V. Cherkassky, University of Minnesota, Minneapolis, MN, USA; **Y. Ma**, Honeywell International, Minneapolis, MN, USA

Introduction to Predictive Learning

The subject of data-driven modeling has been addressed in various disciplines such as statistics, pattern recognition, signal processing, genomics, artificial neural networks, machine learning, and data mining, which adopt specialized terminology and conceptual frameworks to motivate various learning algorithms, in spite of the close similarity (equivalence) between actual algorithms. The main commonality between these methodologies is that they all develop algorithms for estimating predictive models from data, albeit providing quite different motivation for these algorithms. This textbook, accessible to undergraduate students and practitioners, emphasizes the methodology and principles of predictive learning, rather than specialized terminology or detailed description of learning algorithms. Introduction to Predictive Learning adopts the conceptual framework developed in Vapnik-Chervonenkis (VC) theory, focusing on the methodological and practical aspects of VC-theory rather than its technical details.

Features

► Offers a unifying/coherent treatment of learning from data in terms of fundamental underlying concepts ► Presents a mixture of mathematical and philosophical concepts related to predictive learning and induction ► Explains technical aspects of learning methods by means of examples and application studies ► Provides problems, programming assignments and writing assignments at the end of each chapter

Fields of interest
Pattern Recognition; Artificial Intelligence (incl. Robotics); Data Mining and Knowledge Discovery

Target groups
Lower undergraduate

Type of publication
Undergraduate textbook

O. Grillmeyer, University of California, Berkeley, CA, USA

Exploring Computer Science with Scheme

The aim of this textbook is to present the central and basic concepts, techniques, and tools of computer science. The emphasis is on presenting a problem-solving approach and on providing a survey of all of the most important topics covered in computer science degree programmes. Scheme is used throughout as the programming language and the author stresses a functional programming approach which concentrates on the creation of simple functions that are composed to obtain the desired programming goal. Such simple functions are easily tested individually. This greatly helps in producing programs that work right first time. Throughout, the author presents techniques to aid in the writing of programs and makes liberal use of boxes which present "Mistakes to Avoid." Many programming examples are discussed in detail which illustrate general approaches to programming. These include: - abstracting a problem; - creating pseudo code as an intermediate solution; - top-down and bottom-up design; - building procedural and data abstractions; - writing progams in modules which are easily testable. Numerous exercises help the readers test their understanding of the material and develop some ideas in greater depth. As a result this text will make an ideal first course for all students coming to computer science for the first time.

Features

► Expanded later chapters make this book a state-of-the art reference for Scheme ► Revisions based on feedback from 10 years of teaching experience with the first edition ► Author is a respected authority in the field of Scheme

Fields of interest
Theory of Computation; Artificial Intelligence (incl. Robotics)

Target groups
Lower undergraduate

Type of publication
Undergraduate textbook

A. Roscoe, Oxford University, Oxford, UK; **C. B. Jones**, Newcastle University, Newcastle upon Tyne, UK; **K. Wood**, Microsoft Research, Cambridge, UK (Eds.)

Reflections on the Work of C.A.R. Hoare

Written in honor of Sir Tony Hoare's 75th Birthday, this book provides a discussion of the influence of Tony Hoare's work on current research from an international selection of expert contributors. Also supplied is a scientific biography, listing the most influential work by Tony Hoare.

Features

► Written in honor of Sir Tony Hoare's 75th birthday ► Discusses the influence of Sir Tony Hoare's work on current research ► Provides a scientific biography of Sir Tony Hoare

Contents
Insight, Inspiration and Collaboration.- From CSP to Game Semantics.- On Mereologies in Computing Science.- Roles, Stacks, Histories: A Triple for Hoare.- Forward with Hoare.- Probabilistic Programming with Coordination.- The Operational Principle and Problem Frames.- The Role of Auxiliary Variables in the Formal Development of Concurrent Programs.- Avoid a Void: The Eradication of Null Dereferencing.- Unfolding CSP.- Quicksort: Combining Concurrency, Recursion, and Mutable Data Structures.- The Thousand-and-One Cryptographers.- On Process-algebraic Extensions of Metric Temporal Logic.- Fun with Type Functions.- On CSP and the Algebraic Theory of Effects.- CSP is Expressive Enough for π.- The Tokeneer Experiments.

Fields of interest
History of Computing

Target groups
Research

Type of publication
Contributed volume

K. Schmidt, Copenhagen Business School, Denmark

Cooperative Work and Coordinative Practices

Contributions to the Conceptual Foundations of Computer Supported Cooperative Work

Information technology has been used in organisational settings and for organisational purposes such as accounting, for a half century, but IT is now increasingly being used for the purposes of mediating and regulating complex activities in which multiple professional users are involved, such as in factories, hospitals, architectural offices, and so on The economic importance of such coordination systems is enormous but their design often inadequate. The problem is that our understanding of the coordinative practices for which these systems are developed is deficient, leaving systems developers and software engineers to base their designs on commonsensical requirements analyses.

The research reflected in this book addresses these very problems. It is a collection of articles which establish a conceptual foundation for the research area of Computer-Supported Cooperative Work.

Features
► Brings together both classic and less well known articles on the phenomena of cooperative work and coordinative practices - providing critical elements of the conceptual foundation of CSCW

From the contents
Cooperative work and coordinative practices: a progress report.- The texts: genealogical data and credits.- Riding a tiger, or Computer-Supported Cooperative Work.- Taking CSCW seriously: Supporting articulation work.- Modes and mechanisms of interaction in cooperative work.- The organisation of cooperative work.

Fields of interest
Information Systems and Communication Service; Models and Principles; Database Management

Target groups
Research

Type of publication
Reference work

Computer Science

Due July 2010

C. P. Williams, Pasadena, CA, USA

Explorations in Quantum Computing

By the year 2020, the basic memory components of a computer will be the size of individual atoms. At such scales, the current theory of computation will become invalid.

"Quantum computing" is reinventing the foundations of computer science and information theory in a way that is consistent with quantum physics - the most accurate model of reality currently known. Remarkably, this theory predicts that quantum computers can perform certain tasks breathtakingly faster than classical computers – and, better yet, can accomplish mind-boggling feats such as teleporting information, breaking supposedly "unbreakable" codes, generating true random numbers, and communicating with messages that betray the presence of eavesdropping.

Features
► Provides an introduction to the basic mathematical formalism of quantum computing, and the quantum effects that can be harnessed for non-classical computation ► Concludes each chapter with exercises and a summary of the material covered ► Examines the potential applications of quantum computers, investigates the uses of quantum information, and reviews the advancements made towards practical quantum computers

From the contents
Part I: What Is Quantum Computing.- Introduction.- Quantum Gates.- Quantum Circuits.- Quantum Universality, Computability, & Complexity.- Part II: What Can You Do With A Quantum Computer?- Performing Search With A Quantum Computer.- Code Breaking With A Quantum Computer.- Solving NP-Complete Problems With A Quantum Computer.- Quantum Simulation With A Quantum Computer.

Fields of interest
Programming Techniques; Software Engineering; Quantum Physics

Target groups
Professional/practitioner

Type of publication
Graduate/Advanced undergraduate textbook

Computer Science

Due May 2010