

Introduction to R Programming

Course ISI-1493B Two days Instructor-led, Hands-on

Introduction

Over the past few years, R has been steadily gaining popularity with business analysts, statisticians and data scientists as a tool of choice for conducting statistical analysis of data as well as supervised and unsupervised machine learning.

This two-day, instructor-led course helps students learn the practical aspects of the R programming language. The course is supplemented by many hands-on labs which allow attendees to immediately apply their theoretical knowledge in practice.

Topics include high octane introduction to R programming; learning about R data structures; working with R functions; statistical data analysis with R.

This course introduces the R language, the RStudio editor, and common R packages used to import, manipulate, and analyze data. The 20773 class assumes basic familiarity with R and builds on that foundation by adding visualization, machine learning algorithms (including regression, clustering, and decision trees), support for very large data sets using Microsoft R Server, and ways to access Microsoft SQL Server database data.

This course is appropriate for business analysts, technical managers and programmers.

At Course Completion

At course completion, students will be able to use an R development environment, access data stored in files, import package code, and write their own R functions.

Prerequisites

This course is appropriate for students new to R. Participants should have the general knowledge of statistics and programming.

Student Materials

The student kit includes a comprehensive workbook and other necessary materials for this class.

Course Outline

Module 1: What is R

- What is R?
- Positioning of R in the Data Science Space
- The Legal Aspects

Contact ISInc for more information at 916.920.1700 or by visiting our website at <http://www.isinc.com>

- Microsoft R Open
- R Integrated Development Environments
- Running R
- Running RStudio
- Getting Help
- General Notes on R Commands and Statements
- Assignment Operators
- R Objects and Workspace
- Printing Objects
- Arithmetic Operators
- Logical Operators
- System Date and Time
- Operations
- User-defined Functions
- Control Statements
- Conditional Execution
- Repetitive execution
- Built-in Functions

Module 2: Introduction to Functional Programming with R

- What is functional Programming (FP)?
- Terminology: Higher-Order Functions
- A Short List of Languages that Support FP
- Functional Programming in R
- Vector and Matrix Arithmetic
- Vector Arithmetic Example
- More Examples of FP in R

Module 3: Managing Your Environment

- Getting and Setting the Working Directory
- Getting the List of Files in a Directory
- The R Home Directory
- Executing External R commands
- Loading External Scripts in RStudio
- Listing Objects in Workspace
- Removing Objects in Workspace
- Saving Your Workspace in R
- Saving Your Workspace in RStudio
- Saving Your Workspace in R GUI
- Loading Your Workspace
- Diverting Output to a File
- Batch (Unattended) Processing
- Controlling Global Options

Contact ISInc for more information at 916.920.1700 or by visiting our website at <http://www.isinc.com>

Module 4: R Type System and Structures

- The R Data Types
- System Date and Time
- Formatting Date and Time
- Using the mode() Function
- R Data Structures
- What is the Type of My Data Structure?
- Creating Vectors
- Logical Vectors
- Character Vectors
- Factorization
- Multi-Mode Vectors
- The Length of the Vector
- Getting Vector Elements
- Lists
- A List with Element Names
- Extracting List Elements
- Adding to a List
- Matrix Data Structure
- Creating Matrices
- Creating Matrices with cbind() and rbind()
- Working with Data Frames
- Matrices vs Data Frames
- A Data Frame Sample
- Creating a Data Frame
- Accessing Data Cells
- Getting info About a Data Frame
- Selecting Columns in Data Frames
- Selecting Rows in Data Frames
- Getting a subset of a Data Frame
- Sorting (ordering) Data in Data Frames by Attribute(s)
- Editing Data Frames
- The str() Function
- Type Conversion (coercion)
- The summary() Function
- Checking an Object's Type

Module 5: Extending R

- The Base R Packages
- Loading Packages
- What is the Difference between Package and Library?
- Extending R

- The CRAN Web Site
- Extending R in R GUI
- Extending R in RStudio
- Installing and Removing Packages from Command-Line

Module 6: Read-Write and Import-Export Operations in R

- Reading Data from a File into a Vector
- Example of Reading Data from a File into a Vector
- Writing Data to a File
- Example of Writing Data to a File
- Reading Data into A Data Frame
- Writing CSV Files
- Importing Dat into R
- Exporting Data from R

Module 7: Statistical Computing Features in R

- Statistical Computing Features
- Descriptive Statistics
- Basic Statistical Functions
- Examples of Using Basic Statistical Functions
- Non-uniformity of a Probability Distribution
- Writing Your Own skew and kurtosis Functions
- Generating Normally Distributed Random Numbers
- Generating Uniformly Distributed Random Numbers
- Using the summary() Function
- Math Functions Used in Data Analysis
- Examples of Using Math Functions
- Correlations
- Correlation Example
- Testing Correlation Coefficient for Significance
- The cor.test Function
- The cor.test Example
- Regression Analysis
- Types of Regression
- Simple Linear Regression Model
- Least-Squares Method (LSM)
- LSM Assumptions
- Fitting Linear Regression Models in R
- Example of Using lm()
- Confidence intervals for Model Parameters
- Example of Using lm() with a Data Frame
- Regression Models in Excel
- Multiple Regression Analysis

Contact ISInc for more information at 916.920.1700 or by visiting our website at <http://www.isinc.com>

Module 8: Data Manipulation and Transformation in R

- Applying Functions to Matrices and Data Frames
- The `apply()` Function
- Using `apply()`
- Using `apply()` with a User-Defined Function
- `Apply()` Variants
- Using `tapply()`
- Adding a Column to a Data Frame
- Dropping a Column in a Data Frame
- The `attach90` and `detach90` Functions
- Sampling
- Using `sample90` for Generating Labels
- Set Operations
- Example of Using Set Operations
- The `dplyr` Package
- Object Masking (Shadowing) Considerations
- Getting More Information on `dplyr` in RStudio
- The `search()` or `searchpaths()` Functions
- Handling Large Data Sets in R with the `data.table` Package
- The `fread()` and `fwrite()` functions from the `dat.table` Package
- Using the Data Table Structure

Module 9: Data Visualization in R

- Data Visualization
- Data Visualization in R
- The `ggplot2` Data Visualization Package
- Creating Bar Plots in R
- Creating Horizontal Bar Plots
- Using `barplot()` with Matrices
- Using `barplot()` with Matrices Example
- Customizing Plots
- Histograms in R
- Building Histograms with `hist()`
- Example of using `hist()`
- Pie Charts in R
- Examples of using `pie()`
- Generic X-Y Plotting
- Examples of the `plot()` function
- Dot Plots in R
- Saving Your Work
- Supported Export Options
- Plots in RStudio

- Saving a Plot as an Image

Module 10: Using R Efficiently

- Object Memory Allocation Considerations
- Garbage Collection
- Finding Out About Loaded Packages
- Using the conflicts() Function
- Getting Information About the Object Source Package with the pryu Package
- Using the where90 Function from the pryr Package
- Timing Your Code
- Timing Your Code with system.time()
- Timing Your Code with System.time()
- Sleeping a Program
- Handling Large Data Sets in R with the data.table Package
- Passing System-Level Parameters to R