

Something from an old presentation on the subject

Seamless connection to Fabric data

Microsoft Fabric

1. Create Lakehouse
2. Upload .csv
3. Load as (Delta) Table
4. Add AML compute instance workspace Viewer/Contributor

The screenshot shows the Microsoft Fabric Explorer interface. On the left, the 'Explorer' sidebar lists 'jlaLakehouse' with its 'Tables' and 'Files' sections. Inside 'Tables', there is a 'churn' entry. The main area displays the 'churn' table with the following schema:

	RowNumber	CustomerId	Surname	CreditScore	Geography
1	1489	15625824	Kornilova	596	Spain
2	2873	15671591	Castiglione	439	Spain
3	3366	15764431	Chinwenma	671	Spain
4	4167	15654562	Ma	850	Spain
5	6280	15608338	Chiemenam	757	Spain
6	1255	15610383	Dumetolisa	628	France
7	2125	15603851	Galkin	704	France
8	2453	15704442	Fleming	672	France
9	2500	15634974	Seppelt	614	France
10	2542	15679770	Smith	611	France
11	3153	15724161	Sutton	644	France
12	3842	15750778	Ponomarev	653	France
13	5138	15601594	Ifeanacho	698	France
14	5387	15668283	Gardiner	642	France
15	6876	15665283	Brookes	610	France
16	7258	15648681	Voronoff	747	France
17	7725	15673591	Oluchukwu	842	France
18	7730	15681007	Yen	850	France
19	8591	15633194	Osborne	771	France
20	8684	15593929	Christopher	697	France

The 'Properties' panel on the right shows the table is of type Delta and managed by a Compute instance.

URL: <https://onelake.dfs.fabric.microsoft.com/jlaWorkspace1/jlaLakehouse.Lakehouse/Tables/churn>
ABFS path: abfss://jlaWorkspace1@onelake.dfs.fabric.microsoft.com/jlaLakehouse.Lakehouse/Tables/churn

Azure Machine Learning

1. Create Compute Instance – assign it managed identity
2. Create Notebook
1. Reference delta table using abfss path

The screenshot shows the Azure Machine Learning Studio interface. On the left, the 'Notebooks' section shows a notebook named 'jeffrey.ipynb' is running on 'jlaCompute1'. The notebook code is as follows:

```
1 pip install mltable
...
1 from mltable import from_delta_lake
2 url = "abfss://jlaWorkspace1@onelake.dfs.fabric.microsoft.com/jlaLakehouse.Lakehouse/Tables/churn"
3 df2 = from_delta_lake(url).to_pandas_dataframe()
```

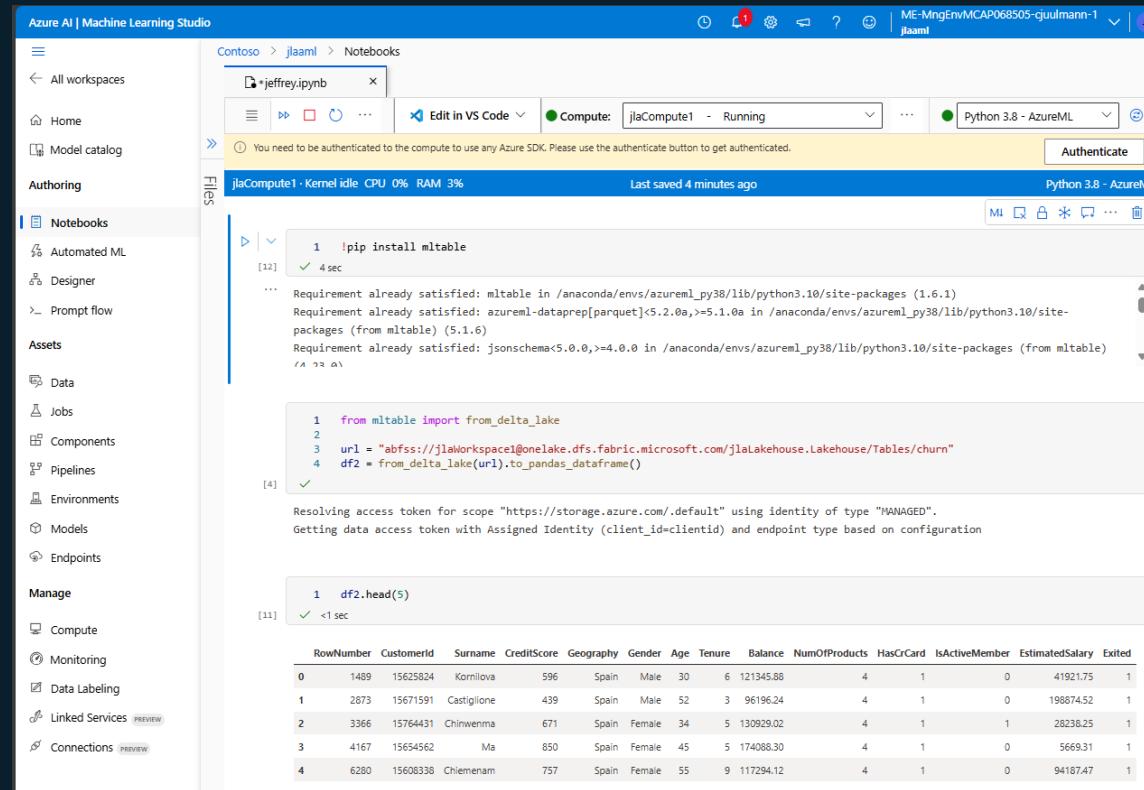
The output shows the data being loaded:

```
1 df2.head(5)
...
RowNumber CustomerId Surname CreditScore Geography Gender Age Tenure Balance NumOfProducts HasCrCard IsActiveMember EstimatedSalary Exited
0 1489 15625824 Kornilova 596 Spain Male 30 6 121345.88 4 1 0 41921.75 1
1 2873 15671591 Castiglione 439 Spain Male 52 3 96196.24 4 1 0 198874.52 1
2 3366 15764431 Chinwenma 671 Spain Female 34 5 130929.02 4 1 1 1 28238.25 1
3 4167 15654562 Ma 850 Spain Female 45 5 174088.30 4 1 0 5669.31 1
4 6280 15608338 Chiemenam 757 Spain Female 55 9 117294.12 4 1 0 94187.47 1
```

There is no need to register onelake as data store

Prove it's not needed:

It's not necessary to register the datastore to reference fabric tables



```

1 !pip install mltable
[12]:    ✓ 4 sec
...
Requirement already satisfied: mltable in /anaconda/envs/azureml_py38/lib/python3.10/site-packages (1.6.1)
Requirement already satisfied: azureml-dataprep[parquet]<5.2.0,>=5.1.0a in /anaconda/envs/azureml_py38/lib/python3.10/site-packages (from mltable) (5.1.6)
Requirement already satisfied: jsonschema<5.0.0,>=4.0.0 in /anaconda/envs/azureml_py38/lib/python3.10/site-packages (from mltable)
[13]:   ✓ 0 sec

1 from mltable import from_delta_lake
2
3 url = "abfss://jlaikeworkspace@onelake.dfs.fabric.microsoft.com/jlaLakehouse.Lakehouse/Tables/churn"
4 df2 = from_delta_lake(url).to_pandas_dataframe()

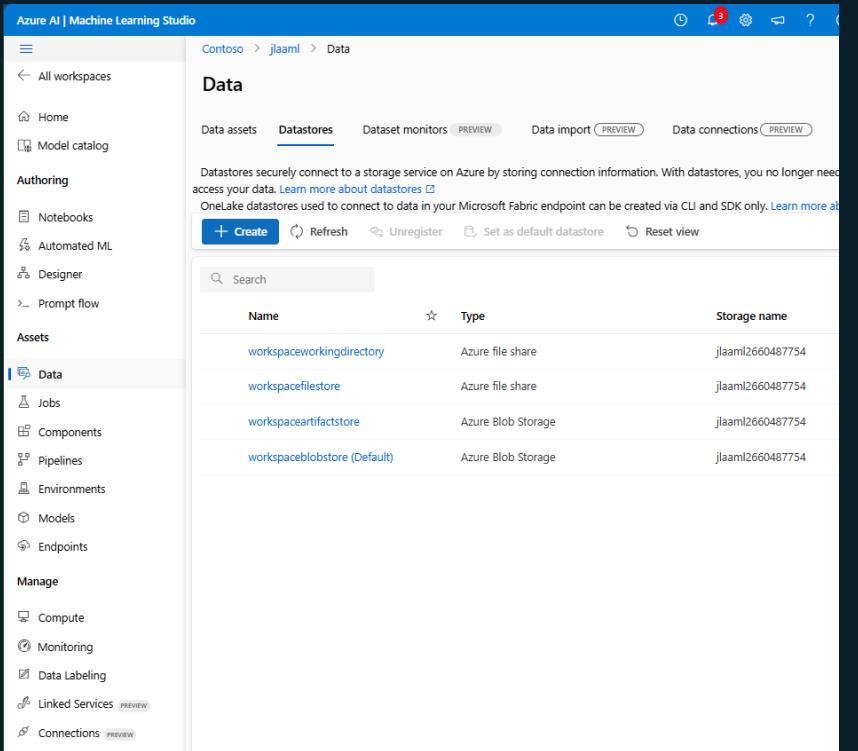
[4]:   ✓ 0 sec

Resolving access token for scope "https://storage.azure.com/.default" using identity of type "MANAGED".
Getting data access token with Assigned Identity (client_id=clientid) and endpoint type based on configuration

1 df2.head(5)
[11]:  ✓ <1 sec

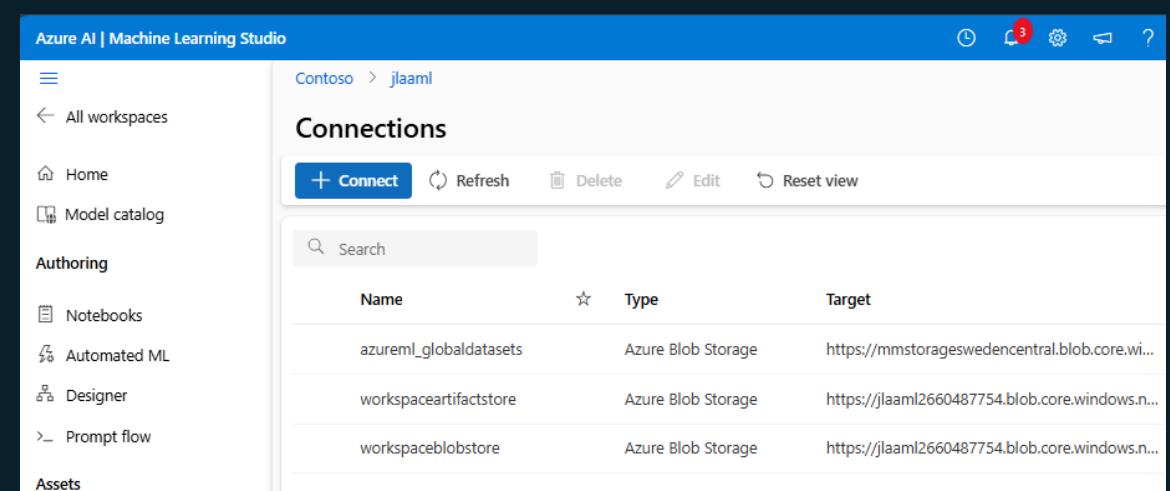
```

RowNumber	CustomerID	Surname	CreditScore	Geography	Gender	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	EstimatedSalary	Exited	
0	1489	15625824	Kornilova	596	Spain	Male	30	6	121345.88	4	1	0	41921.75	1
1	2873	15671591	Castiglione	439	Spain	Male	52	3	96196.24	4	1	0	196874.52	1
2	3366	15764431	Chinwenna	671	Spain	Female	34	5	130929.02	4	1	1	28238.25	1
3	4167	15654562	Ma	850	Spain	Female	45	5	174086.30	4	1	0	5669.31	1
4	6280	15608338	Chiemenam	757	Spain	Female	55	9	117294.12	4	1	0	94187.47	1



Datastores securely connect to a storage service on Azure by storing connection information. With datastores, you no longer need to access your data. [Learn more about datastores](#) [OneLake datastores used to connect to data in your Microsoft Fabric endpoint can be created via CLI and SDK only. Learn more about OneLake datastores](#)

Name	Type	Storage name
workspaceworkingdirectory	Azure file share	jlaaml2660487754
workspacefilestore	Azure file share	jlaaml2660487754
workspaceartifactstore	Azure Blob Storage	jlaaml2660487754
workspaceblobstore (Default)	Azure Blob Storage	jlaaml2660487754



Connections

Name	Type	Target
azureml_globaldatasets	Azure Blob Storage	https://mmstorageswedcentral.blob.core.windows.net/
workspaceartifactstore	Azure Blob Storage	https://jlaaml2660487754.blob.core.windows.net/
workspaceblobstore	Azure Blob Storage	https://jlaaml2660487754.blob.core.windows.net/

Load specific snapshot of delta table as mltable

The screenshot shows a Jupyter Notebook interface within the Azure AI Machine Learning Studio. The notebook is titled 'jeffrey.ipynb' and is currently running on a compute instance named 'jlaCompute1'. The code cell contains the following Python code:

```
1 import mltable
2
3 # create an MLTable from the data files
4 tbl = mltable.from_delta_lake("abfss://jlaworkspace1@onelake.dfs.fabric.microsoft.com/jlalakehouse.Lakehouse/Tables/churn", timestamp_as_of="2022-10-01T00:00:00Z")
5
6 # show the first 5 records
7 tbl.show(5)
```

The output of this cell shows a table with 5 rows of data:

RowNumber	CustomerID	Surname	CreditScore	Geography	Gender	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	EstimatedSalary	Exited	
0	1469	15625624	Kornilova	596	Spain	Male	30	6	121345.88	4	1	0	41921.75	1
1	2873	15671591	Castiglione	439	Spain	Male	52	3	96196.24	4	1	0	198974.52	1
2	3366	15764431	Chinewenna	671	Spain	Female	34	5	130929.02	4	1	1	28238.25	1
3	4167	15654562	Ma	850	Spain	Female	45	5	174083.30	4	1	0	5669.31	1
4	6280	15608338	Chinenam	757	Spain	Female	55	9	117294.12	4	1	0	94187.47	1

The next cell in the notebook contains the following code:

```
1 # save the data loading steps in an MLTable file
2
3 tbl.save("./covid")
```

The output of this cell shows the paths and transformations used to save the data:

```
paths:
- folder: abfss://jlaworkspace1@onelake.dfs.fabric.microsoft.com/jlalakehouse.Lakehouse/Tables/churn
transformations:
- read_delta_lake:
  - include_path_column: false
  - timestamp_as_of: '2022-10-01T00:00:00Z'
type: mltable
```

The final cell in the notebook contains the following code:

```
1 with open("./covid/MLTable", "r") as f:
2     print(f.read())
```

The output of this cell shows the contents of the saved MLTable file:

```
paths:
- folder: abfss://jlaworkspace1@onelake.dfs.fabric.microsoft.com/jlalakehouse.Lakehouse/Tables/churn
transformations:
- read_delta_lake:
  - include_path_column: false
  - timestamp_as_of: '2022-10-01T00:00:00Z'
type: mltable
```

Creating Data Assets from files inside of Fabric Lakehouse

The screenshot shows the 'Connections' page in Azure AI Machine Learning Studio. It lists four connections:

Name	Type	Target	Authentication type	Added by
jeffreylakehouse	Microsoft OneLake	https://onelake.dfs.fabric.microsoft.com/jaWor...	Microsoft Entra ID	Jeffrey Lai
azureml_globaldatasets	Azure Blob Storage	https://mmsstorageswedencentral.blob.core.wi...	SAS	779301c0-18b2-4cdc-8...
workspaceartifactstore	Azure Blob Storage	https://jaaml2660487754.blob.core.windows.n...	Account key	System
workspaceblobstore	Azure Blob Storage	https://jaaml2660487754.blob.core.windows.n...	Account key	System

The screenshot shows the 'jeffreylakehouse' data store in Azure AI Machine Learning Studio. A file named 'churn.csv' is selected, and a context menu is open with options: Copy URI, Create as data asset, Copy usage code, and Download.

A modal dialog titled 'Create as data asset' is open, prompting for the following details:

- Name: myChurnFromFabricLakehouseCsv
- Description: Data asset description
- Type: File (uri_file)
- Datastore URI: azureml://subscriptions/21cccd1c2-d366-48b0-80d3-ee1a5bfd1...

At the bottom right of the modal are 'Create' and 'Cancel' buttons.

1. Register Lakehouse as Data store

2. Browse files in Data store
3. Choose the file, and "Create as data asset"

The screenshot shows the details page for the data asset 'myChurnFromFabricLakehouseCsv'. It includes tabs for Details, Consume, and Jobs. The Details tab displays the following information:

- Usage: Interactive Development
- Pandas code snippet:

```
import pandas as pd
from azure.ai.ml import MLClient
from azure.identity import DefaultAzureCredential
ml_client = MLClient.from_config(DefaultAzureCredential())
data_asset = ml_client.data.get("myChurnFromFabricLakehouseCsv", version="1")
```

- Dataset tutorials:

 - What is Azure Machine Learning CLI & Python SDK v2?
 - Accessing data store documentation during interactive development
 - Interactive data wrangling with Apache Spark in Azure Machine Learning
 - Submit Spark jobs in Azure Machine Learning
 - Access data in a job