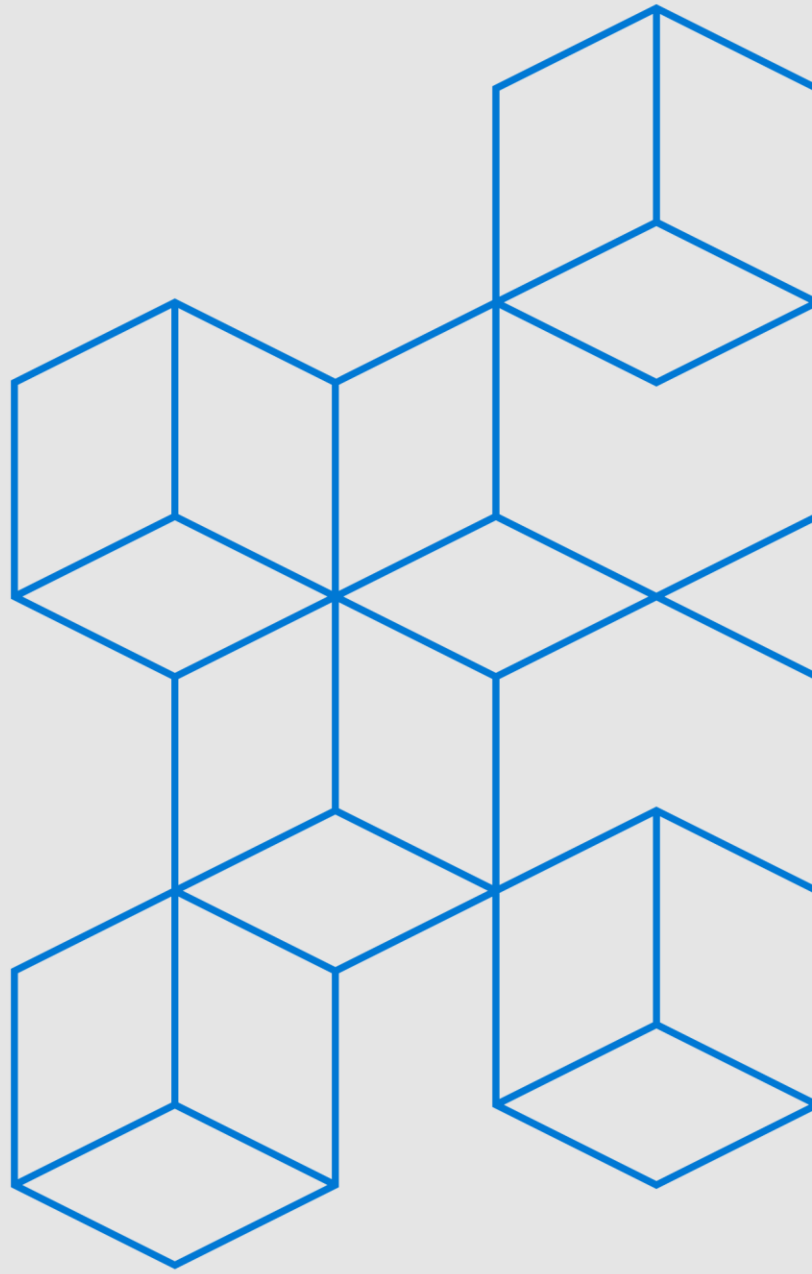
Azure Machine Learning 介绍

Kinfey.Lo
Microsoft MVP/Xamarin MVP
Github：https://github.com/lokinfey
Email: lokinfey@outlook.com

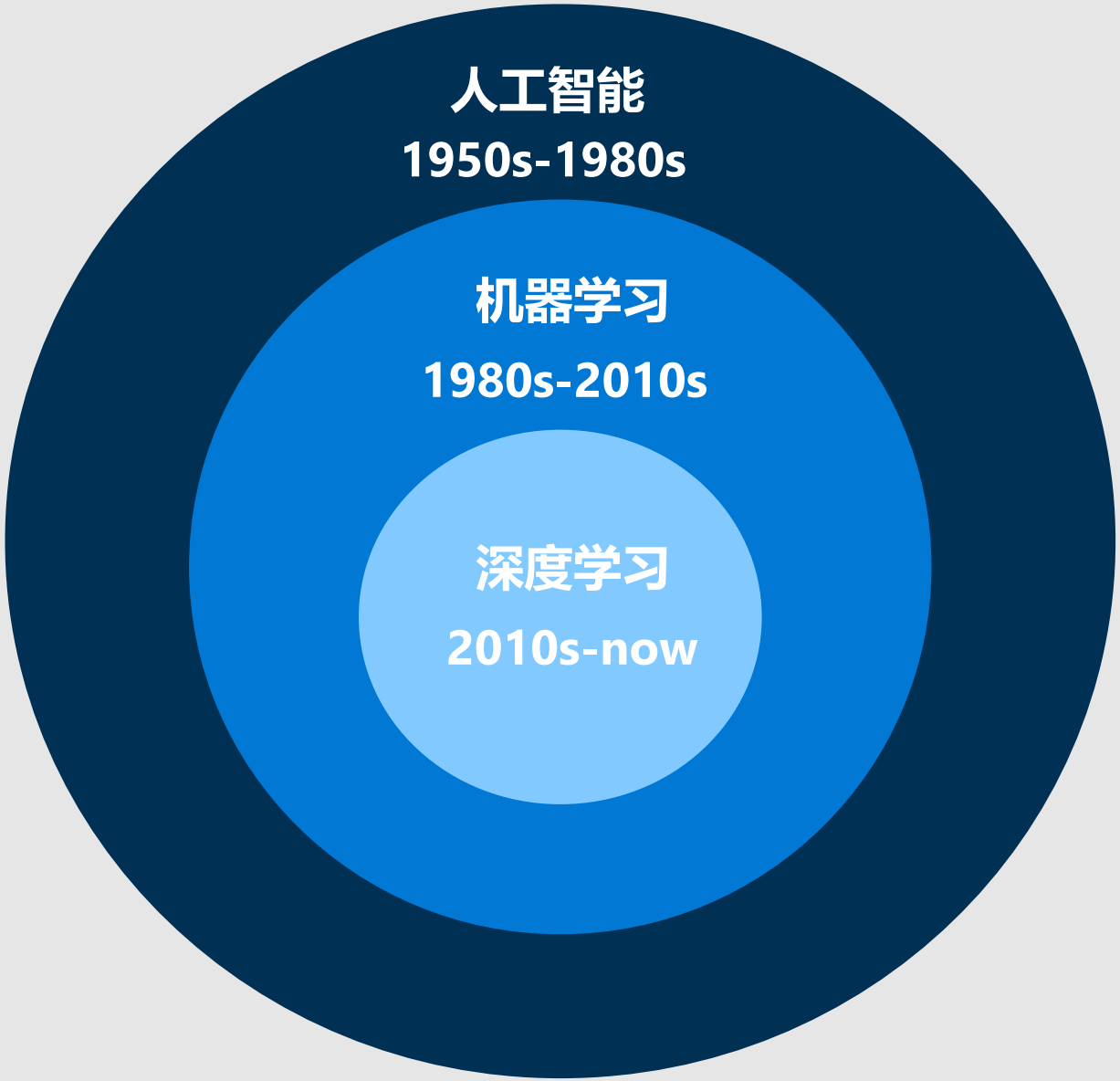


人工智能改变世界

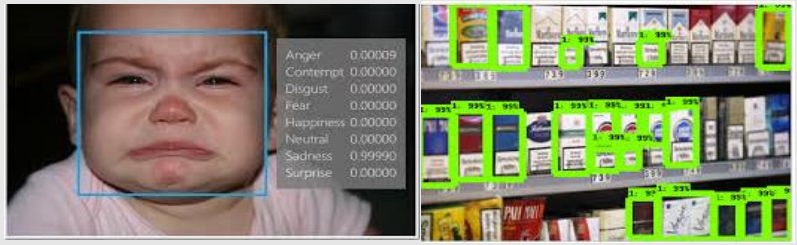

AI is the New Normal


72 FT
52.4932 13.283

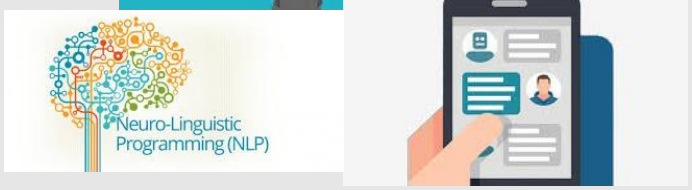
AI 技术不断变化



人工智能的热点方向



计算机视觉



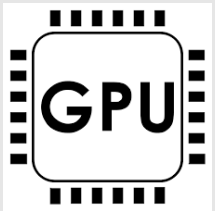
翻译

搜索

人工智能技术栈

语言

框架



硬件

AI@Azure

AI 应用和服务

Azure Bot Service
Azure Cognitive Services

机器学习

Azure Databricks
Azure Machine Learning

知识挖掘

Azure Cognitive Search

Machine Learning on Azure

提供一些预定义的模型
能降低开发门槛

Vision　Speech　Language　Search

兼容不同的开发工具
快速完成模型开发，简化开发流程

PyCharm　Jupyter　Visual Studio Code　Command line

对人工智能开发框架的支持
根据你的需要创建深度学习的解决方案

Pytorch　TensorFlow　Scikit-Learn　Onnx

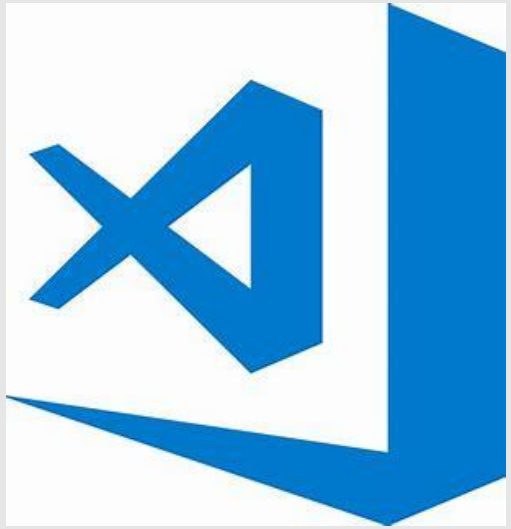
提供生产服务
为开发团队提供数据，和训练环境

Azure Databricks　Azure Machine Learning　Machine Learning VMs

强大的硬件架构支持
加速深度学习环境

CPU　GPU　FPGA

从开发工具谈起



Visual Studio Code

Azure Machine Learning

ms-toolsai.vscode-ai Preview

Microsoft | 271,469 | ★★★★½ | Repository | License

Visual Studio Code extension for Azure Machine Learning

Disable ▾ Uninstall

Details Contributions Changelog Dependencies

Azure Machine Learning for Visual Studio Code

Azure Machine Learning for Visual Studio Code, previously called Visual Studio Code Tools for AI**, is an extension to easily build, train, and deploy machine learning models to the cloud or the edge with Azure Machine Learning service.

Getting Started

- Installation
- Getting started with Azure Machine Learning for Visual Studio Code
- Create and manage Azure compute targets
- Train and tune models
- Deploy and manage models
- Release notes

With Azure Machine Learning service, you can:

- Build and train machine learning models faster, and easily deploy to the cloud or the edge.
- Use the latest open source technologies such as TensorFlow, PyTorch, or Jupyter.
- Experiment locally and then quickly scale up or out with large GPU-enabled clusters in the cloud.
- Speed up data science with automated machine learning and hyper-parameter tuning.
- Track your experiments, manage models, and easily deploy with integrated CI/CD tooling.

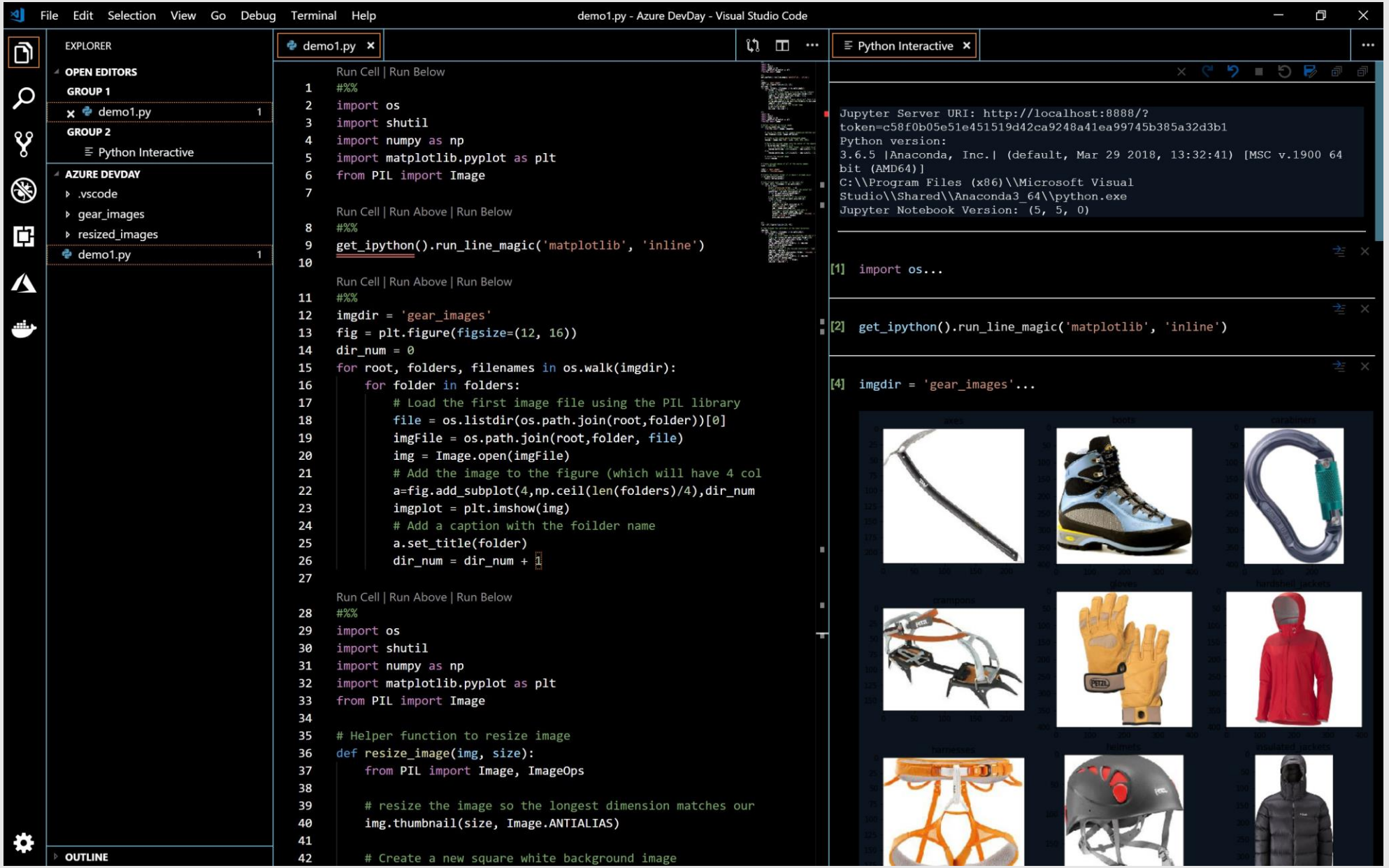
** Previous documentation and vsix installer are moved to the archive folder.

Supported Operating Systems

Currently this extension supports the following 64-bit operating systems:

- Windows
- macOS

示例


File Edit Selection View Go Debug Terminal Help
demo1.py - Azure DevDay - Visual Studio Code
EXPLORER
OPEN EDITORS
GROUP 1
demo1.py 1
GROUP 2
Python Interactive
AZURE DEVDAY
.vscode
gear_images
resized_images
demo1.py 1
OUTLINE
demo1.py
Python Interactive
Run Cell | Run Below
#%%
import os
import shutil
import numpy as np
import matplotlib.pyplot as plt
from PIL import Image
Run Cell | Run Above | Run Below
#%%
get_ipython().run_line_magic('matplotlib', 'inline')
Run Cell | Run Above | Run Below
#%%
imgdir = 'gear_images'
fig = plt.figure(figsize=(12, 16))
dir_num = 0
for root, folders, filenames in os.walk(imgdir):
for folder in folders:
# Load the first image file using the PIL library
file = os.listdir(os.path.join(root,folder))[0]
imgFile = os.path.join(root,folder, file)
img = Image.open(imgFile)
# Add the image to the figure (which will have 4 col
a=fig.add_subplot(4,np.ceil(len(folders)/4),dir_num
imgplot = plt.imshow(img)
# Add a caption with the foilder name
a.set_title(folder)
dir_num = dir_num + 1
Run Cell | Run Above | Run Below
#%%
import os
import shutil
import numpy as np
import matplotlib.pyplot as plt
from PIL import Image
# Helper function to resize image
def resize_image(img, size):
from PIL import Image, ImageOps
# resize the image so the longest dimension matches our
img.thumbnail(size, Image.ANTIALIAS)
# Create a new square white background image
Jupyter Server URI: http://localhost:8888/?
token=c58f0b05e51e451519d42ca9248a41ea99745b385a32d3b1
Python version:
3.6.5 |Anaconda, Inc.| (default, Mar 29 2018, 13:32:41) [MSC v.1900 64
bit (AMD64)]
C:\\Program Files (x86)\\Microsoft Visual
Studio\\Shared\\Anaconda3_64\\python.exe
Jupyter Notebook Version: (5, 5, 0)
[1] import os...
[2] get_ipython().run_line_magic('matplotlib', 'inline')
[4] imgdir = 'gear_images'...

NEW

Azure Cognitive Services

1. Object Detection & Recognition for thousands of objects
2. Video Indexer (Preview)
3. Speech Recognition with customization
4. Speech Synthesis with customizable voice
5. Speech to Speech Translation
6. Text analytics with entity detection
7. Language Understanding (LUIS) with new integrated offer
8. QnA Maker is Generally Available
9. Bing Visual Search with smart identification
10. Bing Search SDK is Generally Available

Vision

Speech

Language

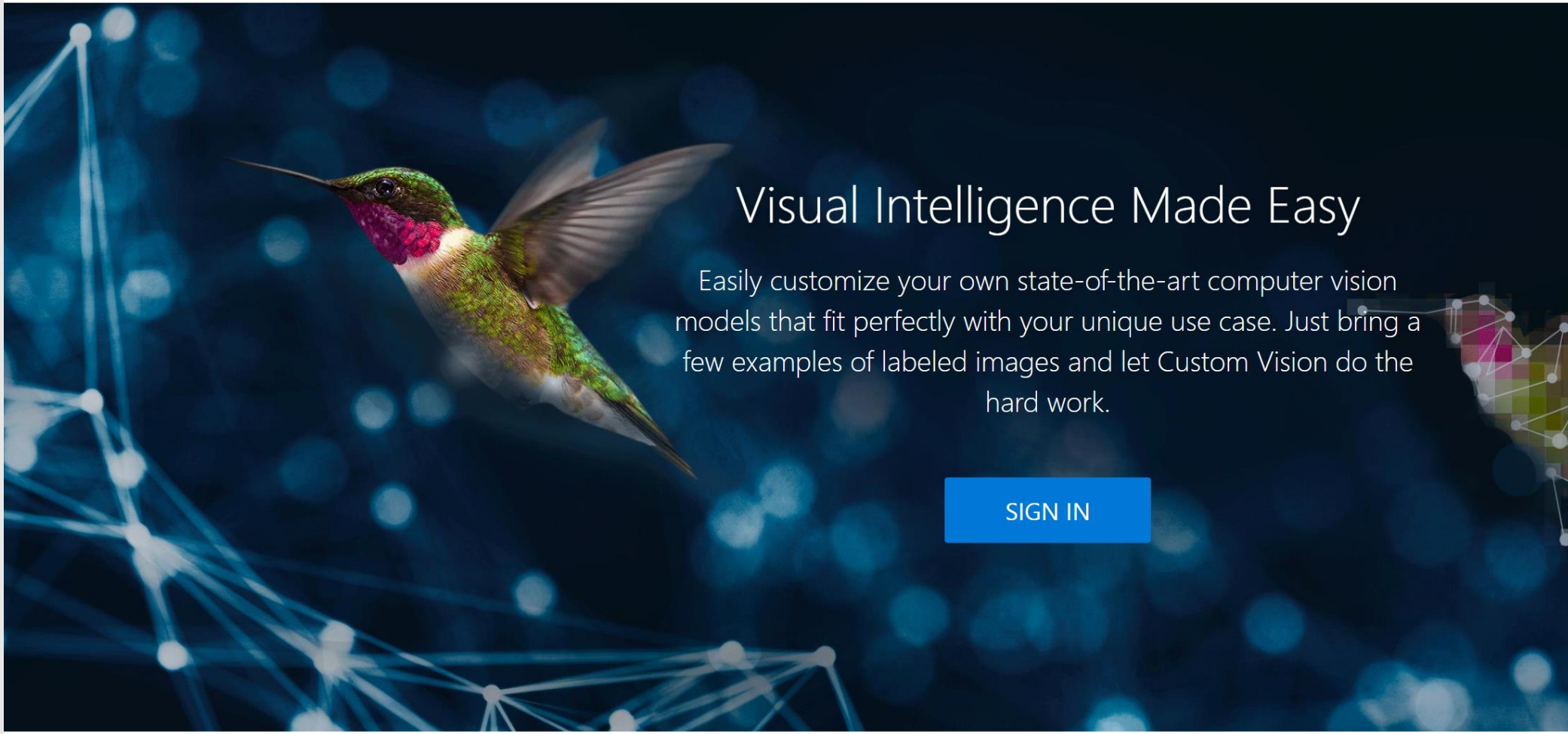
Conversation

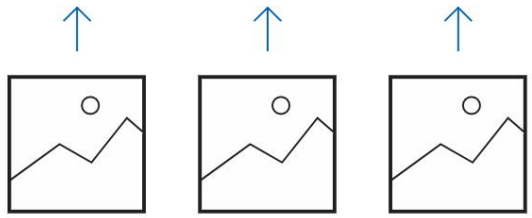
Bing Search

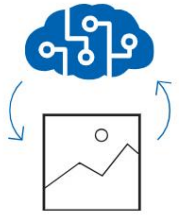
Knowledge

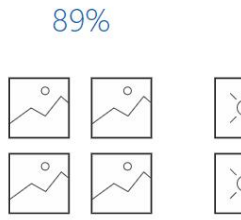
Demo

Custom Vision



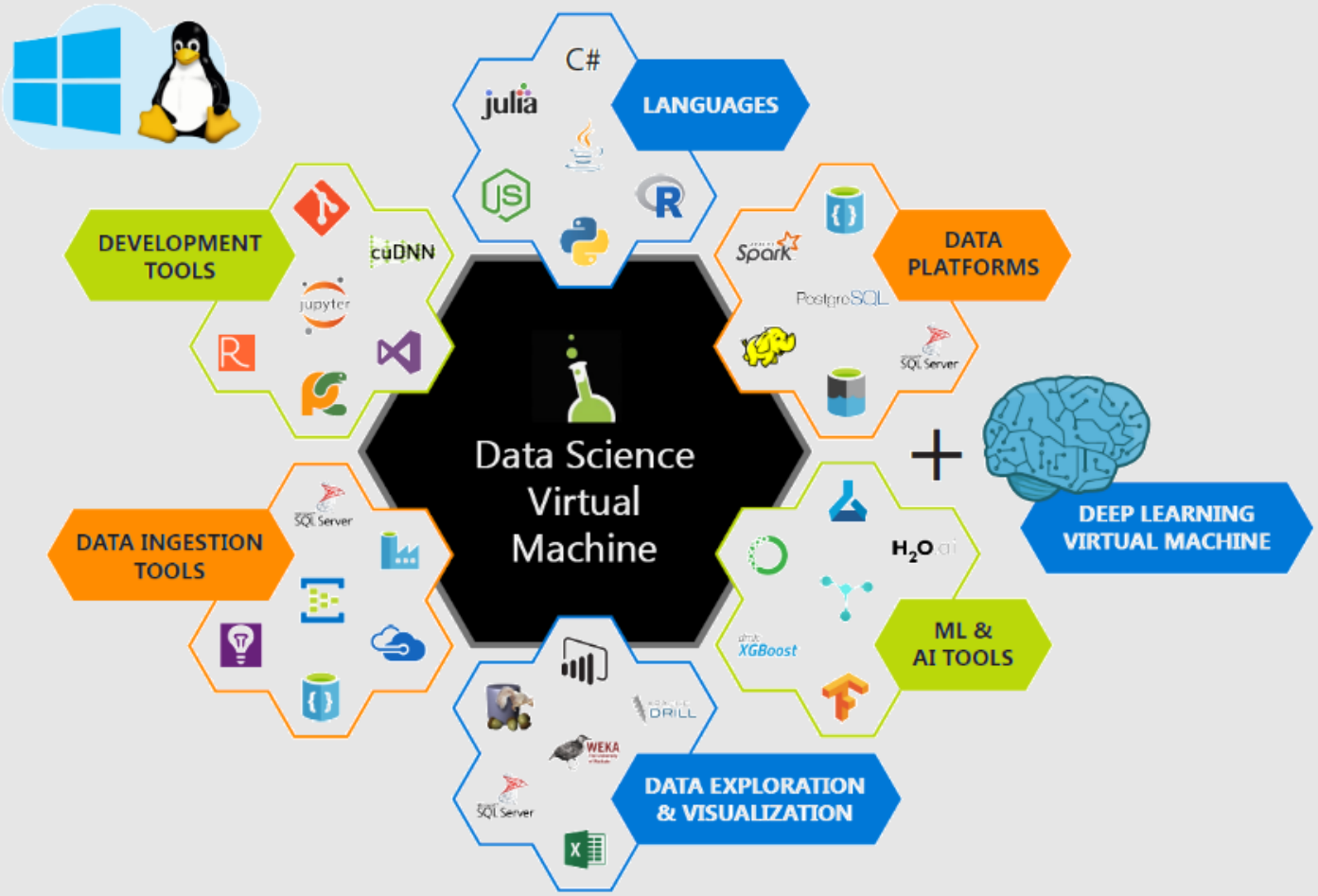






Data Science Virtual Machine(DSVM)

Pre-Configured environments in the cloud for
Data Science & AI Modeling, Development & Deployment.



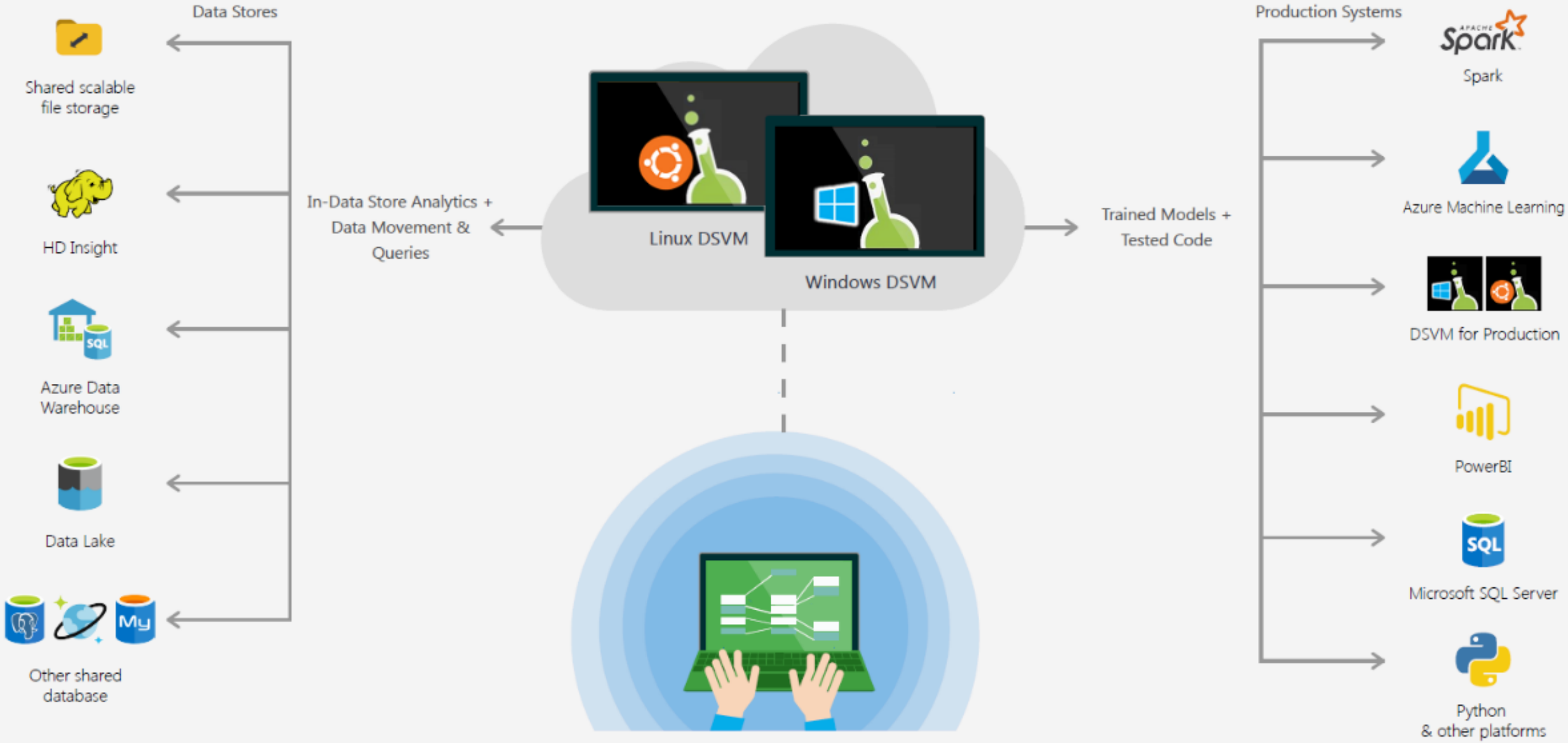
Data Science Virtual Machine(DSVM)

DSVM – Windows Server 2016

DSVM – Linux – Ubuntu

Deep Learning Virtual Machines

End-to-End AI Development Workflow using Data Science Virtual Machines (DSVM)



Demo

Jupyterhub on DSVM

jupyter lab03 Last Checkpoint: 5 hours ago (autosaved)

File Edit View Insert Cell Kernel Widgets Help

Run Code Edit Presentation Show Presentation

```
    return model
```

In [5]:

```
import matplotlib
matplotlib.use("Agg")

from keras.preprocessing.image import ImageDataGenerator
from keras.optimizers import Adam
from keras.preprocessing.image import img_to_array
from keras.utils import np_utils
from sklearn.preprocessing import LabelBinarizer
from sklearn.model_selection import train_test_split
# from imagesearch.smallervggnet import SmallerVGGNet
import matplotlib.pyplot as plt
from imutils import paths
import numpy as np
import argparse
import random
import pickle
import cv2
import os
```

In [6]:

```
EPOCHS = 100
INIT_LR = 1e-3
BS = 32
IMAGE_DIMS = (96, 96, 3)

data = []
labels = []

print("[INFO] loading images...")
imagePaths = sorted(list(paths.list_images('resized_images')))
random.seed(42)
random.shuffle(imagePaths)

for imagePath in imagePaths:
    image = cv2.imread(imagePath)
    image = cv2.resize(image, (IMAGE_DIMS[1], IMAGE_DIMS[0]))
    image = img_to_array(image)
    data.append(image)
```

Azure Machine Learning service

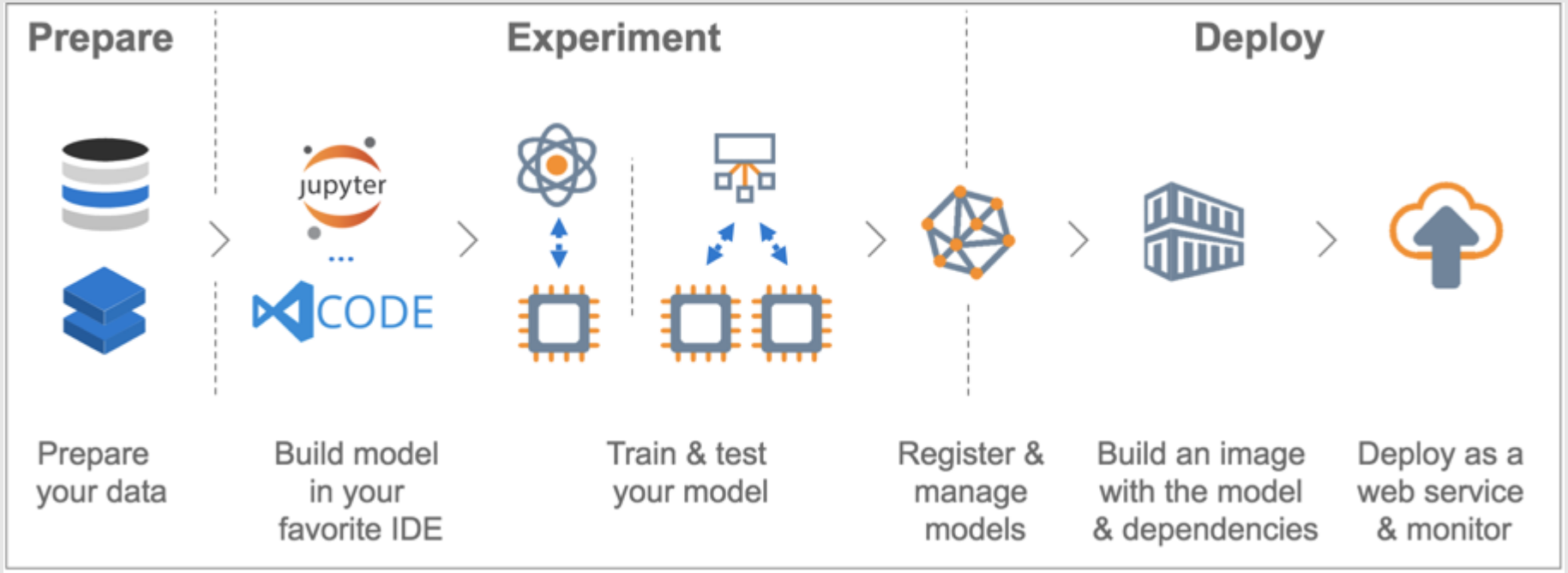
Azure Cloud Services + **Python SDK**

帮助你完成:

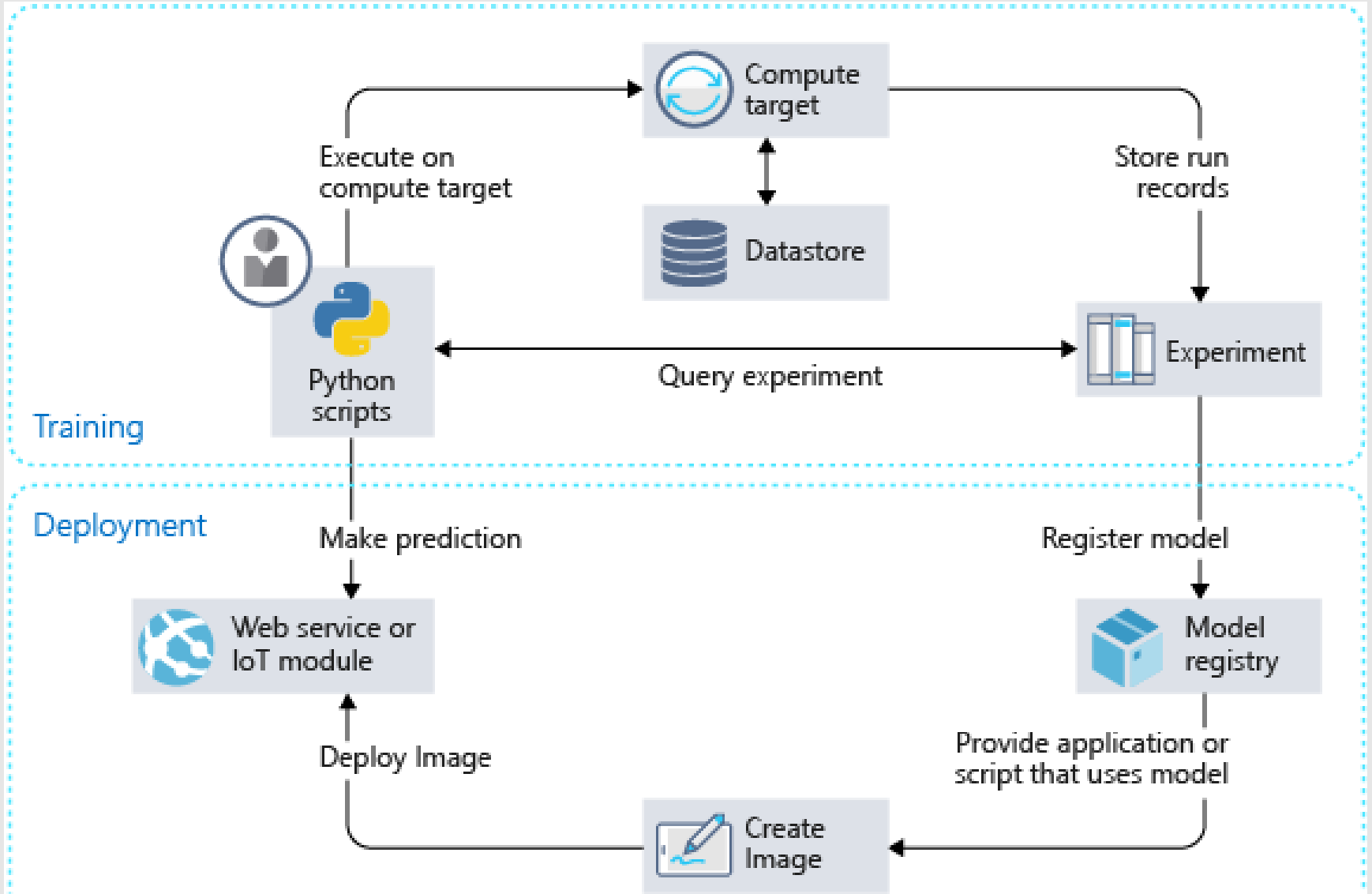
- ✓ 数据准备
- ✓ 编译模型
- ✓ 训练模型
- ✓ 管理模型
- ✓ 跟踪训练
- ✓ 部署模型

Azure Machine Learning Service

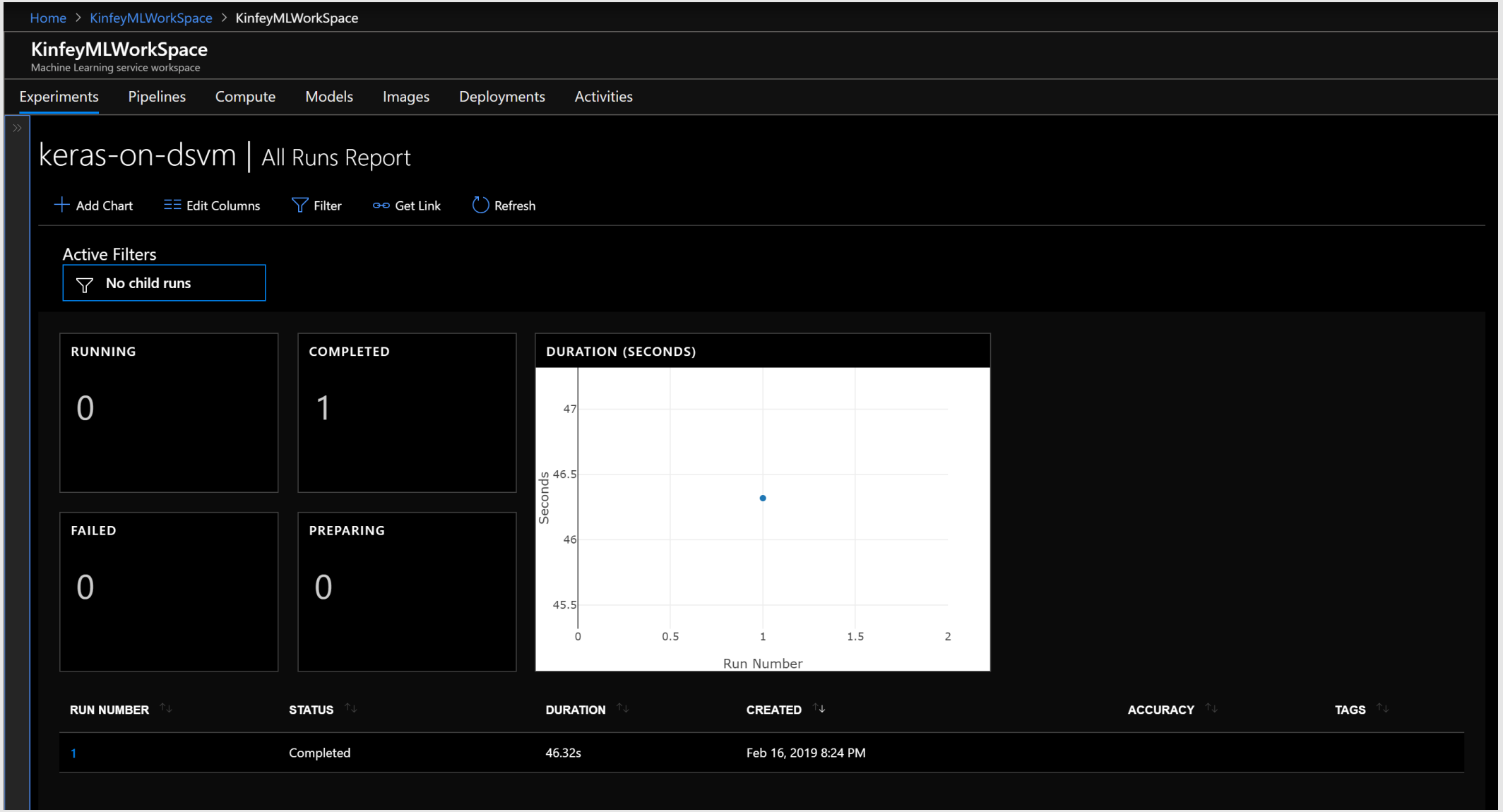
Azure Machine Learning service provides a cloud-based environment you can use to prep data, train, test, deploy, manage, and track machine learning models.



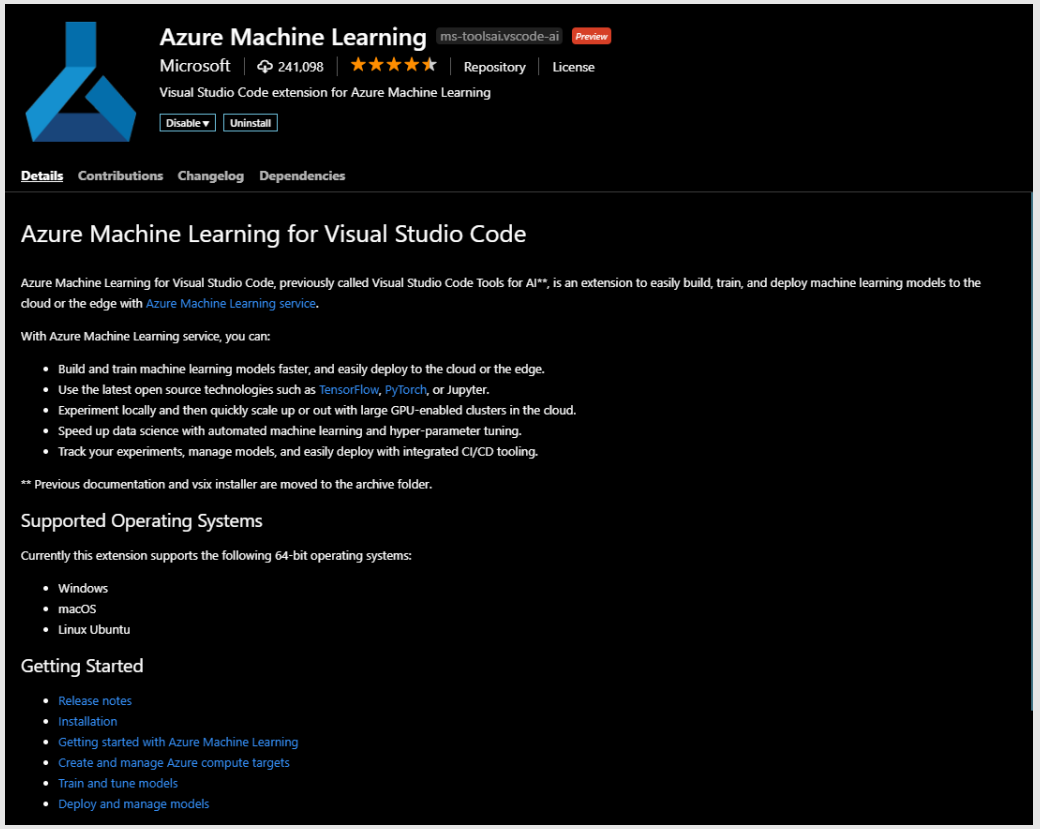
Azure Machine Learning Service

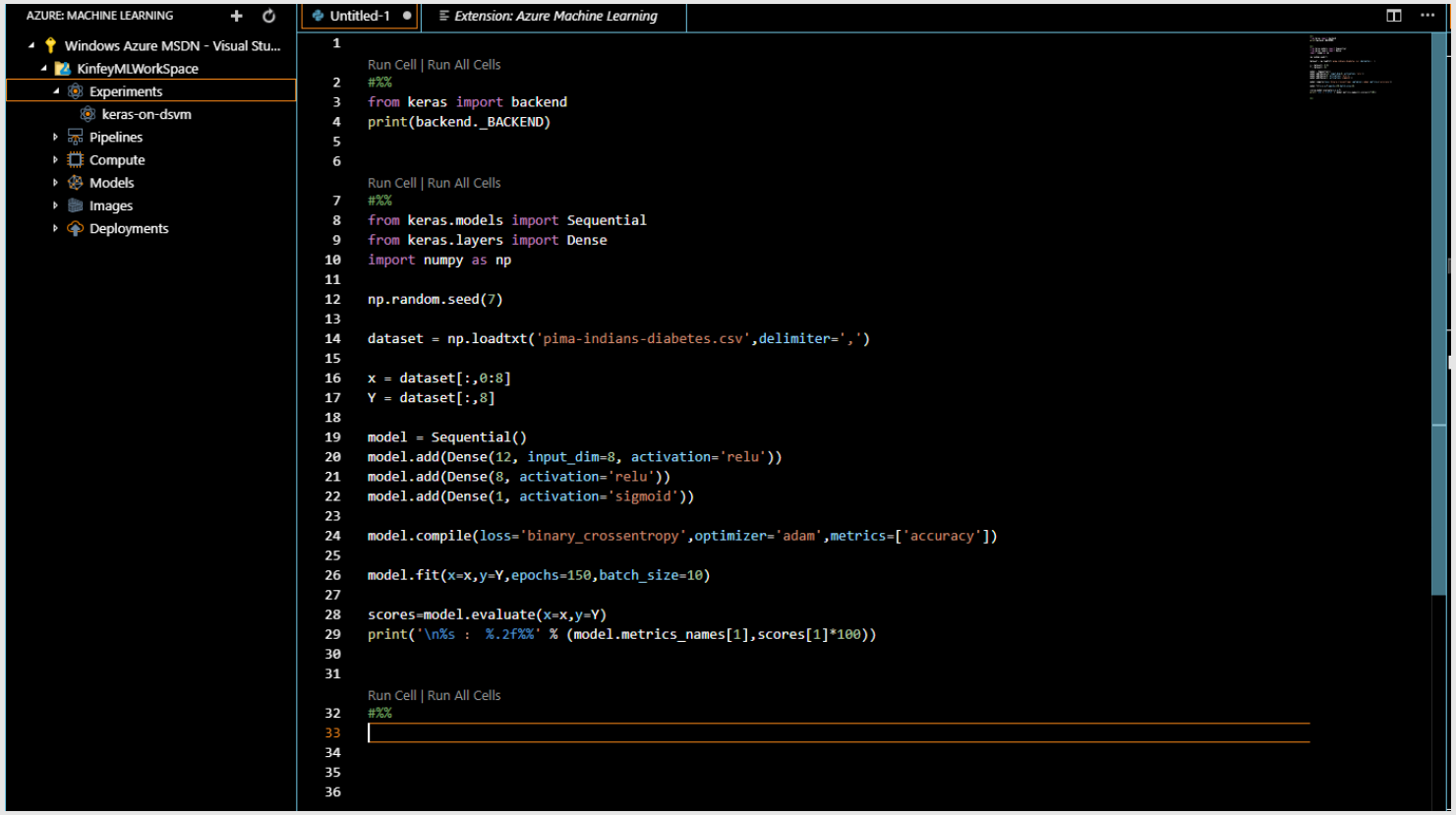


Azure Machine Learning Workbench



Azure Machine Learning Tools for VS Code





File Edit Selection View Go Debug Terminal Help — Python Interactive - ml - Visual Studio Code

AZURE: MACHINE LEARNING

- Windows Azure MSDN - Visual Stu...
  - KinfeyMLWorkSpace
    - Experiments
      - keras-on-dsvm
    - Pipelines
    - Compute
    - Models
    - Images
    - Deployments

Untitled-1 ● | Extension: Azure Machine Learning

```
Run Cell | Run All Cells
#%%
from keras import backend
print(backend._BACKEND)


Run Cell | Run All Cells
#%%
from keras.models import Sequential
from keras.layers import Dense
import numpy as np

np.random.seed(7)

dataset = np.loadtxt('pima-indians-diabetes.csv',delimiter='

x = dataset[:,0:8]
Y = dataset[:,8]

model = Sequential()
model.add(Dense(12, input_dim=8, activation='relu'))
model.add(Dense(8, activation='relu'))
model.add(Dense(1, activation='sigmoid'))

model.compile(loss='binary_crossentropy',optimizer='adam',me

model.fit(x=x,y=Y,epochs=150,batch_size=10)

scores=model.evaluate(x=x,y=Y)
print('\n%s :  %.2f%%' % (model.metrics_names[1],scores[1]*1

Run Cell | Run All Cells
#%%
```

Python Interactive

```
Jupyter Server URI: http://localhost:8888/?token=f19ba4f243b52212b37ef26a13e374145de001acfaeb13b3
Python Version:
3.6.6 (v3.6.6:4cf1f54eb7, Jun 27 2018, 03:37:03) [MSC v.1900 64 bit (AMD64)]
C:\\Program Files (x86)\\Microsoft Visual Studio\\Shared\\Python36_64\\python.exe
Jupyter Notebook Version: (5, 7, 4, '')
```

[1] from keras import backend...

```
Using TensorFlow backend.
tensorflow
```

[2] from keras.models import Sequential...

```
768/768 [==============================] - 0s 251us/step - loss: 0.6174 - acc: 0.6992
Epoch 15/150
768/768 [==============================] - 0s 242us/step - loss: 0.6015 - acc: 0.6979
Epoch 16/150
768/768 [==============================] - 0s 260us/step - loss: 0.5877 - acc: 0.7018
Epoch 17/150
768/768 [==============================] - 0s 264us/step - loss: 0.5850 - acc: 0.6979
Epoch 18/150
768/768 [==============================] - 0s 311us/step - loss: 0.5999 - acc: 0.6901
Epoch 19/150
768/768 [==============================] - 0s 259us/step - loss: 0.5806 - acc: 0.7122
Epoch 20/150
768/768 [==============================] - 0s 245us/step - loss: 0.5797 - acc: 0.7240
Epoch 21/150
768/768 [==============================] - 0s 260us/step - loss: 0.5712 - acc: 0.7148
Epoch 22/150
768/768 [==============================] - 0s 238us/step - loss: 0.5831 - acc: 0.6992
Epoch 23/150
768/768 [==============================] - 0s 276us/step - loss: 0.5751 - acc: 0.7122
Epoch 24/150
768/768 [==============================] - 0s 299us/step - loss: 0.5704 - acc: 0.7266
Epoch 25/150
768/768 [==============================] - 0s 255us/step - loss: 0.5577 - acc: 0.7383
Epoch 26/150
768/768 [==============================] - 0s 239us/step - loss: 0.5720 - acc: 0.7044
Epoch 27/150
```

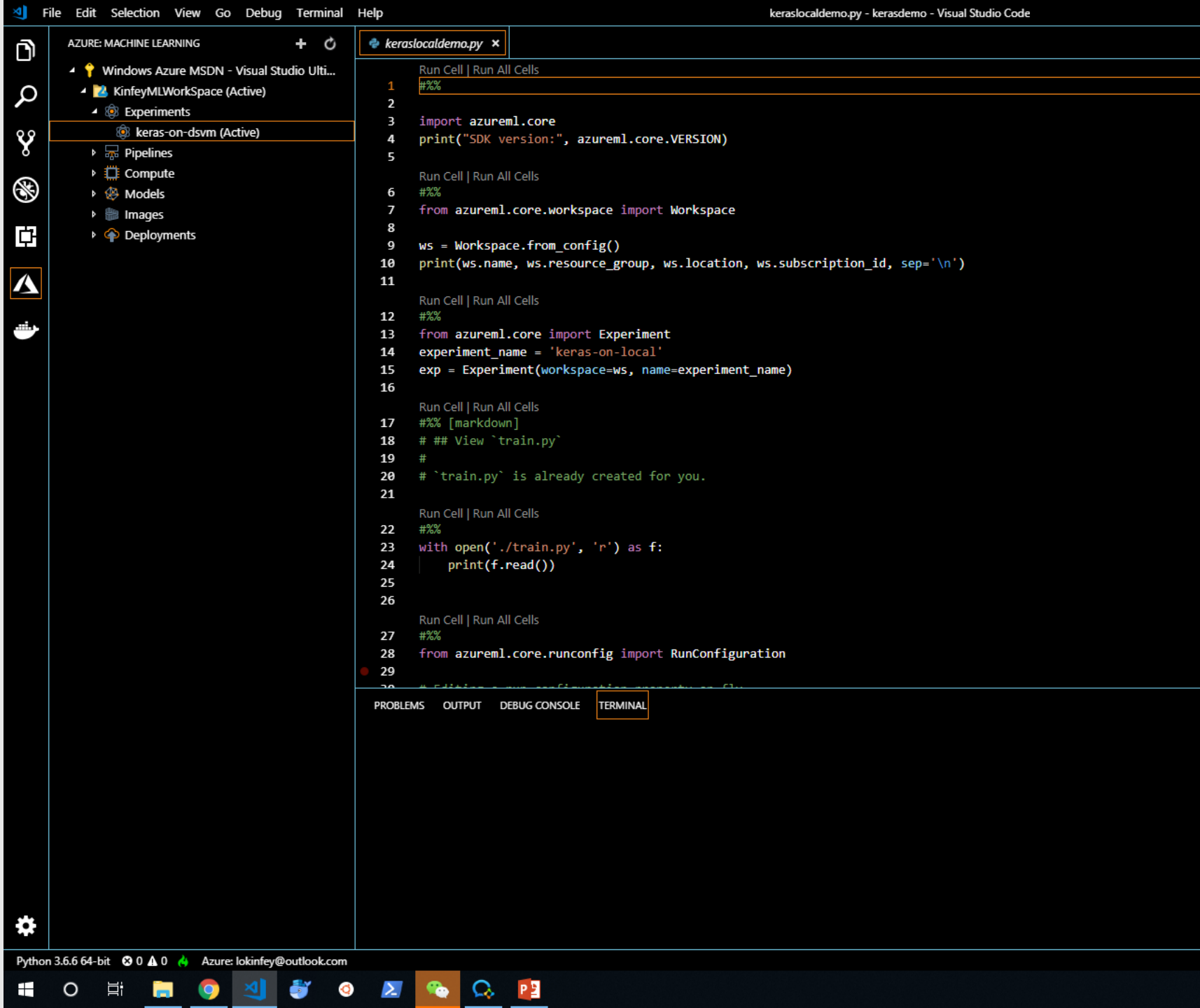
>>> Shift-enter to run

Python 3.6.6 64-bit ⊗ 0 ⚠ 0 Azure: lokinfey@outlook.com

11:49 AM 2/18/2019

Demo

Azure ML Workbench Demo



Thank you