

# Introducing Microsoft Data Science Virtual Machine

Rate this article ★★★★★



Vanja P (<https://social.msdn.microsoft.com/profile/Vanja+P>) November 21, 2016

 0 (<https://blogs.msdn.microsoft.com/intel/archives/835#respond>)

[Share 3](#) 7 0

Microsoft Data Science Virtual Machine (DSVM) is a custom virtual machine on Microsoft's Azure cloud build specifically for doing data science. It's a powerful data science development sandbox equipped with the most popular tools for data exploration and modelling. You can provision a DSVM with a few clicks on Microsoft Azure website (<https://azure.microsoft.com/en-us/marketplace/partners/microsoft-ads/standard-data-science-vm/>), and within 10-20 minutes you will have a running virtual machine to which you can connect via the standard Windows Remote Desktop Connection tool.

DSVM runs on a Windows Server 2012 and contains the following data science and development tools (taken from the product page ([## Follow Us](https://azure.microsoft.com/en-us/marketplace/partners/microsoft-ads/standard-data-science-</a></p></div><div data-bbox=)



(<https://blogs.msdn.microsoft.com/>)

## Popular Tags

- dynamics (<https://blogs.msdn.microsoft.com/>)
- Machine learning (<https://blogs.msdn.microsoft.com/learning>)
- Cortana Intelligence (<https://blogs.msdn.microsoft.com/intelligence>)
- advanced analytics (<https://blogs.msdn.microsoft.com/advanced-analytics>)
- Dynamics 365 (<https://blogs.msdn.microsoft.com/365>)

vm/)): "The main tools include **Microsoft R Server** Developer Edition, **Anaconda Python** distribution, **Jupyter notebooks for Python and R**, **Visual Studio** Community Edition with Python, R and node.js tools, **Power BI** desktop, **SQL Server 2016** Developer edition. It also includes **deep learning tools like CNTK** (an Open Source Deep Learning toolkit from Microsoft) and **mxnet**; **ML algorithms** like **xgboost**, **Vowpal Wabbit**. The **Azure SDK** and libraries on the VM allows you to build your applications using various services in the cloud that are part of the **Cortana Analytics Suite** which includes **Azure Machine Learning**, **Azure Data Factory**, **Azure Stream Analytics** and **SQL Data Warehouse**, **Hadoop**, **Azure Data Lake**, **Spark** and **more**." There is also an equivalent Linux based DSVM (<https://azure.microsoft.com/en-us/marketplace/partners/microsoft-ads/linux-data-science-vm/>).

In this post, we will briefly describe how to provision a data science VM, then go over a simple use case of exploring a large order demand data set using Microsoft R Server. At the end of the post we provide several helpful resources for further reading.

## Provisioning a Data Science VM

To provision a DSVM, go to Azure portal (<https://portal.azure.com/>), and search for New -> Data Science Virtual Machine, and select Create (as shown in the picture below). This will take you through a configuration wizard, asking for inputs, such as Azure subscription and resource group, VM account credentials, VM size, etc. After completing the wizard and selecting the Buy button, it takes 10-20 minutes for the VM to be created and ready for use.

Dynamics AX  
(<https://blogs.msdn.microsoft.com/ax>)

AX2012  
(<https://blogs.msdn.microsoft.com/ax2012>)

AX  
(<https://blogs.msdn.microsoft.com/ax>)

Cortana  
(<https://blogs.msdn.microsoft.com/cortana>)

manufacturing  
(<https://blogs.msdn.microsoft.com/manufacturing>)

distribution  
(<https://blogs.msdn.microsoft.com/distribution>)

retail  
(<https://blogs.msdn.microsoft.com/retail>)

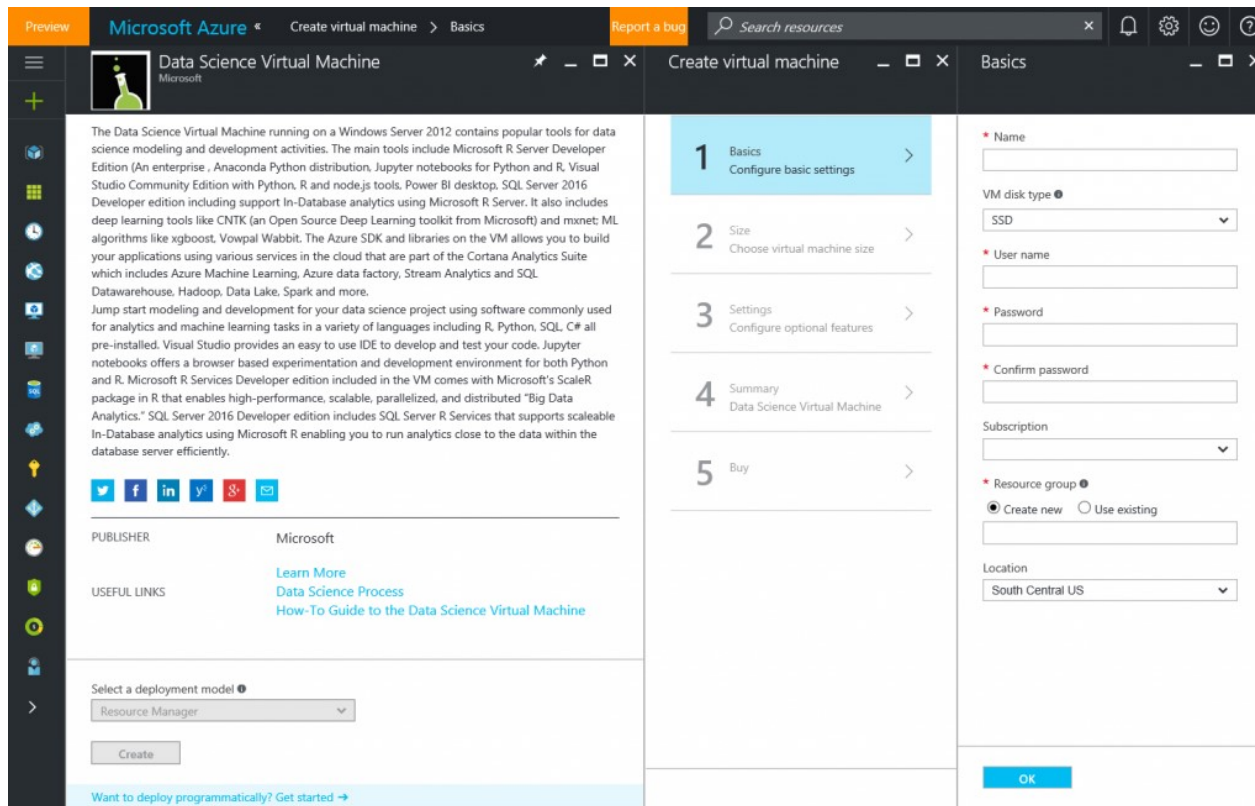
d365  
(<https://blogs.msdn.microsoft.com/d365>)

XAPT  
(<https://blogs.msdn.microsoft.com/xapt>)

Azure ML  
(<https://blogs.msdn.microsoft.com/azureml>)

ML  
(<https://blogs.msdn.microsoft.com/ml>)

CIS  
(<https://blogs.msdn.microsoft.com/cis>)



([https://msdnshared.blob.core.windows.net/media/2016/11/dsvm\\_provisioning.png](https://msdnshared.blob.core.windows.net/media/2016/11/dsvm_provisioning.png))

## Microsoft R Server

Of particular interest to data scientists who use R for common data science tasks on large data will be Microsoft R Server (MRS) that comes with DSVM. Microsoft R Server is enterprise-class distribution of R that allows for scalable data wrangling and modeling for data that does not fit into available RAM. MRS stores the data in external data frames (xdf) limited only by space on disk in a binary file format that is optimized to work with the MRS libraries. It retrieves data in blocks for efficient reading of arbitrary columns and contiguous rows, and optimizes the block sizes depending on individual computer I/O bandwidth. This results in significant speed-ups with respect to interactive data wrangling and modeling.

predictive maintenance  
(<https://blogs.msdn.microsoft.com/maintenance>)

forecasting  
(<https://blogs.msdn.microsoft.com/forecasting>)

demand forecasting  
(<https://blogs.msdn.microsoft.com/demand-forecasting>)

## Archives

February 2017

(<https://blogs.msdn.microsoft.com/february-2017>)

December 2016

(<https://blogs.msdn.microsoft.com/december-2016>)

November 2016

(<https://blogs.msdn.microsoft.com/november-2016>)

October 2016

(<https://blogs.msdn.microsoft.com/october-2016>)

All of 2017

(<https://blogs.msdn.microsoft.com/all-of-2017>)

All of 2016

(<https://blogs.msdn.microsoft.com/all-of-2016>)

To develop R code with Microsoft R Server, we used RStudio (<https://www.rstudio.com/products/RStudio/>), a popular IDE for R, which comes pre-installed on the DSVM.

*NOTE: To make sure that RStudio loads MRS (instead of MRO – a more limited distribution of MRS), go to Tools -> Global Options -> General, and change the R Version to point to C:\Program Files\Microsoft SQL Server\130\R\_SERVER.*

Other popular environments for developing R code available on DSVM are R Tools for Visual Studio (<https://www.visualstudio.com/vs/rtvs/>) and Jupyter Notebooks (<https://jupyter.org/>).

## Example use case: exploration of product demand data

In demand forecasting scenarios, the data is commonly very large, with millions of records indicating shipment records, customer orders, or consumption data. If we wanted to explore the data, look at the variable types, distributions, or visualize the demand over time for hints about trends, seasonality, etc., we might run into issues with the base R version, as the data might be larger than the available memory. Let's look at one such example below, where we used Microsoft R Server to explore and visualize such large data.

We downloaded our data set in a .csv format from a sharepoint site onto the VM. One can easily connect to an external data source (sharepoint, Azure blob, Azure SQL database, etc.), and access the data directly from MRS, or download it onto the VM, and access it locally.

Let's import the data into the .xdf format.

```
in.file <- file.path('C:/shared/demand_dataset.csv')
xdf.file <- "demand_dataset.xdf"

# Read data from csv
dataset <- rxImport(inData = in.file,
                    outFile = xdf.file,
                    overwrite = TRUE,
                    type = "text",
                    missingValueString = "N/A",
                    stringsAsFactors = TRUE)
```

Let's look at the number of rows and columns in our dataset.

```
> dim(dataset)
[1] 41034690    20
```

It's a relatively large data set with 41,034,690 orders and 20 variables describing those orders.

To get information about the variables in the data set, such as variable names, data types, low/high values for numeric variables, unique values and number of unique values for categorical variables, etc., we can run the following command:

```
# Get variable info
rxGetVarInfo(dataset)
```

Due to the privacy of the data, we omit the output here.

Let's look at the summary statistics for the volume variable in our data set.

```

> # Get summary stats
> rxSummary(~ volume, data = dataset)

Rows Processed: 41034690
Call:
rxSummary(formula = ~volume, data = dataset, reportProgress = 1)

Summary Statistics Results for: ~volume
Data: dataset (RxxdfData Data Source)
File name: demand_dataset.xdf
Number of valid observations: 41034690

Name      Mean      StdDev  Min Max      ValidObs MissingObs
volume    48214.28 219421.2 0   135384480 41034690 0

```

We can see that the mean order volume was 48,214.28, and the largest order had 135,384,480 volume.

Let's aggregate the volume data and plot it over time. In order to aggregate the data, we will take advantage of R package `dplyrXdf` (<https://github.com/RevolutionAnalytics/dplyrXdf>), which allows for streamlined and simplified data manipulation.

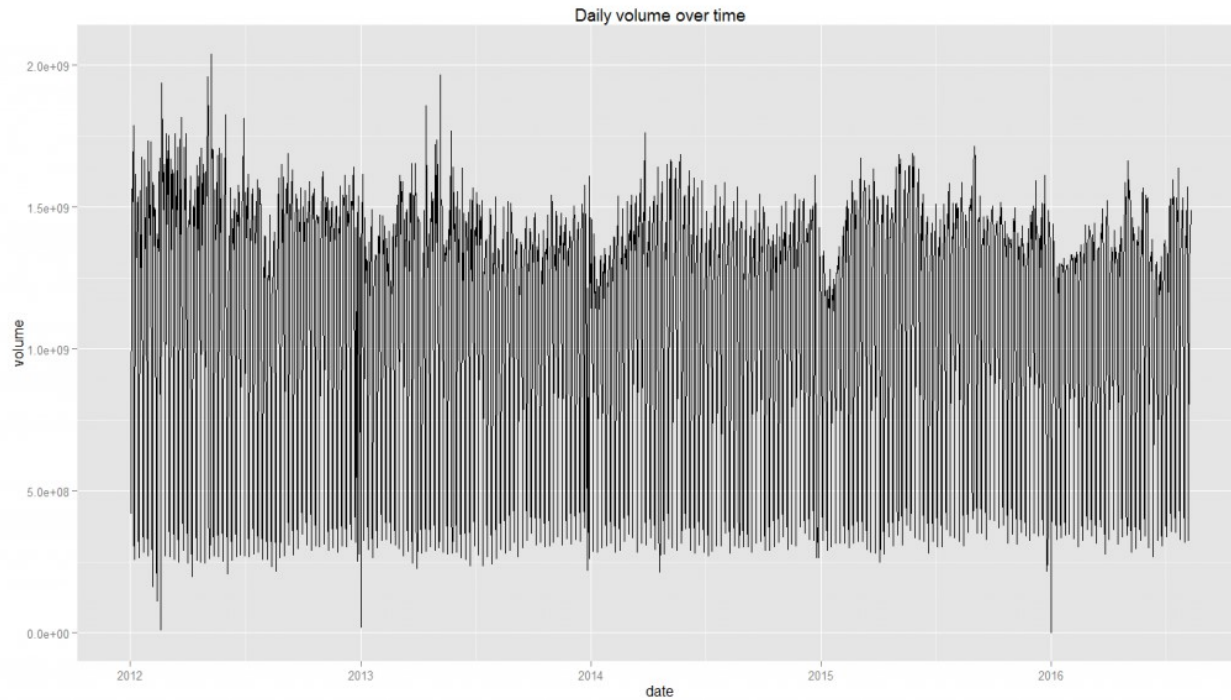
```

# Load dplyrXdf package
library(dplyrXdf)

# Aggregate volume over time (per yyyy-mm-dd)
volume.by.ymd <- dataset %>%
  select(date, volume) %>%
  group_by(date) %>%
  summarise(volume = sum(volume)) %>%
  as.data.frame

# Plot the aggregated data
ggplot(volume.by.ymd, aes(date, volume)) + geom_line() + ggtitle("Daily volume over time")

```

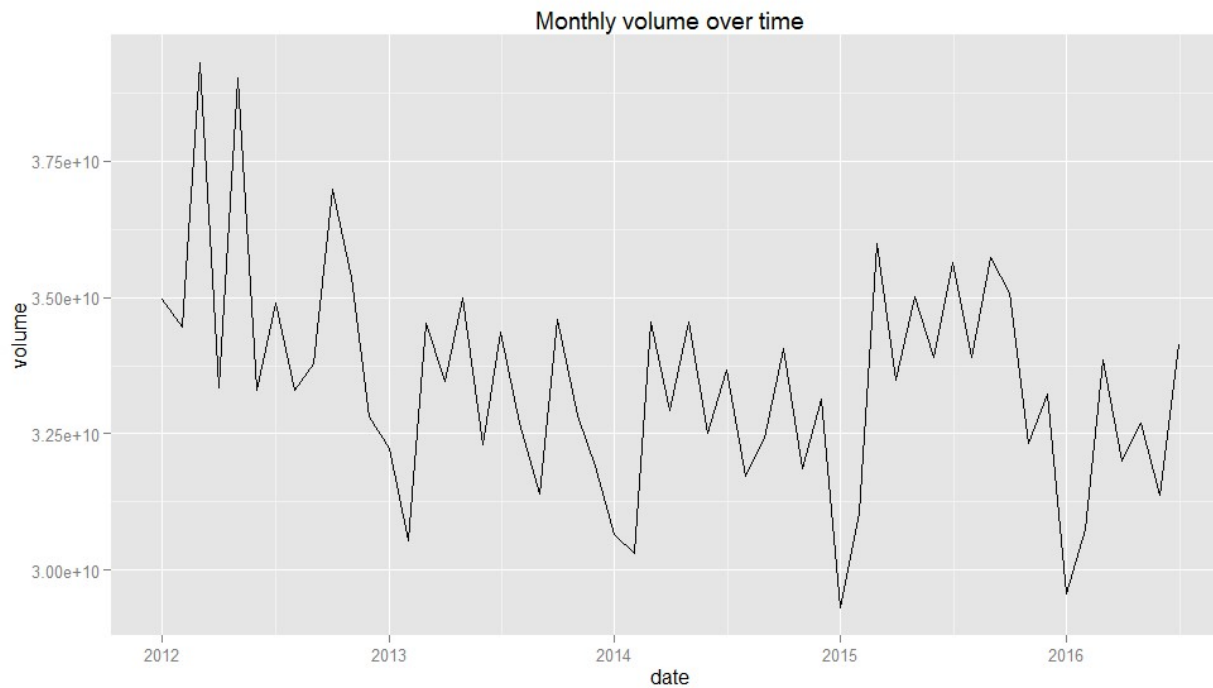


(<https://msdnshared.blob.core.windows.net/media/2016/11/>,

This plot may be too dense to visually inspect and detect any trends in the daily plot, so let's look at the data aggregated at the monthly level.

```
# Aggregate volume over time (per yyyy-mm)
volume.by.ym <- volume.by.ymd %>%
  select(date, volume) %>%
  mutate(date = as.Date(as.yearmon(date))) %>%
  group_by(date) %>%
  summarise(volume = sum(volume)) %>%
  as.data.frame

# plot the aggregated data
ggplot(volume.by.ym, aes(date, volume)) + geom_line() + ggtitle("Monthly volume over
time")
```



([https://msdnshared.blob.core.windows.net/media/2016/11/monthly\\_volume.png](https://msdnshared.blob.core.windows.net/media/2016/11/monthly_volume.png))



This plot is much clearer and it's easier to see trends and seasonality. There is a decreasing trend in the data in early years, potentially reflecting some changes in data reporting. We can also see a clear seasonal pattern in the data with the major volume dips around the end of each year, reflecting patterns around the winter holidays.

We have shown a simple use case of how to explore a large demand data set using Microsoft R Server on Data Science Virtual Machine. Within less than an hour, we were able to provision a virtual machine with MRS already installed (and other popular data science tools), and gain valuable insights from our data without a need for any deep statistics or machine learning skills.

## Further Reading

For more information on what's included in the DSVM, please see links provided under References. Ten common data science tasks you can do on the DSVM are detailed in this document. (<https://azure.microsoft.com/en-us/documentation/articles/machine-learning-data-science-vm-do-ten-things/>) It walks you through how to use DSVM to perform common data science task, such as—data exploration and modeling using Python, Microsoft R Server, and Jupyter Notebooks, operationalizing your model using Azure ML, sharing your code via Github, and visualizing your reports using Power BI.

DSVM usage is charged Azure usage fees only (monthly usage starts at \$76.63), and depends on the size of the provisioned virtual machine. There are no additional software charges for the proprietary software installed on the VM. More information on the pricing can be found here (<https://azure.microsoft.com/en-us/marketplace/partners/microsoft-ads/standard-data-science-vm/>).

## References

1. Introduction to the cloud-based Data Science Virtual Machine for Linux and Windows (<https://azure.microsoft.com/en-us/documentation/articles/machine-learning-data-science-virtual-machine-overview/>)
2. Data Science Virtual Machine – Product page (<https://azure.microsoft.com/en-us/marketplace/partners/microsoft-ads/standard-data-science-vm/>)
3. Provision the Microsoft Data Science Virtual Machine (<https://azure.microsoft.com/en-us/documentation/articles/machine-learning-data-science-provision-vm/>)

4. Ten things you can do on the Data science Virtual Machine (<https://azure.microsoft.com/en-us/documentation/articles/machine-learning-data-science-vm-do-ten-things/>)
5. Microsoft R Server (<https://www.microsoft.com/en-us/cloud-platform/r-server>)
6. dplyrXdf – R package for xdf data manipulation (<https://github.com/RevolutionAnalytics/dplyrXdf>)

Author: Vanja Paunić, Data Scientist, Microsoft

Tags [advanced analytics](https://blogs.msdn.microsoft.com/intel/archives/tag/advanced-analytics) (<https://blogs.msdn.microsoft.com/intel/archives/tag/advanced-analytics>) [Azure VM](https://blogs.msdn.microsoft.com/intel/archives/tag/azure-vm) (<https://blogs.msdn.microsoft.com/intel/archives/tag/azure-vm>) [CIS](https://blogs.msdn.microsoft.com/intel/archives/tag/cis) (<https://blogs.msdn.microsoft.com/intel/archives/tag/cis>) [Cortana Intelligence](https://blogs.msdn.microsoft.com/intel/archives/tag/cortana-intelligence) (<https://blogs.msdn.microsoft.com/intel/archives/tag/cortana-intelligence>) [data science](https://blogs.msdn.microsoft.com/intel/archives/tag/data-science) (<https://blogs.msdn.microsoft.com/intel/archives/tag/data-science>) [data science virtual machine](https://blogs.msdn.microsoft.com/intel/archives/tag/data-science-virtual-machine) (<https://blogs.msdn.microsoft.com/intel/archives/tag/data-science-virtual-machine>) [demand forecasting](https://blogs.msdn.microsoft.com/intel/archives/tag/demand-forecasting) (<https://blogs.msdn.microsoft.com/intel/archives/tag/demand-forecasting>) [dplyrXdf](https://blogs.msdn.microsoft.com/intel/archives/tag/dplyrxdf) (<https://blogs.msdn.microsoft.com/intel/archives/tag/dplyrxdf>) [DSVM](https://blogs.msdn.microsoft.com/intel/archives/tag/dsvm) (<https://blogs.msdn.microsoft.com/intel/archives/tag/dsvm>) [microsoft r server](https://blogs.msdn.microsoft.com/intel/archives/tag/microsoft-r-server) (<https://blogs.msdn.microsoft.com/intel/archives/tag/microsoft-r-server>) [r server](https://blogs.msdn.microsoft.com/intel/archives/tag/r-server) (<https://blogs.msdn.microsoft.com/intel/archives/tag/r-server>)

---

## Comments (0)

Name \*

Email \*

Website

Post Comment

Privacy & Cookies (<https://msdn.microsoft.com/dn529388>)  
Terms of Use (<https://msdn.microsoft.com/cc300389>)  
Trademarks (<https://www.microsoft.com/en-us/legal/intellectualproperty/1>)  
© 2017 Microsoft