

Registering models

5 minutes

After running an experiment that trains a model you can use a reference to the **Run** object to retrieve its outputs, including the trained model.

Retrieving model files

After an experiment run has completed, you can use the run objects **get_file_names** method to list the files generated. Standard practice is for scripts that train models to save them in the run's **outputs** folder.

You can also use the run object's **download_file** and **download_files** methods to download output files to the local file system.

Python

Copy

```
# "run" is a reference to a completed experiment run

# List the files generated by the experiment
for file in run.get_file_names():
    print(file)

# Download a named file
run.download_file(name='outputs/model.pkl', output_file_path='model.pkl')
```

Registering a model

Model registration enables you to track multiple versions of a model, and retrieve models for *inferencing* (predicting label values from new data). When you register a model, you can specify a name, description, tags, framework (such as Scikit-Learn or PyTorch), framework version, custom properties, and other useful metadata. Registering a model with the same name as an existing model automatically creates a new version of the model, starting with 1 and increasing in units of 1.

To register a model from a local file, you can use the **register** method of the **Model** object as shown here:

Python

Copy

```
from azureml.core import Model

model = Model.register(workspace=ws,
                        model_name='classification_model',
                        model_path='model.pkl', # local path
                        description='A classification model',
                        tags={'data-format': 'CSV'},
                        model_framework=Model.Framework.SCIKITLEARN,
                        model_framework_version='0.20.3')
```

Alternatively, if you have a reference to the **Run** used to train the model, you can use its **register_model** method as shown here:

Python

 Copy

```
run.register_model( model_name='classification_model',
                    model_path='outputs/model.pkl', # run outputs path
                    description='A classification model',
                    tags={'data-format': 'CSV'},
                    model_framework=Model.Framework.SCIKITLEARN,
                    model_framework_version='0.20.3')
```

Viewing registered models

You can view registered models in Azure Machine Learning studio. You can also use the **Model** object to retrieve details of registered models like this:

Python

 Copy

```
from azureml.core import Model

for model in Model.list(ws):
    # Get model name and auto-generated version
    print(model.name, 'version:', model.version)
```

Next unit: Exercise - Training and registering a model

[Continue >](#)