

**Analyzing Big Data with R Programming**

**Course Number:** RPROG-112  
**Duration:** 4 days

**Overview**

Accelebrate's Analyzing Big Data with R Programming training teaches attendees how to use In-memory/on-disk, distributed analysis using H20, Hadoop, and Apache Spark, and how to integrate Microsoft Machine Learning Server and R.

**Prerequisites**

In addition to their professional experience, students who attend this course should have:

* Programming experience using R, and familiarity with common R packages
* Knowledge of common statistical methods and data analysis best practices
* Basic knowledge of the Microsoft Windows operating system and its core functionality

**Materials**

All R training students receive comprehensive courseware.

**Software Needed on Each Student PC**

* A recent release of R 4.x
* IDE or text editor of your choice (RStudio recommended)

**Objectives**

* Understand how R works with big data sets
* Manage big data in memory with data.table
* Conduct exploratory data analysis with data.table
* Learn big data management strategies such as sampling, chunk-and-pull, and pushing compute to the database
* Run SQL queries directly against R dataframes using DuckDB
* Use DuckDB as an out-of memory backend for R dataframes
* Perform machine learning operations using mlr3
* Interface with Apache Spark using Sparklyr or SparkR
* Use H2O for data munging and machine learning

**Outline**

* Introduction:
  + Does R work with big datasets?
  + What challenges does big data introduce when using R?
  + ETL and descriptive data tasks
  + Modeling tasks, optimization challenges
* In-memory Big Data: Data.table
  + Why do we need data.table?
  + The i and the j arguments in data.table
  + Renaming columns
  + Adding new columns
  + Binning data (continuous to categorical)
  + Combining categorical values
  + Transforming variables
  + Group-by functions with data.table
  + Chaining commands with data.table
  + Data.table pronouns .N, .SD, SDCols
  + Handling missing data
* EDA with Data.table
  + Data subsetting, splitting, and merging
  + Managing datasets
  + Long to wide and back
  + Merging datasets together
  + Stacking datasets together (concatenation)
  + Data summarization
    - Numerical summaries
    - Categorical summaries
    - Multivariate summaries
  + Creating visualizations
* Big Three Strategies for dealing with Big Data in R
  + https://rviews.rstudio.com/2019/07/17/3-big-data-strategies-for-r/
  + 1. Sampling
  + 2. Chunk-and-pull
  + 3. Push compute to DB
* DuckDB
  + Overview: DuckDB works nicely with R
  + Basic SQL commands for working with DuckDB
  + Understanding query performance optimizations
  + Using dbplyr to work with DuckDB
* mlr3 for Machine Learning in R
  + Overview of mlr3
  + Goals of machine learning
  + mlr3 R6 object-oriented R and methods
  + Defining a task
  + Assigning roles to data
  + Performing a classification
  + Performing a regression
  + Visualization with mlr3
  + Pipelines
  + Model assessment
  + Model optimization
  + Implementing general linear models
  + Establishing and leveraging partitions/clusters
  + Fitting regression models and making predictions
  + Decision trees and random forests
  + Naïve bayes
  + Implementing stacked models via pipelines
  + Implementing an AutoML model via pipelines
  + Managing resource utilization through parallelization
* Apache Spark
  + Overview of Spark
  + APIs to use Apache Spark with R
  + Sparklyr versus SparkR
  + R, Python, Java and Scala APIs to Spark
  + Applied Examples using SparkR
  + Spark and H2O together: sparklingwater
  + Data import and manipulation in Spark(R)
  + The Spark machine learning library MLlib:
    - General linear models
    - Random forest
    - Naïve bayes
  + Data Munging and Machine Learning Via H20
    - Intro to H20
    - Launching the cluster, checking status
    - Data Import, manipulation in H20
    - Fitting models in H20
    - Generalized Linear Models
    - Naïve bayes
    - Random forest
    - Gradient boosting machine (GBM)
    - Ensemble model building
    - AutoML
    - Methods for explaining modeling output
* Conclusion