The Labeled Multi-Bernoulli Filter

The Labeled Multi-Bernoulli Filter:

A Comprehensive Guide for Practitioners

By

Reza Hoseinnezhad

Cambridge Scholars Publishing

The Labeled Multi-Bernoulli Filter:
A Comprehensive Guide for Practitioners

By Reza Hoseinnezhad

This book first published 2024

Cambridge Scholars Publishing

Lady Stephenson Library, Newcastle upon Tyne, NE6 2PA, UK

British Library Cataloguing in Publication Data
A catalogue record for this book is available from the British Library

ISBN: 978-1-0364-1182-4
ISBN (Ebook): 978-1-0364-1183-1

MATLAB source codes of the LMB filter and multisensor fusion methods using LMB filter are provided in:
https://github.com/reza-hoseinnezhad/LMB-book

To my beloved daughter, Helia, who illuminates my world with the brightest of lights.

This book is an essential read for anyone interested in modern tracking technology. This book demystifies the complex world of multi-object tracking by focusing on the labeled multi-Bernoulli filter, a cutting-edge tool used in areas like autonomous vehicles, robotics, and surveillance.

Written with clarity, the guide breaks down technical concepts into easy-to-understand language, making it accessible even to those with limited technical background. It delves into how this filter can accurately track multiple objects simultaneously, a key ability in our increasingly automated world.

The book not only covers the theoretical aspects but also offers complete coding in MATLAB with examples, demonstrating real-world applications. Whether you're a professional in the field, a student, or simply a tech enthusiast, this comprehensive guide provides a clear understanding of this advanced technology.

Contents

List of Figures

1. Introduction

1.1 Background

The Labeled Multi-Bernoulli (LMB) filter has witnessed significant advancements and found applications in various multi-object tracking scenarios. The LMB filter was first introduced by Reuter et al. [1] after which it emerged as a popular approach for multi-object tracking, providing effective solutions for various tracking challenges.

Numerous works during recent years have addressed challenges such as sensor fusion, occlusion, glint noise, distributed tracking, unknown sensor characteristics, and specific application domains, thereby expanding the capabilities and improving the performance of LMB-based tracking systems. These developments pave the way for more accurate and robust multi-object tracking in diverse real-world scenarios.

In the following sections, a number of recent works are presented, with the purpose of motivating the reader and providing exposure to various extensions and applications of the filter. Dong et al. [2] developed an LMB filter tailored for multi-target tracking in the presence of glint noise. Their method effectively handled glint noise, improving tracking performance for objects affected by glint phenomena.

1.1.1 Variations of LMB filter

Saucan et al. [3] proposed the Merged-Measurement LMB (MM-LMB) filter, a variation of the LMB filter designed to handle merged measurements. Their method effectively addressed situations where measurements from different sensors or object detections overlap or merge, improving tracking accuracy and robustness. Li et al. [4] focused on real-time tracking of multiple extended targets using LMB filters within the box-particle framework. Their work combined the strengths of LMB filters and box-particle filtering techniques to achieve real-time tracking performance for extended targets. Olofsson et al. [5] employed LMB filters for drift ice positioning and tracking in polar region operations. Their work showcased the effectiveness of LMB filters in challenging environmental conditions, providing accurate tracking for drift ice objects in polar regions.

Danzer et al. [6] introduced the Adaptive Labeled Multi-Bernoulli filter, which aimed to enhance target tracking precision. Their adaptive approach improved the estimation accuracy of target states by dynamically adjusting the filter's parameters based on the tracking scenario. Herrmann et al. [7] described an extension to the LMB filter for tracking multiple object reference points in urban environments. Their work focused on scenarios where multiple reference points needed to be simultaneously tracked, enabling more comprehensive tracking in complex urban settings. Rathnayake et al. [8] proposed a Track-Before-Detect (TBD) LMB filter for industrial mobile platform safety applications. Their method addressed safety concerns by incorporating track initiation before object detection, enabling proactive safety measures in industrial environments. Dai et al. [9] developed an improved LMB filter capable of handling intertarget occlusion in Multiextended Target Tracking (METT). Their method effectively addressed the challenges posed by occluded objects, enabling accurate tracking even in situations with partial or complete occlusion.

Papi et al. [10] proposed a solution for multi-target tracking with superpositional measurements using LMB filters and Vo-Vo densities. Their work addressed scenarios with overlapping or superimposed measurements, enabling accurate tracking in complex measurement environments. Qiu et al. [11] introduced the Multiple Model LMB (MM-LMB) filter, which combined the strengths of multiple models for maneuvering target tracking. Their method improved tracking accuracy by adapting to target maneuvers using a set of model hypotheses. Liu et al. [12] proposed a computationally efficient LMB smoother for multi-target tracking. Their method improved tracking accuracy by incorporating a smoother component, enabling more accurate estimation of target trajectories. Reuter et al. [13] presented a fast implementation of the LMB filter based on joint prediction and update schemes. Their work focused on improving computational efficiency without compromising tracking accuracy, enabling real-time tracking capabilities.

Hao et al. [14] introduced an adaptive multi-target tracking algorithm based on LMB filters and variational Bayesian (VB) approximation. Their method enhanced tracking performance by adapting to changing tracking scenarios using variational Bayesian techniques. Wang et al. [15] presented a variational Bayesian LMB (VB-LMB) filter for multi-target tracking under unknown sensor characteristics. Their method incorporated variational Bayesian techniques into the LMB filter framework, enabling robust tracking even in the presence of uncertain sensor characteristics. Wang et al. [15] presented an LMB maneuvering target tracking algorithm using a TSK iterative regression multiple model. Their method employed a TSK iterative regression multiple model to enhance the tracking performance for maneuvering targets, enabling accurate tracking in dynamic scenarios.

Yu et al. [16] addressed the problem of tracking closely spaced targets in groups or formations using LMB filters. Their work aimed to enhance the tracking performance for densely clustered objects, enabling reliable tracking in scenarios with high target density. Liu and Huang [17] proposed an LMB filter for tracking maneuvering objects under glint noise within Jump Markov Systems (JMS). Their method effectively addressed the tracking challenges posed by glint noise, improving tracking performance for maneuvering objects. Zhang et al. [18] proposed a heavy-tailed noise-tolerant LMB (TLMB) filter for improved LMB filter performance under heavy-tailed non-Gaussian measurement noise. Their method effectively handled measurement noise with heavy-tailed distributions, improving tracking robustness in challenging measurement conditions.

Lian et al. [19] offered a robust LMB filter for multi-target tracking with time-varying noise covariances. Their method accounted for time-varying noise covariances to improve the tracking performance in scenarios with changing environmental conditions. Li et al. [20] presented a leader-follower model for tracking multiple resolvable group targets using an LMB filter. Their method addressed the tracking of multiple group targets by incorporating a leader-follower model into the LMB filter framework, enabling accurate tracking of cohesive groups. Wei et al. [21] proposed a robust LMB tracking algorithm based on box particle filtering. Their method integrated box particle filtering techniques into the LMB filter framework, enabling robust and accurate tracking in scenarios with complex measurement distributions. Jiang et al. [22] addressed stealth maneuvering multi-target tracking using an IMM-LMB filter. Their work focused on tracking stealthy maneuvering targets by employing an Interacting Multiple Model (IMM) approach within the LMB filter framework.

1.1.2 Multi-sensor fusion with LMB filters

Kellner et al. [23] proposed efficient approximate formulations of the multi-sensor LMB filter for tracking multiple objects. Their work aimed to improve computational efficiency while maintaining tracking accuracy, enabling scalable multi-sensor tracking capabilities. Gostar et al. [24] introduced an approach for fusing multi-sensor information in multi-object tracking problems using LMB filters. Their method addressed the fusion of diverse sensor inputs to improve tracking accuracy and robustness. Chai et al. [25] focused on tackling distributed multi-object tracking in multi-static radar systems. They proposed a locally run LMB filter framework to handle the challenges posed by distributed sensors, enabling efficient and accurate tracking across multiple radar platforms.

Shen et al. [26, 27] introduced a consensus-based LMB filter for resource-sensitive distributed sensor networks. Their method addressed resource constraints in distributed sensor networks by incorporating consensus-based techniques into the LMB filter framework. Shen et al. [28] also introduced a consensus-based LMB filter with adaptive event-triggered communication. Their method employed adaptive event-triggered communication among distributed sensors to enhance the consensus-based LMB filter's performance in resource-constrained environments. Zhang et al. [28] proposed a consensus-based LMB filter for distributed multi-target tracking using a bounded threshold function. Their method addressed distributed multi-target tracking challenges by employing a bounded threshold function to facilitate consensus among distributed sensors. Gao et al. [29] presented a framework for multi-target tracking with multi-detection systems. Their work focused on integrating multidetection systems to improve the tracking performance in complex environments with diverse detection sources.

1.1.3 LMB filter for specific applications

Huang et al. [30] introduced the Joint Probabilistic Hypergraph Matching LMB (JPHGM-LMB) filter, designed for rigid target tracking. Their work incorporated hypergraph matching techniques to enhance the tracking performance for rigid objects. Li et al. [31] addressed multi-source direction-of-arrival (DOA) tracking using LMB filters. Their work focused on estimating the DOA of multiple sources using LMB filters, enabling accurate source localization in diverse environments. Liu et al. [32] proposed an improved Mask RCNN and LMB algorithm for target detection and tracking. Their method combined object detection using Mask RCNN with LMB filtering, enabling accurate detection and tracking of objects in complex scenes.

Zhu et al. [33] addressed multi-object joint detection and tracking using LMB filters and image observations from multiple sensors. Their work focused on integrating information from multiple sensors to improve joint detection and tracking performance. Zhang et al. [34] presented a multi-AUV bearings-only multi-target tracking method based on the fast LMB filter, enabling accurate tracking in underwater environments. Deusch et al. [35] proposed a Rao-Blackwellized implementation of the LMB-SLAM filter for simultaneous localization and mapping (SLAM). Their work focused on integrating SLAM capabilities into the LMB filter framework, enabling joint tracking and mapping in complex environments.

Dai et al. [36] focused on multi-vehicle tracking with inter-target occlusion using a corner tracking algorithm based on the LMB filter. Their work addressed the challenges of occlusion in multi-vehicle tracking scenarios, enabling accurate tracking even when objects are partially occluded. Yu and Ye [37] proposed a method for generating reference data for multiple extended object tracking in the evaluation of autonomous driving systems. Their method provided a benchmarking approach for evaluating the performance of autonomous driving systems in tracking multiple extended objects.

Ishtiaq et al. [38] explored an interaction-aware LMB filter for vehicle tracking. Their work focused on capturing the interactions between vehicles and incorporating them into the LMB filter framework, enabling more accurate tracking of interacting vehicles. Cao et al. [39] developed an LMB filter for tracking multiple weak targets in marine radar. Their work focused on enhancing the tracking performance for weak targets in marine radar scenarios, enabling reliable tracking in challenging maritime environments.

Yang et al. [40] presented a road-map aided GMLMB filter for ground multi-target tracking. Their method incorporated road-map information to improve ground multi-target tracking performance, enabling accurate tracking in road-based scenarios.

1.2 The Problem of Interest

State estimation is a fundamental problem in many technology applications, including power systems, robotics, autonomous vehicles, aerospace systems, industrial process control, and environmental monitoring. In power systems, state estimation is used to estimate the current state of the system, including voltage magnitudes and angles, power flows, and line currents, which are crucial for monitoring and controlling the power grid. In robotics, state estimation is used to estimate the position, orientation, and velocity of a robot in its environment, which is essential for navigation, mapping, and localization tasks. Similarly, in autonomous vehicles, state estimation is used to estimate the vehicle's position, velocity, and orientation relative to its surroundings, which is crucial for safe and efficient navigation. In aerospace systems, such as aircraft and spacecraft, state estimation is used to estimate the vehicle's position, velocity, and attitude, which is vital for navigation, guidance, and control. In industrial process control, state estimation is used to estimate the current state variables of a process, such as temperature, pressure, and flow rates, which is used for monitoring and optimizing the process. In environmental monitoring systems, state estimation is used to estimate the state variables of the environment, such as temperature, humidity, air quality, and pollutant concentrations, which is crucial for understanding and managing environmental conditions.

The mathematical framework for state estimation typically involves the use of probabilistic models, such as State Space Models (SSMs) or Hidden Markov Models (HMMs). These models represent the dynamic system using a set of differential equations and latent states that are associated with the observed data. The objective of state estimation is to estimate the optimal latent states given the observed data and the knowledge encapsulated within the system equations. In the case of linear and Gaussian systems, the Kalman filter is commonly used for state estimation. The Kalman filter is an optimal recursive estimator that estimates the state of a linear dynamic system given noisy measurements. It combines the predictions from the system model with the measurements to provide an optimal estimate of the current state. For non-linear and non-Gaussian systems, particle filtering methods, such as sequential importance sampling or particle filtering, are often used. These methods approximate the posterior distribution of the latent states by sampling a set of particles and updating their weights based on the likelihood of the measurements.

The above filters are all examples of a single-object Bayesian filter, in which from time $k-1$ to k, the object's state vector changes from x_{k-1} to x_k according to a statistical model (also called motion model). The motion model encapsulates various uncertainties involved in state transition. For instance, if the object is a vehicle on the road, and its state is its location, speed and orientation on the 2D road plane, the motion model should encapsulate information such as the fact that next location of the vehicle is most likely on the direction of speed vector. Such information are not part of the measurement process but in the fabric of the nature of the application. The motion model is used in the prediction step of the filter.

After prediction, the measurement acquired by the sensor at time k, denoted by z_k, is used along with another statistical model called the object likelihood, that encapsulates uncertainties in the measurement (see Figure 1.1. The likelihood function is used in the second step of the filter, called the update step to refine the predicted density of the object's state. In such filters, it is the state density that is propagated from each time to the next via the prediction then update steps.

The difficulty increases substantially when there are multiple potential objects with their states being estimated. This situation is particularly prominent in radar multi-target tracking, which

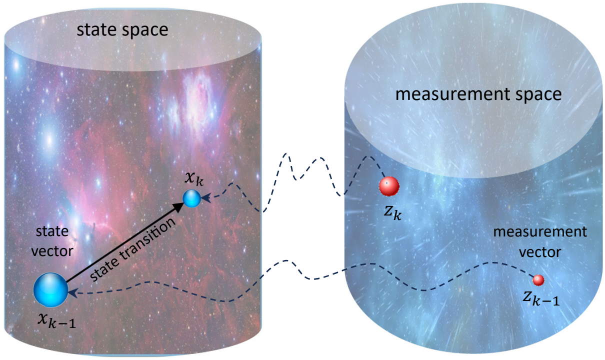

Figure 1.1: Evolution of object's state and measurement in their spaces.

is a widely recognized application that requires solving such a problem. In this context, targets can enter and leave the radar's field-of-view unpredictably, and the measurements derived from processing the radar scan, known as point detections, may contain false alarms. Additionally, it is possible that some targets may go undetected. Notably, both the multi-object state and the multi-object measurement comprise sets of vectors, and the number of elements in these sets can vary randomly over time.

In Figure 1.2, we can observe a basic scenario in which there are 4 objects and 6 sensor detections at time $k-1$. Some of these detections may correspond to the 4 objects, while others may be false alarms. As we transition to time k, one of the 4 objects is no longer present, and the remaining 3 objects have changed their positions in the state space. At this time step, the sensor provides 4 detections, with a similar combination of associations between some of the detections and the 3 existing objects, alongside additional false alarms. Note that the figure displays multi-object states, represented as capital letter X_{k-1} and X_k, which are each a *set* of multiple individual single-object states. Similarly, the multi-object measurements, denoted as capital letter Z_{k-1} and Z_k, are each a *set* composed of multiple individual detections.

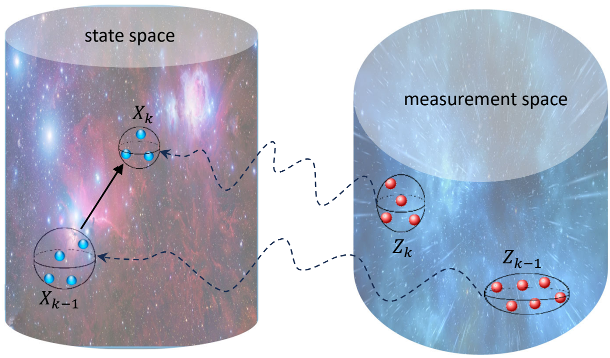

Figure 1.2: Evolution of multi-object's state (set) and multi-object measurement (set) in single-object spaces.

The multi-object estimation/tracking problem is not limited to radar. In computer Vision, object tracking in video surveillance, autonomous driving, and augmented reality applications requires accurately tracking multiple objects simultaneously. In robotics, multi-robot systems, such as swarm robotics or collaborative robotics, often require tracking multiple targets to coordinate their actions and achieve collective objectives. Even in social Behavior analysis, tracking individuals or groups in crowded scenes can help analyze social dynamics, crowd behavior, and pedestrian movement patterns in areas like public spaces, stadiums, or transportation hubs. In medical imaging applications, multi-target tracking is utilized for tracking multiple anatomical structures or lesions over time, aiding in diagnosis, treatment planning, and monitoring of diseases. In traffic surveillance, monitoring and tracking multiple vehicles helps in traffic management, congestion detection, and incident analysis.

1.2.1 Distinction between state localization and tracking: the importance of labels

In the conventional state space model, the terms "state estimation"and "tracking"are often used interchangeably because they both involve estimating the trajectory of a single object. Consistent with this terminology, the terms "multi-object state estimation" and "multi-object tracking" should both refer to the estimation of trajectories for multiple objects, and are hence referred to as "multi-object estimation" here.

To distinguish the task of solely estimating the states of multiple objects without considering trajectory information, we use the term "multi-object localization". It is important to note that unlike single-object systems, the historical states of multiple objects do not necessarily represent their respective trajectories—See Figure 1.3(a) and (b). As a result, it may not be possible to achieve multi-object estimation through filtering methods.

To effectively represent multiple objects in a manner that simulates single-object systems, it is essential to ensure that the state history accurately reflects the trajectory. This is achieved by introducing distinct labels or provisional identities to each object state, as depicted in Figure 1.3(c). The inclusion of labels in the multi-object representation is pivotal as it enables the utilization of filtering techniques for multi-object estimation. By associating a unique label with each object state, it becomes possible to track and estimate the behavior of multiple objects simultaneously. This approach allows for the estimation of object trajectories and their respective states over time.

Without the use of labels, even if it is feasible to estimate segments of the trajectories within shorter moving windows, a significant challenge arises when attempting to link these trajectories from one window to another. In the absence of proper labeling, there is no mechanism for establishing correspondences or connections between object states across different windows. As a result, multi-object estimation can only be performed within a window that grows with time, making it computationally infeasible, even for a single trajectory. Employing a labeled multi-object representation ensures that the state history accurately captures the trajectory of multiple objects. This facilitates the application of filtering techniques for multi-object estimation and overcomes the limitations

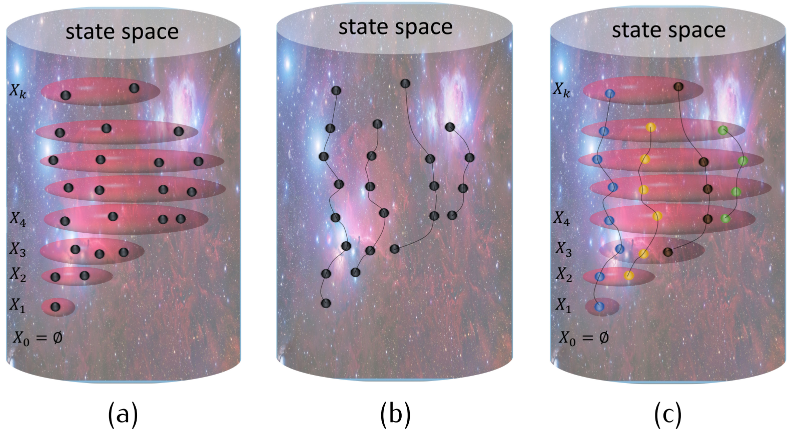

Figure 1.3: (a) The multi-object states do not convey any trajectory information. Indeed, the trajectories are what are shown in (b). With labeled multi-object states and trajectories in (c), the single-object states are augmented with labels (colors), and when the filter estimates those as part of the state, trajectories are automatically obtained.

associated with tracking objects across consecutive windows. By incorporating distinct labels, the representation allows for more efficient and comprehensive analysis of multiple objects in various applications.

1.3 The Labeled Random Finite Set Filters

Multi-object tracking (MOT) is an intricate task due to various factors such as false alarms, miss-detections, and uncertainties associated with data association. Unlike its single-object counterpart, MOT poses significantly greater challenges [41, 42, 43]. Over time, researchers have developed numerous solutions to address MOT, primarily based on three main approaches: Multiple Hypothesis Tracking (MHT), Joint Probabilistic Data Association (JPDA), and Random Finite Set (RFS) [41, 42, 43]. For a concise overview, one can also refer to [44]. It is worth noting that while MOT involves tracking a finite set of objects, only the RFS approach treats the multi-object state as a finite set. In other words, RFS considers the collective state of multiple objects rather than treating them individually. This key distinction makes the RFS approach particularly well-suited for the complexities of multi-object tracking.

Mahler's groundbreaking article on the Random Finite Set (RFS) framework and the Probability Hypothesis Density (PHD) filter was published two decades ago [45]. However, it is important to note that the PHD filter, as originally formulated, primarily focused on multi-object localization, requiring additional heuristics to generate trajectory estimates [46, pp. 505-508]. Despite this limitation, the PHD filter gained immense popularity and became synonymous with the RFS framework. Unfortunately, its inability to handle trajectories necessary for multi-object estimation was mistakenly attributed to the overall framework.

Contrary to this misconception, Mahler had already proposed an RFS formulation for multi-object estimation back in 1997. He introduced the concept of augmenting distinct labels to individual states to represent trajectories [47, pp. 135, 196-197]. This labeled multi-object representation proves to be the appropriate approach for multi-object estimation. However, at the time, it was considered computationally impractical and was eventually abandoned in favor of the unlabeled representation that enabled tractable approximations like the PHD filter.

Nevertheless, the PHD filter served as a catalyst for a new research area, leading to the development of a comprehensive set of analytical and numerical tools designed specifically for multi-object systems. These tools have significantly advanced the field of multi-object estimation, offering various techniques to address the challenges posed by complex scenarios involving multiple objects.

The concept of labeled Random Finite Set (RFS) theory was introduced by Vo et al. [48], who developed the Generalized Labeled Multi-Bernoulli (GLMB) filter. This innovative filtering technique was specifically designed to handle a vast number of trajectories, demonstrating its capability to manage scenarios involving over a million trajectories [48].

Building upon the GLMB filter, Reuter et al. [1] introduced a specialized variant known as the Labeled Multi-Bernoulli (LMB) filter in 2014. The introduction of the GLMB and LMB filters has significantly advanced the field of multi-object estimation. These filters have proven to be powerful tools in handling complex situations with a large number of trajectories, enabling more accurate and reliable estimations in various real-world applications. Their development has contributed to the continual progress of the labeled RFS theory and its practical implementation in multi-object tracking and estimation systems.

The LMB filter provides enhanced computational efficiency when accurate trajectory estimation is essential. As previously stated, the LMB filter has found widespread use across various applications, and its intuitive and straightforward structure has led to the development of numerous application-specific variations.

1.4 Who Is This Book for?

This book is meticulously designed as an exhaustive guide that paves the way to understanding the Labeled Multi-Bernoulli (LMB) filters, presenting a lucid approach to both grasping and implementing solutions based on this profound concept. The LMB filter, with its effortless structure and user-friendly design, is distinctive due to its ability to model the existence and state statistics of individual objects independently. This remarkable feature has rendered it a favorite tool among global researchers who have successfully applied it across a range of applications, often requiring minimal customization.

A successful engagement with this book only necessitates a basic familiarity with probability theory and a working knowledge of MATLAB coding. This deliberate minimal prerequisite design is aimed to make this book accessible and an ideal starting point for both burgeoning and established researchers as well as practitioners who have set their sights on exploring the uncharted territories of multi-target tracking or multi-sensor fusion applications using the LMB filter.

The book has been thoughtfully structured in a progressive format to ensure that the readers are not overwhelmed but instead gradually led towards mastering the topic. The journey commences with Chapter 2, which lays the groundwork by introducing the rudimentary yet critical concept of particle approximation of densities. It also acquaints readers with the bedrock of particle filters and motion models, setting the stage for the more advanced topics that follow.

Moving on to Chapter 3, it delves into the complex world of multi-object systems. Here, readers will find a thorough explanation of how these systems are parametrically modeled, providing them with a comprehensive understanding of the underlying mechanics.

Chapter 4 then marks the culmination of this journey, focusing entirely on the LMB filter. Here, readers are introduced to the basic theoretical framework that underpins the LMB filter, followed by a detailed walk-through of the relevant code. This approach is aimed at ensuring that the readers are not only acquainted with the theory but also gain practical exposure by understanding how the theory translates into code, enhancing their ability to apply the knowledge in a real-world setting.

An additional feature of this book is the accompanying code, provided for readers to download and try their hands on. This unique inclusion serves the dual purpose of bridging the gap between theory and practice and allowing readers to cement their understanding through actual experimentation, thereby providing a holistic learning experience.