

1. Record Nr.	UNINA9910813335003321
Autore	Lim Aloysius
Titolo	R high performance programming : overcome performance difficulties in R with a range of exciting techniques and solutions // Aloysius Lim, William Tjhi
Pubbl/distr/stampa	Birmingham, England : , : Packt Publishing, , 2015 ©2015
ISBN	1-78398-927-0
Descrizione fisica	1 online resource (176 p.)
Collana	Community Experience Distilled
Disciplina	519.502855133
Soggetti	R (Computer program language)
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Note generali	Includes index.
Nota di contenuto	Cover; Copyright; Credits; About the Authors; About the Reviewers; www.PacktPub.com; Table of Contents; Preface; Chapter 1: Understanding R's Performance - Why Are R Programs Sometimes Slow?; Three constraints on computing performance - CPU, RAM, and disk I/O; R is interpreted on the fly; R is single-threaded; R requires all data to be loaded into memory; Algorithm design affects time and space complexity; Summary; Chapter 2: Profiling - Measuring Code's Performance; Measuring the total execution time; Measuring execution time with system.time(); Repeating time measurements with rbenchmark Measuring distribution of execution time with microbenchmarkProfiling the execution time; Profiling a function with Rprof(); The profiling results; Profiling the memory utilization; Monitoring memory utilization, CPU utilization, and disk I/O using OS tools; Identifying and resolving bottlenecks; Summary; Chapter 3: Simple Tweaks to Make R Run Faster; Vectorization; Use of built-in functions; Preallocating memory; Use of simpler data structures; Use of hash tables for frequent lookups on large data; Seek fast alternative packages in CRAN; Summary Chapter 4: Using Compiled Code for Greater SpeedCompiling R code before execution; Compiling functions; Just-in-time (JIT) compilation of R code; Using compiled languages in R; Prerequisites; Including

compiled code inline; Calling external compiled code; Considerations for using compiled code; The R APIs; R data types versus native data types; Creating R objects and garbage collection; Allocating memory for non-R objects; Summary; Chapter 5: Using GPUs to Run R Even Faster; General purpose computing on GPUs; R and GPUs; Installing gputools; Fast statistical modeling in R with gputools
SummaryChapter 6: Simple Tweaks to Use Less RAM; Reusing objects without taking up more memory; Removing intermediate data when it is no longer needed; Calculating values on the fly instead of storing them persistently; Swapping active and non-active data; Summary; Chapter 7: Processing Large Datasets with Limited RAM; Using memory-efficient data structures; Smaller data types; Sparse matrices; Symmetric matrices; Bit vectors; Using memory-mapped files and processing data in chunks; The bigmemory package; The ff package; Summary; Chapter 8: Multiplying Performance with Parallel Computing
Data parallelism versus task parallelismImplementing data parallel algorithms; Implementing task parallel algorithms; Running the same task on workers in a cluster; Running different tasks on workers in a cluster; Executing tasks in parallel on a cluster of computers; Shared memory versus distributed memory parallelism; Optimizing parallel performance; Summary; Chapter 9: Offloading Data Processing to Database Systems; Extracting data into R versus processing data in a database; Preprocessing data in a relational database using SQL; Converting R expressions into SQL; Using dplyr
Using PivotalR

Sommario/riassunto

This book is for programmers and developers who want to improve the performance of their R programs by making them run faster with large data sets or who are trying to solve a pesky performance problem.
