

*Bayesian
Optimization
in Action*

QUAN NGUYEN

FOREWORDS BY LUIS SERRANO AND DAVID SWEET



MANNING
SHELTER ISLAND

Куан Нгуен

БАЙЕСОВСКАЯ ОПТИМИЗАЦИЯ

С ПРИМЕРАМИ
ИЗ БИБЛИОТЕК PYTHON

Астана
«АЛИСТ»
2024

УДК 004.4
ББК 32.973.26-018.1
Н37

Нгуен К.

Н37 Байесовская оптимизация с примерами из библиотек Python: Пер. с англ. — Астана: АЛИСТ, 2024. — 416 с.: ил.

ISBN 978-601-08-4437-7

Книга рассказывает об оптимизации моделей машинного обучения на основе байесовского метода и теории вероятностей. Примеры составлены на языке Python с использованием библиотек PyTorch, GPyTorch и BoTorch. Описаны приемы настройки гиперпараметров, гауссовское распределение, использование политик машинного обучения, в частности на основе задачи о многоруких бандитах, повышение производительности вычислений, многовариантная оптимизация с учетом полезности и затрат, а также применение байесовского метода в специализированных сценариях.

*Для специалистов по машинному обучению
и математическим алгоритмам*

УДК 004.4
ББК 32.973.26-018.1

© 2024 ALIST LLP

Authorized translation of the English edition © 2023 Manning Publications. This translation is published and sold by permission of Manning Publications, the owner of all rights to publish and sell the same.

Авторизованный перевод английской редакции книги © 2023 Manning Publications.
Перевод опубликован и продается с разрешения компании-правообладателя Manning Publications.
Все права защищены.

ISBN 978-1-63343-907-8 (англ.)
ISBN 978-601-08-4437-7 (каз.)

© Manning Publications, 2023
© Издание на русском языке. ТОО "АЛИСТ", 2024

Оглавление

Введение	11
Пролог	13
Благодарности	15
О книге	17
Кому следует прочитать эту книгу?	17
Как организована эта книга: дорожная карта.....	17
О коде.....	20
Дискуссионный форум liveBook.....	21
Архив с иллюстрациями к книге	21
Об авторе	22
О техническом редакторе	22
Об изображении на обложке	23
Глава 1. Введение в байесовскую оптимизацию	24
1.1. Нахождение оптимума дорогостоящей функции «черного ящика»	25
1.1.1. Настройка гиперпараметров как пример дорогостоящей задачи оптимизации «черного ящика».....	25
1.1.2. Проблема дорогостоящей оптимизации «черного ящика»	27
1.1.3. Другие реальные примеры дорогостоящих задач оптимизации «черного ящика»	28
1.2. Введение в байесовскую оптимизацию	30
1.2.1. Моделирование с помощью гауссовского процесса.....	31
1.2.2. Принятие решений с помощью политики БО	34
1.2.3. Объединение ГП и политики БО для формирования цикла оптимизации	35
1.2.4. Байесовская оптимизация в действии.....	37
1.3. Что вы узнаете в этой книге?	43
Заключение	44
ЧАСТЬ I. МОДЕЛИРОВАНИЕ С ПОМОЩЬЮ ГАУССОВСКИХ ПРОЦЕССОВ	45
Глава 2. Гауссовский процесс как распределение по функциям	47
2.1. Как продать свой дом по методу Байеса.....	49
2.2. Моделирование корреляций с помощью многомерных гауссовских распределений и байесовских обновлений	51
2.2.1. Использование многомерных распределений Гаусса для совместного моделирования нескольких переменных.....	51

2.2.2. Обновление нормальных распределений.....	55
2.2.3. Моделирование множества переменных с помощью многомерных гауссовских распределений	58
2.3. Переход от конечной гауссианы к бесконечной.....	60
2.4. Реализация ГП в Python.....	65
2.4.1. Настройка обучающих данных.....	65
2.4.2. Реализация класса ГП	67
2.4.3. Создание прогнозов с помощью ГП.....	69
2.4.4. Визуализация прогнозов ГП	70
2.4.5. Выход за рамки одномерных целевых функций.....	73
2.5. Упражнение.....	75
Заключение.....	77

Глава 3. Настройка ГП с помощью функций среднего значения и ковариации 78

3.1. Важность априорных знаний в байесовских моделях.....	79
3.2. Включение предварительных знаний в ГП.....	82
3.3. Определение поведения целевой функции с помощью функции среднего значения	83
3.3.1. Использование функции нулевого среднего в качестве базовой стратегии	84
3.3.2. Использование постоянной функции с градиентным спуском	86
3.3.3. Использование линейной функции с градиентным спуском.....	90
3.3.4. Использование квадратичной функции с помощью реализации пользовательской функции среднего.....	92
3.4. Определение изменчивости и гладкости с помощью ковариационной функции	95
3.4.1. Настройка масштабов функции ковариации	95
3.4.2. Управление гладкостью с помощью различных ковариационных функций.....	99
3.4.3. Моделирование различных уровней изменчивости с помощью множественных масштабов длины.....	101
3.5. Упражнение.....	104
Заключение.....	106

ЧАСТЬ II. ПРИНЯТИЕ РЕШЕНИЙ С ПОМОЩЬЮ БАЙЕСОВСКОЙ ОПТИМИЗАЦИИ 107

Глава 4. Достижение наилучшего результата с помощью политик, основанных на улучшениях..... 109

4.1. Навигация по пространству поиска в БО.....	110
4.1.1. Цикл БО и политики	110
4.1.2. Баланс между исследованием и эксплуатацией	120
4.2. Поиск улучшений в БО.....	122
4.2.1. Измерение улучшения с помощью ГП.....	123
4.2.2. Вычисление вероятности улучшения	127
4.2.3. Запуск политики PoI	131
4.3. Оптимизация ожидаемого значения улучшения.....	135
4.4. Упражнения.....	139
4.4.1. Упражнение 1: поощрение исследования с помощью PoI.....	139
4.4.2. Упражнение 2: БО для настройки гиперпараметров	140
Заключение.....	142

Глава 5. Исследование пространства поиска с помощью «многорукого бандита»..... 143

5.1. Введение в задачу многорукого бандита	144
5.1.1. Как найти наилучший игровой автомат в казино	144
5.1.2. От MAB к БО	147

5.2. Как быть оптимистом в условиях неопределенности с помощью политики верхней доверительной границы	149
5.2.1. Оптимизм в условиях неопределенности	149
5.2.2. Баланс между исследованием и эксплуатацией	152
5.2.3. Реализация с помощью VoTorch	154
5.3. Интеллектуальная выборка с помощью политики выборки Томпсона	157
5.3.1. Одна выборка для представления неизвестного	157
5.3.2. Реализация с помощью VoTorch	160
5.4. Упражнения	166
5.4.1. Упражнение 1: Настройка плана эксплуатации для UCS	167
5.4.2. Упражнение 2: БО для настройки гиперпараметров	167
Заключение	168

Глава 6. Использование теории информации с политиками на основе энтропии 170

6.1. Измерение знаний с помощью теории информации	171
6.1.1. Измерение неопределенности с помощью энтропии	171
6.1.2. Поиск пульты дистанционного управления с помощью энтропии	174
6.1.3. Бинарный поиск с применением энтропии	177
6.2. Энтропийный поиск в БО	183
6.2.1. Поиск оптимума с помощью теории информации	183
6.2.2. Реализация энтропийного поиска с помощью VoTorch	187
6.3. Упражнения	190
6.3.1. Упражнение 1: Включение предварительных (априорных) знаний в энтропийный поиск	190
6.3.2. Упражнение 2: Байесовская оптимизация для настройки гиперпараметров	192
Заключение	193

ЧАСТЬ III. РАСШИРЕНИЕ БАЙЕСОВСКОЙ ОПТИМИЗАЦИИ НА СПЕЦИАЛИЗИРОВАННЫЕ СЦЕНАРИИ 195

Глава 7. Максимизация производительности за счет оптимизации пакетной обработки 197

7.1. Одновременное выполнение нескольких оценок функций	198
7.1.1. Параллельное использование всех доступных ресурсов	198
7.1.2. Почему нельзя использовать обычные политики БО в пакетном режиме?	201
7.2. Вычисление улучшения и верхней доверительной границы для набора точек	203
7.2.1. Расширение эвристик оптимизации для пакетного режима	203
7.2.2. Реализация пакетных политик Pol, EI и UCS	210
7.3. Упражнение 1: Расширение TS до пакетной оптимизации с помощью повторной выборки	216
7.4. Вычисление ценности пакета точек с помощью теории информации	217
7.4.1. Нахождение наиболее информативного пакета точек с помощью циклического уточнения	217
7.4.2. Реализация пакетного энтропийного поиска с помощью VoTorch	220
7.5. Упражнение 2: Оптимизация конструкции самолета	222
Заключение	225

Глава 8. Удовлетворение особых ограничений с помощью ограниченной оптимизации 227

8.1. Учет ограничений в задаче ограниченной оптимизации	228
8.1.1. Ограничения могут изменить решение задачи оптимизации	228
8.1.2. Структура БО, учитывающая ограничения	231

8.2. Принятие решений с учетом ограничений в БО	232
8.3. Упражнение 1: Ручное вычисление ограниченной EI	238
8.4. Реализация ограниченной EI с помощью BoTorch	239
8.5. Упражнение 2: Ограниченная оптимизация при конструировании самолета	243
Заключение	245

Глава 9. Балансирование между полезностью и затратами с помощью многовариантной оптимизации	246
9.1. Использование приближений с низкой точностью для изучения дорогостоящих явлений	248
9.2. Разноточное моделирование с помощью ГП	251
9.2.1. Форматирование разноточного набора данных	251
9.2.2. Обучение разноточного ГП	255
9.3. Баланс между информацией и затратами в разноточной оптимизации	260
9.3.1. Моделирование затрат на запросы различной точности	260
9.3.2. Оптимизация количества информации для управления оптимизацией	262
9.4. Измерение производительности в разноточной оптимизации	267
9.5. Упражнение 1: Визуализация средней производительности в разноточной оптимизации	272
9.6. Упражнение 2: Разноточная оптимизация с использованием нескольких низкоточных приближений	273
Заключение	275

Глава 10. Обучение на парных сравнениях с помощью оптимизации предпочтений	277
10.1. Оптимизация «черного ящика» с помощью парных сравнений	279
10.2. Формулировка задачи оптимизации предпочтений и форматирование данных парного сравнения	282
10.3. Обучение ГП, основанного на предпочтениях	286
10.4. Оптимизация предпочтений с помощью игры «Царь горы»	290
Заключение	294

Глава 11. Оптимизация нескольких целевых функций одновременно	296
11.1. Балансирование нескольких целей оптимизации с помощью БО	297
11.2. Нахождение границы наиболее оптимальных точек данных	299
11.3. Стремление к улучшению оптимальной границы данных	307
11.4. Упражнение: Многоцелевая оптимизация для конструирования самолета	313
Заключение	314

ЧАСТЬ IV. СПЕЦИАЛЬНЫЕ МОДЕЛИ ГАУССОВСКОГО ПРОЦЕССА

Глава 12. Масштабирование гауссовского процесса для больших наборов данных	317
12.1. Обучение ГП на большом наборе данных	319
12.1.1. Постановка задачи обучения	319
12.1.2. Обучение обычного ГП	322
12.1.3. Проблемы с обучением обычного ГП	324
12.2. Автоматический выбор репрезентативных точек из большого набора данных	327
12.2.1. Минимизация различий между двумя ГП	327
12.2.2. Обучение модели небольшими пакетами	329
12.2.3. Реализация приближенной модели	332

12.3. Достижение лучшей оптимизации с учетом геометрии поверхности потерь	338
12.4. Упражнение.....	344
Заключение	346

Глава 13. Комбинирование гауссовских процессов с нейросетями

13.1. Данные, имеющие структуру	348
13.2. Нахождение сходства в структурированных данных	351
13.2.1. Использование ядра с GPyTorch	352
13.2.2. Работа с изображениями в PyTorch	352
13.2.3. Вычисление ковариации двух изображений	354
13.2.4. Обучение ГП на изображениях	355
13.3. Использование нейросетей для обработки сложных структурированных данных	359
13.3.1. Зачем использовать нейросети для моделирования?	359
13.3.2. Реализация комбинированной модели в GPyTorch	361
Заключение	368

ПРИЛОЖЕНИЕ. Решения для упражнений.....

П.1. Глава 2: Гауссовский процесс как распределение по функциям	369
П.2. Глава 3: Объединение предварительных знаний с функциями среднего и ковариации.....	372
П.3. Глава 4: Достижение наилучшего результата с помощью политик, основанных на улучшениях	374
П.3.1. Упражнение 1: Поощрение исследований с помощью вероятности улучшения.....	375
П.3.2. Упражнение 2: БО для настройки гиперпараметров.....	376
П.4. Глава 5: Исследование пространства поиска с помощью политик «многоорукого бандита»	382
П.4.1. Упражнение 1: Настройка плана исследования для верхней доверительной границы	383
П.4.2. Упражнение 2: БО для настройки гиперпараметров.....	384
П.5. Глава 6: Использование теории информации с политиками, основанными на энтропии	387
П.5.1. Упражнение 1: Включение предварительных знаний в энтропийный поиск.....	388
П.5.2. Упражнение 2: БО для настройки гиперпараметров.....	391
П.6. Глава 7: Увеличение пропускной способности с помощью пакетной оптимизации	392
П.6.1. Упражнение 1: Расширение TS для пакетного режима с помощью повторной выборки.....	392
П.6.2. Упражнение 2: Оптимизация конструкции самолета	393
П.7. Глава 8: Удовлетворение дополнительных ограничений с помощью ограниченной оптимизации.....	397
П.7.1. Упражнение 1: Ручное вычисление ограниченной политики EI	397
П.7.2. Упражнение 2: Ограниченная оптимизация конструкции самолета	400
П.8. Глава 9: Баланс между полезностью и затратами с помощью разноточной оптимизации	402
П.8.1. Упражнение 1: Визуализация средней производительности при разноточной оптимизации.....	403
П.8.2. Упражнение 2: Разноточная оптимизация с использованием нескольких низкоточных приближений.....	405
П.9. Глава 11: Оптимизация нескольких целевых функций одновременно	407
П.10. Глава 12: Масштабирование гауссовских процессов на большие наборы данных	409

Предметный указатель

Моему лучшему другу, Ньюнгу

Поскольку сложность задач, с которыми мы сталкиваемся в *машинном обучении* (**Machine Learning, ML**) и смежных областях, продолжает расти, все более важным становится умение оптимизировать использование ресурсов и способность эффективно принимать обоснованные решения. *Байесовская оптимизация* — мощный метод, помогающий находить максимумы и минимумы целевых функций, оценка которых требует больших затрат. Это очень полезное решение данной проблемы, поскольку любую функцию можно рассматривать как «черный ящик», что позволяет исследователям и практикам решать очень сложные функции, используя *байесовский вывод* в качестве основного метода оптимизации.

Из-за своей сложности байесовская оптимизация является менее доступной для начинающих специалистов в машинном обучении, чем другие методы. Однако такой инструмент должен присутствовать в наборе каждого, кто хочет достичь наилучших результатов. Для освоения этой темы нужно иметь очень хорошее представление о математических вычислениях и теории вероятностей.

Здесь данная книга и приходит на помощь. В ней Кван красиво и успешно демистифицирует эти сложные концепции. Практический подход, четкие диаграммы, примеры из реальной жизни и полезные листинги кода помогают автору приоткрыть завесу над этой темой, как с теоретической, так и с практической точки зрения.

Автор использует свой обширный опыт специалиста по данным (Data Science) и преподавателя с целью дать читателю очень четкое представление об этих методах и о том, как их можно применять для решения реальных проблем. Начиная с принципов байесовского вывода, в книге постепенно развиваются концепции байесовской оптимизации и модели *гауссовского процесса*. Кван рассказывает о современных библиотеках, таких как GPyTorch и BoTorch, исследуя их использование сразу в нескольких областях.

Эта книга является обязательной к прочтению для любого специалиста по data science или ML, который хочет использовать истинную мощь байесовской оптимизации для решения реальных задач. Я настоятельно рекомендую ее всем, кто хочет овладеть искусством оптимизации с помощью байесовского вывода.

— Луис Серрано,
доктор философии,
ученый и популяризатор искусственного интеллекта,
автор книги «*Grokking Machine Learning*»

Инженеры и ученые сталкиваются с общим вызовом, который важен для закрепления ценности их исследований и творчества. Им нужна оптимизация. Инженеры по машинному обучению работают с *гиперпараметрами*, которые делают модели генерализованными (обобщенными). Физики настраивают лазеры на свободных электронах на максимальную энергию импульса. Программисты настраивают сборщики мусора JVM (Java Virtual Machine) с целью максимизировать пропускную способность сервера. Материаловеды выбирают морфологию микроструктуры, которая максимально увеличивает светопоглощение солнечной батареи. В каждом примере есть проектные решения, которые невозможно сделать на основе лишь базовых принципов. Они выполнены с помощью экспериментальных измерений.

Чтобы оценить что-то экспериментально, нужно создать программное обеспечение, запустить оборудование или построить новый объект, одновременно измеряя его производительность. Чтобы найти хороший дизайн, нужно провести его оценку. Такие вычисления требуют времени, стоят денег и могут быть сопряжены с риском. Поэтому крайне важно, чтобы их количество, необходимое для поиска оптимальной конструкции, было минимальным. В этом и есть вся суть байесовской оптимизации.

В своей работе на протяжении последних 20 лет я использовал данный подход и связанные с ним методы-предшественники. За это время академические исследования и отчеты о промышленном применении улучшили производительность и расширили возможности применения байесовской оптимизации. В настоящее время существуют высококачественные программные инструменты и методы, которые используются при создании оптимизаторов для конкретных проектов.

Текущее состояние можно рассматривать как аналог прогнозирования с помощью линейных моделей. Инженеры, желающие построить линейную модель, могут обнаружить, что такие программные инструменты, как *sklearn*, позволяют им разрабатывать модели для разных типов (например, непрерывных или категориальных) и различного количества входных и выходных переменных, выполнять автоматический их выбор и измерять качество генерализации. Точно так же инженеры, желающие создать байесовский оптимизатор, могут убедиться, что *BoTorch* (построенный на *GPuTorch*, *Pyro* и *PyTorch*) предоставляет инструменты, которые способны оптимизировать различные типы переменных, максимизировать несколько целей, обрабатывать ограничения и многое другое.

В этой книге изучение байесовской оптимизации начинается с ее самых основных компонентов (*регрессия гауссовского процесса* и *численная оптимизация функции сбора оценки*) и заканчивается новейшими методами по обработке большого количества оценок (наблюдений) и экзотических пространств проектирования. Попутно она охватывает все специализации, которые могут вам понадобиться в определенном проекте: обработка ограничений, множественные цели, параллельная оценка, а также оценка через парные сравнения. Вы найдете много технической информации, чтобы освоить инструменты и методы, а также увидите много примеров реального кода, которые можно использовать в работе.

Несмотря на весь успех байесовской оптимизации, существует мало литературы, ориентированной на новичков. Эта книга прекрасно заполняет данную нишу.

Осенью 2019 года я был аспирантом на первом курсе и не знал, над какой проблемой работать в своем исследовании. Я знал, что хочу сосредоточиться на искусственном интеллекте (ИИ). Было что-то привлекательное в автоматизации мыслительных процессов с помощью компьютеров. Но это огромная область, и мне было трудно сузить свою работу до конкретной темы.

Вся неопределённость исчезла, когда я прошёл курс «*Байесовские методы машинного обучения*». К этому моменту я уже несколько раз встречался с теоремой Байеса, когда учился на бакалавриате, но именно на первых лекциях этого курса все стало получаться! Теорема Байеса может предложить интуитивный способ поразмышлять о вероятностях, и, как по мне, это элегантная модель человеческих убеждений. У каждого из нас есть *априорное убеждение* (изначальное мнение о чем-либо), с которого мы начинаем. Оно обновляется по мере того, как мы встречаем доказательства за или против него. Результатом является *апостериорное убеждение*, отражающее как априорное предположение (изначальное), так и данные. Тот факт, что теорема Байеса дает элегантный способ поддерживать убеждения в ИИ и находит применение во многих задачах, стал для меня мощным сигналом, что байесовское машинное обучение — это тема, которой стоит заниматься.

К тому времени, когда мы добрались до лекции по байесовской оптимизации (БО), решение было уже принято: теория интуитивно понятна, приложений много, возможностей для построения еще больше. Опять же, что-то врожденное привлекло (и продолжает привлекать) меня к автоматизации мышления, а если точнее, к принятию решений. БО стала идеальным аттракционом. Я попал в исследовательскую лабораторию Романа Гарнетта (Roman Garnett), профессора, ведущего курс. И мое путешествие по БО началось!

Перенесемся в 2021 год. Я потратил некоторое время на исследование и внедрение решений этого метода, и мое понимание БО только выросло. Я бы порекомендовал его друзьям и коллегам при решении сложных задач оптимизации с полной уверенностью, что он будет работать хорошо. Была только одна проблема: отсутствовал хороший ресурс, на который я мог бы указать. Исследовательские работы изобиловали математикой, онлайн-уроки были слишком короткими, чтобы дать содержательную информацию, а учебные пособия по программному обеспечению БО были разрозненными и не содержали хорошего описания.

Тогда мне в голову пришла идея в виде цитаты Тони Моррисона (Toni Morrison): «*Если есть книга, которую вы хотите прочитать, но она еще не написана, тогда*

вы должны ее написать». Как верно сказано! Эта перспектива взволновала меня по двум причинам. Во-первых, я мог бы написать книгу о чем-то близком и дорогим моему сердцу. Во-вторых, писательство, несомненно, помогло бы мне достичь еще более глубокого понимания. Я подготовил предложение и связался с Manning (издателем моих любимых книг) в том стиле, который я себе представлял.

В ноябре 2021 года мой редактор по сбору данных Энди Уолдрон (Andy Waldron) прислал мне электронное письмо, ознаменовав этим самое первое сообщение от Manning. В декабре 2021 года я подписал контракт и начал писать, что, как позже выяснилось, потребовало больше времени, чем я изначально предполагал (я уверен, что это касается любой книги). В апреле 2023 года я написал это предисловие как один из последних этапов перед публикацией!

Благодарности

Чтобы воспитать ребенка и написать книгу, нужна целая деревня. Ниже представлена лишь небольшая часть ее жителей, которые мне очень помогли в процессе работы.

Моя первая и главная благодарность моим родителям Бангу (Bang) и Лан (Lan), чья постоянная поддержка позволила мне бесстрашно исследовать неизведанное: отучиться за границей, получить докторскую степень и, конечно же, написать книгу. И я хотел бы искренне поблагодарить мою сестру и доверенное лицо – Ню (Nhu), которая всегда находилась рядом, чтобы помочь мне в самые трудные времена.

Байесовская оптимизация составляет значительную часть моих докторских исследований, и я хотел бы поблагодарить участников программы, которые действительно сделали мой опыт бесценным. Особая благодарность моему консультанту, Роману Гарнетту, который без особых усилий убедил меня продолжить исследования в области байесовского машинного обучения. Ты тот, кто все это начал. Я также благодарю своих друзей из лаборатории активного обучения: Еу Чен (Yehu Chen), Шаян Монаджеми (Shayan Monadjemi) и профессора Алвитта Оттли (Alvitta Ottley). Награда за докторскую степень не очень велика, и работа с вами составляет большую ее часть.

Также я выражаю благодарность замечательной команде издательства Manning. Я благодарю моего редактора по разработке Марину Майклс (Marina Michaels), которая руководила этим проектом с самого первого дня, проявляя высочайший уровень профессионализма, заботы, поддержки и терпения. Мне очень повезло, что я встретил вас в нашем проекте. Спасибо моему редактору по сбору данных Энди Уолдрону за веру в эту идею, несмотря на то, что над подобной темой уже работал другой автор (который лучше меня), а также Ивану Мартиновичу (Ivan Martinovic) за помощь с вопросами по AsciiDoc и терпеливое исправление моего кода.

Я хотел бы также поблагодарить следующих рецензентов: Аллан Макура (Allan Makura), Андрей Палейес (Andrei Paleyes), Карлос Айя-Морено (Carlos Ayala-Moreno), Клаудиу Шиллер (Claudiu Schiller), Козимо Аттанаси (Cosimo Attanasi), Денис Лапчев (Denis Lapchev), Гэри Бэйк (Gary Bake), Джордж Онофрей (George Onofrei), Говард Бэнди (Howard Bandy), Иоаннис Атсониос (Ioannis Atsonios), Хесус Антонино Хуарес Герреро (Jesús Antonino Juárez Guerrero), Джош МакАдамс (Josh McAdams), Квеку Реджинальд Уэйд (Kweku Reginald Wade), Кайл Питерсон (Kyle Peterson), Локеш Кумар (Lokesh Kumar), Люциан Мирча Сасу (Lucian Mircea Sasu), Марк-Энтони Тейлор (Marc-Anthony Taylor), Марсио Николау (Marcio

Nicolau), Макс Део (Max Dehaut), Максим Волгин (Maxim Volgin), Мишель Ди Педе (Michele Di Pede), Мирерфан Геиби (Mirerfan Gheibi), Ник Декроос (Nick Decroos), Ник Васкес (Nick Vazquez), Ор Голан (Or Golan), Питер Хенсток (Peter Henstock), Филип Вайсс (Philip Weiss), Рави Киран Бамиди (Ravi Kiran Bamidi), Ричард Тобиас (Richard Tobias), Рохит Госвами (Rohit Goswami), Серджио Говони (Sergio Govoni), Шаби Икбал (Shabie Iqbal), Шриша Джагадиш (Shreesha Jagadeesh), Симон Стуацца (Simone Sguazza), Шрирам Мачарла (Sriram Macharla), Шимон Харабаз (Szymon Harabasz), Томас Форис (Thomas Forys) и Влад Навицкий (Vlad Navitski).

При написании книги неизбежно возникают «слепые пятна». Именно рецензенты помогают их заполнить и сосредоточить внимание автора на том, что действительно важно. Выражаю особую благодарность Керри Койчу (Kerry Koitzsch) за его содержательные отзывы и Джеймсу Байлеки (James Byleckie) за его отличные предложения, как по коду, так и по его написанию.

Наконец, я благодарю команды, создавшие невероятные инструменты GPU Torch и BoTorch. Я пробовал и другие различные библиотеки для гауссовских процессов (ГП) и БО, но всегда возвращался к ним. Я надеюсь, что эта книга сможет создать и без того замечательное сообщество вокруг этих инструментов.

Раньше узнать о байесовской оптимизации (БО) можно было из онлайн-статей и туториалов к определенной библиотеке. Они были разрознены и по своей природе не углублялись в специфику. И их поиск отнимал время и силы. Вы также могли обратиться к техническим учебникам, но они, как правило, были слишком объемными и содержали много математики. Это представляло проблему, если вы практикующий специалист, который хотел бы сразу же приступить к делу.

Эта книга заполняет данный пробел, предлагая сочетание практик, ссылок на более глубокие материалы для заинтересованного читателя и готовых к использованию примеров кода. Она работает, сначала создавая интуитивно понятное представление о компонентах БО, а затем реализуя их на Python с использованием современного программного обеспечения.

Цель книги — предоставить доступное введение в байесовскую оптимизацию, основанное на глубоком понимании математики и теории вероятностей. Кроме того, заинтересованный читатель может найти дополнительные технические тексты, на которые в книге имеются ссылки, для более глубокого погружения в интересующую тему.

Кому следует прочитать эту книгу?

Эта книга принесет пользу специалистам по data science и ML, которые интересуются настройкой гиперпараметров, A/B-тестированием или экспериментируют с принятием решений (в общем смысле).

Она также будет полезна исследователям в таких научных областях, как химия, материаловедение и физика, которые сталкиваются со сложными задачами по оптимизации. Несмотря на то что в книге будет охвачена большая часть базовых знаний, необходимых для понимания содержания, аудитория должна быть знакома с такими общими концепциями машинного обучения, как *обучающие данные*, *прогностические модели*, *многомерные нормальные распределения* и прочие.

Как организована эта книга: дорожная карта

Книга разделена на четыре части. Каждая из них содержит несколько глав, охватывающих соответствующую тему:

- В главе 1 представлена байесовская оптимизация с использованием реальных сценариев применения. Она также включает (не вдаваясь в технические детали) наглядный пример, как БО может ускорить поиск глобального оптимума дорогостоящей функции.

Первая часть рассматривает гауссовские процессы как прогностическую модель функции, которую необходимо оптимизировать. Центральный тезис заключается в том, что гауссовские процессы предлагают калиброванную количественную оценку неопределенности, что важно в рамках нашей байесовской оптимизации. Эта часть состоит из двух глав:

- Глава 2 показывает, что гауссовские процессы являются естественным решением задачи обучения регрессионной модели на основе некоторых наблюдаемых данных. Такой процесс определяет распределение по функциям и может быть обновлен, чтобы отразить наше убеждение о значении функции, учитывая некоторые наблюдаемые данные.
- Глава 3 содержит два основных способа включить априорную информацию в гауссовский процесс: *функция нахождения среднего значения* и *функция ковариации*. Первая определяет общую тенденцию, а вторая показывает гладкость функции.

Во второй части перечислены политики байесовской оптимизации, которые представляют собой процедуры принятия решений, как следует выполнять оценки функций, чтобы глобальный оптимум можно было определить наиболее эффективно. Несмотря на то что разные политики мотивированы разными задачами, все они направлены на баланс между *исследованием* и *эксплуатацией*. Эта часть состоит из трех глав:

- В главе 4 обсуждается естественный способ принятия решений, какую оценку функции наиболее выгодно выполнить, учитывая улучшение, которое можно было бы получить от текущего наилучшего значения функции. Благодаря предположению об этой функции, основанному на гауссовском процессе, мы можем вычислить связанные с улучшением величины в закрытой форме и с минимальными затратами, что позволяет использовать две конкретные политики БО: *вероятность улучшения (Probability of Improvement, PoI)* и *ожидаемое улучшение (Expected Improvement, EI)*.
- В главе 5 исследуется связь между байесовской оптимизацией и другим распространенным классом задач, которое называется «*Многорукый бандит*» (**Multi-Armed Bandit, MAB**). Мы научимся переносить политики многорукого бандита в настройку байесовской оптимизации и получим соответствующие стратегии: *верхнюю доверительную границу (Upper Confidence Bound, UCB)* и *выборку Томпсона (Thompson Sampling, TS)*.
- В главе 6 рассматривается стратегия, которая уменьшает наибольшую неопределенность в наших убеждениях относительно глобального оптимума функции. Она представляет собой политику, основанную на энтропии, которая использует область математики, называемую *теорией информации*.

В третьей части представлены некоторые наиболее распространенные варианты использования, которые не совсем вписываются в рабочий процесс, разобранный к этому моменту в книге. А также показано, как можно модифицировать БО для решения различных задач оптимизации:

- В главе 7 представлена пакетная оптимизация, при которой для увеличения пропускной способности мы позволяем экспериментам выполняться параллельно. Например, можно обучать несколько экземпляров большой нейросети одновременно на кластере графических процессоров. Для этого политика оптимизации должна возвращать более одной рекомендации одновременно.
- В главе 8 обсуждаются критически важные для безопасности случаи использования, когда мы не можем свободно исследовать пространство поиска, поскольку некоторые оценки функций могут иметь пагубные последствия. Это приводит к сценарию, в котором существуют ограничения на то, как должна вести себя рассматриваемая функция, а также к потребности учитывать данные ограничения при разработке политики оптимизации.
- Глава 9 показывает, что при наличии доступа к множеству способов наблюдать за значениями функции на разных уровнях стоимости и точности (широко известному как *разноточность байесовской оптимизации*), учет переменных затрат может привести к повышению производительности оптимизации.
- В главе 10 рассматриваются парные сравнения, которые более точно отражают различные предпочтения человека, чем числовые оценки или рейтинги, поскольку они являются более простыми для понимания и создают меньшую когнитивную нагрузку на маркировщика. Эта глава расширяет представление о БО сначала с помощью специальной модели гауссовского процесса, а затем через использование существующих политик таким образом, чтобы они вписывались в рабочий процесс парного сравнения.
- В главе 11 рассматривается проблема многоцелевой оптимизации, где мы стремимся оптимизировать сразу несколько целевых функций, а также как БО может быть распространена на подобные задачи.

Четвертая часть посвящена специальным вариантам моделей гауссовских процессов, демонстрируя их гибкость и эффективность при моделировании и обеспечении прогнозов с калибровкой по неопределенности, даже вне контекста БО:

- В главе 12 мы узнаем, что в некоторых случаях получить решения обученного гауссовского процесса в замкнутой форме невозможно. Тем не менее высокоточные приближения все же могут быть получены с использованием сложных стратегий аппроксимации.
- Глава 13 демонстрирует, что благодаря экосистеме Torch объединение нейросетей PyTorch с гауссовскими процессами GPuTorch является простым процессом. Это позволяет нашим моделям стать более гибкими и выразительными.

Первые шесть глав принесут новичку большую пользу. Опытные специалисты, желающие внедрить байесовскую оптимизацию в свой вариант использования, могут найти ценность в главах с 7 по 11, которые можно читать независимо друг от друга

и в любом порядке. Давним пользователям гауссовских процессов, скорее всего, будут интересны последние две главы, в которых мы разрабатываем специализированные гауссовские процессы.

О коде

Исполняемые фрагменты кода находятся в онлайн-версии этой книги в liveBook по адресу: <https://livebook.manning.com/book/bayesian-optimization-in-action/>. Код доступен для загрузки с веб-сайта издательства (<https://www.manning.com/books/bayesian-optimization-in-action/>) и GitHub (<https://github.com/KrisNguyen135/bayesian-optimization-in-action/>). Последний будет включать ошибки и *пул-реквесты* (**Pull request**).

Для запуска кода вам понадобится **Jupyter Notebook**. Он предлагает удобный способ динамической работы с кодом, позволяя исследовать, как каждый объект ведет себя и взаимодействует с другими объектами. Дополнительную информацию, как начать работу с Jupyter Notebook, можно найти на их официальном сайте. Возможность динамически исследовать объекты особенно полезна в нашем случае, поскольку многие компоненты в рабочем процессе БО реализованы в виде объектов Python с помощью GPyTorch и BoTorch (основные библиотеки, которые мы будем использовать).

GPyTorch и BoTorch — лучшие инструменты для гауссовского моделирования процессов и байесовской оптимизации в Python. Существуют и другие варианты, такие как расширение scikit-optimize для библиотеки scikit-Learn или GPflow и GPflowOpt для фреймворка TensorFlow. Однако комбинация GPyTorch и BoTorch образует наиболее полную и гибкую кодовую базу, включающую множество современных алгоритмов, которые получены в результате исследований. На собственном опыте использования ПО для байесовской оптимизации я обнаружил, что GPyTorch и BoTorch обеспечивают хороший баланс между удобством для новичков и предоставлением самых современных методов.

Необходимо отметить еще одну вещь. Поскольку эти библиотеки имеют активную поддержку, API, показанные в книге, могут быть немного изменены в более новых версиях. Поэтому важно установить именно версии библиотек, указанные в файле `require.txt`, чтобы код запускался без ошибок. Дополнительные инструкции, как создавать среды Python, используя файл `require.txt`, можно найти, например, в официальной документации языка. При этом для работы с более новыми версиями вам, скорее всего, потребуется лишь внести незначительные изменения в код.

Читая книгу, вы также можете заметить, что текст имеет тенденцию сосредотачиваться только на ключевых компонентах кода, упуская из виду многие детали, такие как импорт библиотек и бухгалтерский учет. Разумеется, при первом появлении фрагмента они будут представлены в тексте корректно. Но именно краткость наших рассуждений поможет сосредоточиться на том, что нового привносится в каждой главе, и избежать необходимости повторяться. С другой стороны, код в Jupyter Notebook является автономным, и каждый фрагмент можно запускать отдельно, без каких-либо изменений.

Дискуссионный форум liveBook

Покупка книги включает бесплатный доступ к liveBook — онлайн-платформе для чтения от издательства Manning. Используя эксклюзивные функции, вы можете добавлять комментарии ко всей книге или к определенным разделам и абзацам. Здесь также легко можно оставлять для себя заметки, задавать технические вопросы и отвечать на них, а также получать помощь от автора и других пользователей. Вы также можете узнать больше о форумах Manning и правилах поведения на их странице обсуждения.

Manning предоставляет читателям место для обсуждения между отдельными читателями, а также между читателями и автором. Это совсем не обязывает к какому-то определенному объему участия со стороны последнего, чей вклад в форум остается добровольным (и неоплачиваемым). Предлагаем вам самим попробовать задать автору несколько интересных вопросов, чтобы его интерес не угас! Форум и архив предыдущих обсуждений будут доступны на сайте издательства до тех пор, пока книга находится в печати.

Архив с иллюстрациями к книге

Для полноценного изучения материала иллюстрации этой книги наиболее наглядны в цветном, а не черно-белом исполнении. На страницах данного печатного издания иллюстрации монохромные, а архив с их цветными версиями выложен на сайте издательства. Его можно скачать по ссылке

https://files.alistbooks.kz/3015_Nguyen_Python.zip



Кван Нгуен — Python-программист и энтузиаст машинного обучения. Он интересуется задачами о принятии решений, связанных с неопределенностью. Кван является автором нескольких книг по программированию на Python и научным вычислениям. В настоящее время он работает над степенью доктора компьютерных наук в Вашингтонском университете в Сент-Луисе, где занимается исследованием байесовских методов машинного обучения.

О техническом редакторе

Технический редактор этой книги — Керри Койч. Он является автором и архитектором ПО с более чем тридцатилетним опытом по внедрению корпоративных приложений и решений информационной архитектуры. Керри — автор книги по распределенной обработке, а также множества более коротких технических публикаций и обладатель патента на инновационную технологию *оптического распознавания символов* (**Optical Character Recognition, ORC**). Он также является лауреатом медали Армии США «За достижения».

Об изображении на обложке

Рисунок на обложке книги подписан *Polonnois (лат.)* или «Поляк». Он взят из сборника Жака Грассе де Сен-Совера, опубликованного в 1797 году. Каждая иллюстрация прекрасно нарисована и раскрашена вручную.

В те времена просто по одежде было легко определить, где живут люди и какое они имеют занятие или общественное положение. Manning отмечает изобретательность и инициативность компьютерного бизнеса, создавая обложки книг, основанные на богатом разнообразии региональной культуры много веков назад, возвращая их к жизни с помощью фотографий из подобных коллекций.

1

Введение в байесовскую оптимизацию

Темы главы:

- Что мотивирует использовать байесовскую оптимизацию и как она работает?
- Реальные примеры задач с применением байесовской оптимизации.
- «Игрушечный» пример байесовской оптимизации в действии.

В глобальном смысле байесовская оптимизация (БО) — это метод оптимизации, применяемый к функции (или любому процессу, генерирующему выходные данные), которая представляет собой «черный ящик», и ее оценка требует больших затрат с точки зрения времени, денег или других ресурсов. Данный подход содержит множество важных задач, включая настройку гиперпараметров, которые будут затронуты чуть позже. Использование БО может ускорить процедуру поиска и помочь найти оптимум функции как можно быстрее.

Несмотря на то что БО вызывает интерес со стороны ML-сообщества, на практике она используется не так часто, как другие методы, связанные с машинным обучением. Но почему так происходит? Некоторые находят ответ в том, что байесовская оптимизация сложна в изучении: нужно разбираться в вычислениях и теории вероятностей, а также иметь некий опыт в ML, позволяющий использовать метод в приложениях. Цель книги — развеять представление, что БО сложно использовать, а также показать, что она более интуитивна и доступна, чем можно было бы подумать.

На протяжении всей книги вы будете встречать множество иллюстраций, сюжетов и кода, которые призваны сделать эту тему более простой и конкретной. Вы узнаете, как работает каждый компонент байесовской оптимизации на высоком уровне и как реализовать их с помощью современных библиотек Python. Кроме того, примеры помогут вам приступить к реализации собственных проектов, поскольку основа подхода является универсальной и работает по принципу «подключи и играй». В этом отношении также полезны упражнения.

В общем, я надеюсь, что книга окажется полезной для ваших потребностей в машинном обучении и в целом будет интересна для чтения. Прежде чем углубиться в содержание, уделим некоторое время описанию проблемы, которую БО призвана решить.

1.1. Нахождение оптимума дорогостоящей функции «черного ящика»

Настройка гиперпараметров в машинном обучении является одним из наиболее распространенных применений байесовской оптимизации. В этом разделе будет исследована данная задача (а также несколько других) как пример общей задачи оптимизации «черного ящика». Это поможет понять, зачем нужна БО.

1.1.1. Настройка гиперпараметров как пример дорогостоящей задачи оптимизации «черного ящика»

Допустим, вы хотите обучить нейросеть на большом наборе данных, но не уверены, сколько слоев она должна иметь. Ее архитектура является решающим фактором при *глубоком обучении* (**Deep Learning, DL**), поэтому необходимо провести некоторое первоначальное тестирование (результаты представлены в табл. 1.1).

Таблица 1.1. Пример задачи по настройке гиперпараметров

Число слоев	Точность на тестовом наборе
5	0.72
10	0.81
20	0.75

Наша задача — определить, сколько слоев должна иметь нейросеть, чтобы быть наиболее точной. Есть некоторая сложность при выборе числа слоев, которое следует попробовать. Наилучшая обнаруженная точность согласно таблице — 81%. Это хороший результат, но есть возможность добиться большего с другим количеством. Предположим, руководство установило крайний срок, когда нужно завершить внедрение модели. Поскольку обучение нейросети занимает несколько дней, в нашем распоряжении есть некоторое конечное число испытаний, прежде чем нужно будет решить, сколько слоев должна иметь сеть. Имея это в виду, нам необходимо знать, какие еще значения следует попробовать следующими, чтобы найти количество слоев, обеспечивающее максимально возможную точность.

Подобная задача, где нужно найти наилучшие настройки (значения гиперпараметров) модели с целью оптимизировать показатели производительности (например, точность прогнозирования), обычно называется *настройкой гиперпараметров* (**Hyperparameter tuning**). В данном примере гиперпараметром нейросети является ее *глубина* (количество слоев). Если мы работаем с деревом решений, общими

гиперпараметрами будут максимальная глубина, минимальное количество точек на узел и критерий разделения. Используя *машину опорных векторов (Support Vector Machine, SVM)*, можно было бы настроить *регуляризацию* и *ядро*. Поскольку производительность модели во многом зависит от ее гиперпараметров, их настройка является важным компонентом любого ML-конвейера.

Если бы это был типичный набор реальных данных, процесс мог бы занять много времени и ресурсов. Рис. 1.1 от OpenAI показывает, что по мере того, как нейросети становятся больше и глубже, объем необходимых вычислений (измеряется в петафлопс-днях) увеличивается экспоненциально.

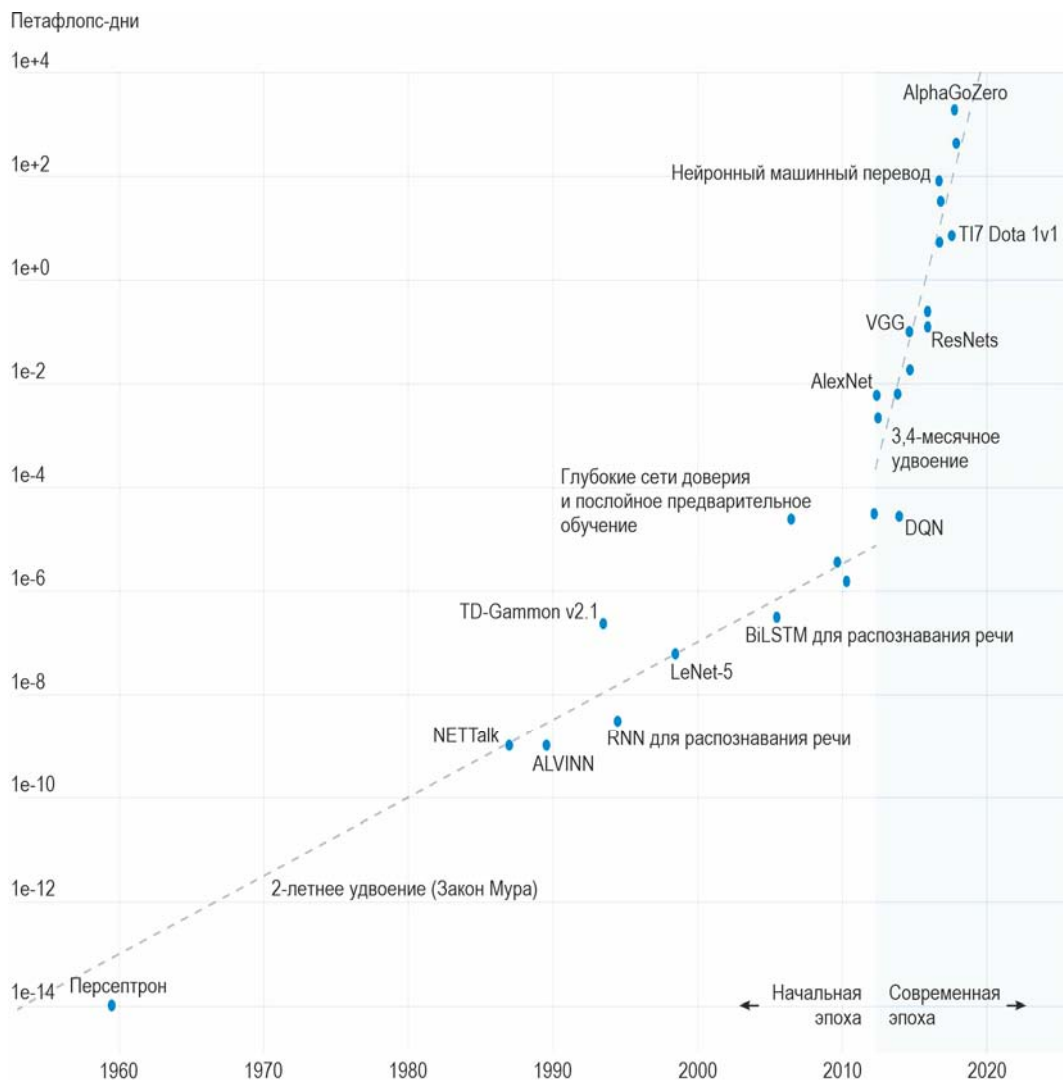


Рис. 1.1. График, демонстрирующий затруднения при настройке гиперпараметров из-за неуклонного роста вычислительных затрат на обучение больших нейросетей