


## CREATING A PROJECT:

[← Back](#)

### Create a project

Choose whether to create an empty project or to preload your project with data and analytical assets. Add collaborators and data, and then choose the right tools to accomplish your goals. Add services as necessary.




#### Create an empty project

Add the data you want to prepare, analyze, or model. Choose tools based on how you want to work: write code, create a flow on a graphical canvas, or automatically build models.

USE TO

- Prepare and visualize data
- Analyze data in notebooks
- Train models



#### Create a project from a sample or file

Get started fast by loading existing assets. Choose a project file from your system, or choose a curated sample project.

USE TO

- Learn by example
- Build on existing work
- Run tutorials

## CREATING A NEW ENVIRONMENT:

### NEW NOTEBOOK

BlankFrom fileFrom URL

Name

Train\_the\_model

Description (optional)

Type your description here

Select runtime

Runtime 22.1 on Python 3.9 XS (2 vCPU 8 GB RAM) ▾

The selected runtime has 2 vCPU and 8 GB RAM. It consumes 1 capacity unit per hour. [Learn more](#) about capacity unit hours and Watson Studio pricing plans.

Notebook file

Upload only .ipynb files. 52 MB max file size.

Drag and drop files here or upload.

Train\_the\_model.ipynb ✕

Cancel

Create

## CREATING CLOUD SPACE:

New project

Define details

Name

A Novel Method for Handwritten Digit Recognition

Description

handwritten digits. We use Artificial neural networks to train these images and build a deep learning model. Web application is created where the user can upload an image of a handwritten digit. this image is **analyzed** by the model and the detected result is returned on to UI

Choose project options

☐ Restrict who can be a collaborator ⓘ

☐ Mark as sensitive ⓘ

Project includes integration with [Cloud Object Storage](#) for storing project assets.

Storage

Cloud Object Storage-kr

Cancel Create

## TRAINING THE MODEL ON IBM CLOUD:

```
Projects / A Novel Method for Handwritten ... / Handwritten Digit Recognition
File Edit View Insert Cell Kernel Help Not Trusted | Python 3.9
Import the necessary packages
In [1]: import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

from keras.utils import np_utils
from tensorflow.keras.datasets import mnist
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Conv2D, Dense, Flatten
from tensorflow.keras.optimizers import Adam
from tensorflow.keras.models import load_model
from PIL import Image, ImageOps

Load the data
In [5]: (X_train, y_train), (X_test, y_test) = mnist.load_data()

Data Analysis
In [6]: print(X_train.shape)
print(X_test.shape)
```



```
Projects / A Novel Method for Handwritten ... / Handwritten Digit Recognition

File Edit View Insert Cell Kernel Help Trusted | Python 3.5

Data Pre-Processing

In [10]: X_train = X_train.reshape(50000, 28, 28, 1).astype('float32')
X_test = X_test.reshape(10000, 28, 28, 1).astype('float32')

In [11]: number_of_classes = 10
Y_train = np_utils.to_categorical(y_train, number_of_classes)
Y_test = np_utils.to_categorical(y_test, number_of_classes)

Create Model

In [12]: model = Sequential()
model.add(Conv2D(64, (3, 3), input_shape=(28, 28, 1), activation='relu'))
model.add(Conv2D(32, (3, 3), activation='relu'))
model.add(Flatten())
model.add(Dense(number_of_classes, activation='softmax'))

In [13]: model.compile(loss='categorical_crossentropy', optimizer='Adam', metrics=['accuracy'])

Train the Model
```

```
Projects / A Novel Method for Handwritten ... / Handwritten Digit Recognition

File Edit View Insert Cell Kernel Help Trusted | Python 3.5

Train the Model

In [*]: model.fit(X_train, Y_train, batch_size=32, epochs=5, validation_data=(X_test, Y_test))

Epoch 1/5
1875/1875 [=====] - 190s 181ms/step - loss: 0.2821 - accuracy: 0.9473 - val_loss: 0.0984 - val_accuracy: 0.9678
Epoch 2/5
1875/1875 [=====] - 191s 182ms/step - loss: 0.0737 - accuracy: 0.9774 - val_loss: 0.0700 - val_accuracy: 0.9703
Epoch 3/5
1875/1875 [=====] - 186s 99ms/step - loss: 0.0504 - accuracy: 0.9834 - val_loss: 0.0846 - val_accuracy: 0.9755
Epoch 4/5
1875/1875 [=====] - 188s 100ms/step - loss: 0.0373 - accuracy: 0.9881 - val_loss: 0.1391 - val_accuracy: 0.9625
Epoch 5/5
1267/1875 [=====>.....] - ETA: 59s - loss: 0.0256 - accuracy: 0.9923 ETA: 1:00 - loss: 0.0255 - a

Test the Model

In [ ]: metrics = model.evaluate(X_test, Y_test, verbose=0)
print('Metrics (Test Loss & Test Accuracy): ')
print(metrics)

In [ ]: prediction = model.predict(X_test[:4])
print(prediction)
```

```
Projects / A Novel Method for Handwritten ... / Handwritten Digit Recognition

File Edit View Insert Cell Kernel Help Trusted | Python 3.9

Test the Model

In [15]: metrics = model.evaluate(X_test, Y_test, verbose=0)
print("Metrics (Test Loss & Test Accuracy): ")
print(metrics)

Metrics (Test Loss & Test Accuracy):
[0.88687877655029297, 0.9807999730518168]

In [16]: prediction = model.predict(X_test[:4])
print(prediction)

[[5.89432931e-15 1.56521345e-20 7.35486906e-12 1.36783318e-09
 5.79286134e-22 1.02446433e-15 1.01128972e-21 1.00000000e+00
 9.58406006e-15 1.10001279e-11]
[1.01669286e-08 7.29183843e-08 0.99993801e-01 3.05165288e-13
 7.06790235e-16 2.02896846e-17 6.04845673e-06 3.74402691e-14
 1.11660945e-13 8.82754992e-14]
[3.54005977e-07 9.08027857e-01 3.29728266e-07 4.19751123e-09
 3.01197371e-03 3.50851333e-05 1.20187156e-06 2.09555239e-07
 2.37075274e-05 2.90604186e-10]
[1.00000000e+00 2.00214140e-18 1.09388729e-12 1.19740111e-16
 1.63756283e-10 8.57702418e-13 3.01977536e-08 1.70557578e-12
 1.17479572e-12 1.02323587e-08]]

In [17]: print(numpy.argmax(prediction, axis=1))
```

```
Projects / A Novel Method for Handwritten ... / Handwritten Digit Recognition

File Edit View Insert Cell Kernel Help Trusted | Python 3.9

Metrics (Test Loss & Test Accuracy):
[0.88687877655029297, 0.9807999730518168]

In [16]: prediction = model.predict(X_test[:4])
print(prediction)

[[5.89432931e-15 1.56521345e-20 7.35486906e-12 1.36783318e-09
 5.79286134e-22 1.02446433e-15 1.01128972e-21 1.00000000e+00
 9.58406006e-15 1.10001279e-11]
[1.01669286e-08 7.29183843e-08 0.99993801e-01 3.05165288e-13
 7.06790235e-16 2.02896846e-17 6.04845673e-06 3.74402691e-14
 1.11660945e-13 8.82754992e-14]
[3.54005977e-07 9.08027857e-01 3.29728266e-07 4.19751123e-09
 3.01197371e-03 3.50851333e-05 1.20187156e-06 2.09555239e-07
 2.37075274e-05 2.90604186e-10]
[1.00000000e+00 2.00214140e-18 1.09388729e-12 1.19740111e-16
 1.63756283e-10 8.57702418e-13 3.01977536e-08 1.70557578e-12
 1.17479572e-12 1.02323587e-08]]

In [19]: print(np.argmax(prediction, axis=1))
print(Y_test[:4])

[7 2 1 0]
[[0. 0. 0. 0. 0. 0. 1. 0. 0.]
 [0. 0. 1. 0. 0. 0. 0. 0. 0.]
 [0. 1. 0. 0. 0. 0. 0. 0. 0.]
 [1. 0. 0. 0. 0. 0. 0. 0. 0.]]
```

```
Projects / A Novel Method for Handwritten ... / Handwritten Digit Recognition

File Edit View Insert Cell Kernel Help Trusted | Python 3.9

Save the model

In [20]: model.save("model.h5")

Converting to tar format

In [21]: !tar -czvf Handwritten-Digit-Recognition_new.tar.gz model.h5
model.h5

In [22]: !ls -l
Handwritten-Digit-Recognition_new.tar.gz
model.h5

Installing Watson Machine Learning

In [ ]: !pip install watson-machine-learning-client --upgrade

Watson API credentials
```

```
Projects / A Novel Method for Handwritten ... / Handwritten Digit Recognition

File Edit View Insert Cell Kernel Help Trusted | Python 3.9

Installing watson machine Learning

In [23]: !pip install watson-machine-learning-client --upgrade

Collecting watson-machine-learning-client
  Downloading watson-machine-learning-client-1.0.391-py3-none-any.whl (538 kB)
    538 kB 18.6 MB/s eta 0:00:01
Requirement already satisfied: tqdm in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson-machine-learning-client) (4.62.3)
Requirement already satisfied: requests in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson-machine-learning-client) (2.26.0)
Requirement already satisfied: lsmom in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson-machine-learning-client) (0.3.3)
Requirement already satisfied: certifi in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson-machine-learning-client) (2022.9.24)
Requirement already satisfied: urllib3 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson-machine-learning-client) (1.26.7)
Requirement already satisfied: pandas in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson-machine-learning-client) (1.3.4)
Requirement already satisfied: ibm-cos-sdk in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson-machine-learning-client) (2.11.0)
Requirement already satisfied: boto3 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson-machine-learning-client) (1.18.21)
Requirement already satisfied: tabulate in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson-machine-learning-client) (0.8.9)
Requirement already satisfied: botocore<1.22.0,>=1.21.21 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from boto3->watson-machine-learning-client) (1.21.41)
Requirement already satisfied: jmespath<1.0.0,>=0.7.1 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from boto3->watson-machine-learning-client) (0.10.0)
Requirement already satisfied: s3transfer<0.6.0,>=0.5.0 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from boto3->watson-machine-learning-client) (0.5.0)
Requirement already satisfied: python-dateutil<3.0.0,>=2.1 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from botocore<1.22.0,>=1.21.21->boto3->watson-machine-learning-client) (2.8.2)
Requirement already satisfied: six<=1.5 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from python-dateutil<3.0.0,>=2.1->botocore<1.22.0,>=1.21.21->boto3->watson-machine-learning-client) (1.15.0)
Requirement already satisfied: ibm-cos-sdk-core<=2.11.0 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-cos-sdk->watson-machine-learning-client) (2.11.0)
```

## WATSON API CONFIGU

```
Projects / A Novel Method for Handwritten ... / Handwritten Digit Recognition

Watson API credentials

In [48]: from ibm_watson_machine_learning import APIClient
credentials = {
    "url": "https://us-south.ml.cloud.ibm.com",
    "apikey": "79d8p6c6fy3d2f1270w32npg022a1aFus08Tlpud27j"
}
client = APIClient(credentials)

In [49]: def guid_from_space_name(client, space_name):
space=client.spaces.get_details()
for item in space['resources']:
    if item['entity']['name'] == space_name['metadata']['id']:
        return item['id']

In [50]: space_uid = guid_from_space_name(client, 'HandwrittenDigitRecognition')
print('space_uid = ' + space_uid)
space_uid = wada184f-43ea-4552-9509-61a24b551a41

In [51]: client.set_default_space(space_uid)

Out[51]: "SUCCESS"

In [52]: client.software_specifications.list()

-----
NAME                                ASSET_ID                                TYPE
default_py3.6                      00620d89-8b7d-44a0-w9b9-46a416adb0d9 base
kernel-spark3.2-scala1.12          820069ce-7ac1-9e08-9c1e-31189b67356a base
pytorch-anna_1.3-py3.7-cudt       009a13a4-3d4d-5748-0513-40120a15c288 base
spark-mllib_2.4-r_3.6             08c5a100-9c1e-4a73-a344-e97b665ff687 base
spark-mllib_3.0-scala_2.12        09fa4ff8-98a7-5899-bbed-1e7348aebdee base
```

```
Projects / A Novel Method for Handwritten ... / Handwritten Digit Recognition

pytorch-anna_1.2-py3.6-cudt       40580a0e-7019-4a28-8daa-f88586f4fa32 base
default_py3.6                     41c247d3-45f8-5a71-b005-0500229facf0 base
autnai-ta_r12.1-py3.9             4203d25e-07ea-5d40-8fde-2d495bdc71f7 base
autnai-omn_3.0                   42b82e18-09a0-567f-988a-42a00a1e05f7 base
pml-3.0-4.3                       491bcb05-16f1-9b3c-baed-01baf08e9e7 base
spark-mllib_2.4-r_3.6             4b4810ff-92e9-4c87-a3d7-a4200021c095 base
apbnoet_8.90-py3.8               4ff60bc2-1543-4c18-85a1-689c953004d3 base
pytorch-anna_1.1-py3.6           50f9582a-bc16-430b-bc94-b0e0200c600 base
autnai-ta_3.9-py3.8              52c57218-80fa-972e-8728-9ea7c3b42cde base
spark-mllib_2.4-scala_2.11        55a70f99-7338-40e5-97b9-bc050a13af9 base
spark-mllib_3.0                   5c1bbaa2-4077-5c2e-9439-ff644eafffa9 base
autnai-omn_2.0                   5c2e377a-880b-5e77-940f-d911469614ee base
spss-modeler_18.1                5c3cad7e-987f-482a-e9a7-eb55a21deeb0 base
cud9-py3.8                       5d32328f-c90b-5d94-a2c0-70b870a1c64e base
autnai-hb_3.1-py3.7              832d4022-18aa-5180-88f9-f12dfb444a07 base
pytorch-anna_1.7-py3.8           83a0d50c-0562-5079-a234-e9b09a78456b base
spark-mllib_2.3-r_3.6            8588b0a3-cce8-4f92-90ef-0f8c32bd0f0c base
tensorflow_2.4-py3.7             85e171d7-72d1-55d9-b0bb-f813620c90b base
spss-modeler_18.2                887add99-028e-4117-b0dd-e57036f1efa5 base

Note: Only first 50 records were displayed. To display more use 'limit' parameter.

In [57]: software_spec_uid=client.software_specifications.get_uid_by_name('tensorflow_2.4-py3.6')
software_spec_uid

Out[57]: "2b75a275-7cbf-428b-w912-aa7f435a0bc"

In [58]: model_details=client.repository.store_model(model='Handwritten-Digit-Recognition_new.tgi', meta_props={
client.repository.ModelMetadata.NAME:'OM',
client.repository.ModelMetadata.TYPE:'kars_2.2.4',
client.repository.ModelMetadata.SOFTWARE_SPEC_UID:software_spec_uid
})
model_id=client.repository.get_model_uid(model_details)
```

