

1. Record Nr.	UNINA9910523765803321
Autore	Livshin Igor
Titolo	Artificial neural networks with Java : tools for building neural network applications / / Igor Livshin
Pubbl/distr/stampa	[Place of publication not identified] : , : Apress, , [2022] ©2022
ISBN	1-5231-5092-0 1-4842-7368-0
Edizione	[2nd ed.]
Descrizione fisica	1 online resource (635 pages)
Disciplina	006.32
Soggetti	Neural networks (Computer science) Java (Computer program language)
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Note generali	Includes index.
Nota di contenuto	<p>Intro -- Table of Contents -- About the Author -- About the Technical Reviewers -- Acknowledgments -- Introduction -- Part I: Getting Started with Neural Networks -- Chapter 1: Learning About Neural Networks -- Biological and Artificial Neurons -- Activation Functions -- Summary -- Chapter 2: Internal Mechanics of Neural Network Processing -- Function to Be Approximated -- Network Architecture -- Forward Pass Calculation -- Input Record 1 -- Input Record 2 -- Input Record 3 -- Input Record 4 -- Back-Propagation Pass -- Function Derivative and Function Divergent -- Most Commonly Used Function Derivatives -- Summary -- Chapter 3: Manual Neural Network Processing -- Example: Manual Approximation of a Function at a Single Point -- Building the Neural Network -- Forward Pass Calculation -- Hidden Layers -- Output Layer -- Backward Pass Calculation -- Calculating Weight Adjustments for the Output-Layer Neurons -- Calculating Adjustment for W211 -- Calculating Adjustment for W212 -- Calculating Adjustment for W213 -- Calculating Weight Adjustments for Hidden-Layer Neurons -- Calculating Adjustment for W111 -- Calculating Adjustment for W112 -- Calculating Adjustment for W121 -- Calculating Adjustment for W122 -- Calculating Adjustment for W131 -- Calculating Adjustment for W132 -- Updating Network Biases -- Back to the Forward Pass -- Hidden Layers -- Output Layer</p>

-- Matrix Form of Network Calculation -- Digging Deeper -- Mini-Batches and Stochastic Gradient -- Summary -- Part II: Neural Network Java Development Environment -- Chapter 4: Configuring Your Development Environment -- Installing the Java Environment and NetBeans on Your Windows Machine -- Installing the Encog Java Framework -- Installing the XChart Package -- Summary -- Chapter 5: Neural Networks Development Using the Java Encog Framework. Example: Function Approximation Using Java Environment -- Network Architecture -- Normalizing the Input Datasets -- Building the Java Program That Normalizes Both Datasets -- Building the Neural Network Processing Program -- Program Code -- Debugging and Executing the Program -- Processing Results for the Training Method -- Testing the Network -- Testing Results -- Digging Deeper -- Summary -- Chapter 6: Neural Network Prediction Outside of the Training Range -- Example: Approximating Periodic Functions Outside of the Training Range -- Network Architecture for the Example -- Program Code for the Example -- Testing the Network -- Example: Correct Way of Approximating Periodic Functions Outside of the Training Range -- Preparing the Training Data -- Network Architecture for the Example -- Program Code for Example -- Training Results for Example -- Log of Testing Results for Example 3 -- Summary -- Chapter 7: Processing Complex Periodic Functions -- Example: Approximation of a Complex Periodic Function -- Data Preparation -- Reflecting Function Topology in the Data -- Network Architecture -- Program Code -- Training the Network -- Testing the Network -- Digging Deeper -- Summary -- Chapter 8: Approximating Noncontinuous Functions -- Example: Approximating Noncontinuous Functions -- Network Architecture -- Program Code -- Code Fragments for the Training Process -- Unsatisfactory Training Results -- Approximating the Noncontinuous Function Using the Micro-Batch Method -- Program Code for Micro-Batch Processing -- Program Code for the getChart() Method -- Code Fragment 1 of the Training Method -- Code Fragment 2 of the Training Method -- Training Results for the Micro-Batch Method -- Testing the Processing Logic -- Testing the Results for the Micro-Batch Method -- Digging Deeper -- Summary. Chapter 9: Approximation of Continuous Functions with Complex Topology -- Example: Approximation of Continuous Functions with Complex Topology Using a Conventional Neural Network Process -- Network Architecture for the Example -- Program Code for the Example -- Training Processing Results for the Example -- Approximation of Continuous Functions with Complex Topology Using the Micro-Batch Method -- Program Code for the Example Using the Micro-Batch Method -- Example: Approximation of Spiral-like Functions -- Network Architecture for the Example -- Program Code for Example -- Approximation of the Same Functions Using Micro-Batch Method -- Summary -- Chapter 10: Using Neural Networks for the Classification of Objects -- Example: Classification of Records -- Training Dataset -- Network Architecture -- Testing Dataset -- Program Code for Data Normalization -- Program Code for Classification -- Training Results -- Testing Results -- Summary -- Chapter 11: The Importance of Selecting the Correct Model -- Example: Predicting Next Month's Stock Market Price -- Including the Function Topology in the Dataset -- Building Micro-Batch Files -- Network Architecture -- Program Code -- Training Process -- Training Results -- Testing Dataset -- Testing Logic -- Testing Results -- Analyzing Testing Results -- Summary -- Chapter 12: Approximation Functions in 3D Space -- Example: Approximation Functions in 3D Space -- Data Preparation -- Network Architecture -- Program Code --

Processing Results -- Summary -- Part III: Introduction to Computer Vision -- Chapter 13: Image Recognition -- Classification of Handwritten Digits -- Preparing the Input Data -- Input Data Conversion -- Building the Conversion Program -- Summary -- Chapter 14: Classification of Handwritten Digits -- Network Architecture -- Program Code -- Programming Logic -- Execution. Convolution Neural Network -- Summary -- Index.

#### Sommario/riassunto

Develop neural network applications using the Java environment. After learning the rules involved in neural network processing, this second edition shows you how to manually process your first neural network example. The book covers the internals of front and back propagation and helps you understand the main principles of neural network processing. You also will learn how to prepare the data to be used in neural network development and you will be able to suggest various techniques of data preparation for many unconventional tasks. This book discusses the practical aspects of using Java for neural network processing. You will know how to use the Encog Java framework for processing large-scale neural network applications. Also covered is the use of neural networks for approximation of non-continuous functions. In addition to using neural networks for regression, this second edition shows you how to use neural networks for computer vision. It focuses on image recognition such as the classification of handwritten digits, input data preparation and conversion, and building the conversion program. And you will learn about topics related to the classification of handwritten digits such as network architecture, program code, programming logic, and execution. The step-by-step approach taken in the book includes plenty of examples, diagrams, and screenshots to help you grasp the concepts quickly and easily. What You Will Learn Use Java for the development of neural network applications Prepare data for many different tasks Carry out some unusual neural network processing Use a neural network to process non-continuous functions Develop a program that recognizes handwritten digits Who This Book Is For Intermediate machine learning and deep learning developers who are interested in switching to Java.

2. Record Nr.	UNINA9911047799303321
Autore	Downey R. G. q (Rod G.)
Titolo	Computable structure theory : a unified approach / / Rodney G. Downey, Alexander Melnikov
Pubbl/distr/stampa	Cham : , : Springer, , [2026] ©2026
ISBN	9783031924330
Edizione	[1st ed.]
Descrizione fisica	1 online resource (xii, 540 pages) : illustrations
Collana	Theory and applications of computability, In cooperation with the Association Computability in Europe, , 2190-6203
Altri autori (Persone)	MelnikovAlexander
Disciplina	511.352
Soggetti	Computer science Computer arithmetic and logic units Algorithms Computable functions Recursion theory Theory of Computation Arithmetic and Logic Structures Computability and Recursion Theory
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Nota di bibliografia	Includes bibliographical references and index.
Nota di contenuto	Part I Foundation of Computability -- 1. Introduction -- 2. Basics of Computability Theory -- 3. Computable Algebraic Structures -- 4. Computable Separable Spaces -- Part II Computable Duality -- 5. Computable Boolean Algebras -- 6. Computable Stone Spaces -- 7. Computable Abelian Groups -- 8. Computable Connected Compact Spaces -- Part III Computability and Classification Problems. 9. The Analytical Hierarchy and 11-completeness -- 10. Computable Categoricity -- 11. Computable Banach Spaces with Applications -- 12. Resource Bounded Computation -- Part IV Non-computability and Randomness. 13. Randomness -- 14. Degree Spectra -- 15 -- Computable Transfinite Analysis.
Sommario/riassunto	This is the first book which gives a unified theory for countable and uncountable computable structures. The work treats computable linear orderings, graphs, groups and Boolean algebras unified with

computable metric and Banach spaces, profinite groups, and the like. Further, it provides the first account of these that exploits effective versions of dualities, such as Stone and Pontryagin dualities. The themes are effective classification and enumeration. Topics and features: · Delivers a self-contained, gentle introduction to priority arguments, directly applying them in algebraic contexts · Includes extensive exercises that both cement and amplify the materials · Provides complete introduction to the basics of computable analysis, particularly in the context of computable structures · Offers the first monograph treatment of computable Polish groups, effective profinite groups via Stone duality, and effective abelian groups via Pontryagin duality · Presents the first book treatment of Friedberg enumerations of structures This unique volume is aimed at graduate students and researchers in computability theory, as well as mathematicians seeking to understand the algorithmic content of structure theory. Being self-contained, it provides ample opportunity for self-study.

---