Course Name	Oracle Machine Learning using R
Course Duration	3 Days (24 hours)
Target Audience	Data Analyst, Data Scientist, ML Engineer
Course Outcomes	The OML4R Transparency Layer
	Creating and Managing R Objects in Oracle Database
	Data Preparation and Data Manipulation
	OML4R Embedded R Execution
	Modeling in OML4R
	Working with ROracle
	OML4R Statistics Engine

Module 01: Overview of Oracle Machine Learning for R	
1.1	Overview of Oracle Machine Learning for R
1.2	Oracle Machine Learning Notebooks
1.3	Oracle Machine Learning: Key Attributes
1.4	Oracle R Distribution
1.5	Oracle Machine Learning for R: Installation Steps
1.6	Practice I-I: Understanding the Lab Environment
1.7	Practice 1-2: Connect to the R console
1.8	Practice 1-3: Connect to RStudio Server
1.9	Practice 1-4: Run the Sample Scripts
Module 02: OML4R Transparency Layer: Introduction	
2.1	OML4R Transparency Layer: Introduction
2.2	Using ore.connect Function
2.3	In-Database Sampling
2.4	Ordering Framework Creatine Ordered and Unordered ore.frame Objects
2.5	Practice 2-1: Using the ore.connect Function
2.6	Practice 2-2: Working with OML4R Transparency Layer
2.7	Practice 2-3: Additional Transparency Layer Functions
2.8	Practice 2-4: Working with ore.Frame Database Table Proxy Object

2.9	Practice 2-5: Using Scale() and Transform() Functions
2.11	Practice 2-6: Sampling Data
2.12	Practice 2-7: Using the core.disconnect Function
Module 03: OML4R Transparency Layer: Create and Manage R Objects in Oracle Database	
3.1	Introduction to OML4R Transparency Layer
3.2	ore.save() Function
3.3	Create R Objects for In-Database Data
3.4	Move Data to and from the Database
3.5	Practice 3-1: Get Objects with ore.get Function
3.6	Practice 3-2: using ore.Create, ore.Push and ore.Drop
3.7	Practice 3-3: Create and Manage R Object in Oracle Database using R Data Store
Module 04: OML4R Transparency Layer: Data Preparation and Data Manipulation	
4.1	Introduction
4.2	Using row indexing
4.3	Exploratory Data Analysis Functions
4.4	Using Third-Party Packages on the R Client
4.5	Practice 4-1: Basic data manipulation using OREdplyr
4.6	Practice 4-2: Stacking and Grouping using OREdplyr
4.7	Practice 4-3: Chaining Using OREdplyr
4.8	Practice 4-4: Implementing Rank Function using OREdplyr
4.9	Practice 4-5: Aggregate the Column values using
4.11	the OREdplyr
4.12	Practice 4-6: Joining data using OREdplyr
Module 05: OML4R Embedded R Execution — R Interface	
5.1	Introduction
5.2	User-Defined R Functions for Embedded R Execution
5.3	Functionality of Automatic Connection
5.4	Using the ore.doEval Function
5.5	OML4R-Defined Graphics Function
5.6	Practice 5-1: Using ore.doEval() with R Script Repository
5.7	Practice 5-2: Using ore.table.Apply()
5.8	Practice 5-3: Using Embedded R Execution Functions: ore.eroup Apply, ore.index Apply

5.9	Practice 5-4: Best Practice Workflow for Developing Deployable User-Defined Functions
Module 06: OML4R Embedded R Execution — SQL Interface	
6.1	OML4R Embedded R Execution — SQL Interface
6.2	SQL API for Oracle Machine Learning for R
6.3	Returning R Statistical Results as a Database Table
6.4	Manage User-Defined R Functions using the SQL
6.5	Interface
6.6	PL/SQL Procedures for Managing R Scripts and Datastores
6.7	Practice 6-1: Build an 1m Model by using rqTabIe Eval()
6.8	Practice 6-2: Score Data in Batch Mode Using rqTabIe Eval()
6.9	Practice 6-3: Passing Arguments to User-Defined Functions using SQL API
6.11	Practice 64: Using the SQL Datastore API
6.12	Practice 6-5: Additional Code Samples
Module 07: Modeling in OML4R: Part 1	
7.1	Modeling in OML4R: Part I
7.2	OREdm Package
7.3	Feature Extraction - Explicit Semantic Analysis (ESA)
7.4	Generalized Linear Models
7.5	Clustering - K-Means
7.6	Classification - Naive Bayes
7.7	Feature Extraction Non-negative Matrix Factorization (NMF)
7.8	Partitioned Models
7.9	Practice 7-1: using Ore.odmGLM
7.11	Practice 7-2: Using Ore_odmSVM
7.12	Practice 7-3: using Ore_odmKMeans
7.13	Practice 74: using Ore.odmAssocRuIes
7.14	Practice 7-5: Using ERE Framework algorithms
7.15	Practice 7-6: Identifying Frequently-Purchased Groceries
Module 08: Modeling in OML4R: Part 2	
8.1	Introduction
8.2	Neural Networks
8.3	Singular Value Decomposition

8.4	Practice 8-1: using ore.glm Function
8.5	Practice 8-2: Using ore.neural Function
8.6	Practice 8-3: using ore.randomForest Function
8.7	Practice 8-4: Usecase-Estimating Wine Quality
Module 09: Working with ROracle	
9.1	Introduction
9.2	Connect to an extproc for Use within OML4R Embedded R Execution
9.3	Read/Write Table Methods
9.4	Practice 9-1: using the ROracIe Interface
Module 10: OML4R Statistics Engine	
10.1	OML4R Statistics Engine
10.2	ore_summary
10.3	ore.rank
10.4	ore.sort
10.5	ore.corr
10.6	ore.crosstab
10.7	ore.freq
10.8	ore.esm
10.9	ore.univariate
10.11	Practice 10-1: Working with OML4R Statistical Functions
10.12	Practice 10-2: Using ore.summary Function
10.13	Practice 10-3: Using ore.rank Function
10.14	Practice 10-4: using ore.sort Function
Module 11: OML4R Best Practices	
11.1	Introduction
11.2	Open Source Packages
11.3	Machine Learning Interface: Benefits
11.4	Explicitly Specifying Oracle Database Parallelism
11.5	Embedded R Execution Initial Memory Management Considerations
11.6	Datastore: Benefits
11.7	Object Migration