.3.1. Дрейф ковариат.....	205
19.2.3.2. Дрейф концепта	205
19.2.3.3. Дрейф меток/априорного распределения.....	206
19.2.3.4. Дрейф проявления	206
19.2.4. Смещение выбора	206
19.3. Обнаружение дрейфа распределения.....	207
19.3.1. Обнаружение дрейфа путем двухвыборочного критерия	207
19.3.2. Обнаружение не принадлежащих распределению входов	208
19.3.2.1. Методы идентификации ID/OOD с учителем (выявление выбросов).....	209
19.3.2.2. Методы, предсказывающие уверенность классификации....	209
19.3.2.3. Конформное предсказание	209
19.3.2.4. Методы обучения без учителя.....	210
19.3.3. Избирательное предсказание	211
19.3.3.1. Пример: SGLD и CGC для MСP.....	211
19.3.4. Распознавание открытого множества и открытого мира.....	212
19.4. Робастность к дрейфу распределения	213
19.4.1. Пополнение данных.....	213
19.4.2. Устойчивая к изменению распределения оптимизация.....	213
19.5. Адаптация к дрейфу распределения	214
19.5.1. Адаптация с учителем с применением переноса обучения	214
19.5.1.1. Предобучение и дообучение	214
19.5.1.2. Дообучение с подсказками (обучение в контексте)	215
19.5.2. Взвешенная ERM для дрейфа ковариат	215
19.5.2.1. Почему дрейф ковариат является проблемой для дискриминантных моделей?	216
19.5.2.2. Как оценивать веса ERM?	216
19.5.3. Адаптация к домену без учителя для дрейфа ковариат.....	217
19.5.4. Методы без учителя для дрейфа меток	217
19.5.5. Адаптация на этапе тестирования.....	218
19.6. Обучение на примерах из нескольких распределений	219
19.6.1. Многозадачное обучение	220
19.6.2. Обобщение домена	221
19.6.3. Минимизация инвариантного риска.....	222
19.6.4. Метаобучение.....	223
19.6.4.1. Метаобучение как вероятностный вывод для предсказания.....	223
19.6.4.2. Нейронные процессы.....	225
19.6.4.3. Градиентное метаобучение (MAML)	226
19.6.4.4. Метрическое обучение на нескольких примерах (прототипические сети).....	226

19.7. Непрерывное обучение	226
19.7.1. Дрейф домена	227
19.7.2. Дрейф концепта.....	227
19.7.3. Инкрементное обучение задач.....	228
19.7.4. Катастрофическое забывание	230
19.7.5. Онлайнное обучение	232
19.8. Состязательные примеры.....	234
19.8.1. Градиентные атаки на белый ящик	235
19.8.2. Безградиентные атаки на черный ящик	236
19.8.3. Состязательные атаки в реальном мире	237
19.8.4. Защита, основанная на робастной оптимизации.....	238
19.8.5. Почему для моделей имеются состязательные примеры	239

ЧАСТЬ IV. ПОРОЖДЕНИЕ..... 241

Глава 20. Порождающие модели: общий обзор..... 243

20.1. Введение	243
20.2. Типы порождающих моделей	243
20.3. Цели порождающего моделирования.....	246
20.3.1. Генерирование данных	246
20.3.2. Оценивание плотности.....	248
20.3.3. Подстановка	248
20.3.4. Обнаружение структуры.....	250
20.3.5. Интерполяция латентного пространства	250
20.3.6. Арифметика в латентном пространстве	251
20.3.7. Порождающее проектирование	252
20.3.8. Обучение с подкреплением на основе модели	252
20.3.9. Обучение представлений	252
20.3.10. Сжатие данных	253
20.4. Оценивание порождающих моделей.....	253
20.4.1. Оценивание на основе правдоподобия.....	253
20.4.1.1. Вычисление логарифмического правдоподобия.....	254
20.4.1.2. Иногда правдоподобие трудно вычислить.....	254
20.4.1.3. Правдоподобие не связано с качеством выборки	254
20.4.2. Расстояния и расхождения в пространстве признаков	255
20.4.3. Метрики точности и полноты	257
20.4.4. Статистические критерии	257
20.4.5. Проблемы использования предобученных классификаторов.....	258
20.4.6. Использование выборок из модели для обучения классификаторов	258
20.4.7. Оценка переобучения	259
20.4.8. Оценивание человеком	260

Глава 21. Вариационные автокодировщики 261

21.1. Введение	261
21.2. Основы VAE.....	262
21.2.1. Модельные предположения	262
21.2.2. Обучение модели	263

21.2.3. Сравнение VAE и автокодировщиков	263
21.2.4. VAE оптимизируют в дополненном пространстве	265
21.3. Обобщения VAE	267
21.3.1. β -VAE	267
21.3.1.1. Разделенные представления	268
21.3.1.2. Связь с информационным бутылочным горлышком	269
21.3.2. InfoVAE	269
21.3.2.1. Связь с MMD VAE	270
21.3.2.2. Связь с β -VAE	271
21.3.2.3. Связь с состязательными автокодировщиками	271
21.3.3. Мультимодальные VAE	271
21.3.4. VAE, обучаемые с частичным привлечением учителя	274
21.3.5. VAE с последовательными кодировщиками/декодерами	275
21.3.5.1. Модели	275
21.3.5.2. Приложения	277
21.4. Избегание схлопывания апостериорного распределения	279
21.4.1. Отжиг расхождения КЛ	279
21.4.2. Ограничение скорости снизу	280
21.4.3. Бесплатные биты	280
21.4.4. Добавление прямых связей	280
21.4.5. Улучшенный вариационный вывод	281
21.4.6. Альтернативные целевые функции	281
21.5. VAE с иерархической структурой	282
21.5.1. Восходящий и нисходящий вывод	283
21.5.2. Пример: очень глубокие VAE	284
21.5.3. Связь с моделями авторегрессии	285
21.5.4. Вариационная обрезка	286
21.5.5. Другие трудности оптимизации	287
21.6. VAE с векторным квантованием	288
21.6.1. Автокодировщик с бинарным кодом	288
21.6.2. Модель VQ-VAE	288
21.6.3. Обучение априорного распределения	290
21.6.4. Иерархическое обобщение (VQ-VAE-2)	291
21.6.5. Дискретный VAE	291
21.6.6. VQ-GAN	292

Глава 22. Модели авторегрессии..... 293

22.1. Введение	293
22.2. Нейронные авторегрессионные оценщики плотности (NADE)	294
22.3. Каузальные СНС	295
22.3.1. Одномерные каузальные СНС (сверточные марковские модели)	295
22.3.2. Двумерная каузальная СНС (PixelCNN)	296
22.4. Трансформеры	296
22.4.1. Генерирование текста (GPT и т. д.)	297
22.4.2. Генерирование изображений (DALL-E и т. д.)	299
22.4.3. Другие применения	300

Глава 23. Нормализующие потоки.....	301
23.1. Введение	301
23.1.1. Предварительные сведения	301
23.1.2. Как обучать потоковую модель	303
23.1.2.1. Оценивание плотности.....	303
23.1.2.2. Вариационный вывод	303
23.2. Конструирование потоков	304
23.2.1. Аффинные потоки.....	304
23.2.2. Поэлементные потоки	305
23.2.2.1. Аффинная скалярная биекция	305
23.2.2.2. Возмущения высшего порядка	305
23.2.2.3. Комбинации строго монотонных скалярных функций.....	306
23.2.2.4. Скалярные биекции, являющиеся результатом интегрирования.....	306
23.2.2.5. Сплайны.....	307
23.2.3. Связывающие потоки	308
23.2.4. Авторегрессионные потоки.....	310
23.2.4.1. Аффинные авторегрессионные потоки.....	310
23.2.4.2. Замаскированные авторегрессионные потоки.....	312
23.2.4.3. Обратные авторегрессионные потоки	313
23.2.4.4. Связь с моделями авторегрессии.....	314
23.2.5. Остаточные потоки	315
23.2.5.1. Сжимающие остаточные блоки.....	315
23.2.5.2. Остаточные блоки с якобианом низкого ранга.....	316
23.2.6. Потоки с непрерывным временем	317
23.3. Приложения.....	319
23.3.1. Оценивание плотности.....	319
23.3.2. Порождающее моделирование	320
23.3.3. Вывод	320
Глава 24. Модели на основе энергии.....	322
24.1. Введение	322
24.1.1. Пример: производство экспертов (PoE).....	323
24.1.2. Вычислительные трудности	323
24.2. Обучение методом максимального правдоподобия	324
24.2.1. Градиентные МСМС-методы.....	325
24.2.2. Сопоставительное расхождение	325
24.2.2.1. Обучение ограниченных машин Больцмана методом CD	326
24.2.2.2. Сохраняемое сопоставительное расхождение	328
24.2.2.3. Другие методы.....	329
24.3. Приравнивание вкладов (SM)	329
24.3.1. Базовое приравнивание вкладов	330
24.3.2. Шумоподавляющее приравнивание вкладов (DSM).....	330
24.3.2.1. Трудности	331
24.3.3. Слоистое приравнивание вкладов (SSM).....	332
24.3.4. Связь с сопоставительным расхождением	333
24.3.5. Порождающие модели на основе вклада	334

24.4. Шумосопоставительное оценивание	334
24.4.1. Связь с приравнением вкладов	336
24.5. Другие методы	337
24.5.1. Минимизация разностей и производных расхождений КЛ	337
24.5.2. Минимизация расхождения Штейна	338
24.5.3. Состязательное обучение	338

Глава 25. Диффузионные модели 341

25.1. Введение	341
25.2. Шумоподавляющие диффузионные вероятностные модели (DDPM)	341
25.2.1. Кодировщик (прямая диффузия)	342
25.2.2. Декодер (обратная диффузия)	343
25.2.3. Обучение модели	344
25.2.4. Обучение плана изменения шума	346
25.2.5. Пример: генерирование изображений	348
25.3. Порождающие модели на основе вклада	349
25.3.1. Пример	349
25.3.2. Прибавление шума при нескольких масштабах	349
25.3.3. Эквивалентность DDPM	351
25.4. Модели с непрерывным временем и дифференциальные уравнения	352
25.4.1. СДУ прямой диффузии	352
25.4.2. ОДУ прямой диффузии	353
25.4.3. СДУ обратной диффузии	354
25.4.4. ОДУ обратной диффузии	355
25.4.5. Сравнение подходов на основе СДУ и ОДУ	356
25.4.6. Пример	357
25.5. Ускорение диффузионных моделей	357
25.5.1. DDIM-отборщик	357
25.5.2. Негауссовы сети-декодеры	358
25.5.3. Дистилляция	358
25.5.4. Диффузия в латентном пространстве	359
25.6. Условное порождение	360
25.6.1. Условная диффузионная модель	360
25.6.2. Руководство со стороны классификатора	361
25.6.3. Руководство без классификатора	361
25.6.4. Генерирование изображений высокого разрешения	362
25.7. Диффузия для пространств дискретных состояний	363
25.7.1. Дискретные шумоподавляющие диффузионные вероятностные модели	363
25.7.2. Выбор переходных матриц марковской цепи для прямых процессов	364
25.7.3. Параметризация обратного процесса	365
25.7.4. План изменения шума	366
25.7.5. Связи с другими вероятностными моделями дискретных последовательностей	366

Глава 26. Порождающие состязательные сети	368
26.1. Введение	368
26.2. Обучение путем сравнения	370
26.2.1. Руководящие принципы	370
26.2.2. Оценивание отношения плотностей с помощью бинарных классификаторов	371
26.2.3. Границы f-расхождений	374
26.2.4. Интегральные вероятностные метрики	376
26.2.5. Приравнивание моментов	378
26.2.6. Об отношениях и разностях плотностей	379
26.3. Порождающие состязательные сети	381
26.3.1. От принципов обучения к функциям потерь	381
26.3.2. Градиентный спуск	383
26.3.3. Проблемы обучения GAN	384
26.3.4. Улучшение оптимизации GAN	385
26.3.5. Сходимость обучения GAN	386
26.4. Условные GAN	390
26.5. Вывод с помощью GAN	391
26.6. Нейронные архитектуры в GAN	392
26.6.1. Важность архитектуры дискриминатора	392
26.6.2. Архитектурные индуктивные смещения	393
26.6.3. Механизмы внимания в GAN	394
26.6.4. Прогрессивное генерирование	394
26.6.5. Регуляризация	395
26.6.6. Масштабирование моделей GAN	396
26.7. Приложения	396
26.7.1. Применение GAN для генерирования изображений	397
26.7.1.1. Условное генерирование изображений	397
26.7.1.2. Парное генерирование изображений	398
26.7.1.3. Непарное генерирование изображений	398
26.7.2. Генерирование видео	400
26.7.3. Генерирование звука	400
26.7.4. Генерирование текста	401
26.7.5. Имитационное обучение	402
26.7.6. Адаптация домена	403
26.7.7. Дизайн, искусство и творчество	403
ЧАСТЬ V. ОБНАРУЖЕНИЕ	405
Глава 27. Методы обнаружения: обзор	407
27.1. Введение	407
27.2. Обзор части V	408
Глава 28. Модели латентных факторов	409
28.1. Введение	409
28.2. Смесовые модели	409
28.2.1. Модели гауссовой смеси (GMM)	411

28.2.2. Модели бернуллиевой смеси.....	412
28.2.3. Масштабированные гауссовы смеси (GSM).....	412
28.2.3.1. t-распределение Стьюдента как GSM	413
28.2.3.2. Распределение Лапласа как GSM	413
28.2.3.3. Импульсно-плоское распределение	414
28.2.3.4. Подковообразное распределение	414
28.2.4. Использование GMM в качестве априорного распределения для задач обращения изображений	414
28.2.4.1. Почему этот метод работает?	416
28.2.4.2. Ускорение вывода с помощью дискриминантных моделей	417
28.2.4.3. Задача обращения вслепую	417
28.2.5. Использование смесовых моделей в задачах классификации	418
28.2.5.1. Гибридное обучение порождающе-дискриминантной модели.....	418
28.2.5.2. Проблемы оптимизации.....	419
28.2.5.3. Численные проблемы	419
28.3. Факторный анализ	420
28.3.1. Факторный анализ: основы.....	420
28.3.1.1. ФА как гауссово распределение с ковариационной матрицей в виде суммы диагональной матрицы и матрицы низкого ранга.....	420
28.3.1.2. Вычисление апостериорного распределения	422
28.3.1.3. Вычисление правдоподобия	422
28.3.1.4. Обучение модели с помощью EM-алгоритма	422
28.3.1.5. Обработка отсутствия данных	423
28.3.1.6. Неидентифицируемость параметров	423
28.3.2. Вероятностный PCA	425
28.3.2.1. Вывод MLE	425
28.3.2.2. PCA – предел в отсутствие шума	426
28.3.2.3. Вычисление апостериорного распределения	426
28.3.2.4. Обучение модели с помощью EM-алгоритма	426
28.3.3. Смесь факторных анализаторов	427
28.3.3.1. Определение модели	428
28.3.3.2. Обучение модели с помощью EM-алгоритма	428
28.3.3.3. Обучение модели с помощью СГС	430
28.3.3.4. Выбор модели.....	430
28.3.3.5. Применение MixFA для генерирования изображений	430
28.3.4. Модели факторного анализа для спаренных данных.....	434
28.3.4.1. PCA с учителем	434
28.3.4.2. Метод частичных наименьших квадратов.....	435
28.3.4.3. Канонический корреляционный анализ	436
28.3.5. Факторный анализ с правдоподобиями из экспоненциального семейства	437
28.3.5.1. Пример: бинарный PCA.....	438
28.3.5.2. Пример: категориальный PCA	438
28.3.6. Факторный анализ с ГНС-правдоподобиями (VAE).....	438
28.3.7. Факторный анализ с ГП-правдоподобиями (GP-LVM).....	439

28.4. Модели латентных факторов с негауссовыми априорными распределениями	441
28.4.1. Неотрицательное матричное разложение (НМР)	441
28.4.2. Мультиномиальный PCA	442
28.4.2.1. Пример: данные о поименном голосовании	443
28.4.2.2. Преимущество априорного распределения Дирихле перед гауссовым	444
28.4.2.3. Связь со смесовыми моделями	444
28.5. Тематические модели	445
28.5.1. Латентное распределение Дирихле (LDA)	445
28.5.1.1. Определение модели	445
28.5.1.2. Многозначность	447
28.5.1.3. Вывод апостериорного распределения	447
28.5.1.4. Определение числа тем	448
28.5.2. Коррелированная тематическая модель	448
28.5.3. Динамическая тематическая модель	449
28.5.4. LDA-НММ	450
28.6. Анализ независимых компонент (ICA)	454
28.6.1. Незашумленная модель ICA	454
28.6.2. Необходимость негауссовых априорных распределений	455
28.6.3. Оценка максимального правдоподобия	456
28.6.4. Альтернативы MLE	457
28.6.4.1. Максимизация негауссовости	457
28.6.4.2. Минимизация полной корреляции	458
28.6.4.3. Максимизация взаимной информации (InfoMax)	458
28.6.5. Разреженное кодирование	459
28.6.6. Нелинейный ICA	460
Глава 29. Модели пространства состояний	461
29.1. Введение	461
29.2. Скрытые марковские модели (НММ)	462
29.2.1. Свойства условной независимости	462
29.2.2. Модель переходов состояний	463
29.2.3. Дискретные правдоподобия	463
29.2.4. Гауссовы правдоподобия	464
29.2.5. Авторегрессионные правдоподобия	464
29.2.6. Нейросетевые правдоподобия	466
29.3. НММ: приложения	467
29.3.1. Сегментация временных рядов	467
29.3.2. Выравнивание последовательностей белков	469
29.3.3. Исправление орфографических ошибок	470
29.3.3.1. Базовая модель	470
29.3.3.2. Модель НММ	471
29.3.3.3. Расширенная модель НММ	472
29.4. НММ: обучение параметров	472
29.4.1. Алгоритм Баума–Велша	473
29.4.1.1. Логарифмическое правдоподобие	473

29.4.1.2. E-шаг	474
29.4.1.3. M-шаг	474
29.4.1.4. Инициализация	475
29.4.1.5. Пример: НММ для казино.....	476
29.4.2. Оценивание параметров с помощью СГС	476
29.4.2.1. Пример: НММ казино	477
29.4.3. Оценивание параметров спектральными методами	477
29.4.4. Байесовские НММ	478
29.4.4.1. Блочная выборка Гиббса для НММ	479
29.5. НММ: обобщения	480
29.5.1. Скрытая полумарковская модель (HSMM)	480
29.5.2. Иерархические НММ.....	483
29.5.3. Факториальные НММ.....	484
29.5.4. Спаренные НММ	486
29.5.5. Динамические байесовские сети (DBN).....	486
29.5.6. Обнаружение точки перемены.....	487
29.5.6.1. Пример.....	489
29.5.6.2. Обобщения	490
29.6. Линейные динамические системы (ЛДС).....	490
29.6.1. Свойства условной независимости.....	490
29.6.2. Параметризация.....	491
29.7. ЛДС: применения	491
29.7.1. Прослеживание объектов и оценивание состояний	491
29.7.2. Онлайн-байесовская линейная регрессия (рекурсивный метод наименьших квадратов)	492
29.7.3. Адаптивная фильтрация	495
29.7.4. Прогнозирование временных рядов.....	495
29.8. LDS: обучение параметров	495
29.8.1. EM-алгоритм для ЛДС.....	495
29.8.2. Методы идентификации подпространства	498
29.8.3. Обеспечение устойчивости динамической системы.....	498
29.8.4. Байесовские ЛДС	498
29.8.4.1. Блочная выборка Гиббса для ЛДС	499
29.9. Переключательные линейные динамические системы (ПЛДС)	500
29.9.1. Параметризация.....	500
29.9.2. Вывод апостериорного распределения	500
29.9.3. Приложение: сопровождение нескольких целей.....	501
29.9.3.1. Прогрев	501
29.9.3.2. Привязка данных	502
29.9.3.3. Венгерский алгоритм аппроксимации по ближайшему соседу	503
29.9.3.4. Другие схемы приближенного вывода	503
29.9.3.5. Обработка неизвестного числа целей.....	504
29.10. Нелинейные SSM.....	505
29.10.1. Пример: сопровождение объекта и оценивание состояния	505
29.10.2. Вывод апостериорного распределения	505

29.11. Негауссовы SSM	506
29.11.1. Пример: моделирование последовательности импульсов.....	506
29.11.2. Пример: стохастические модели волатильности.....	507
29.11.3. Вывод апостериорного распределения	508
29.12. Структурные модели временных рядов	508
29.12.1. Введение	508
29.12.2. Структурные строительные блоки	509
29.12.2.1. Модель локального уровня	509
29.12.2.2. Локальная линейная модель	509
29.12.2.3. Добавление ковариат	510
29.12.2.4. Моделирование сезонности	510
29.12.2.5. Сведение воедино	511
29.12.3. Обучение модели	511
29.12.4. Прогнозирование	512
29.12.5. Примеры	512
29.12.5.1. Пример: прогнозирование уровней CO ₂ из-за выбросов вулкана Мауна-Лоа	512
29.12.5.2. Пример: прогноз потребления энергии (вещественного)	513
29.12.5.3. Пример: предсказание объема продаж (целочисленного)	514
29.12.5.4. Пример: иерархическая SSM для данных электоральной панели	516
29.12.6. Каузальные последствия вмешательства во временные ряды.....	517
29.12.6.1. Вычисление контрфактического предсказания.....	517
29.12.6.2. Предположения, при которых метод работает	518
29.12.6.3. Пример	519
29.12.6.4. Сравнение с синтетическим управлением	520
29.12.7. Библиотека Prophet	520
29.12.8. Нейронные методы прогнозирования	521
29.13. Глубокие SSM	522
29.13.1. Глубокие марковские модели	523
29.13.2. Рекуррентная SSM	524
29.13.3. Улучшение многошаговых предсказаний	524
29.13.4. Вариационные PHS	526
Глава 30. Обучение графов	528
30.1. Введение	528
30.2. Модели с латентными величинами для графов.....	528
30.3. Обучение структуры графовой модели	528
Глава 31. Непараметрические байесовские модели	531
31.1. Введение	531
Глава 32. Обучение представлений	533
32.1. Введение	533
32.2. Оценивание и сравнение обученных представлений	534
32.2.1. Последующее качество	534
32.2.1.1. Линейные классификаторы и линейное оценивание	535

32.2.1.2. Дообучение	536
32.2.1.3. Разделение	536
32.2.2. Сходство представлений	537
32.2.2.1. Анализ сходства представлений и выравнивание центрированных ядер	537
32.2.2.2. Канонический корреляционный анализ и родственные методы	538
32.2.2.3. Сравнение мер сходства представлений	540
32.3. Подходы к обучению представлений	541
32.3.1. Обучение представлений с учителем и их передача	543
32.3.2. Обучение порождающих представлений	545
32.3.2.1. Модели с латентными величинами	545
32.3.2.2. Полностью наблюдаемые модели	546
32.3.2.3. Автокодировщики	547
32.3.2.4. Проблемы обучения порождающих представлений	548
32.3.3. Обучение представлений с самоконтролем	548
32.3.3.1. Шумоподавление и замаскированное предсказание	548
32.3.3.2. Предсказание преобразования	550
32.3.4. Многокурсное обучение представлений	551
32.3.4.1. Выбор ракурса	552
32.3.4.2. Сопоставительные потери	554
32.3.4.3. Потери без отрицательных примеров	555
32.3.4.4. Секреты ремесла	556
32.4. Теория обучения представлений	557
32.4.1. Идентифицируемость	557
32.4.2. Максимизация информации	558

Глава 33. Интерпретируемость 561

33.1. Введение	561
33.1.1. Роль интерпретируемости: неизвестные и недоспецификация	562
33.1.2. Терминология и экосистема	563
33.2. Методы интерпретируемого машинного обучения	567
33.2.1. Внутренне интерпретируемые методы: модель сама является своим объяснением	568
33.2.2. Внутренне полуинтерпретируемые модели: методы на основе примеров	570
33.2.3. Ретроспективное применение или совместное обучение: объяснение дает частичный вид модели	571
33.2.3.1. Из чего состоит объяснение?	571
33.2.3.2. Как вычисляется объяснение	573
33.2.4. Прозрачность и визуализация	575
33.3. Свойства: абстракция посередине между контекстом и методом	576
33.3.1. Свойства объяснений из интерпретируемого машинного обучения	577
33.3.2. Свойства объяснений, заимствованные из когнитивистики	580
33.4. Оценивание моделей интерпретируемого машинного обучения	581

33.4.1. Вычислительное оценивание: обладает ли метод желательными свойствами?	582
33.4.2. Оценивание с участием пользователей: помогает ли метод пользователю решить задачу?.....	586
33.4.2.1. Исследования с участием пользователей в реальных контекстах.....	586
33.4.2.2. Основные элементы исследования с участием пользователей ...	587
33.4.2.3. Исследования с участием пользователей в синтетических контекстах.....	590
33.5. Обсуждение: как рассматривать интерпретируемое машинное обучение	591

ЧАСТЬ V. ДЕЙСТВИЕ 597

Глава 34. Принятие решений в условиях неопределенности599

34.1. Теория статистических решений.....	599
34.1.1. Основы.....	599
34.1.2. Частотная теория принятия решений	600
34.1.3. Байесовская теория принятия решений.....	600
34.1.4. Частотная оптимальность байесовского подхода.....	601
34.1.5. Примеры задач однократного принятия решения.....	602
34.1.5.1. Классификация.....	602
34.1.5.2. Регрессия	602
34.1.5.3. Оценивание параметров	603
34.1.5.4. Оценивание дискретных параметров	603
34.1.5.5. Структурное предсказание	604
34.1.5.6. Справедливость.....	606
34.2. Диаграммы решений (влияния).....	606
34.2.1. Пример: бурение нефтяных скважин наугад	606
34.2.2. Информационные ребра	607
34.2.3. Ценность информации	608
34.2.4. Вычисление оптимальной стратегии	609
34.3. А/В-тестирование.....	610
34.3.1. Байесовский подход.....	610
34.3.1.1. Оптимальная стратегия.....	610
34.3.1.2. Оптимальный размер выборки.....	611
34.3.1.3. Сожаление.....	613
34.3.1.4. Ожидаемая частота ошибок	613
34.3.2. Пример.....	614
34.4. Контекстуальные бандиты	615
34.4.1. Типы бандитов	615
34.4.2. Применения.....	617
34.4.3. Компромисс между исследованием и использованием.....	618
34.4.4. Оптимальное решение	618
34.4.5. Верхние доверительные границы (ВДГ).....	620
34.4.5.1. Частотный подход	620
34.4.5.2. Байесовский подход.....	621

34.4.5.3. Пример.....	622
34.4.6. Выборка Томпсона	622
34.4.7. Сожаление	623
34.5. Марковские задачи принятия решений	625
34.5.1. Основные положения	625
34.5.2. Частично наблюдаемые МППР	627
34.5.3. Эпизоды и доходы.....	627
34.5.4. Функции ценности.....	628
34.5.5. Оптимальные функции ценности и стратегии.....	629
34.5.5.1. Пример.....	630
34.6. Планирование МППР	631
34.6.1. Итерация по ценности.....	631
34.6.2. Итерация по стратегиям.....	633
34.6.3. Линейное программирование.....	634
34.7. Активное обучение.....	635
34.7.1. Сценарии активного обучения.....	635
34.7.2. Связь с другими формами последовательного принятия решений.....	636
34.7.3. Стратегии селекции.....	637
34.7.3.1. Выборка по степени неопределенности	637
34.7.3.2. Коллективный запрос.....	638
34.7.3.3. Теоретико-информационные методы	638
34.7.4. Пакетное активное обучение.....	639
34.7.4.1. BatchBALD	640
34.7.4.2. Оптимизация BatchBALD.....	640
34.7.4.3. Вычисление BatchBALD.....	641
34.7.4.4. Экспериментальное сравнение BALD и BatchBALD на наборе данных MNIST	642

Глава 35. Обучение с подкреплением 643

35.1. Введение	643
35.1.1. Обзор методов.....	644
35.1.2. Методы на основе ценности.....	645
35.1.3. Методы поиска стратегии	645
35.1.4. ОП на основе модели	646
35.1.5. Компромисс между исследованием и использованием.....	646
35.1.5.1. ϵ -жадная стратегия	646
35.1.5.2. Исследование Больцмана	647
35.1.5.3. Верхние доверительные границы и выборка Томпсона	647
35.1.5.4. Оптимальные решения с использованием байесовски-адаптивных МППР	648
35.2. ОП на основе ценности.....	649
35.2.1. ОП Монте-Карло.....	649
35.2.2. Обучение на основе временных различий (TD-обучение).....	649
35.2.3. TD-обучение со следами приемлемости	651
35.2.4. SARSA: TD-обучение с единой стратегией.....	652

35.2.5. Q-обучение: TD-обучение с разделенной стратегией.....	652
35.2.5.1. Пример.....	653
35.2.5.2. Двойное Q-обучение.....	654
35.2.6. Глубокая Q-сеть (DQN).....	655
35.3. ОП на основе стратегии.....	656
35.3.1. Теорема о градиенте стратегии.....	656
35.3.2. REINFORCE.....	657
35.3.3. Метод типа исполнитель–критик.....	658
35.3.3.1. A2C и A3C.....	659
35.3.3.2. Следы приемлемости.....	659
35.3.4. Методы ограниченной оптимизации.....	660
35.3.5. Детерминированные методы градиента стратегии.....	662
35.3.6. Безградиентные методы.....	663
35.4. ОП на основе модели.....	664
35.4.1. Управление на основе предсказательной модели (MPC).....	664
35.4.1.1. Эвристический поиск.....	665
35.4.1.2. Поиск Монте-Карло по дереву (MCTS).....	665
35.4.1.3. Оптимизация траекторий для непрерывных действий.....	666
35.4.2. Комбинирование безмодельных методов с методами на основе модели.....	666
35.4.3. MBRL с применением гауссовских процессов.....	667
35.4.3.1. PILCO.....	667
35.4.3.2. GP-MPC.....	668
35.4.4. MBRL с использованием ГНС.....	669
35.4.5. MBRL с использованием моделей с латентными величинами.....	669
35.4.5.1. Модели мира.....	669
35.4.5.2. PlaNet и Dreamer.....	671
35.4.6. Устойчивость к ошибкам модели.....	672
35.5. Обучение с разделенной стратегией.....	672
35.5.1. Простые методы.....	673
35.5.1.1. Прямой метод.....	673
35.5.1.2. Выборка по значимости.....	674
35.5.1.3. Дважды робастная оценка.....	676
35.5.1.4. Метод с регуляризованным поведением.....	676
35.5.2. Проклятие горизонта.....	677
35.5.3. Смертельная триада.....	679
35.6. Управление как вывод.....	680
35.6.1. Обучение с подкреплением с максимальной энтропией.....	680
35.6.2. Другие подходы.....	683
35.6.3. Имитационное обучение.....	685
35.6.3.1. Имитационное обучение путем клонирования поведения.....	685
35.6.3.2. Имитационное обучение посредством обратного обучения с подкреплением.....	685
35.6.3.3. Имитационное обучение посредством минимизации расхождения.....	686

Глава 36. Каузальность.....	688
36.1. Введение	688
36.2. Каузальный формализм	690
36.2.1. Структурные каузальные модели	690
36.2.2. Каузальные ОАГ.....	692
36.2.3. Идентификация.....	694
36.2.4. Контрфактические вопросы и каузальная иерархия.....	696
36.3. Рандомизированные контролируемые испытания.....	698
36.4. Поправка на искажающие факторы.....	699
36.4.1. Каузальный оцениваемый показатель, статистический оцениваемый показатель и идентификация.....	700
36.4.2. Оценивание АТЕ с наблюдаемыми искажающими факторами... 703	
36.4.2.1. Корректировка модели исходов	703
36.4.2.2. Корректировка коэффициента предрасположенности	704
36.4.2.3. Двойное машинное обучение	706
36.4.2.4. Перекрестное обучение	708
36.4.3. Количественное выражение неопределенности.....	709
36.4.4. Сопоставление.....	709
36.4.5. Практические соображения и процедуры.....	711
36.4.5.1. Что корректировать	711
36.4.5.2. Перекрытие	713
36.4.5.3. Выбор оцениваемого показателя и среднего эффекта воздействия на получивших лечение	713
36.4.6. Резюме и практические рекомендации	714
36.5. Стратегии, основанные на инструментальных величинах.....	716
36.5.1. Аддитивное ненаблюдаемое искажение	718
36.5.2. Монотонность инструментальных величин и локальный средний эффект воздействия	720
36.5.2.1. Оценивание	722
36.5.3. Двухэтапный метод наименьших квадратов	724
36.6. Разность разностей	725
36.6.1. Оценивание	729
36.7. Проверки правдоподобности	729
36.7.1. Проверки на плацебо	730
36.7.2. Анализ чувствительности к ненаблюдаемому искажению	731
36.7.2.1. Калибровка с использование наблюдаемых данных	737
36.7.2.2. Практическое применение	738
36.8. do-исчисление	740
36.8.1. Три правила.....	740
36.8.2. Еще раз о поправке на черные ходы.....	741
36.8.3. Поправка на парадные ходы	742
36.9. Для дополнительного чтения.....	744
Предметный указатель	746

Часть III

Предсказание

Предсказательные модели: общий обзор

14.1. ВВЕДЕНИЕ

Большая часть машинного обучения посвящена решению единственной задачи – научиться предсказывать выходы \mathbf{y} по входам \mathbf{x} с помощью некоторой функции f , которая оценивается на основе размеченного обучающего набора $\mathcal{D} = \{(\mathbf{x}_n, \mathbf{y}_n) : n = 1 : N\}$, где $\mathbf{x}_n \in \mathcal{X} \subseteq \mathbb{R}^D$, $\mathbf{y}_n \in \mathcal{Y} \subseteq \mathbb{R}^C$. Недостоверность правильного выхода для заданного входа можно смоделировать условной вероятностной моделью вида $p(\mathbf{y}|\mathbf{x})$. Когда \mathcal{Y} является дискретным множеством меток, это называется (в литературе по МО) **дискриминантной моделью**, потому что она различает (дискриминирует) возможные значения \mathbf{y} . Если выход вещественный, $\mathcal{Y} = \mathbb{R}$, то модель называется **моделью регрессии**. (В литературе по статистике термин «модель регрессии» используется в обоих случаях, даже если \mathcal{Y} – дискретное множество.) Мы будем использовать более общий термин «**предсказательная модель**», говоря о таких моделях.

Предсказательную модель можно рассматривать как частный случай условной порождающей модели (см. главу 20). В предсказательной модели размерность выхода обычно мала и существует единственный наилучший ответ, который мы и хотим предсказать. Однако в большинстве порождающих моделей выход обычно имеет высокую размерность, как, например, в случае изображений или предложений языка, и для заданного входа может быть много правильных выходов. Мы обсудим разнообразие предсказательных моделей в разделе 14.1.1, но детали отложим до последующих глав. Далее в этой главе обсуждаются проблемы, свойственные всем типам предсказательных моделей, независимо от формы, в частности оценка их качества.

14.1.1. Типы моделей

Существует много разных видов предсказательных моделей $p(\mathbf{y}|\mathbf{x})$. Самым существенным является различие между **параметрическими моделями**, имеющими фиксированное число параметров, не зависящее от размера обучающего набора, и **непараметрическими моделями**, имеющими переменное число параметров, которое растет вместе с размером обучающего набора. Непараметрические модели обычно более гибкие, но предсказывать могут медленнее.

Большинство непараметрических моделей основаны на сравнении тестового входа \mathbf{x} с некоторыми или со всеми хранимыми обучающими примерами $\{\mathbf{x}_n, n = 1 : N\}$ путем применения некоторой формы сходства, $s_n = \mathcal{K}(\mathbf{x}, \mathbf{x}_n) \geq 0$, и последующего предсказания выхода с помощью некоторой взвешенной комбинации обучающих меток, например $\hat{\mathbf{y}} = \sum_{n=1}^N s_n \mathbf{y}_n$. Типичный пример – гауссовский процесс, который мы будем обсуждать в главе 18. Другие примеры, например модели K ближайших соседей, обсуждаются в первом томе этой книги, [Mur22].

Большинство параметрических моделей имеют вид $p(\mathbf{y}|\mathbf{x}) = p(\mathbf{y}|f(\mathbf{x}; \boldsymbol{\theta}))$, где f – некоторая функция предсказания параметров (например, среднего или логитов) выходного распределения (например, гауссова или категориального). К нашим услугам много видов функций. Если f – линейная функция $\boldsymbol{\theta}$ (т. е. $f(\mathbf{x}; \boldsymbol{\theta}) = \boldsymbol{\theta}^T \boldsymbol{\phi}(\mathbf{x})$) для некоторого фиксированного преобразования признаков $\boldsymbol{\phi}$, то модель называется обобщенной линейной моделью (*англ.* generalized linear model – GLM), такие модели обсуждаются в главе 15. Если f – нелинейная, но дифференцируемая функция $\boldsymbol{\theta}$ (например, $f(\mathbf{x}; \boldsymbol{\theta}) = \boldsymbol{\theta}_2^T \boldsymbol{\phi}(\mathbf{x}; \boldsymbol{\theta}_1)$) для некоторой допускающей обучение функции $\boldsymbol{\phi}(\mathbf{x}; \boldsymbol{\theta}_1)$, то принято представлять f с помощью нейронной сети (глава 16). Другие типы предсказательных моделей, например решающие деревья и случайные леса, обсуждаются в первом томе этой книги, [Mur22].

14.1.2. Обучение модели с помощью ERM, MLE и MAP

В этом разделе мы кратко обсудим некоторые методы, применяемые для обучения (параметрических) моделей. Самый распространенный подход – использовать **оценку максимального правдоподобия**, или **MLE**, которая сводится к решению следующей задачи оптимизации:

$$\hat{\boldsymbol{\theta}} = \operatorname{argmax}_{\boldsymbol{\theta} \in \Theta} p(\mathcal{D}|\boldsymbol{\theta}) = \operatorname{argmax}_{\boldsymbol{\theta} \in \Theta} \log p(\mathcal{D}|\boldsymbol{\theta}). \quad (14.1)$$

Если набор данных состоит из N независимых и одинаково распределенных примеров, то правдоподобие разлагается в произведение $p(\mathcal{D}|\boldsymbol{\theta}) = \prod_{n=1}^N p(\mathbf{y}_n|\mathbf{x}_n, \boldsymbol{\theta})$. Таким образом, мы можем вместо этого минимизировать следующее (масштабированное) **отрицательное логарифмическое правдоподобие**:

$$\hat{\boldsymbol{\theta}} = \operatorname{argmin}_{\boldsymbol{\theta} \in \Theta} \frac{1}{N} \sum_{n=1}^N [-\log p(\mathbf{y}_n|\mathbf{x}_n, \boldsymbol{\theta})]. \quad (14.2)$$

Это можно обобщить, заменив **логарифмическую потерю** $l_n(\boldsymbol{\theta}) = -\log p(\mathbf{y}_n|\mathbf{x}_n, \boldsymbol{\theta})$ более общей функцией потерь, так что

$$\hat{\boldsymbol{\theta}} = \operatorname{argmin}_{\boldsymbol{\theta} \in \Theta} r(\boldsymbol{\theta}), \quad (14.3)$$

где $r(\boldsymbol{\theta})$ – **эмпирический риск**,

$$r(\boldsymbol{\theta}) = \frac{1}{N} \sum_{n=1}^N \ell_n(\boldsymbol{\theta}). \quad (14.4)$$

Такой подход называется **минимизацией эмпирического риска** (*англ.* empirical risk minimization – ERM).

ERM часто приводит к **переобучению**, поэтому стандартной практикой является прибавление члена штрафа, или регуляризатора:

$$\hat{\theta} = \operatorname{argmin}_{\theta \in \Theta} r(\theta) + \lambda C(\theta), \quad (14.5)$$

где $\lambda \geq 0$ управляет степенью регуляризации, а $C(\theta)$ – некоторая мера сложности. Если использовать логарифмическую потерю, определить $C(\theta) = -\log \pi_0(\theta)$, где $\pi_0(\theta)$ – некоторое априорное распределение, и положить $\lambda = 1$, то мы вернемся к **оценке MAP**

$$\hat{\theta} = \operatorname{argmax}_{\theta \in \Theta} \log p(\mathcal{D}|\theta) + \log \pi_0(\theta). \quad (14.6)$$

Эту задачу можно решить стандартными методами оптимизации (см. главу 6).

14.1.3. Обучение модели байесовскими методами, методами вариационного вывода и обобщенными байесовскими методами

Еще один способ предотвратить переобучение – оценивать *распределение вероятностей параметров*, $q(\theta)$, вместо вычисления точечной оценки. То есть мы можем попытаться оценить ERM в математическом ожидании:

$$\hat{q} = \operatorname{argmin}_{q \in \mathcal{P}(\Theta)} \mathbb{E}_{q(\theta)} [r(\theta)]. \quad (14.7)$$

Если $\mathcal{P}(\Theta)$ – пространство всех распределений вероятностей параметров, то решение будет сходиться к дельта-функции, которая отдает всю вероятность MLE. Таким образом, этот подход сам по себе не предотвращает переобучения. Однако мы можем регуляризовать задачу, не давая распределению слишком далеко отклоняться от априорного. Измерив расхождение КЛ между q и априорным распределением, получим

$$\hat{q} = \operatorname{argmin}_{q \in \mathcal{P}(\Theta)} \mathbb{E}_{q(\theta)} [r(\theta)] + \frac{1}{\lambda} D_{\text{KL}}(q \parallel \pi_0). \quad (14.8)$$

Решение этой задачи называется **апостериорным распределением Гиббса** и имеет вид

$$\hat{q}(\theta) = \frac{e^{-\lambda r(\theta)} \pi_0(\theta)}{\int e^{-\lambda r(\theta')} \pi_0(\theta') d\theta'}. \quad (14.9)$$

Оно широко используется в сообществе, занимающемся **РАС-байесовскими методами** (см., например, [Alq21]).

Теперь, предположив, что используется логарифмическая потеря, и положив $\lambda = N$, получим

$$\hat{q}(\theta) = \frac{e^{\sum_{n=1}^N \log p(\mathbf{y}_n | \mathbf{x}_n, \theta)} \pi_0(\theta)}{\int e^{\sum_{n=1}^N \log p(\mathbf{y}_n | \mathbf{x}_n, \theta')} \pi_0(\theta') d\theta'}. \quad (14.10)$$

Тогда получающееся распределение эквивалентно байесовскому апостериорному распределению:

$$\hat{q}(\boldsymbol{\theta}) = \frac{p(\mathcal{D}|\boldsymbol{\theta})\pi_0(\boldsymbol{\theta})}{\int p(\mathcal{D}|\boldsymbol{\theta}')\pi_0(\boldsymbol{\theta}')d\boldsymbol{\theta}'} \quad (14.11)$$

Часто вычислить байесовское апостериорное распределение невозможно. Мы можем упростить задачу, сосредоточив внимание на ограниченном семействе распределений, $\mathcal{Q}(\Theta) \subset \mathcal{P}(\Theta)$. Это приводит к следующей целевой функции:

$$\hat{q} = \operatorname{argmin}_{q \in \mathcal{Q}(\Theta)} \mathbb{E}_{q(\boldsymbol{\theta})} [-\log p(\mathcal{D}|\boldsymbol{\theta})] + D_{\text{KL}}(q \parallel \pi_0). \quad (14.12)$$

Такой подход называется **вариационным выводом**, детали см. в главе 10.

Его можно обобщить, заменив отрицательное логарифмическое правдоподобие риском общего вида, $r(\boldsymbol{\theta})$. Кроме того, расхождение КЛ можно заменить общим расхождением, $D(q|\pi_0)$, которому можно назначить вес λ . В результате получится такая целевая функция:

$$\hat{q} = \operatorname{argmin}_{q \in \mathcal{Q}(\Theta)} \mathbb{E}_{q(\boldsymbol{\theta})} [r(\boldsymbol{\theta})] + \lambda D(q|\pi_0). \quad (14.13)$$

Это называется **обобщенным байесовским выводом** [BHW16; KJD19; KJD21].

14.2. ВЫЧИСЛЕНИЕ ПРЕДСКАЗАТЕЛЬНЫХ МОДЕЛЕЙ

В этом разделе мы обсудим, как оценить качество обученной дискриминантной модели.

14.2.1. Собственные скоринговые правила

Общепринято измерять качество предсказательной модели с помощью **собственного скорингового правила** [GR07a], определяемого следующим образом. Пусть $S(p_\theta, (y, \mathbf{x}))$ – оценка предсказательного распределения $p_\theta(y|\mathbf{x})$ при заданном событии $y|\mathbf{x} \sim p^*(y|\mathbf{x})$, где p^* – истинное условное распределение. (Если мы хотим оценить байесовскую модель, где параметр $\boldsymbol{\theta}$ маргинализируется вместо обусловливания по нему, то просто заменяем $p_\theta(y|\mathbf{x})$ на $p(y|\mathbf{x}) = \int p_\theta(y|\mathbf{x})p(\boldsymbol{\theta}|\mathcal{D})d\boldsymbol{\theta}$.) Ожидаемая оценка определяется так:

$$S(p_\theta, p^*) = \int p^*(\mathbf{x})p^*(y|\mathbf{x})S(p_\theta, (y, \mathbf{x}))dyd\mathbf{x}. \quad (14.14)$$

Собственное скоринговое правило – это такая оценка, для которой $S(p_\theta, p^*) \leq S(p^*, p^*)$, причем равенство достигается тогда и только тогда, когда $p_\theta(y|\mathbf{x}) = p^*(y|\mathbf{x})$. Таким образом, максимизация такого собственного скорингового правила заставляет модель находить истинные вероятности.

Логарифмическое правдоподобие, $S(p_\theta, (y, \mathbf{x})) = \log p_\theta(y|\mathbf{x})$, является собственным правилом оценивания. Это следует из неравенства Гиббса:

$$S(p_\theta, p^*) = \mathbb{E}_{p^*(\mathbf{x})p^*(y|\mathbf{x})} [\log p_\theta(y|\mathbf{x})] \leq \mathbb{E}_{p^*(\mathbf{x})p^*(y|\mathbf{x})} [\log p^*(y|\mathbf{x})]. \quad (14.15)$$

Поэтому минимизация отрицательного логарифмического правдоподобия (иначе говоря, логарифмической потери) должна давать хорошо откалиброванные вероятности. Однако на практике логарифмическая потеря может придавать чрезмерный вес хвостовым вероятностям [QC+06].

Распространенная альтернатива – использовать **оценку Брайера**, определяемую следующим образом:

$$S(p_{\theta}, (y, \mathbf{x})) \triangleq \frac{1}{C} \sum_{c=1}^C (p_{\theta}(y = c | \mathbf{x}) - \mathbb{I}(y = c))^2. \quad (14.16)$$

Это просто квадратичная ошибка предсказательного распределения $\mathbf{p} = p(1 : C | \mathbf{x})$ по сравнению с унитарным распределением меток y . Поскольку оценка Брайера основана на квадратичной ошибке, она менее чувствительна к исключительно редким или исключительно частым классам. Она также является собственным скоринговым правилом.

14.2.2. Калибровка

Модель, для которой предсказанные вероятности совпадают с эмпирическими частотами, называется **откалиброванной** [Daw82; NMC05; Guo+17]. Например, если классификатор предсказывает $p(y = c | \mathbf{x}) = 0.9$, то мы ожидаем, что эта метка будет истинной в 90 % случаев. Хорошо откалиброванная модель полезна, поскольку позволяет избегать неверных решений, когда исход слишком неопределенный. В разделах ниже мы обсудим некоторые способы измерить и улучшить калибровку.

14.2.2.1. Ожидаемая ошибка калибровки

Для оценивания калибровки мы разбиваем предсказанные вероятности на конечное множество интервалов и оцениваем расхождение между эмпирической и предсказанной вероятностями путем подсчета. Точнее, предположим, что имеется B интервалов. Обозначим \mathcal{B}_b множество индексов примеров, для которых уверенность в правильности предсказания попадает в интервал $I_b = (\frac{b-1}{B}, \frac{b}{B}]$. Здесь мы использовали интервалы равной ширины, но можно было бы определить интервалы так, что в каждый попадает одинаковое число примеров.

Обозначим $f(\mathbf{x})_c = p(y = c | \mathbf{x})$, $\hat{y}_n = \operatorname{argmax}_{c \in \{1, \dots, C\}} f(\mathbf{x}_n)_c$ и $\hat{p}_n = \max_{c \in \{1, \dots, C\}} f(\mathbf{x}_n)_c$. Точность в интервале b определяется как

$$\operatorname{acc}(\mathcal{B}_b) = \frac{1}{|\mathcal{B}_b|} \sum_{n \in \mathcal{B}_b} \mathbb{I}(\hat{y}_n = y_n). \quad (14.17)$$

Средняя достоверность в этом интервале определяется как

$$\operatorname{conf}(\mathcal{B}_b) = \frac{1}{|\mathcal{B}_b|} \sum_{n \in \mathcal{B}_b} \hat{p}_n. \quad (14.18)$$

Построив график зависимости точности от достоверности, мы получим **диаграмму надежности**, представленную на рис. 14.1. Расхождение между точностью и достоверностью показано красными столбиками. Его можно изме-

рять с помощью **ожидаемой ошибки калибровки** (англ. expected calibration error – **ECE**) [NCH15]:

$$\text{ECE}(f) = \sum_{b=1}^B \frac{|\mathcal{B}_b|}{B} |\text{acc}(\mathcal{B}_b) - \text{conf}(\mathcal{B}_b)|. \quad (14.19)$$

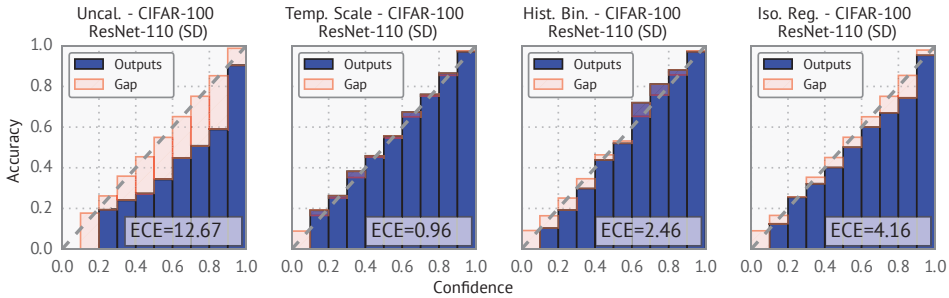


Рис. 14.1. Диаграммы надежности для классификатора изображений на основе CHC ResNet CNN [He+16b] в применении к набору данных CIFAR-100. ECE (expected calibration error) – ожидаемая ошибка калибровки, которая измеряет размер выделенных красным цветом расхождений. Методы слева направо: исходные вероятности, после температурного масштабирования, после распределения по интервалам, после изотонической регрессии. На основе рис. 4 из работы [Guo+17]. Печатается с разрешения Чуань Гуо

В случае нескольких классов ECE смотрит только на ошибку в предсказании MAP (лучшая метка). Мы можем обобщить эту метрику, так чтобы она принимала во внимание все классы, введя в рассмотрение **маргинальную ошибку калибровки** (MCE), предложенную в работе [KLM19]:

$$\text{MCE} = \sum_{c=1}^C w_c \mathbb{E} [(p(Y = c | f(\mathbf{x})_c) - f(\mathbf{x})_c)^2] \quad (14.20)$$

$$= \sum_{c=1}^C w_c \sum_{b=1}^B \frac{|\mathcal{B}_{b,c}|}{B} (\text{acc}(\mathcal{B}_{b,c}) - \text{conf}(\mathcal{B}_{b,c}))^2, \quad (14.21)$$

где $\mathcal{B}_{b,c}$ – b -й интервал для класса c , а $w_c \in [0, 1]$ обозначает значимость класса c . (Мы можем положить $w_c = 1/C$, если все классы одинаково значимы.) В работе [Nix+19] эта метрика названа **статической ошибкой калибровки** и показано, что некоторые методы с хорошей ECE могут иметь плохую MCE. Другие метрики многоклассовой калибровки обсуждаются в работе [WLZ19].

14.2.2.2. Улучшение калибровки

В принципе, при любой методике обучения классификатора, которая оптимизирует собственное скоринговое правило (например, NLL), автоматически должен получаться хорошо откалиброванный классификатор. Однако на практике несбалансированные наборы данных могут приводить к плохо откалиброванным предсказаниям. Ниже мы обсудим различные способы улучшения калибровки вероятностных классификаторов, следуя работе [Guo+17].

14.2.2.3. Масштабирование Платта

Пусть z – логарифмическое отношение шансов, или логит, а значение $p = \sigma(z)$ порождено вероятностным бинарным классификатором. Мы хотим преобразовать его в лучше откалиброванное значение q . Простейший способ сделать это называется **масштабированием Платта** и был предложен в работе [Pla00]. Идея в том, чтобы вычислить $q = \sigma(az + b)$, где a и b оцениваются по максимальному правдоподобию на контрольном наборе.

В случае нескольких классов мы можем обобщить масштабирование Платта, воспользовавшись масштабированием матриц: $\mathbf{q} = \text{softmax}(\mathbf{Wz} + \mathbf{b})$, где \mathbf{W} и \mathbf{b} оцениваются с помощью максимального правдоподобия на контрольном наборе. Поскольку \mathbf{W} имеет $K \times K$ параметров, где K – количество классов, этот метод склонен к переобучению, поэтому на практике мы ограничиваемся диагональными матрицами \mathbf{W} .

14.2.2.4. Непараметрические (гистограммные) методы

Метод масштабирования Платта делает сильное предположение о форме калибровочной кривой. Более гибкий, непараметрический метод заключается в том, чтобы распределить предсказанные вероятности по интервалам, p_m , и оценивать эмпирическую вероятность q_m для каждого такого интервала; затем мы заменяем p_m на q_m ; это называется **построением гистограммы** [ZE01a]. Данный метод можно регуляризовать, потребовав, чтобы функция $q = f(p)$ была кусочно-постоянной и неубывающей; это называется **изотонической регрессией** [ZE01a]. Альтернативный подход, называемый **калибровкой с масштабированием и распределением по интервалам** (англ. *scaling-binning calibrator*), заключается в том, чтобы применить какой-нибудь метод масштабирования (например, Платта), а затем к результату – построение гистограммы. Преимущество в том, что в каждом интервале оказываются среднее масштабированных вероятностей вместо среднего наблюдаемых бинарных меток (см. рис. 14.2). В работе [KLM19] доказано, что это приводит к лучшей калибровке вследствие меньшей дисперсии оценки.

В случае нескольких классов \mathbf{z} – вектор логитов, а $p = \text{softmax}(\mathbf{z})$ – вектор вероятностей. Мы хотим преобразовать его в лучше откалиброванную версию, q . В работе [ZE01b] предлагается обобщить построение гистограммы и изотоническую регрессию на этот случай, применив описанный выше бинарный метод к каждой из K задач «один против остальных», где K – количество классов. Однако для этого требуется K отдельных моделей калибровки, и в результате получается ненормированное распределение вероятностей.

14.2.2.5. Температурное масштабирование

В работе [Guo+17] эмпирически установлено, что диагональная версия масштабирования Платта применительно к разнообразным глубоким нейронным сетям часто заканчивалась обучением вектора вида $\mathbf{w} = (c, c, \dots, c)$ для некоторой постоянной c . Это наводит на мысль о существовании более простой формы масштабирования, которую авторы называют **температурным**: $\mathbf{q} = \text{softmax}(\mathbf{z}/T)$, где $T > 0$ – параметр температуры, который можно оценить с помощью максимального правдоподобия на контрольном наборе. Этот па-

раметр делает пики распределения менее выраженными, как показано на рис. 14.3. В работе [Guo+17] эмпирически показано, что этот метод порождает наименьшую ESE в различных задачах классификации с применением ГНС (см. визуализацию на рис. 14.1). Кроме того, он гораздо проще и быстрее других методов.

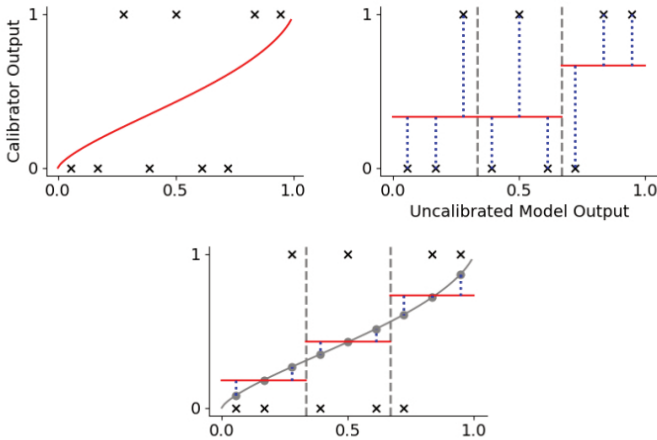


Рис. 14.2. Визуализация трех разных подходов к калибровке бинарного вероятностного классификатора. Черные крестики – наблюдаемые бинарные метки, красные линии – откалиброванные выходы: (а) масштабирование Платта; (б) построение гистограммы с тремя интервалами. Выход в каждом интервале – среднее попавших в него бинарных меток; (с) калибровка с масштабированием и построением гистограммы. Сначала применяется масштабирование Платта, а затем вычисляется среднее масштабированных точек (серые кружочки) в каждом интервале. На основе рис. 1 из работы [KLM19]. Печатается с разрешения Аниши Кумара

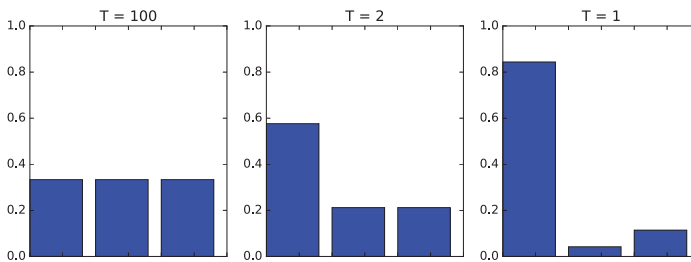


Рис. 14.3. Распределение $\text{softmax}(a/T)$, где $a = (3, 0, 1)$, при температурах $T = 100$, $T = 2$ и $T = 1$. Когда температура высока (слева), распределение равномерное, а когда мала (справа), распределение «остроконечное», и наибольшая масса вероятности приходится на самый большой элемент. Построено программой `softmax_plot.ipynb`

Отметим, что масштабирование Платта и температурное масштабирование не влияют на метку самого вероятного класса, так что эти методы не оказывают влияния на верность классификации. Однако они улучшают качество калибровки. Более современный метод многоклассовой калибровки обсуждается в работе [Kul+19].

14.2.2.6. Сглаживание меток

При обучении классификаторов истинная целевая метка обычно представляется унитарным вектором – скажем, $\mathbf{y} = (0, 1, 0)$ представляет класс 2 из трех. Результаты можно улучшить, если «размазать» часть массы вероятности по всем интервалам. Например, можно положить $\mathbf{y} = (0.1, 0.8, 0.1)$. Это называется **сглаживанием меток** и часто приводит к лучше откалиброванным моделям [МКН19].

14.2.2.7. Байесовские методы

Байесовские подходы к обучению классификаторов часто приводят к лучше откалиброванным предсказаниям, т. к. представляют неопределенность в параметрах. Пример см. в разделе 17.3.8. Однако в работе [Ova+19] показано, что хорошо откалиброванные модели (даже байесовские) часто становятся откалиброванными неправильно, если применяются к входам, происходящим из другого распределения (детали см. в разделе 19.2).

14.2.3. За пределами вычисления маргинальных вероятностей

Калибровка (раздел 14.2.2) относится в первую очередь к оценке свойств маргинального предсказательного распределения $p(\mathbf{y}|\mathbf{x})$. Но иногда этого недостаточно для различения хорошей и плохой моделей, особенно в контексте онлайн-обучения и последовательного принятия решений, как отмечено в работах [Lu+22; Osb+21; WSG21; KKG22]. Рассмотрим, к примеру, двух обучающихся агентов, которые наблюдают последовательность подбрасываний монеты. Обозначим исход в момент t $Y_t \sim \text{Ver}(\theta)$, где θ – неизвестный параметр. Агент 1 полагает, что $\theta = 2/3$, а агент 2 – что $\theta = 0$ или $\theta = 1$, но не уверен, какое именно значение истинно, и назначает этим событиям вероятности $1/3$ и $2/3$. Таким образом, оба агента, несмотря на различные модели, делают одинаковые предсказания следующего исхода: $p(Y_i^1 = 0) = 1/3$ для агентов $i = 1, 2$. Однако предсказания агентов о *последовательности* τ будущих исходов сильно различаются: агент 1 предсказывает, что каждое отдельное подбрасывание монеты – событие, имеющее распределение Бернулли, вероятностный характер которого обусловлен неустранимым шумом, или **алеаторической неопределенностью**:

$$p(Y_1^1 = 0, \dots, Y_\tau^1 = 0) = \frac{1}{3^\tau}. \quad (14.22)$$

С другой стороны, агент 2 предсказывает, что выпадут либо все орлы, либо все решки, а вероятностный характер обусловлен **эпистемической неопределенностью** истинных параметров:

$$p(Y_1^2 = y_1, \dots, Y_\tau^2 = y_\tau) = \begin{cases} 1/3, & \text{если } y_1 = \dots = y_\tau = 0 \\ 2/3, & \text{если } y_1 = \dots = y_\tau = 1. \\ 0 & \text{в противном случае.} \end{cases} \quad (14.23)$$

Различие предположений этих агентов влияет на их поведение. Например, в казино агент 1 считает небольшим риском раз за разом ставить на выпадение

орлов в конечном итоге, но для агента 2 такая стратегия была бы крайне неразумной, и стоило бы для начала собрать информацию (провести исследование).

Исходя из вышеизложенного, мы видим, что при оценивании предсказательных моделей полезно вычислять совместные предсказательные распределения. В работах [Lu+22; Osb+21] предлагается вычислять апостериорные предсказательные распределения τ исходов $\mathbf{y} = Y_{T+1:T+\tau}$ при условии множества τ входов $\mathbf{x} = X_{T:T+\tau-1}$ и прошлых T примеров данных, $\mathcal{D}_T = \{(X_t, Y_{t+1}) : t = 0, 1, \dots, T-1\}$. Байесовское оптимальное предсказательное распределение имеет вид

$$P_T^B = p(\mathbf{y}|\mathbf{x}, \mathcal{D}_T). \quad (14.24)$$

Обычно вычислить его невозможно. Вместо этого агент использует приближенное распределение, называемое **доверительным состоянием**, которое мы обозначим

$$Q_T = p(\mathbf{y}|\mathbf{x}, \mathcal{D}_T). \quad (14.25)$$

Естественной метрикой качества является расхождение КЛ между этими распределениями. Поскольку она зависит от входов \mathbf{x} и $\mathcal{D}_T = (X_{0:T-1}, Y_{1:T})$, мы усредним расхождение КЛ по этим значениям, выбираемым независимо из истинного распределения, порождающего данные, которое мы обозначим

$$Q_T = p(\mathbf{y}|\mathbf{x}, \mathcal{D}_T), \quad (14.26)$$

где \mathcal{E} – истинное, но неизвестное окружение. Таким образом, мы определяем нашу метрику как

$$d_{B,Q}^{KL} = \mathbb{E}_{P(\mathbf{x}, \mathcal{D}_T)} [D_{\text{KL}}(P^B(\mathbf{y}|\mathbf{x}, \mathcal{D}_T) \parallel Q(\mathbf{y}|\mathbf{x}, \mathcal{D}_T))], \quad (14.27)$$

где

$$P(\mathbf{x}, \mathcal{D}_T, \mathcal{E}) = P(\mathcal{E}) \underbrace{\left[\prod_{t=0}^{T-1} P(X_t|\mathcal{E})P(Y_{t+1}|X_t, \mathcal{E}) \right]}_{P(\mathcal{D}_T|\mathcal{E})} \underbrace{\left[\prod_{t=T}^{T+\tau-1} P(x_t|\mathcal{E}) \right]}_{P(\mathbf{x}|\mathcal{E})}, \quad (14.28)$$

а $P(\mathbf{x}, \mathcal{D}_T)$ – результат его маргинализации по окружениям.

К сожалению, вычислить точное байесовское апостериорное распределение P_T^B обычно невозможно, поэтому мы не можем вычислить $d_{B,Q}^{KL}$. Однако в разделе 14.2.3.1 мы покажем, что

$$d_{B,Q}^{KL} = d_{\mathcal{E},Q}^{KL} - \mathbb{I}(\mathcal{E}; \mathbf{y}|\mathcal{D}_T, \mathbf{x}), \quad (14.29)$$

где второй член не зависит от агента, а первый равен

$$d_{\mathcal{E},Q}^{KL} = \mathbb{E}_{P(\mathbf{x}, \mathcal{D}_T, \mathcal{E})} [D_{\text{KL}}(P(\mathbf{y}|\mathbf{x}, \mathcal{E}) \parallel Q(\mathbf{y}|\mathbf{x}, \mathcal{D}_T))] \quad (14.30)$$

$$= \mathbb{E}_{P(\mathbf{y}|\mathbf{x}, \mathcal{E})P(\mathbf{x}, \mathcal{D}_T, \mathcal{E})} \left[\log \frac{P(\mathbf{y}|\mathbf{x}, \mathcal{E})}{Q(\mathbf{y}|\mathbf{x}, \mathcal{D}_T)} \right]. \quad (14.31)$$

Поэтому если ранжировать агентов в терминах $d_{\mathcal{E},Q}^{KL}$, то получим те же самые результаты, что при ранжировании по $d_{B,Q}^{KL}$.

Чтобы вычислить $d_{\mathcal{E},Q}^{KL}$ на практике, можно использовать аппроксимацию Монте-Карло: нужно просто выбрать J окружений, $\mathcal{E}^j \sim P(\mathcal{E})$, выбрать обучающий набор \mathcal{D}_T из каждого окружения, $\mathcal{D}_T^j \sim P(\mathcal{D}_T | \mathcal{E}^j)$, а затем выбрать N векторов данных длины τ , $(\mathbf{x}_n^j, \mathbf{y}_n^j) \sim P(X_{T:T+\tau-1}, Y_{T+1:T+\tau} | \mathcal{E}^j)$. После этого можно вычислить

$$\hat{d}_{\mathcal{E},Q}^{KL} = \frac{1}{JN} \sum_{j=1}^J \sum_{n=1}^N \left[\log P(\mathbf{y}_n^j | \mathbf{x}_n^j, \mathcal{E}^j) - \log Q(\mathbf{y}_n^j | \mathbf{x}_n^j, \mathcal{D}_T^j) \right], \quad (14.32)$$

где

$$p_{jn} = P(\mathbf{y}_n^j | \mathbf{x}_n^j, \mathcal{E}^j) = \prod_{t=T}^{T+\tau-1} P(Y_{n,t+1}^j | X_{n,t}^j, \mathcal{E}^j), \quad (14.33)$$

$$q_{jn} = Q(\mathbf{y}_n^j | \mathbf{x}_n^j, \mathcal{D}_T^j) = \int Q(\mathbf{y}_n^j | \mathbf{x}_n^j, \boldsymbol{\theta}) Q(\boldsymbol{\theta} | \mathcal{D}_T^j) d\boldsymbol{\theta} \quad (14.34)$$

$$\approx \frac{1}{M} \sum_{m=1}^M \prod_{t=T}^{T+\tau-1} Q(Y_{n,t+1}^j | X_{n,t}^j, \boldsymbol{\theta}_m^j), \quad (14.35)$$

где $\boldsymbol{\theta}_m^j \sim Q(\boldsymbol{\theta} | \mathcal{D}_T^j)$ – выборка из апостериорного распределения окружений агента.

Выше предполагалось, что $P(Y|X)$ известно; так будет, если мы используем генератор синтетических данных, как в «нейронном испытательном стенде», описанном в работе [Osб+21]. Если мы имеем только J эмпирических распределений $P^j(X, Y)$, то можем заменить расхождение КЛ перекрестной энтропией, которая отличается только аддитивной постоянной:

$$\hat{d}_{\mathcal{E},Q}^{KL} = \mathbb{E}_{P(\mathbf{x}, \mathcal{D}_T, \mathcal{E})} [D_{\text{KL}}(P(\mathbf{y} | \mathbf{x}, \mathcal{E}) \parallel Q(\mathbf{y} | \mathbf{x}, \mathcal{D}_T))] \quad (14.36)$$

$$= \underbrace{\mathbb{E}_{P(\mathbf{x}, \mathbf{y}, \mathcal{E})} [\log P(\mathbf{y} | \mathbf{x}, \mathcal{E})]}_{\text{const}} - \underbrace{\mathbb{E}_{P(\mathbf{x}, \mathbf{y}, \mathcal{D}_T | \mathcal{E}) P(\mathcal{E})} [\log Q(\mathbf{y} | \mathbf{x}, \mathcal{D}_T)]}_{d_{\mathcal{E},Q}^{CE}}, \quad (14.37)$$

где последний член – просто эмпирическое отрицательное логарифмическое правдоподобие (NLL) агента на примерах из окружения. Следовательно, если мы ранжируем агентов в терминах их NLL или перекрестной энтропии $d_{\mathcal{E},Q}^{CE}$, то получим такие же результаты, как при ранжировании их по $d_{\mathcal{E},Q}^{KL}$, что, в свою очередь, дает такие же результаты, как при ранжировании их по $d_{B,Q}^{KL}$.

На практике перекрестную энтропию можно аппроксимировать следующим образом:

$$\hat{d}_{\mathcal{E},Q}^{CE} = -\frac{1}{JN} \sum_{j=1}^J \sum_{n=1}^N \log Q(\mathbf{y}_n^j | \mathbf{x}_n^j, \mathcal{D}_T^j), \quad (14.38)$$

где $\mathcal{D}_T^j \sim P^j$ и $(\mathbf{x}_n^j, \mathbf{y}_n^j) \sim P^j$.

Альтернативой оцениванию расхождения КЛ или NLL является оценка совместного предсказательного распределения путем использования его в последующем задании. В работе [Osб+21] показано, что хорошая верность предсказаний (для $\tau > 1$) коррелирует с хорошим качеством на задаче о бандите (см. раздел 34.4). В работе [WSG21] показано, что хорошая верность предсказаний (для $\tau > 1$) ведет к хорошему качеству на задаче трансдуктивного активного обучения.

14.2.3.1. Доказательство утверждения

Теперь мы докажем формулу (14.29), следуя работе [Lu+21a]. Прежде всего отметим, что

$$d_{\mathcal{E},Q}^{KL} = \mathbb{E}_{P(\mathbf{x}, \mathcal{D}_T, \mathcal{E})P(\mathbf{y}|\mathbf{x}, \mathcal{E})} \left[\log \frac{P(\mathbf{y}|\mathbf{x}, \mathcal{E})}{Q(\mathbf{y}|\mathbf{x}, \mathcal{D}_T)} \right] \quad (14.39)$$

$$= \mathbb{E} \left[\log \frac{P(\mathbf{y}|\mathbf{x}, \mathcal{D}_T)}{Q(\mathbf{y}|\mathbf{x}, \mathcal{D}_T)} \right] + \mathbb{E} \left[\log \frac{P(\mathbf{y}|\mathbf{x}, \mathcal{E})}{P(\mathbf{y}|\mathbf{x}, \mathcal{D}_T)} \right]. \quad (14.40)$$

Для первого члена (14.40) имеем

$$\mathbb{E} \left[\log \frac{P(\mathbf{y}|\mathbf{x}, \mathcal{D}_T)}{Q(\mathbf{y}|\mathbf{x}, \mathcal{D}_T)} \right] = \sum P(\mathbf{x}, \mathbf{y}, \mathcal{D}_T) \log \frac{P(\mathbf{y}|\mathbf{x}, \mathcal{D}_T)}{Q(\mathbf{y}|\mathbf{x}, \mathcal{D}_T)} \quad (14.41)$$

$$= \sum P(\mathbf{x}, \mathcal{D}_T) \sum P(\mathbf{y}|\mathbf{x}, \mathcal{D}_T) \log \frac{P(\mathbf{y}|\mathbf{x}, \mathcal{D}_T)}{Q(\mathbf{y}|\mathbf{x}, \mathcal{D}_T)} \quad (14.42)$$

$$= \mathbb{E}_{P(\mathbf{x}, \mathcal{D}_T)} [D_{\text{KL}}(P(\mathbf{y}|\mathbf{x}, \mathcal{D}_T) \parallel Q(\mathbf{y}|\mathbf{x}, \mathcal{D}_T))]. \quad (14.43)$$

Теперь покажем, что второй член в (14.40) сводится к взаимной информации. Воспользуемся тем, что

$$P(\mathbf{y}|\mathbf{x}, \mathcal{E}) = P(\mathbf{y}|\mathcal{D}_T, \mathbf{x}, \mathcal{E}) = \frac{P(\mathcal{E}, \mathbf{y}|\mathcal{D}_T, \mathbf{x})}{P(\mathcal{E}|\mathcal{D}_T, \mathbf{x})}, \quad (14.44)$$

т. к. \mathcal{D}_T не содержит новой информации, помимо \mathcal{E} . Отсюда получаем

$$\mathbb{E} \left[\log \frac{P(\mathbf{y}|\mathbf{x}, \mathcal{E})}{P(\mathbf{y}|\mathbf{x}, \mathcal{D}_T)} \right] = \mathbb{E} \left[\log \frac{P(\mathcal{E}, \mathbf{y}|\mathcal{D}_T, \mathbf{x})/P(\mathcal{E}|\mathcal{D}_T, \mathbf{x})}{P(\mathbf{y}|\mathcal{D}_T, \mathbf{x})} \right] \quad (14.45)$$

$$= \sum P(\mathcal{D}_T, \mathbf{x}) \sum P(\mathcal{E}, \mathbf{y}|\mathcal{D}_T, \mathbf{x}) \log \frac{P(\mathcal{E}, \mathbf{y}|\mathcal{D}_T, \mathbf{x})}{P(\mathbf{y}|\mathcal{D}_T, \mathbf{x})P(\mathcal{E}|\mathcal{D}_T, \mathbf{x})} \quad (14.46)$$

$$= \mathbb{I}(\mathcal{E}; \mathbf{y}|\mathcal{D}_T, \mathbf{x}). \quad (14.47)$$

Следовательно,

$$d_{\mathcal{E},Q}^{KL} = d_{B,Q}^{KL} + \mathbb{I}(\mathcal{E}; \mathbf{y}|\mathcal{D}_T, \mathbf{x}), \quad (14.48)$$

что и требовалось доказать.

14.3. КОНФОРМНОЕ ПРЕДСКАЗАНИЕ

В этом разделе мы кратко обсудим **конформное предсказание** [VGS05; SV08; ZFV20; AB21; KSB21; Man22b]. Это простой, но эффективный способ создания интервалов или наборов предсказаний с гарантированной частотной вероятностью покрытия из любого метода предсказания $p(y|x)$. Его можно рассматривать как форму **количественного выражения неопределенности, не зависящую от распределения**, поскольку он не делает никаких предположений (кроме перестановочности данных) об истинном процессе, порождающем данные, или о форме модели¹. Наше изложение основано на прекрасном пособии [AB21]².

При конформном предсказании мы начинаем с некоторого эвристического понятия неопределенности, например оценки softmax для задачи классификации или дисперсии для задачи регрессии, и используем ее, чтобы определить **конформную оценку** $s(\mathbf{x}, y) \in \mathbb{R}$, измеряющую, насколько плохо выход y соответствует («конформен») \mathbf{x} . (Большие значения оценки маловероятны, поэтому лучше считать, что это оценка неконформности.) Затем эта оценка применяется к **калибровочному набору** из n помеченных примеров, которые не использовались во время обучения f , чтобы получить $S = \{s_i = s(\mathbf{x}_i, y_i) : i = 1 : n\}$ ³. Пользователь задает желательный доверительный порог α , скажем 0.1, после чего вычисляется $(1 - \alpha)$ -й квантиль \hat{q} множества S . (На самом деле следует заменить $1 - \alpha$ величиной $\frac{\lceil (n+1)(1-\alpha) \rceil}{n}$, чтобы учесть конечность размера S .) Наконец, получив новый тестовый вход \mathbf{x}_{n+1} , мы вычисляем набор предсказаний

$$\mathcal{T}(\mathbf{x}_{n+1}) = \{y : s(\mathbf{x}_{n+1}, y) \leq \hat{q}\}. \quad (14.49)$$

Интуитивно понятно, что мы включаем все выходы y , правдоподобные при заданном входе. См. иллюстрацию на рис. 14.4.

Удивительно, что можно доказать следующий общий результат

$$1 - \alpha \leq P^*(y^{n+1} \in \mathcal{T}(\mathbf{x}_{n+1})) \leq 1 - \alpha + \frac{1}{n+1}, \quad (14.50)$$

где вероятность берется относительно истинного распределения $P^*(\mathbf{x}_{n+1}, y_{n+1})$. Говорят, что набор предсказаний имеет уровень **покрытия** $1 - \alpha$. Это верно для любого $n \geq 1$ и $\alpha \in [0, 1]$. Единственное предположение – перестановочность значений (\mathbf{x}_i, y_i) , а следовательно, и калибровочных оценок s_i .

¹ Предположение о перестановочности исключает данные из временных рядов, потому что они последовательно коррелированы. Однако были разработаны обобщения конформного предсказания на случай временных рядов, см., например, [Zaf+22]. Кроме того, предположение о перестановочности исключает дрейф распределения, хотя и эта проблема частично решена.

² См. также простую в использовании библиотеку **MAPIE**, написанную на Python, по адресу <https://mapie.readthedocs.io/en/latest/index.html> и список статей в работе [Man22a].

³ Использование калибровочного набора называется **расщепленным конформным предсказанием** (англ. split conformal prediction). Если данных недостаточно для такого подхода с расщеплением, то используется **полное конформное предсказание** [VGS05], для которого необходимо обучать модель n раз, применяя процедуру исключения по одному.

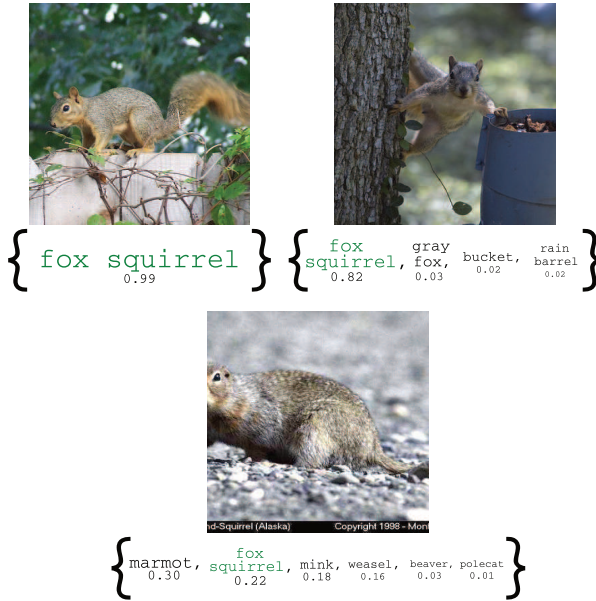


Рис. 14.4. Примеры из набора предсказаний в Imagenet. Показаны три примера увеличивающейся трудности из класса «черная белка» и наборы предсказаний, сгенерированные методом конформного предсказания. (Сравните с рис. 17.9.) На основе рис. 1 из работы [AB21]. Печатается с разрешения Анастасиоса Ангелопулоса

Чтобы понять, почему это верно, отсортируем оценки: $s_1 < \dots < s_n$, так что $\hat{q} = s_i$, где $i = \lfloor d(n+1)(1-\alpha) \rfloor / n$. (Для простоты предполагается, что оценки различны.) Оценка s_{n+1} с одинаковой вероятностью попадает между любой парой калибровочных точек s_1, \dots, s_n , т. к. эти точки перестановочны. Поэтому

$$P^*(s_{n+1} \leq s_k) = \frac{k}{n+1} \tag{14.51}$$

для любых $k \in \{1, \dots, n+1\}$. Событие $\{y_{n+1} \in \mathcal{T}(x_{n+1})\}$ эквивалентно $\{s_{n+1} \leq \hat{q}\}$. Следовательно,

$$P^*(y_{n+1} \in \mathcal{T}(x_{n+1})) = P^*(s_{n+1} \leq \hat{q}) = \frac{\lceil (n+1)(1-\alpha) \rceil}{n+1} \geq 1 - \alpha. \tag{14.52}$$

Доказательство верхней границы см. в работе [Lei+18].

Хотя этот результат может показаться «бесплатным завтраком», стоит отметить, что мы всегда можем добиться желаемого уровня покрытия, определив в качестве набора предсказаний все возможные метки. В таком случае набор предсказаний не будет зависеть от входа, но будет покрывать истинную метку в $1 - \alpha$ случаев. Чтобы исключить некоторые вырожденные случаи, мы ищем как можно меньшие наборы предсказаний (хотя допускаем большие наборы для трудных примеров), удовлетворяющие требованию покрытия. Для достижения этой цели необходимо определить подходящие конформные оценки. Ниже приведено несколько примеров вычисления конформных оценок

$s(\mathbf{x}, y)$ для задач разного рода¹. Важно также отметить, что гарантии покрытия по природе своей частотные и относятся к среднему поведению, а не представляют неопределенность на уровне экземпляра, как при байесовском подходе.

14.3.1. Конформализация классификации

Самый простой способ применить конформное предсказание к многоклассовой классификации – вывести конформную оценку из оценки softmax, назначенной метке, по формуле $s(\mathbf{x}, y) = 1 - f(\mathbf{x})_y$, так чтобы большие значения считались менее вероятными, чем малые. Порог \hat{q} вычисляется, как описано выше, после чего мы определяем набор предсказаний как $\mathcal{T}(\mathbf{x}) = \{y : f(\mathbf{x})_y \geq 1 - \hat{q}\}$, что соответствует формуле (14.49). То есть мы берем набор всех меток класса выше заданного порога, как показано на рис. 14.4.

Хотя описанный выше подход порождает наборы предсказаний наименьшего возможного среднего размера (как доказано в работе [SLW19]), размер набора обычно оказывается слишком большим для легких примеров и слишком малым для трудных. Далее мы представим улучшенный метод, решающий эту проблему; он называется «адаптивные наборы предсказаний» и описан в работе [RSC20]. Идея проста: мы сортируем все оценки softmax, $f(\mathbf{x})_c$ для $c = 1 : C$ и получаем перестановку $\pi_{1:C}$, а затем определяем $s(\mathbf{x}, y)$ как сумму оценок до достижения метки y : $s(\mathbf{x}, y) = \sum_{c=1}^k f(\mathbf{x})_{\pi_c}$, где $k = \pi_y$. Теперь вычисляем \hat{q} , как и раньше, и определяем набор предсказаний $\mathcal{T}(\mathbf{x})$ как множество всех меток, отсортированное в порядке убывания вероятностей, пока не покроем долю q массы вероятности. См. иллюстрацию на рис. 14.5(а). Здесь используются все оценки softmax, выведенные моделью, а не только самая верхняя, и именно поэтому качество оказывается лучше.

14.3.2. Конформализация регрессии

В этом разделе мы рассмотрим конформализованные задачи регрессии. Поскольку теперь $y \in \mathbb{R}$, вычисление набора предсказаний в формуле (14.49) обходится дорого, поэтому взамен мы будем вычислять интервал предсказаний, определяемый нижней и верхней границами.

14.3.2.1. Конформализация квантильной регрессии

В этом разделе мы воспользуемся **квантильной регрессией** для вычисления нижней и верхней границ. Сначала обучим функцию вида $t_\gamma(\mathbf{x})$, предсказывающую квантиль функции плотности вероятности $P(Y|\mathbf{x})$. Например, если положить $\gamma = 0.5$, то получим медиану. Если взять $\gamma = 0.05$ и $\gamma = 0.95$, то получим приближенно 90%-ный интервал предсказаний $[t_{0.05}(\mathbf{x}), t_{0.95}(\mathbf{x})]$, показанный серыми линиями на рис. 14.5(б). Чтобы обучить модель квантильной регрессии, мы просто заменим квадратичную потерю **квантильной потерей**, которая еще называется **пинбольной** и определяется следующим образом:

$$l_\gamma(y, \hat{t}) = (y - \hat{t})\gamma\mathbb{I}(y > \hat{t}) + (\hat{t} - y)(1 - \gamma)\mathbb{I}(y < \hat{t}), \quad (14.53)$$

¹ Можно также обучать конформные оценки «сквозняком», совместно с предсказательной моделью, как обсуждается в работе [Stu+22].

где y – истинный выход, а \hat{t} – предсказанное значение в квантиле γ . См. иллюстрацию на рис. 14.5(c).

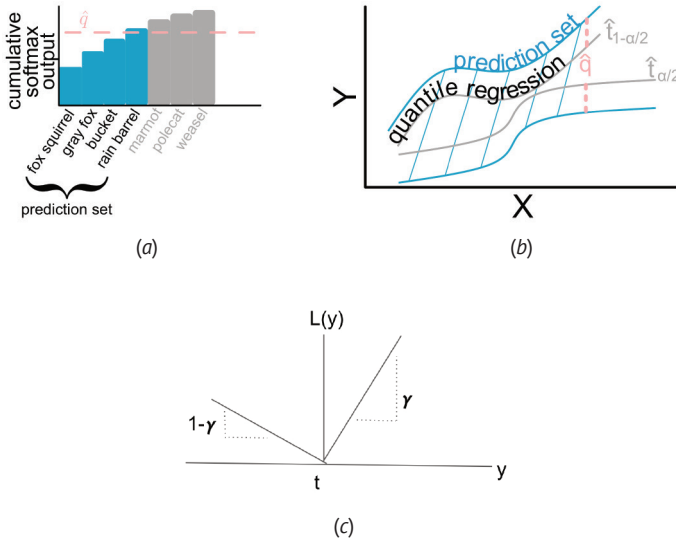


Рис. 14.5. (a) Иллюстрация адаптивного набора предсказаний. На основе рис. 5 из работы [AB21]. Печатается с разрешения Анастасиоса Ангелопулоса; (b) иллюстрация конформализованной квантильной регрессии. На основе рис. 6 из работы [AB21]. Печатается с разрешения Анастасиоса Ангелопулоса; (c) пинбольная функция потерь

Квантили регрессии составляют лишь приблизительно 90%-ный интервал, потому что модель может не соответствовать истинному распределению. Однако мы можем исправить это, воспользовавшись конформным предсказанием. А именно определим конформную оценку:

$$s(\mathbf{x}, y) = \max(\hat{t}_{\alpha/2}(\mathbf{x}) - y, y - \hat{t}_{1-\alpha/2}(\mathbf{x})). \quad (14.54)$$

Иными словами, $s(\mathbf{x}, y)$ – положительная мера того, насколько значение y выходит за пределы интервала предсказания. Мы вычисляем \hat{q} , как и раньше, и определяем интервал конформного предсказания как

$$\mathcal{T}(\mathbf{x}) = [\hat{t}_{\alpha/2}(\mathbf{x}) - \hat{q}, \hat{t}_{1-\alpha/2}(\mathbf{x}) + \hat{q}]. \quad (14.55)$$

Это делает интервал квантильной регрессии шире, если \hat{q} положительно (если базовый метод был чрезмерно уверенным), и уже, если \hat{q} отрицательно (базовый метод был излишне неуверенным). См. иллюстрацию на рис. 14.5(b). Данный подход называется **конформализованной квантильной регрессией** (англ. conformalized quantile regression – **CQR**) [RPC19].

14.3.2.2. Конформализация предсказанных дисперсий

Существует много способов определить оценки неопределенности $u(\mathbf{x})$, например предсказанное стандартное отклонение, из которого можно вывести интервал предсказаний по формуле

$$\mathcal{T}(\mathbf{x}) = [f(\mathbf{x}) - u(\mathbf{x})\hat{q}, f(\mathbf{x}) + u(\mathbf{x})\hat{q}]. \quad (14.56)$$

Здесь \hat{q} выведено из квантилей следующих конформных оценок:

$$s(\mathbf{x}, y) = \frac{|y - f(\mathbf{x})|}{u(\mathbf{x})}. \quad (14.57)$$

Интервал, порождаемый этим методом, обычно оказывается шире вычисленного методом CQR, потому что распространяется на одинаковую величину выше и ниже предсказанного значения $f(\mathbf{x})$. Кроме того, мера неопределенности $u(\mathbf{x})$ может плохо масштабироваться с ростом α . Тем не менее это простой ретроспективный метод, применимый ко многим методам регрессии без необходимости переобучения.