

In this exercise, you will learn how to provision, connect to, and use a DSVM and explore the use of Jupyter notebooks, a popular web-based data science integrated development environment (IDE). You will see that a Data Science VM is a powerful learning tool with software, sample code, and data to demonstrate how to use many of the Azure data and AI services.

## Learning objectives

In this exercise, you will:

- Create an Azure Data Science Virtual Machine using the Azure portal
- Connect to an Azure Data Science Virtual Machine using a Remote Desktop connection
- Learn about the Jupyter Notebook
- Create a Jupyter Notebook
- Learn about the learning tools that are pre-installed on a Data Science Virtual Machine To create an Azure DSVM, you need an Azure subscription. You can create an Azure free account that includes \$200 credit to spend for the first 30 days, free access to Azure's most popular products for 12 months and access to more than 25 products that are always free. (See [Get Azure free trial](#)).

After signing into your account, go to [the DSVM product page](#). Scroll down until you see the DSVM editions as shown below.


### DSVM editions

#### Data Science Virtual Machine - Windows 2016

Exploration, analysis, modeling, and development tools for data science

Software plans start at

**Free**




#### Data Science Virtual Machine for Linux (Ubuntu)

Virtual machine with deep learning frameworks and tools for machine learning and data science

Software plans start at

**Free**




#### Deep Learning Virtual Machine


A deep learning toolkit for the data science virtual machine

Software plans start at

**Free**



Click on the Data Science Virtual Machine – Windows 2016 link. You should then see the screen below.

 Microsoft

Azure Marketplace

Search Marketplace

More







Products > Data Science Virtual Machine - Windows 2016



# Data Science Virtual Machine - Windows 2016

Microsoft

[Overview](#) [Plans](#) [Reviews](#)

Development and modeling tools for AI, data science and analytics

The 'Data Science Virtual Machine (DSVM)' is a 'Windows Server 2016 with Containers' VM & includes popular tools for data exploration, analysis, modeling & development.

Highlights:

- Azure Machine Learning SDK
- Microsoft ML Server - Dev Edition (Scalable R & Python)
- Anaconda Python
- SQL Server 2017 Dev. Edition - With In-Database R and Python analytics
- Microsoft Office 365 ProPlus BYOL - Shared Computer Activation
- Julia Pro + Juno Editor

GET IT NOW


SAVE FOR LAT...

**Categories**  
[Databases](#)  
[Analytics](#)

**Legal**  
[License Agreement](#)  
[Privacy Policy](#)

When you see the screen below, click on the Continue button.

## Create this app in Azure



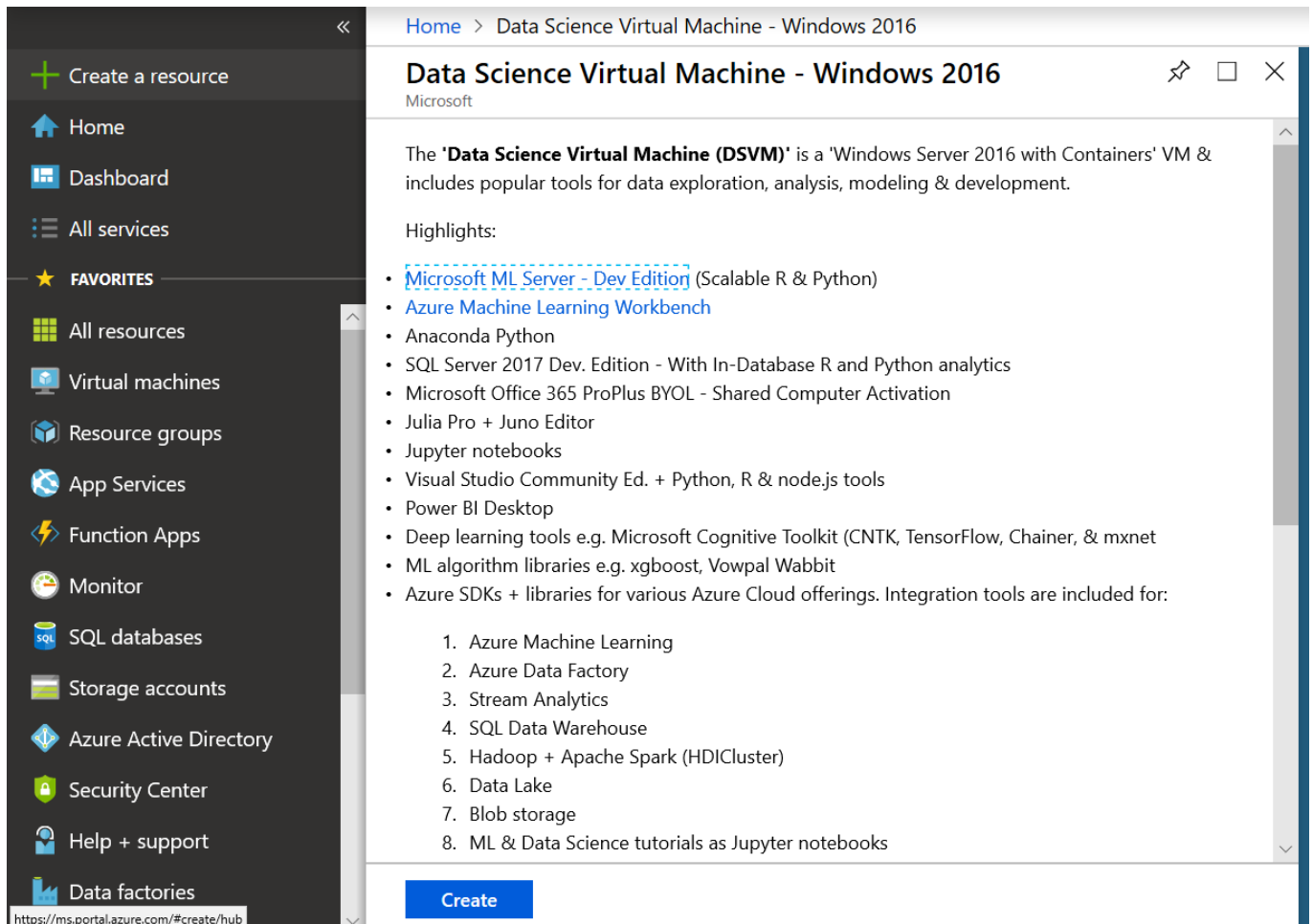
**Data Science Virtual Machine - Windows 2016**  
By Microsoft

**Software plan**  
**Data Science Virtual Machine - Windows 2016**  
Details: Development and modeling tools for AI, data science and analytics

I agree to the provider's [terms of use](#) and [privacy policy](#) and understand that the rights to use this product do not come from Microsoft, unless Microsoft is the provider. Use of Azure Marketplace is governed by separate [terms](#).

Continue

You will be logged into the Azure portal and shown the screen to provision a Data Science Virtual Machine – Windows 2016 as shown below. If you are prompted for your credentials, just enter then and continue.



Click the blue button labeled "Create" to start to configure a VM. You need to fill out the following fields:

- **Subscription.** If you have more than one subscription, select the one on which the machine is to be created and billed.
- **Resource group.** A resource group is a container that holds related resources for an Azure solution. You can create a new one or use an existing group.
- **Virtual Machine name.** The name of the Data Science server you are creating. Win2016 will be used in the example.
- **Region.** Select the data center that's most appropriate. For fastest network access, it's the data center that has most of your data or is closest to your physical locations.
- **Image:** Do not change this. It should say "Data Science Virtual Machine – Windows 2016".
- **Size.** Select one of the server types that meets your functional requirements and cost constraints.
- **VM Disk Type.** Choose SSD (Solid-State Drive) or HDD (Hard Disk Drive).
- **User Name.** The administration account Id to access the VM. You will need this later to access the VM.
- **Password.** The administration account password. You will need this later to access the VM.
- **Already have a Windows license?** Leave this as the default which is No.
- **Use Managed Disks.** Choose **Managed** if you want Azure to manage the disks for the VM. If not, you need to specify a new or existing storage account.

After filling out the fields, click the Disks button. Select the type of disks you want and click on the "Review and Create" button. Don't worry about the section stating **Subscription credits apply**. You will **not** be charged. You may have to wait a few seconds before Review and Create screen appears. Click on the "Create" button and the data science virtual machine will start being provisioned.



The VM provisioning takes a few minutes. You should see a blue line under the bell, icon which indicates that Azure is busy building the VM. When the VM is done being created, the VM main properties screen will appear. Whenever you need to access the VM, you can find it from the Azure portal. In the left panel, click "Virtual Machines" tab, you should see the DSVM you just created. Click on the name of the DSVM which will bring up the Overview screen.

Microsoft Azure

Search resources, services, and docs

Home > Virtual machines

Virtual machines

Default Directory

+ Add

⌚ Reservations

≡ Edit columns

↺ Refresh

⬢ Assign tags

Subscriptions: Free Trial

Filter by name...

All resource groups

All types

1 items

	NAME ↑↓	TYPE ↑↓	STATUS
<input type="checkbox"/>	Win2016	Virtual machine	Running

Then click on "Connect".

Home > Virtual machines > Win2016

Win2016

Virtual machine

Search (Ctrl+ /)

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

⚡ Connect

▶ Start

↺ Restart

■ Stop

📷 Capture

Resource group (change)

DSVMFreeTrial

Status

Running

Location

East US

Subscription (change)

Free Trial

You can open the RDP (Remote Desktop) file directly. Or you can download it and then open it with Remote Desktop Connection app. When the connection asks for the credentials, use the admin account name and password for creating the DSVM.

# Connect to virtual machine



Win2016V2

RDP **SSH**

To connect to your virtual machine via RDP, select an IP address, optionally change the port number, and download the RDP file.

\* IP address

Public IP address (52.175.211.118)

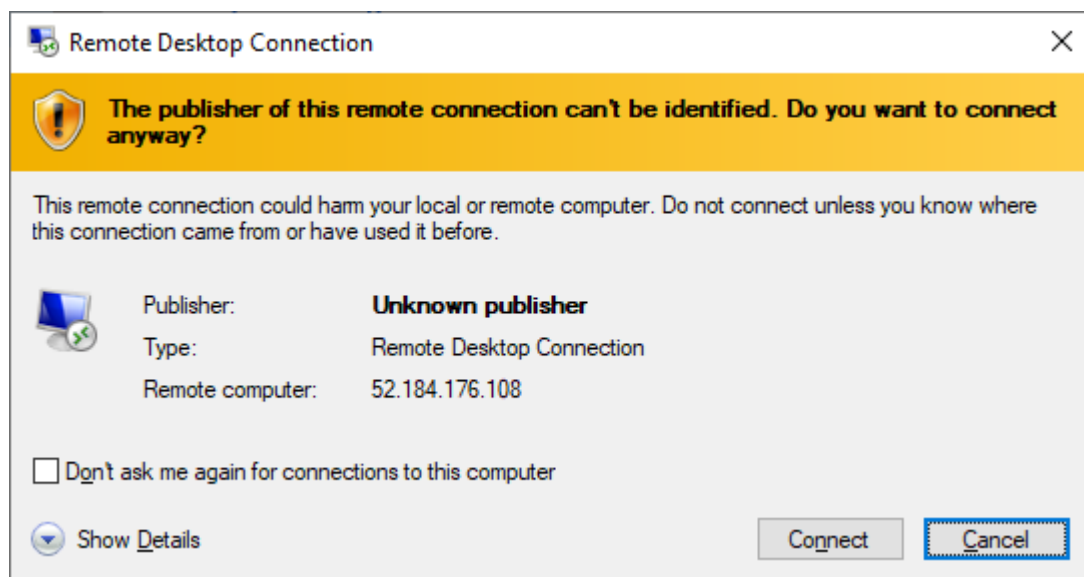


\* Port number

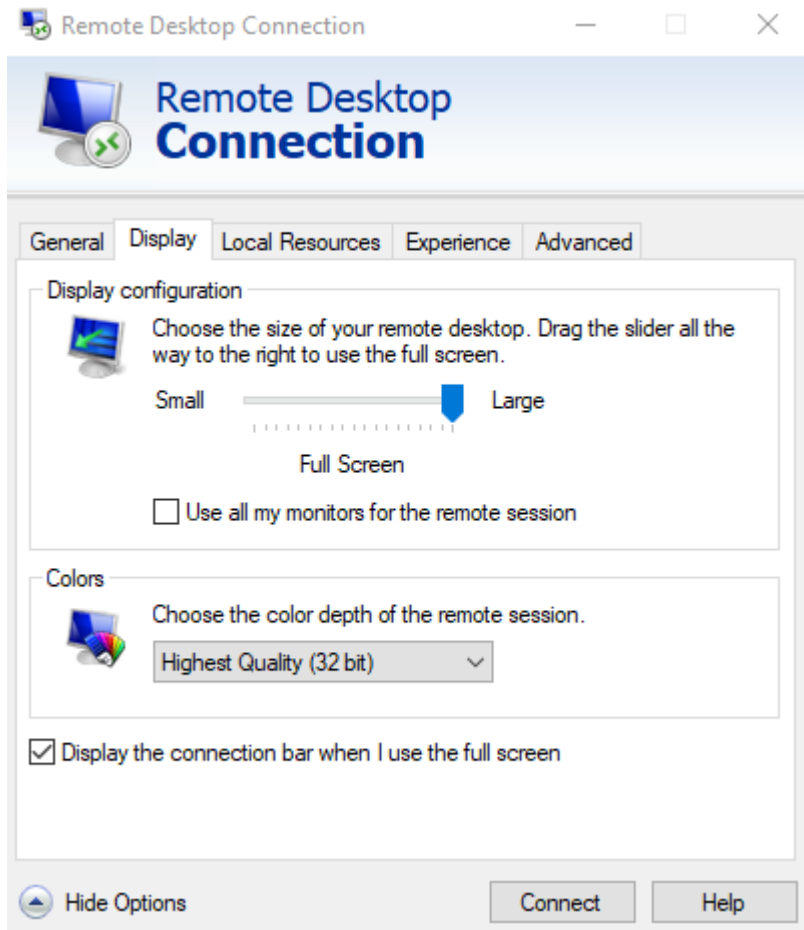
3389

**Download RDP File**

On the Remote Desktop Connection dialog box, click on **Connect**. Then enter the administrator user id and password you used when you created the VM.

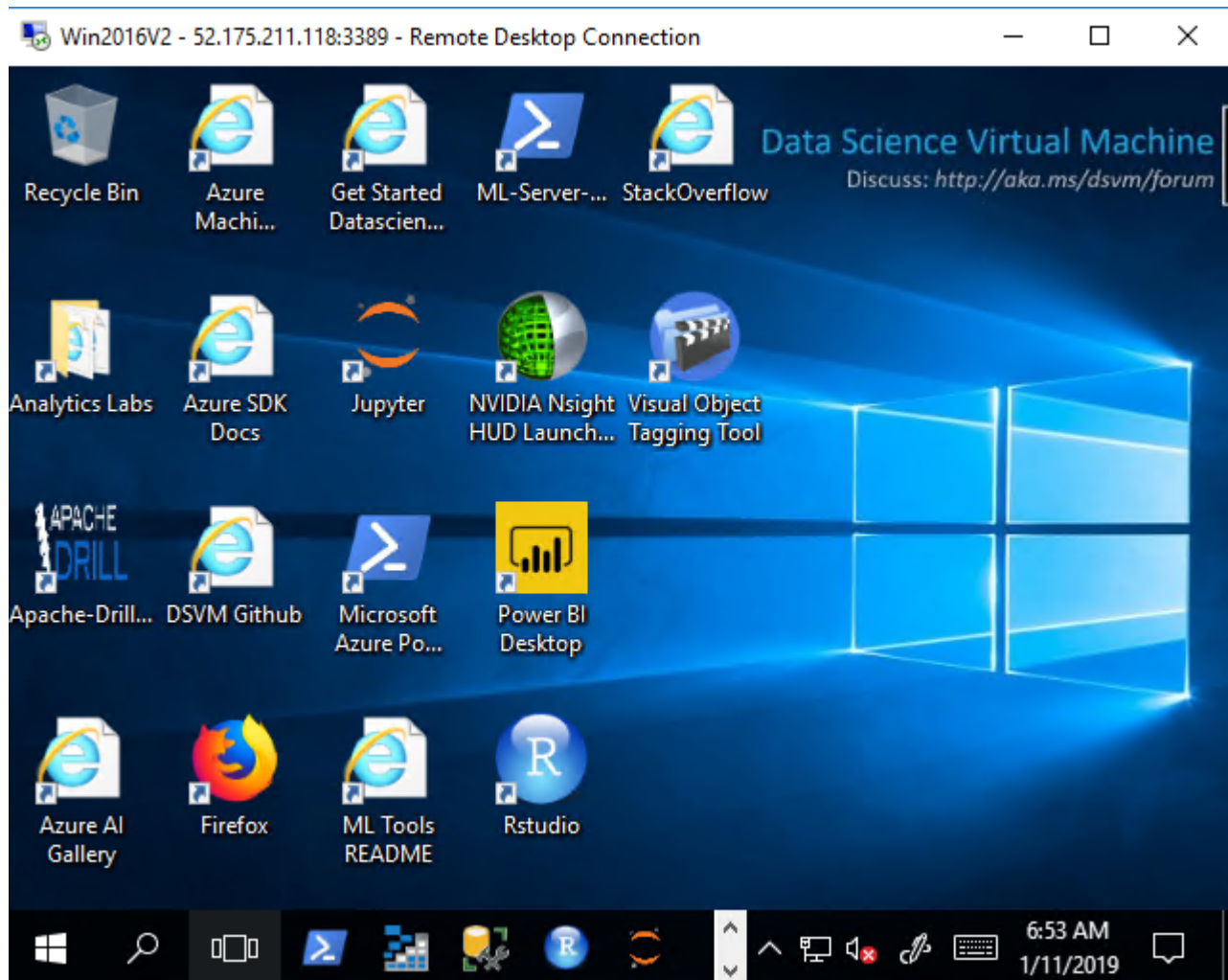


![IMPORTANT] When the screen resolution on your client machine is high, you may get a black screen. If that happens, use Remote Desktop Connection to open the RDP file, and before connecting, click the Show Options link, click the Display tab and set the Display Configuration to a lower resolution as shown below.



The first time you log into the VM, it runs some scripts that install demonstration code and data. Wait for a few minutes to let this compete before continuing.

Now you are ready to start using the DSVM with data science tools installed and configured. Many of the tools have start menu tiles and desktop icons.



The

Jupyter Notebook is a popular browser-based integrated development environment (IDE) for Data Science. You can use Jupyter Notebooks to explore data and train models. It supports both R and Python programming languages. Specifically, Python 2, Python 3, Open Source R, and the Microsoft R Server. Although notebooks support rich annotations, they are programs that can execute complete data science pipelines.

## Why use Jupyter Notebooks?

### Presentation

The Jupyter Notebook integrates code, its output, and explanatory text all into a single document. Comparing to a traditional code editor, it is great for presentation because the file can contain visualizations, tables, narrative texts, equations, etc.

### Collaboration

You can export a Jupyter Notebook to different formats. By downloading it as an HTML file, you can share the results with a small group of people. Alternatively, you can save the notebook to its native format, with the .ipynb file extension, and upload to GitHub for a more direct sharing.

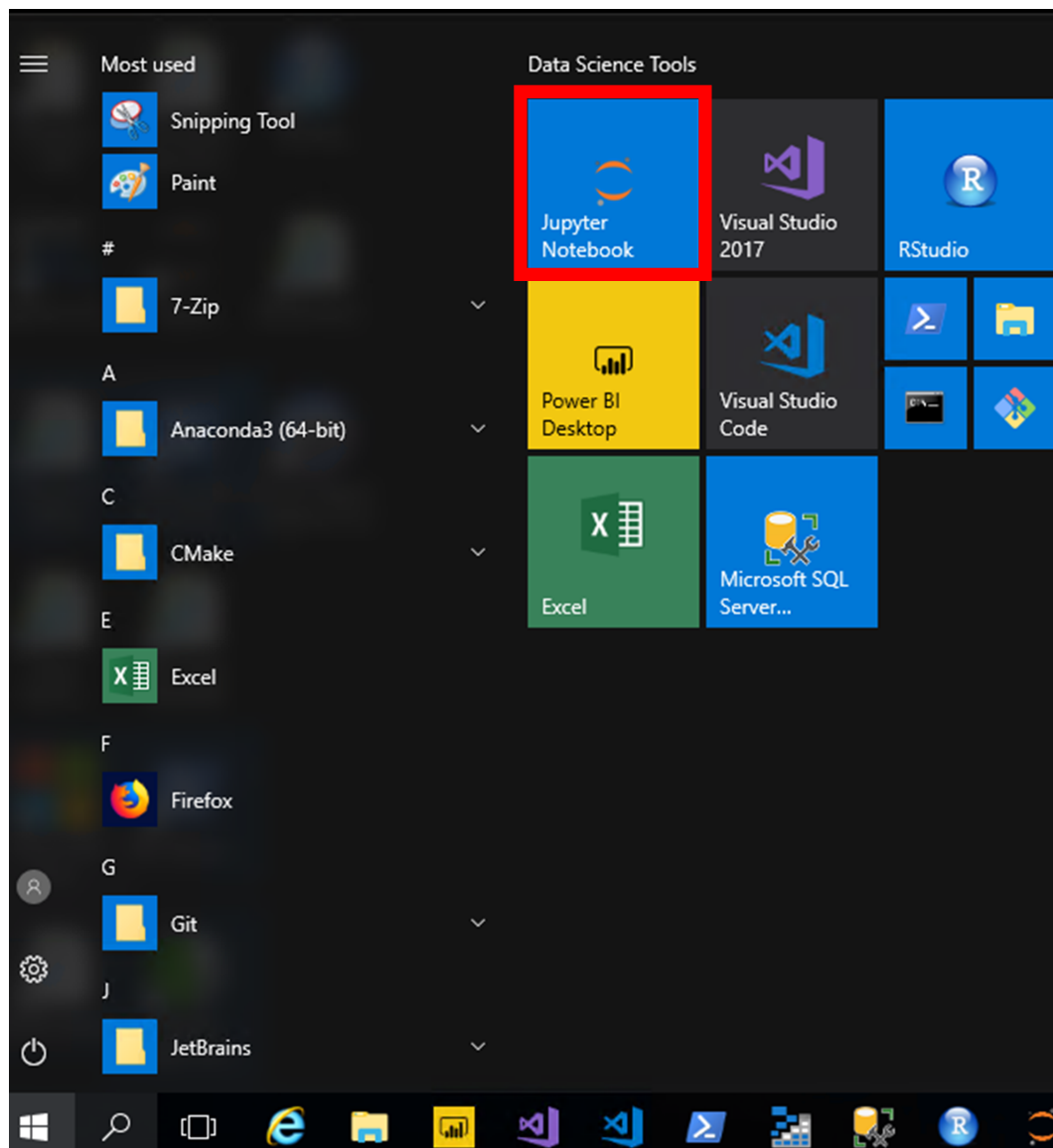
### Exploratory analysis

The development process on Jupyter Notebook is very interactive. You write a small piece of code, observe the output immediately, and write another piece of code. This is consistent with what data scientists do in their day-to-day work because data science is an iterative and exploratory process.

## Many Languages Supported

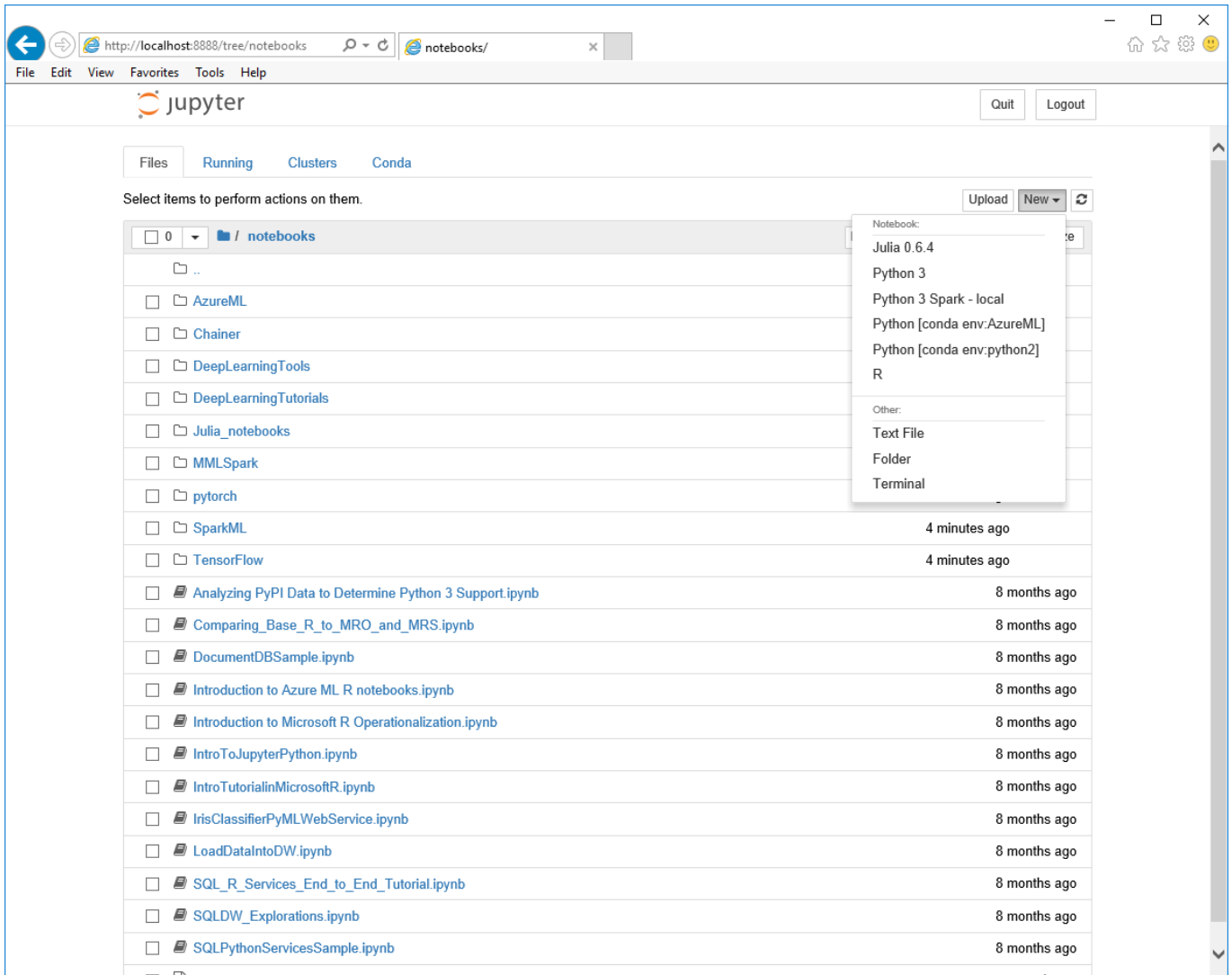
Jupyter Notebooks use a kernel-based architecture which separates the language interpreter from the notebook. Any language that has a Jupyter language kernel can be used in a notebook. All popular data science languages including Python, R, Julia, and Java have kernels available. The Jupyter Notebook is a machine learning development environment for virtually any language.


To Use Jupyter Notebook, click on the start menu icon:



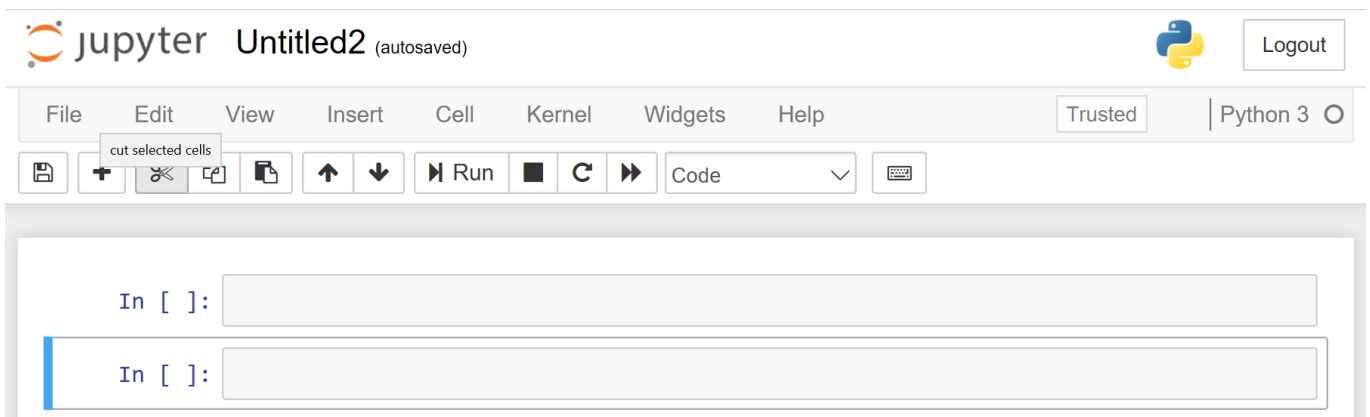
The interface of Jupyter Notebook is intuitive. Under the Files tab, you can see the list of folders and files in the current directory. Select the location where you want to create your notebook. Then click **New** on the right upper corner to create a new notebook and select a language (Python 3 in this example). Notice several language kernels are installed for you but you can install more if you like.



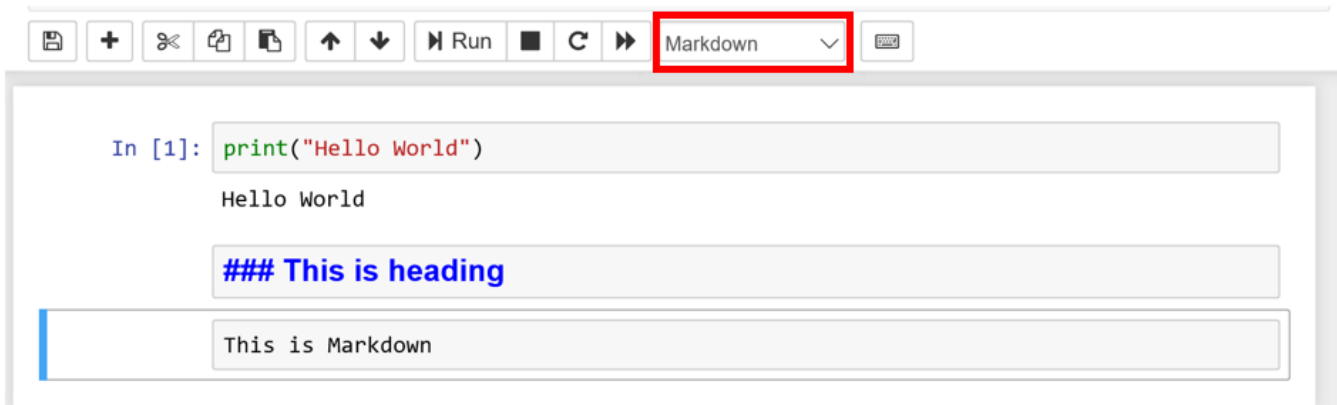


A new notebook is created, and you can see an empty code cell waiting for your input. You can run, add, remove, or edit the cells. The toolbar buttons allow you to do various operations on selected cells, for example, delete, copy, move, run, etc. There are two commonly used keyboard shortcuts listed below, and you can get a full list of keyboard shortcuts by clicking on the keyboard icon in the toolbar, . Alternatively, you can use the drop-down menus to perform actions on your notebook.

- Ctrl + Enter: to run the selected cell
- Shift + Enter: to run the selected cell, move to the next cell when the selected cell is not the last one, otherwise add a new cell.



By default, a cell is executed in the "kernel", which is a computational engine that executes the code contained in the cell. Alternatively, you could change the cell type to be "Markdown" or "Header", which are used annotations and formatted titles. Note: The "Header" type is just a Markdown cell that has '##' characters added to format it as a header.



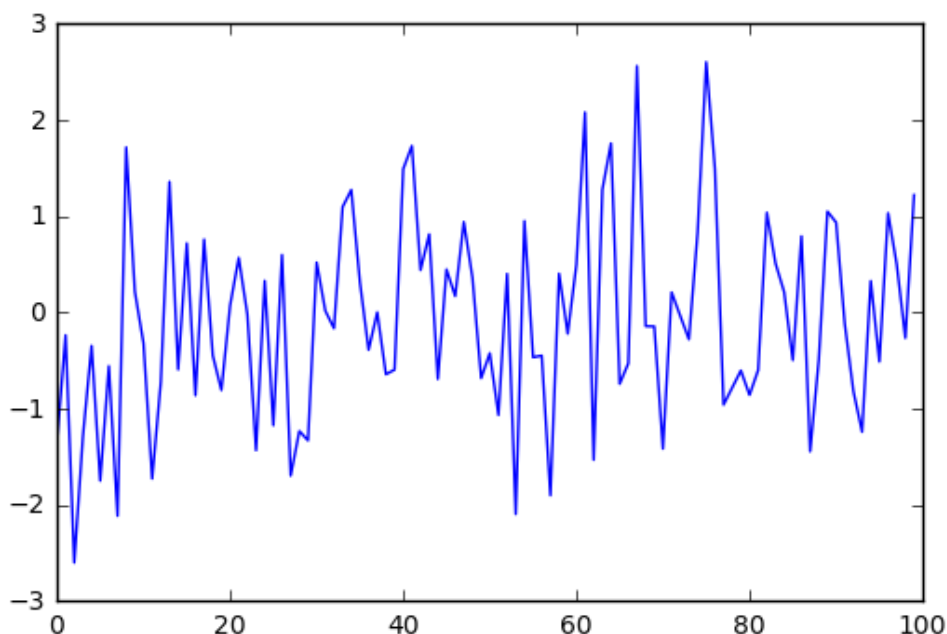
For more information about using Jupyter notebooks, see <http://Jupyter.org>.

The output from code executed in a cell is displayed in the cell immediately below in a cell labeled Out followed by the cell number in brackets as shown below. Note: Don't worry about the raw cell type. This cell type is used to format output when exporting the notebook using the NBConvert utility.

```
In [1]: # This is a code cell. Click on it and press Shift+Enter. While it's executin
%pylab inline
print "Python is easy to learn!"
plot(randn(100))
```

```
Populating the interactive namespace from numpy and matplotlib
Python is easy to learn!
```

```
Out[1]: [<matplotlib.lines.Line2D at 0x49a0d68>]
```



The Windows DSVM provides a wide selection of languages, development tools, machine learning tools, data

visualization tool, and data platforms that enable you to accomplish most of data science tasks. Here are some examples of those tasks:

- To find, load and preprocess data

Use the pre-installed data platform & ingestion tools, such as Azure blob storage, Azure Data Lake, Azure HDInsight (Hadoop), Azure CosmosDB, Azure SQL Data Warehouse & databases.

- To train and test models

Use Jupyter Notebook, Visual Studio, Visual Studio Code as development tools, select your preferred language (R or Python), leverage the powerful tools included in the Microsoft ML Server (for example, Microsoft R Server as we demonstrated in the earlier section)

- To operationalize the mode

Use R and Python Azure Machine Learning to deploy your model so client applications can access your models using a simple web service interface

- To train deep learning models

Use deep learning platforms such as PyTorch, CNKT, TensorFlow that supports GPU computation to accelerate the model training.

- To visualize data

Use the Power BI Desktop to build reports and dashboards and one-click to publish to the cloud

- Share file and code

Create an Azure File storage as a mountable drive on your DSVM to share large-scale datasets/code with your team; use Git clients – Git Bash and Git GUI to access your repository and share code with your team

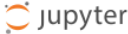
- Cloud administration

Use the Azure portal or PowerShell to manage your Azure resources, such as to restart/stop your VM, dynamically scale your DSVM to meet your project needs, extend your storage space, etc.

## Learn with pre-loaded Jupyter Notebooks

The DSVM has some pre-loaded notebooks for learning purposes. The pre-loaded notebooks are designed to teach about how to use various Azure services, such as the Azure Machine Learning services, SQL Server/Machine Learning Server, and Cognitive Services. To access the notebooks, launch the Jupyter Notebook and go into the notebooks folder. Subject areas are in separate subfolders. Note: SQL Server Developer Edition is installed with Machine Learning Server and data and code is loaded into the local database instance.

You can browse the [DSVM samples](#) for a more comprehensive understanding of what learning resources are available, and can also refer the notebooks by the language you prefer.



Quit

Logout

FilesRunningClustersConda

DuplicateRenameMoveDownloadViewEdit

UploadNew

1 / notebooks

NameLast ModifiedFile size

..seconds ago

AzureMLa day ago

Chainera day ago

DeepLearningToolsa day ago

DeepLearningTutorialsa day ago

Julia\_notebooksa day ago

MMLSparka day ago

pytorcha day ago

SparkMLa day ago

TensorFlowa day ago

Analyzing PyPI Data to Determine Python 3 Support.ipynb8 months ago

Comparing\_Base\_R\_to\_MRO\_and\_MRS.ipynb8 months ago