# Instructions for preparing Premium Linux HDI Spark clusters running MRS (preview) and performing analytical tasks described in the KDD 2016 tutorial: “Scalable R on Spark”

**John-Mark Agosta**, **Debraj GuhaThakurta**, **Robert Horton**, **Mario Inchiosa, Srini Kumar**,

**Vanja Paunic**, **Hang Zhang,** **Mengyue Zhao**

Data Group, Microsoft Corporation

## KDD 2016 tutorial: Scalable R on Spark

In August 2016, we presented a tutorial titled: (links to [KDD tutorial page](http://www.kdd.org/kdd2016/tutorials/view/scalable-r-on-spark); [ACM digital library](http://dl.acm.org/citation.cfm?id=2945398&dl=ACM&coll=DL&CFID=839147418&CFTOKEN=52861733), and [public GitHub repository for tutorial materials](https://github.com/Azure/Azure-MachineLearning-DataScience/tree/master/Misc/KDDCup2016)). This document provides instructions on how to create Spark HDInsight clusters (preview) running MRS, and execute the R code, which was a part of the hands-on exercise at the tutorial.

## Overview of what is provided in this document

For preparing the clusters we provide [Windows PowerShell](https://technet.microsoft.com/en-us/library/bb978526.aspx) scripts with [Azure commandlets](https://msdn.microsoft.com/en-us/library/azure/jj554330.aspx), a set of Windows PowerShell modules that help you to automate your Microsoft Azure resource and service management tasks. For the hands-on exercises running R code, we have provided R markdown files, which can be easily executed to generate the results in an HTML output file.

Here we provide details of the following steps:

### [Pre-requisites – setting up Windows PowerShell and Azure management modules](#_Pre-requisites:_Installing_Azure)

### [Provisioning Premium Azure HDInsight Spark clusters with MRS (preview) and the necessary blob storage accounts](#_Provisioning_Azure_HDInsight)

### [Customizing cluster with script actions: Installing software and R packages, and copying code files from GitHub by running script actions on cluster edge node](#_Customizing_cluster_with)

### [Copying files from a public blob storage using](#_Copying_files_from)

### [Verifying clusters for files and Spark compute environment](#_Verifying_clusters_for)

### [Hands-on exercises: Execution of the R code](#_Hands-on_exercises:_Execution)

### [Deleting clusters and associated storage accounts (when cluster is not needed further)](#_Deleting_clusters_and)

For steps 2, 3, and 7, we provide scripts that will run in parallel, thereby reducing the total cluster preparation and deletion times.

All the scripts and code are available for download from a [public GitHub repository](https://github.com/Azure/Azure-MachineLearning-DataScience/tree/master/Misc/KDDCup2016). Scripts for creating, preparing, and deleting clusters are in the folder “Scripts”. R code for exercises are in the folder “Code”. Slides presented during the tutorial are in the folder “Slides”. Slides provide details about the tutorial, including Microsoft R server architecture on HDInsight Spark clusters (preview).

## Pre-requisites: Installing Azure resource and service management modules

First, download or update your Azure PowerShell commandlets by following the instructions given [here](http://social.technet.microsoft.com/wiki/contents/articles/31127.azure-powershell-cmdlets-version-updates.aspx).

Run Windows PowerShell as administrator and install the Azure Resource Manager and Service Management modules from PowerShell Gallery as described [here](https://azure.microsoft.com/en-us/documentation/articles/powershell-install-configure/).

# Install the Azure Resource Manager modules from the PowerShell Gallery

Install-Module AzureRM

# Install the Azure Service Management module from the PowerShell Gallery

Install-Module Azure

The above commands may ask you to install NuGet provider. Select ‘Yes’ if prompted. It may also ask if you want to install the modules from ‘PSGallery’. Select ‘Yes’.

You will also need to download and install [AzCopy](https://azure.microsoft.com/en-us/documentation/articles/storage-use-azcopy/) on your Windows machine. Instructions are provided [here](https://azure.microsoft.com/en-us/documentation/articles/storage-use-azcopy/).

## Provisioning Premium Azure HDInsight Spark clusters with MRS (preview) and the necessary blob storage accounts

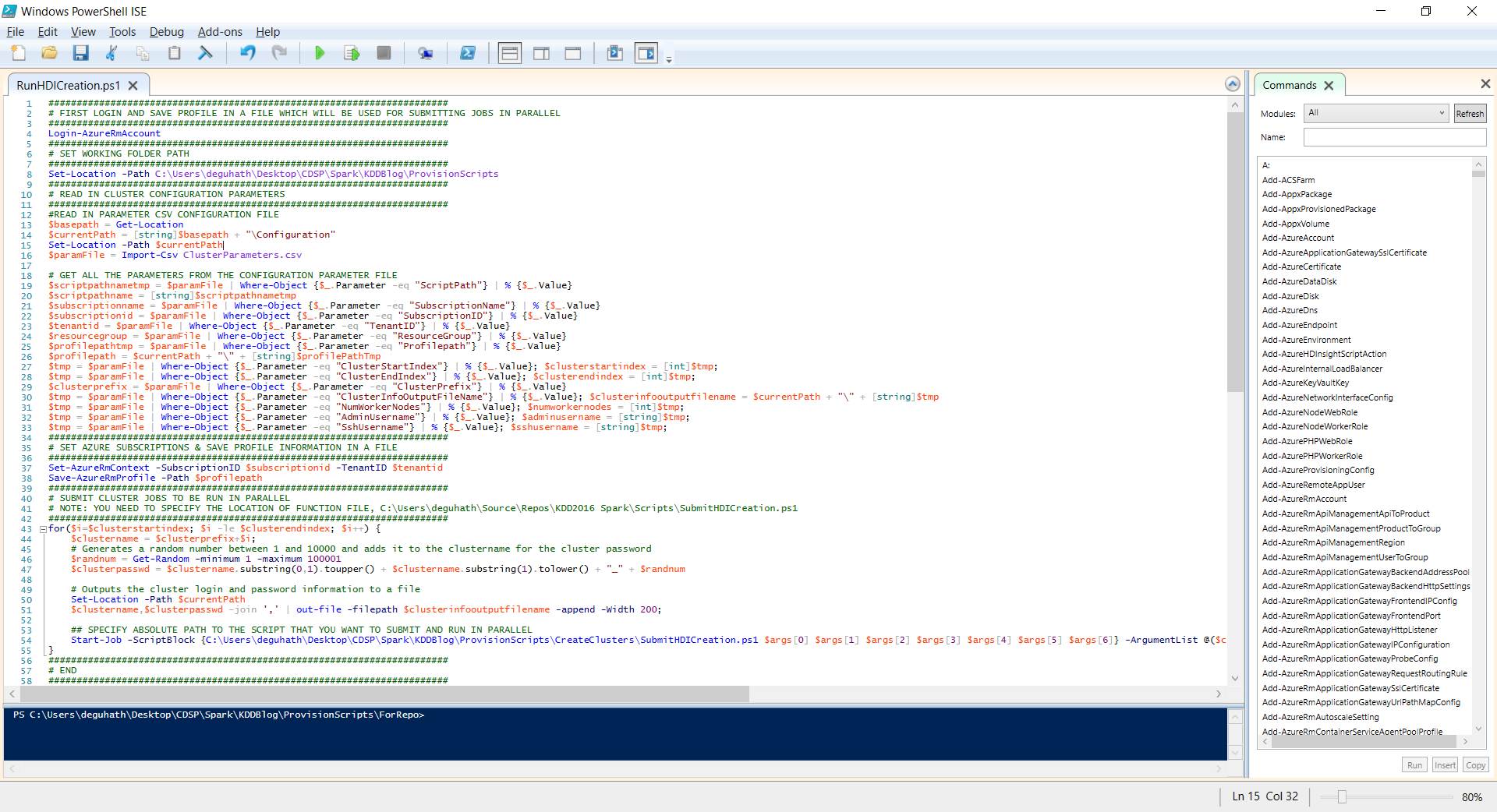
An overview of Microsoft R Server on HDInsight (preview) is provided [here](https://azure.microsoft.com/en-us/documentation/articles/hdinsight-hadoop-r-server-overview/). For provisioning Spark HDInsight clusters using Windows PowerShell, you will need the following configuration files and scripts:

1. A configuration parameter file in csv format to specify cluster names, logins, cluster-size (how many worker nodes are needed), node-sizes (size of the VMs to be used as head, edge and worker nodes), Azure subscription which is to be used for creating the clusters, etc. Windows PowerShell will use this as input for getting all the relevant parameters about the clusters to be created. File name: [ClusterParameters.csv](https://github.com/Azure/Azure-MachineLearning-DataScience/blob/master/Misc/KDDCup2016/Scripts/Configuration/ClusterConfigurationParameters.csv). Location to be specified in the script (see below).
2. An Azure resource deployment schema in json format for creating the clusters, also called an Azure resource management (ARM) template (see for [details](https://azure.microsoft.com/en-us/documentation/templates/101-hdinsight-rserver-ssh-password/)). The parameters from the CSV configuration file are passed along to this schema file to be used for cluster creation. File name: [azuredeploy.json](https://github.com/Azure/Azure-MachineLearning-DataScience/blob/master/Misc/KDDCup2016/Scripts/Configuration/azuredeploy.json)
3. A Windows PowerShell script which submits the job for creating individual clusters. File name: [SubmitHDICreation.ps1](https://github.com/Azure/Azure-MachineLearning-DataScience/blob/master/Misc/KDDCup2016/Scripts/CreateClusters/SubmitHDICreation.ps1)
4. A driver Windows PowerShell file which runs the jobs in parallel using the submission script (#3). File name: [RunHDICreation.ps1](https://github.com/Azure/Azure-MachineLearning-DataScience/blob/master/Misc/KDDCup2016/Scripts/CreateClusters/RunHDICreation.ps1)

In the public GitHub repository, first two files are in “Scripts/Configuration”, and 3rd and 4th files are in “Scripts/CreateClusters”.

Note that clusters will be created for login with SSH authentication with a [username / password](https://azure.microsoft.com/en-us/documentation/templates/101-hdinsight-rserver-ssh-password/).

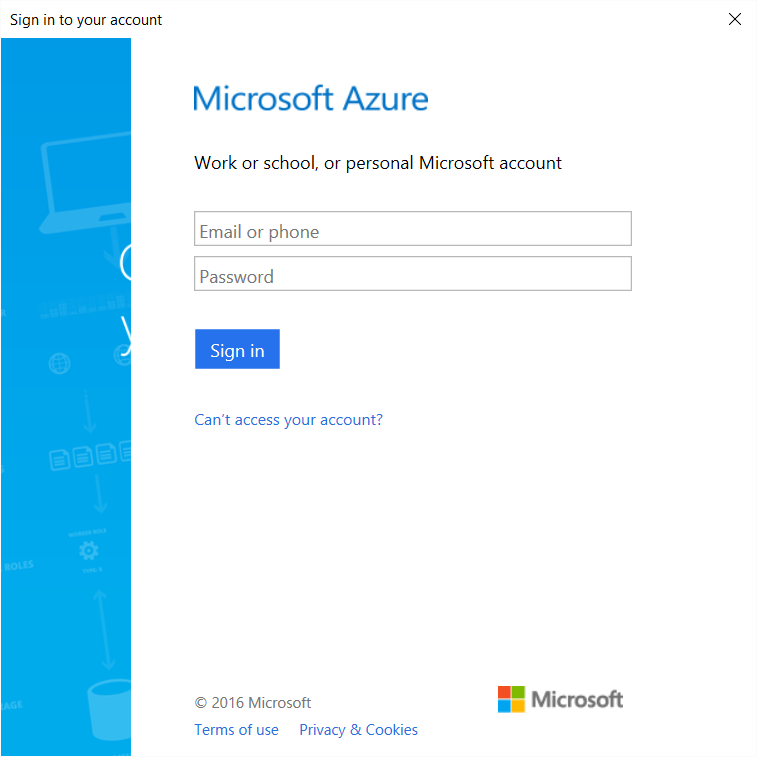
You can copy and paste the code in file: [RunHDICreation.ps1](https://github.com/Azure/Azure-MachineLearning-DataScience/blob/master/Misc/KDDCup2016/Scripts/CreateClusters/RunHDICreation.ps1). To create clusters, open RunHDICreation.ps1. Set the base path. Make sure path to parameter file (#1), as well as the [SubmitHDICreation.ps1](https://github.com/Azure/Azure-MachineLearning-DataScience/blob/master/Misc/KDDCup2016/Scripts/CreateClusters/SubmitHDICreation.ps1) file (#3), are set correctly. See below in Figure 1.



**Figure 1:** Script “[RunHDICreation.ps1](https://github.com/Azure/Azure-MachineLearning-DataScience/blob/master/Misc/KDDCup2016/Scripts/CreateClusters/RunHDICreation.ps1)”, used to submit cluster creation jobs. Note that when running the script, you will have to set the paths to the configuration file and script [SubmitHDICreation.ps1](https://github.com/Azure/Azure-MachineLearning-DataScience/blob/master/Misc/KDDCup2016/Scripts/CreateClusters/SubmitHDICreation.ps1) (shown in red boxes).

The first step is to login to your Azure subscription using the command “Login-AzureRmAccount” in [RunHDICreation.ps1](https://github.com/Azure/Azure-MachineLearning-DataScience/blob/master/Misc/KDDCup2016/Scripts/CreateClusters/RunHDICreation.ps1). When you do that, you will be prompted for login and password. You will need to save your Azure profile information in a file (defined by the $profilepath parameter in the parameter file), and then pass that profile to [SubmitHDICreation.ps1](https://github.com/Azure/Azure-MachineLearning-DataScience/blob/master/Misc/KDDCup2016/Scripts/CreateClusters/SubmitHDICreation.ps1) for submitting the batch jobs for cluster creation. This also applies to the subsequent steps which use batch job submission using the Azure resource and service management commandlets.





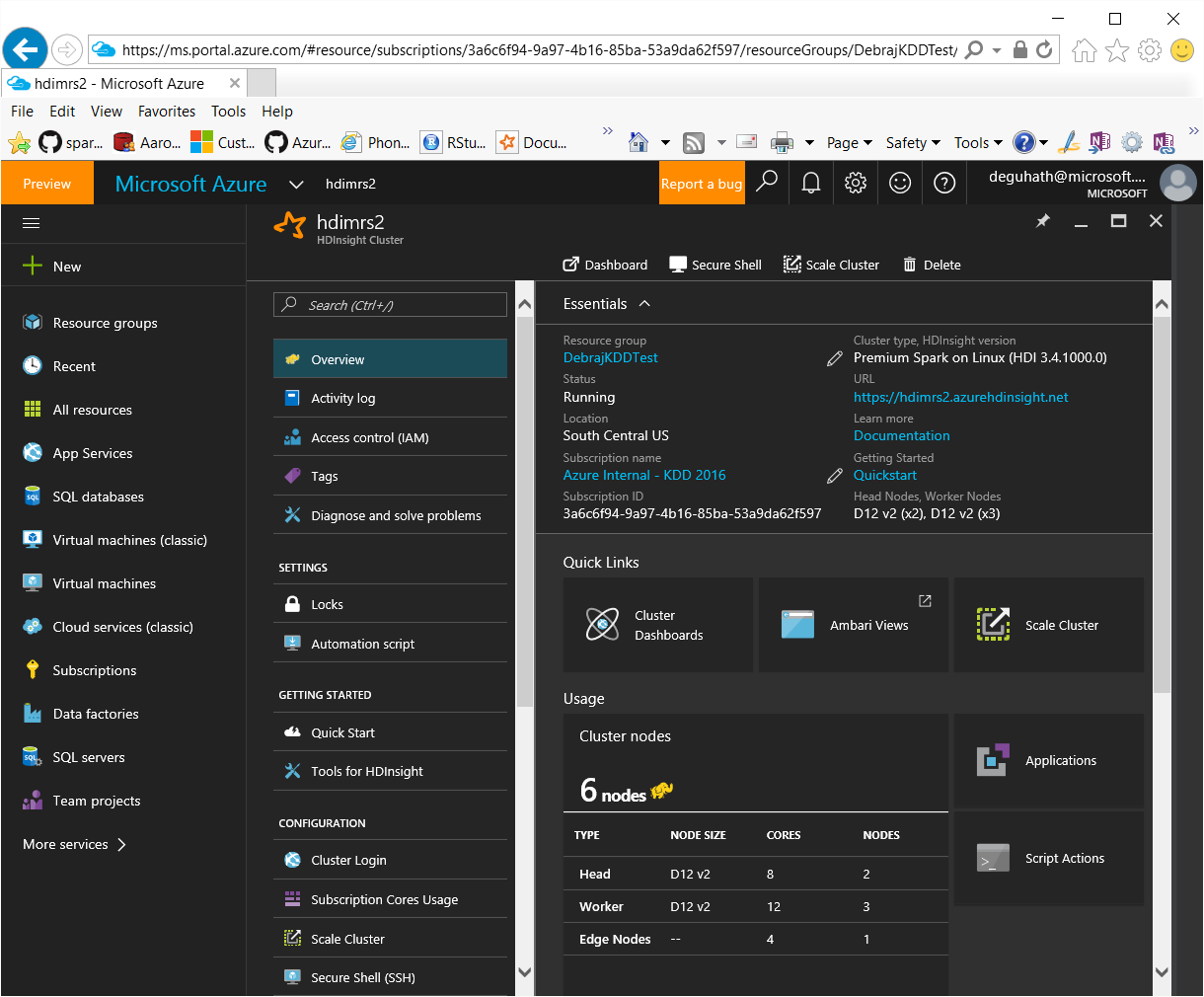
**Figure 2:** Pop-up window for logging into to Azure subscription and getting profile information.

Note that the clusters are currently set to be created using the VM size of “Standard\_D12\_v2”, as specified by the following line in the file [azuredeploy.json](https://github.com/Azure/Azure-MachineLearning-DataScience/blob/master/Misc/KDDCup2016/Scripts/Configuration/azuredeploy.json): "vmSize": "Standard\_D12\_v2".

If you want to change the vmSize – please set the vmSize of your choice in the json file. Details about VM sizes and prices are given [here](https://azure.microsoft.com/en-us/pricing/details/hdinsight/). Currently, the cost of running an HDInsight cluster with MRS is with 1 head node, 1 edge node (for MRS) and 2 worker nodes with Standard D12 V2 VM size is $4.20/hr (5 nodes, 20 cores).

Once the script is submitted, the clusters take about 20-25 mins to provision and be active for use. A blob storage account is created for every cluster. The name of the blob storage account associated with each cluster will be: name\_of\_cluter + “storage”. For example, for a cluster hdimrs1, the corresponding blob storage will be hdimrs1storage. This will serve as the HDFS for the cluster.

Once the cluster is ready to be used, you can navigate and examine it through the GUI on your Azure portal.



**Figure 3:** Dashboard for managing HDI Spark cluster on Azure portal. Once the cluster is created, you can access this dashboard for your cluster. You can change the size of the cluster and scale it up or down using the “scale cluster” tile.

## Customizing cluster with script actions: Installing software and R packages, and copying code files from GitHub by running script actions on cluster edge node

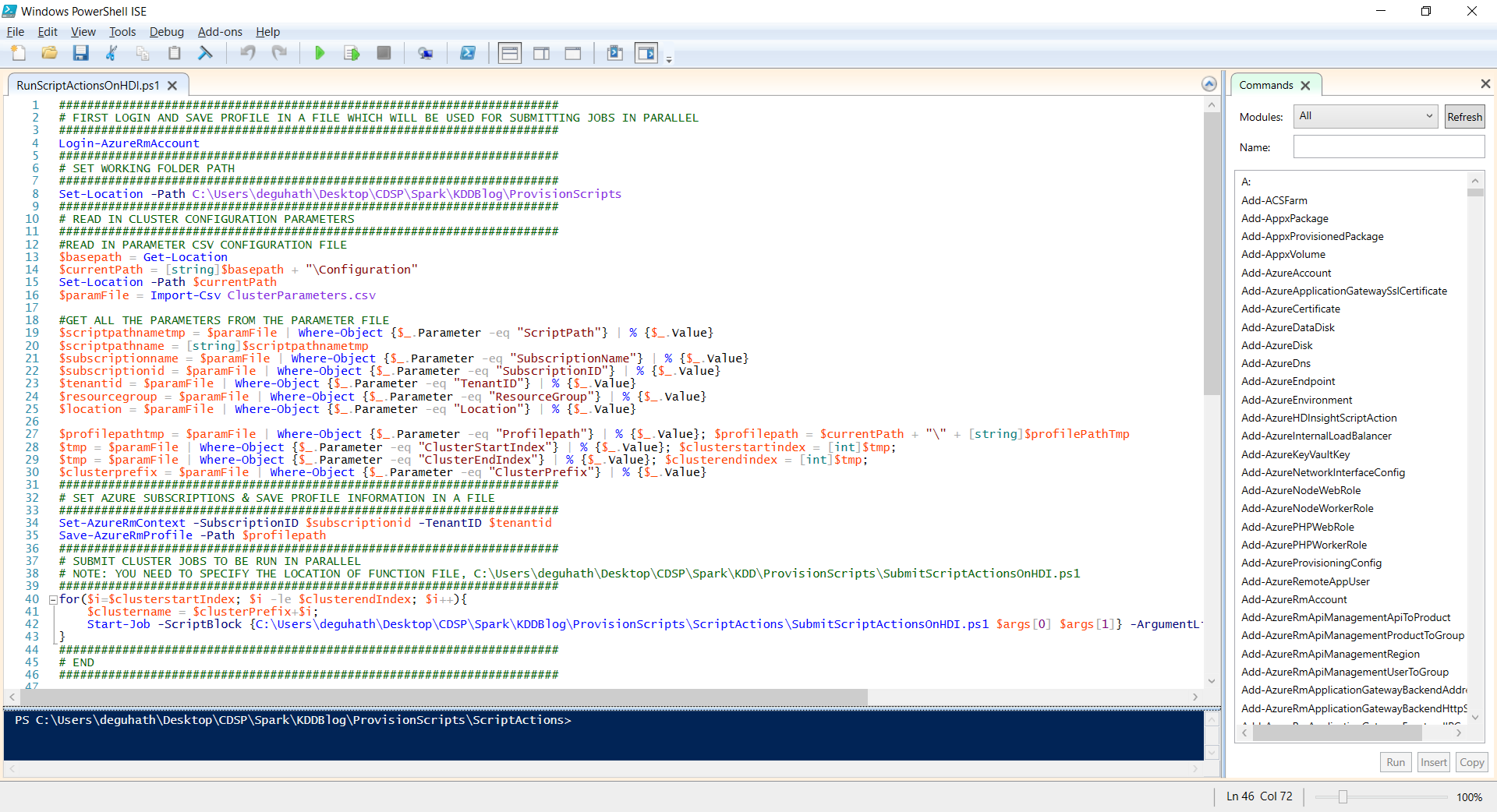
After the cluster is created, you can login to the r-server edge node using a client such as [putty](http://www.putty.org/) or [MobaXterm](http://mobaxterm.mobatek.net/). For example, for a cluster with name hdimrs1, the edge node will have address: r-server.hdimrs1-ssh.azurehdinsight.net. The ssh login and username are needed to login.

Script actions need be run on the edge nodes of the clusters to install necessary software (e.g. RStudio) and R packages, and copy code and script files from the GitHub repository, so that the exercises can run. Data files are copied from a public blob storage location. For more on how to customize HDInsight clusters using script actions, read [this](https://azure.microsoft.com/en-us/documentation/articles/hdinsight-hadoop-customize-cluster-linux/).

For preparing your clusters, you will need the following scripts, which are located in “Scripts/RunningScriptActions” in the public [GitHub repository](https://github.com/Azure/Azure-MachineLearning-DataScience/tree/master/Misc/KDDCup2016):

1. A parameter configuration file in csv format to specify cluster names, logins, Azure subscription which is to be used for creating the clusters, etc. Windows PowerShell will use this as input for getting all the relevant parameters about the clusters. File name: [ClusterParameters.csv](https://github.com/Azure/Azure-MachineLearning-DataScience/blob/master/Misc/KDDCup2016/Scripts/Configuration/ClusterConfigurationParameters.csv)
2. A Windows PowerShell script which submits the job for creating individual clusters. File name: [SubmitScriptActionsOnHDI.ps1](https://github.com/Azure/Azure-MachineLearning-DataScience/blob/master/Misc/KDDCup2016/Scripts/RunningScriptActions/SubmitScriptActionsOnHDI.ps1)
3. A driver Windows PowerShell file which runs the jobs in parallel using the submit script (#2). File name: [RunScriptActionsOnHDI.ps1](https://github.com/Azure/Azure-MachineLearning-DataScience/blob/master/Misc/KDDCup2016/Scripts/RunningScriptActions/RunScriptActionsOnHDI.ps1)

For execution, you can copy and paste scripts from [RunScriptActionsOnHDI.ps1](https://github.com/Azure/Azure-MachineLearning-DataScience/blob/master/Misc/KDDCup2016/Scripts/RunningScriptActions/RunScriptActionsOnHDI.ps1) on your Windows PowerShell command line. You will need to specify the location of your configuration file [ClusterParameters.csv], as well as [SubmitScriptActionsOnHDI.ps1](https://github.com/Azure/Azure-MachineLearning-DataScience/blob/master/Misc/KDDCup2016/Scripts/RunningScriptActions/SubmitScriptActionsOnHDI.ps1). If you have saved your Azure profile in a file earlier, you don’t need to login and save the profile again, you can use that profile file.

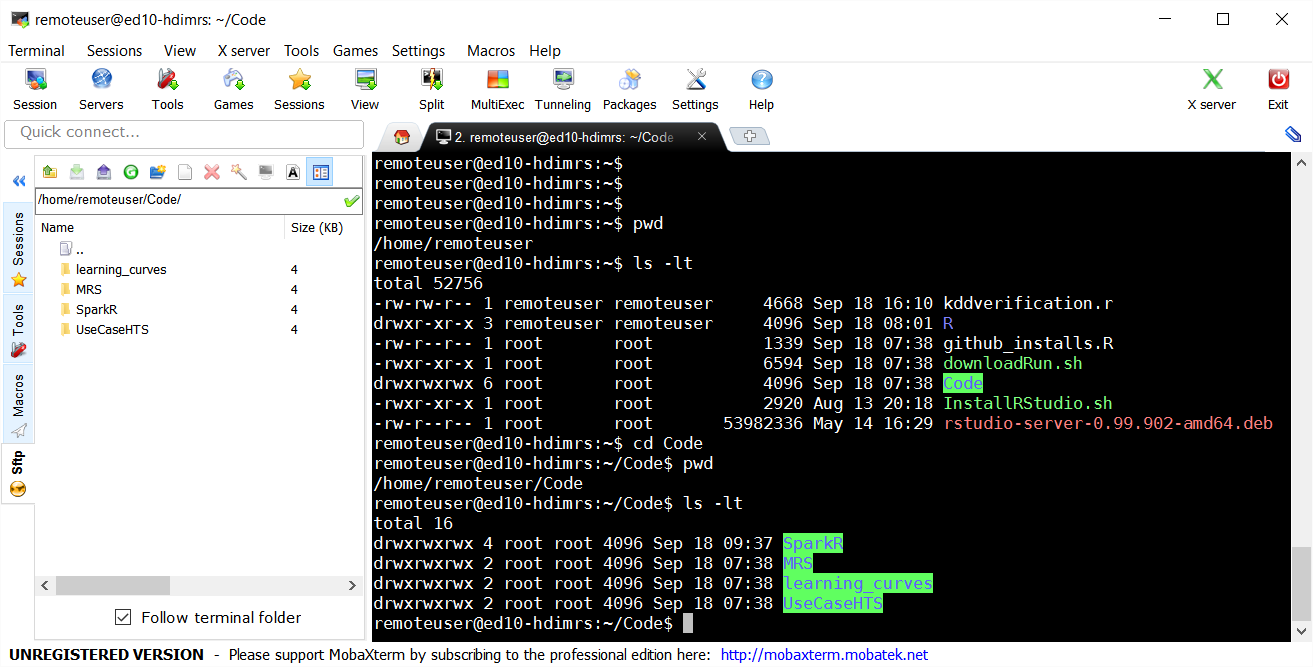


**Figure 4:** Script [RunScriptActionsOnHDI.ps1](https://github.com/Azure/Azure-MachineLearning-DataScience/blob/master/Misc/KDDCup2016/Scripts/RunningScriptActions/RunScriptActionsOnHDI.ps1) in Windows PowerShell. Locations where file paths have to be set are indicated in red boxes.

The scripts will download and use the following two files from GitHub:

1. An R file for necessary package installations from CRAN and GitHub. File name: [github\_installs.R](https://github.com/Azure/Azure-MachineLearning-DataScience/blob/master/Misc/KDDCup2016/Scripts/RunningScriptActions/github_installs.R)
2. A shell script file which calls the R-script file for installing R packages, downloading necessary files and installing RStudio. File name: [downloadRun.sh](https://github.com/Azure/Azure-MachineLearning-DataScience/blob/master/Misc/KDDCup2016/Scripts/RunningScriptActions/downloadRun.sh)

Once the scripts finish execution, you will observe several files/folders in your edge node /remoteuser/home folder:



**Figure 5:** Files and folders that are created on the edge node of a Spark HDI cluster running MRS after script actions are run on that node.

R code for running the hands-on exercises are in the “Code” folder. Brief descriptions of the code files in various folders are provided here:

**SparkR**: Code files in this directory show how to perform data manipulations using [SparkR](https://spark.apache.org/docs/1.6.2/sparkr.html) ([SparkR\_sparklyr\_NYCTaxi.Rmd](https://github.com/Azure/Azure-MachineLearning-DataScience/blob/master/Misc/KDDCup2016/Code/SparkR/SparkR_sparklyr_NYCTaxi.Rmd)), and perform featurization and ML model training using [sparklyr](http://spark.rstudio.com/) ([sparklyr\_NYCTaxi.Rmd](https://github.com/Azure/Azure-MachineLearning-DataScience/blob/master/Misc/KDDCup2016/Code/SparkR/sparklyr_NYCTaxi.Rmd)).

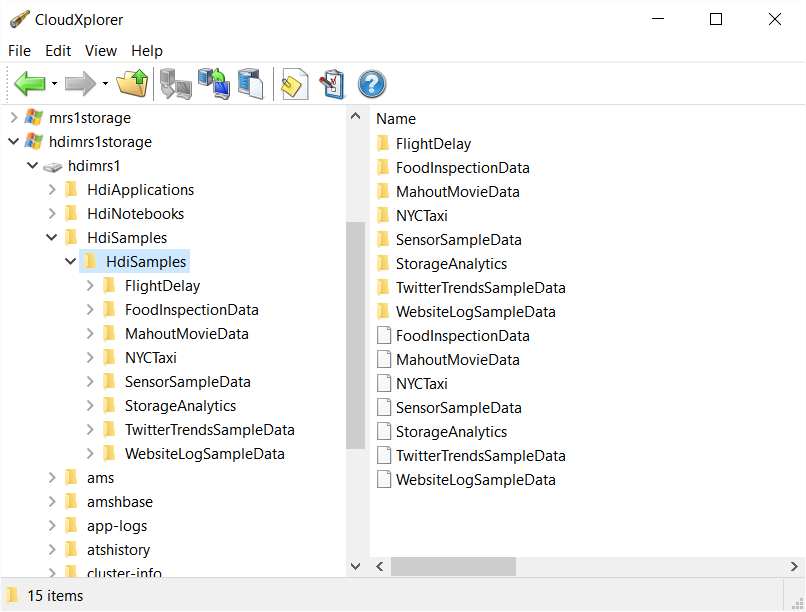
**MRS**: R code files showing how to use SparkR for pre-processing data, build ML models using MRS functions, deploy these models on [Azure ML](https://azure.microsoft.com/en-us/services/machine-learning/).

**learning\_curves**: R code files showing how to create learning curves on big data using MRS.

**UseCastHTS**: Contains a sample R script for the tutorial use case: Training parallel models for hierarchical time series optimization. This script uses Australian tourism data set from 'fpp' package to forecast quarterly visitor nights spent by international tourists to Australia. The quarterly historic data is available for years 1999-2010.

## Copying data files from a public blob storage using

Data-sets for the exercises are copied from a public Azure blob storage [http://cdspsparksamples.blob.core.windows.net/data] using Windows PowerShell script (CopyDataWithAzCopy.ps1) which uses [AzCopy](https://azure.microsoft.com/en-us/documentation/articles/storage-use-azcopy/) to copy files from the public blob storage to the HdiSamples folder in the cluster’s blob storage (HDFS). The figure below shows the files that are copied from the public blob storage. The public blob storage and the container for the files to be copied are specified in the configuration parameters file. You will need to specify the location of AzCopy in the CopyDataWithAzCopy.ps1. It is currently set to: 'C:\Program Files (x86)\Microsoft SDKs\Azure\AzCopy\AzCopy.exe'.





**Figure 6**: Folders that are copied to the cluster blob storage are shown for the example cluster hdimrs1.

Description of data-sets

## Verifying clusters for files and Spark compute environment

## Hands-on exercises

### Brief description of the exercises

#### SparkR and sparklyr

This tutorial describes how to use [SparkR](https://spark.apache.org/docs/1.6.2/sparkr.html) for data manipulation and use <sparklyr> for building ML models. We used the NY City taxi trip and fare data for this exercise. The [NYC Taxi data](https://azure.microsoft.com/en-us/documentation/articles/machine-learning-data-science-spark-overview/#the-nyc-2013-taxi-data) is about 20GB of compressed comma-separated values (CSV) files (~48GB uncompressed), comprising more than 173 million individual trips and the fares paid for each trip. Each trip record includes the pickup and drop-off location and time, anonymized hack (driver's) license number and medallion (taxi’s unique id) number. The data covers all trips in the year 2013 and is provided in the following two datasets for each month (Jan through Dec). Each fare record includes information about the fare amount, tip paid, taxes and tolls etc. We use the trip and fare files from December 2013. We join the data-sets and use the joined data-set to model the amount of tip paid (regression) using three different modeling approaches from [sparklyr](http://spark.rstudio.com/) (viz. regularized regression, random forest, boosted regression tree). Two R markdown files are provided:

[SparkR\_sparklyr\_NYCTaxi.Rmd](https://github.com/Azure/Azure-MachineLearning-DataScience/blob/master/Misc/KDDCup2016/Code/SparkR/SparkR_sparklyr_NYCTaxi.Rmd): Shows how to use SparkR for data manipulations and joining data-sets.

[sparklyr\_NYCTaxi.Rmd](https://github.com/Azure/Azure-MachineLearning-DataScience/blob/master/Misc/KDDCup2016/Code/SparkR/sparklyr_NYCTaxi.Rmd): Shows how to use sparklyr for training ML models.

#### MRS

#### SparkSQL

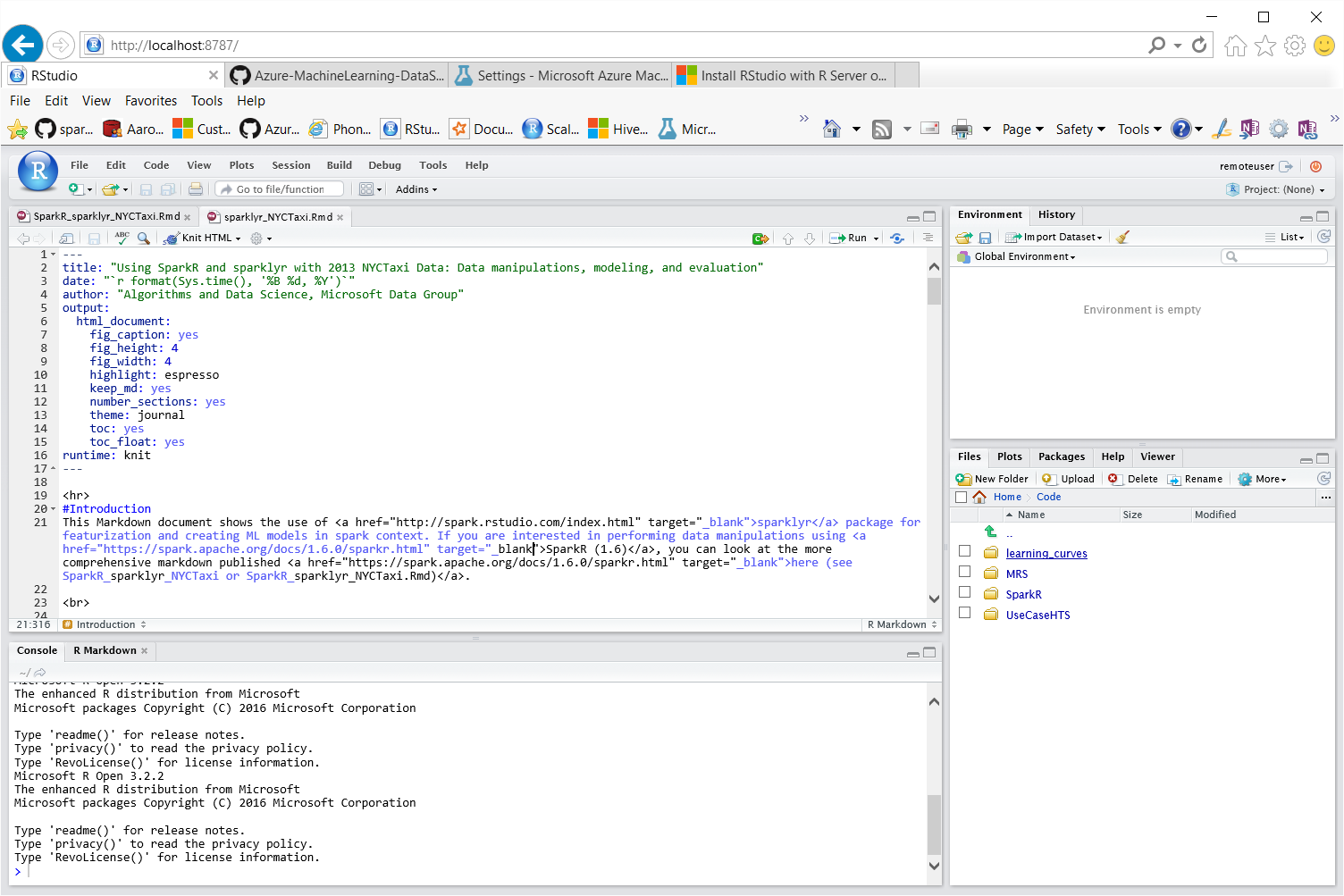
#### Use Case HTS

#### Learning curves

### How to execute code using RStudio server on cluster edge node

Instructions for logging into the RStudio server on the edge-node of the Spark cluster are given [here](https://azure.microsoft.com/en-us/documentation/articles/hdinsight-hadoop-r-server-install-r-studio/#install-rstudio-on-the-cluster-using-a-custom-script). RStudio is already installed on the edge node using script actions. So, you will have to follow instructions on how to login to the RStudio server and run R code. Code files are in GitHub in the “Code” folder.

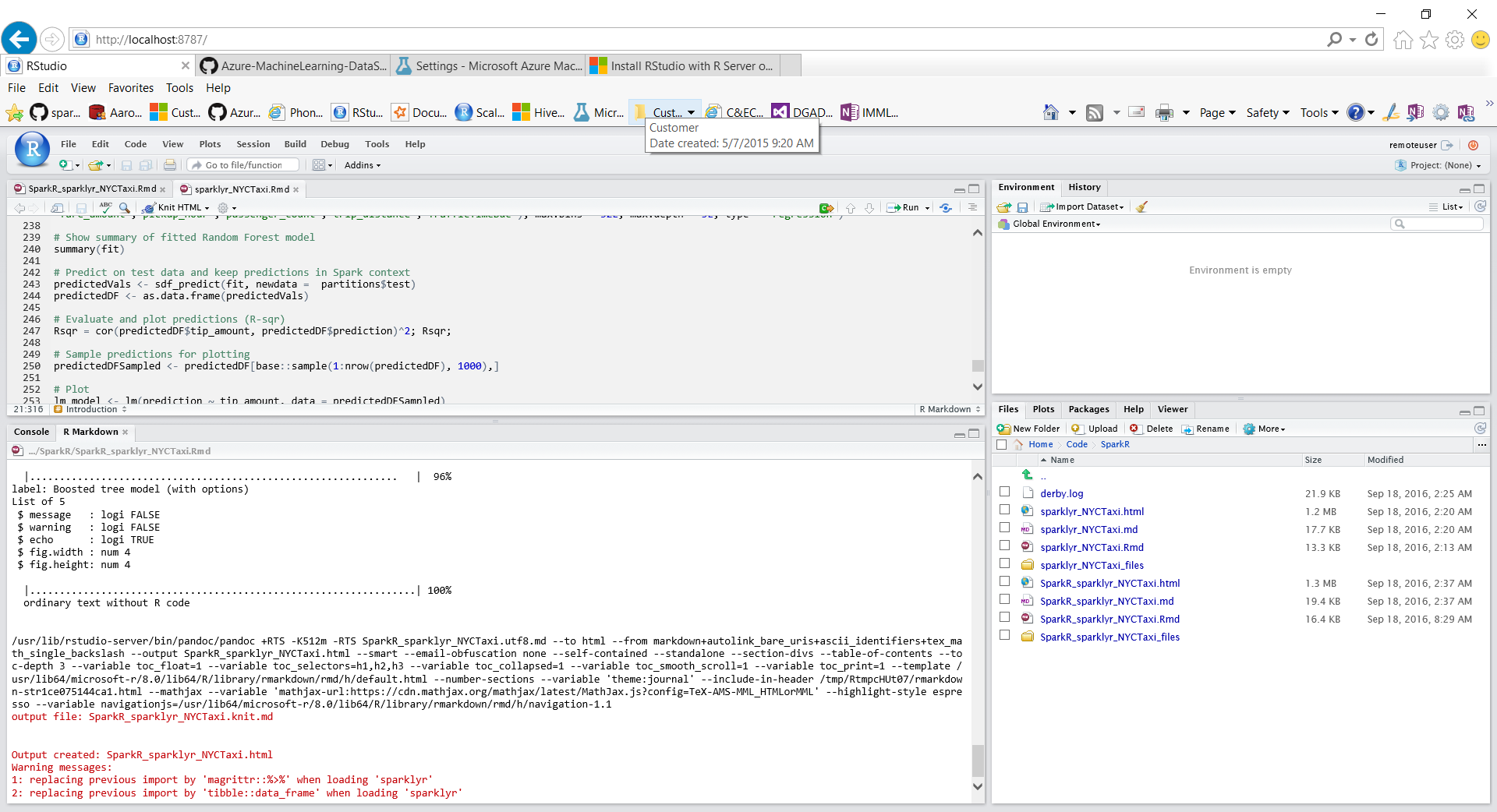
Once you login to the RStudio server, you will see the code files in different directories, as shown below:





**Figure 7:** R markdown file in RStudio. To execute the code in R markdown files (.Rmd), just click on “knit HTML”, and it will execute and produce an HTML output.

For R markdown files (.Rmd), you can click on ‘knit HTML”, and see the code run. After completion of the execution, and HTML output file will be produced. For example, see below.



**Figure 6:** RStudio server snapshot after a R markdown file is successfully executed. Note in the bottom right panel the HTML files which appeared after the code in the R markdown file fully executed.

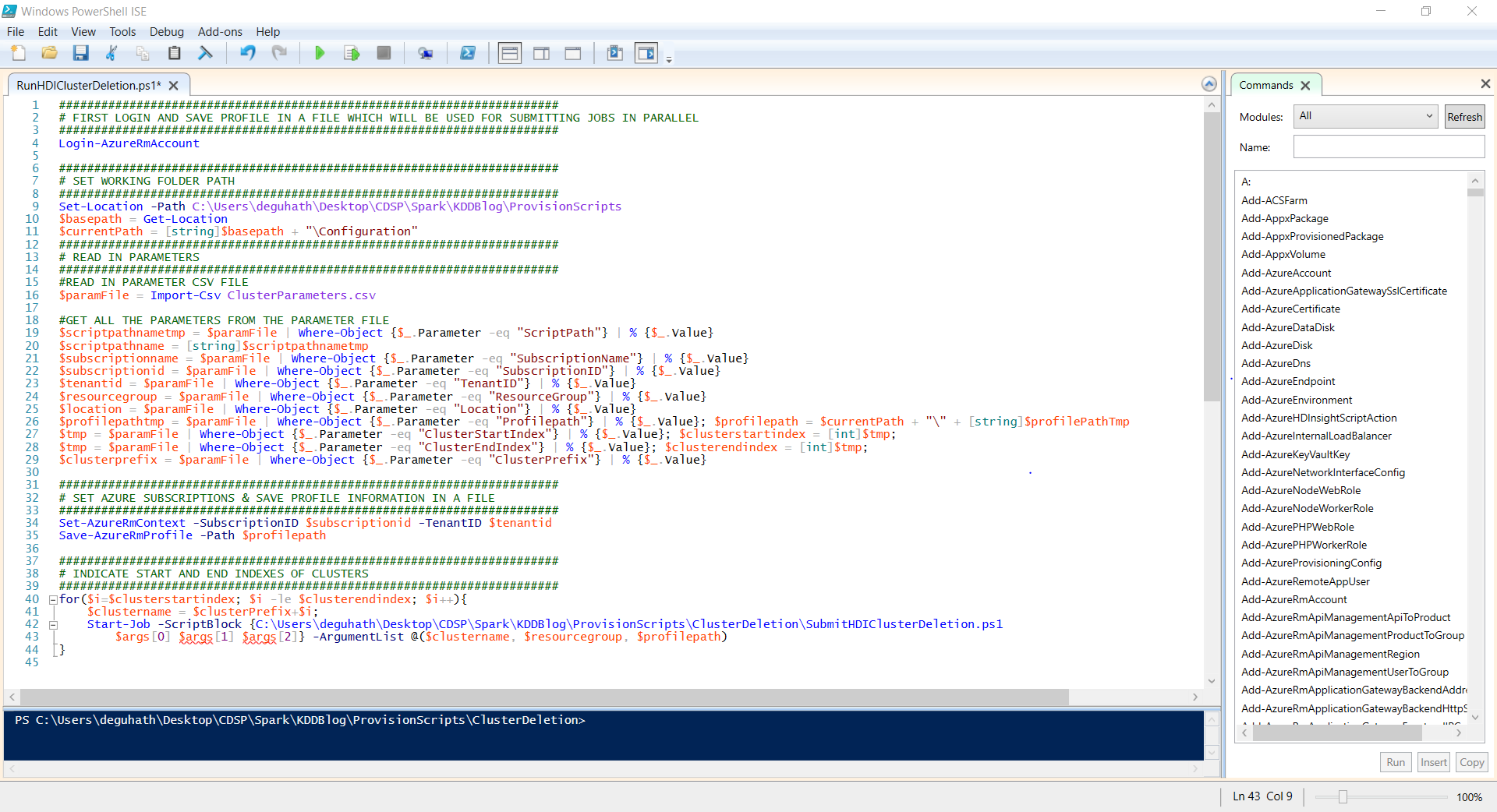
Depending on which file you are running, code can take 3-15 mins to finish. Code that are not in Rmd file, can be loaded in RStudio and executed in the Console.

## Deleting clusters and associated storage accounts

After you are done using the clusters, you can easily delete them using Windows PowerShell scripts and Azure commandlets. For cluster deletion, you will need the following scripts, which are located in “Scripts/DeleteClusters”.

You will need the following three files:

1. A parameter configuration file in csv format to specify cluster names, logins, Azure subscription which is to be used for creating the clusters, etc. Windows PowerShell will use this as input for getting all the relevant parameters about the clusters that are to be deleted. File name: [ClusterParameters.csv](https://github.com/Azure/Azure-MachineLearning-DataScience/blob/master/Misc/KDDCup2016/Scripts/Configuration/ClusterConfigurationParameters.csv)
2. A Windows PowerShell script which submits the job for deleting individual clusters. File name: [SubmitHDIClusterDeletion.ps1](https://github.com/Azure/Azure-MachineLearning-DataScience/blob/master/Misc/KDDCup2016/Scripts/DeleteClusters/SubmitHDIClusterDeletion.ps1)
3. A driver Windows PowerShell script, which reads in the cluster parameters from [ClusterParameters.csv](https://github.com/Azure/Azure-MachineLearning-DataScience/blob/master/Misc/KDDCup2016/Scripts/Configuration/ClusterConfigurationParameters.csv), runs the deletion jobs in parallel using the submit script (#2). File name: [RunHDIClusterDeletion.ps1](https://github.com/Azure/Azure-MachineLearning-DataScience/blob/master/Misc/KDDCup2016/Scripts/DeleteClusters/RunHDIClusterDeletion.ps1)



**Figure 7:** Script [RunHDIClusterDeletion.ps1](https://github.com/Azure/Azure-MachineLearning-DataScience/blob/master/Misc/KDDCup2016/Scripts/DeleteClusters/RunHDIClusterDeletion.ps1) in Windows PowerShell. Locations where file paths have to be set are indicated in red boxes.

Clusters can take 15-20 mins to get deleted. Check Azure portal to make sure clusters are deleted.

## Summary

In this document we have provided the instructions for provisioning, configuring and deleting Spark HDI Premium clusters running MRS, as well as instructions for running the R codes that were provided for the hands-on exercises in the KDD 2016 tutorial on Scalable R on Spark.

We sincerely appreciate feedback. We would find it very helpful if users try our scripts and provide feedback, especially if parts of the documents are not clearly explained, or if some scripts are not working as expected. That will help us improve our documentation and scripts for future users.