

---

# DEPLOYING MACHINE LEARNING MODELS IN MICROSOFT AZURE ML WORKSPACE

---

THAPELO LENZI



14 JUNE 2024  
METTUS/TESSERAI AI ACADEMY  
MLOPS

## Overview

This document provides a brief guide on training and deploying a machine learning model using Microsoft Azure, aimed at data scientists, ml engineers, and/or AI enthusiasts like me. The project makes use of the diabetes dataset from Kaggle, covering steps from data extraction and preparation, through model training and evaluation, to deployment and monitoring. The dataset in use contains several independent (medical predictor variables), with only one target dependent variable. Its main objective is to diagnostically predict whether a patient is diabetic based on the predictor variables. The process involves using Azure ML Studio to build and refine the model and deploying it using Azure's deployment tools.

(<https://www.kaggle.com/datasets/akshaydattatraykhare/diabetes-dataset>)

The system architecture for this project uses Microsoft Azure's suite of tools to well-organize the machine learning workflow. The diabetes dataset is stored in Azure Blob Storage, accessed and processed in the Azure ML Workspace (the centre for all the activities). Compute resources, such as Compute Instances, provide scalable processing power for model training and evaluation. Trained models are registered in Azure's Model Registry for version control and easy deployment. The model is deployed using Azure Kubernetes Services or Azure Container Instances (used in this project), creating a web service with RESTful API endpoints for real-time predictions. Performance and health of the deployed model is monitored using Azure Monitor and Application Insights, which ensures maintenance and updates.

## Azure Workspace

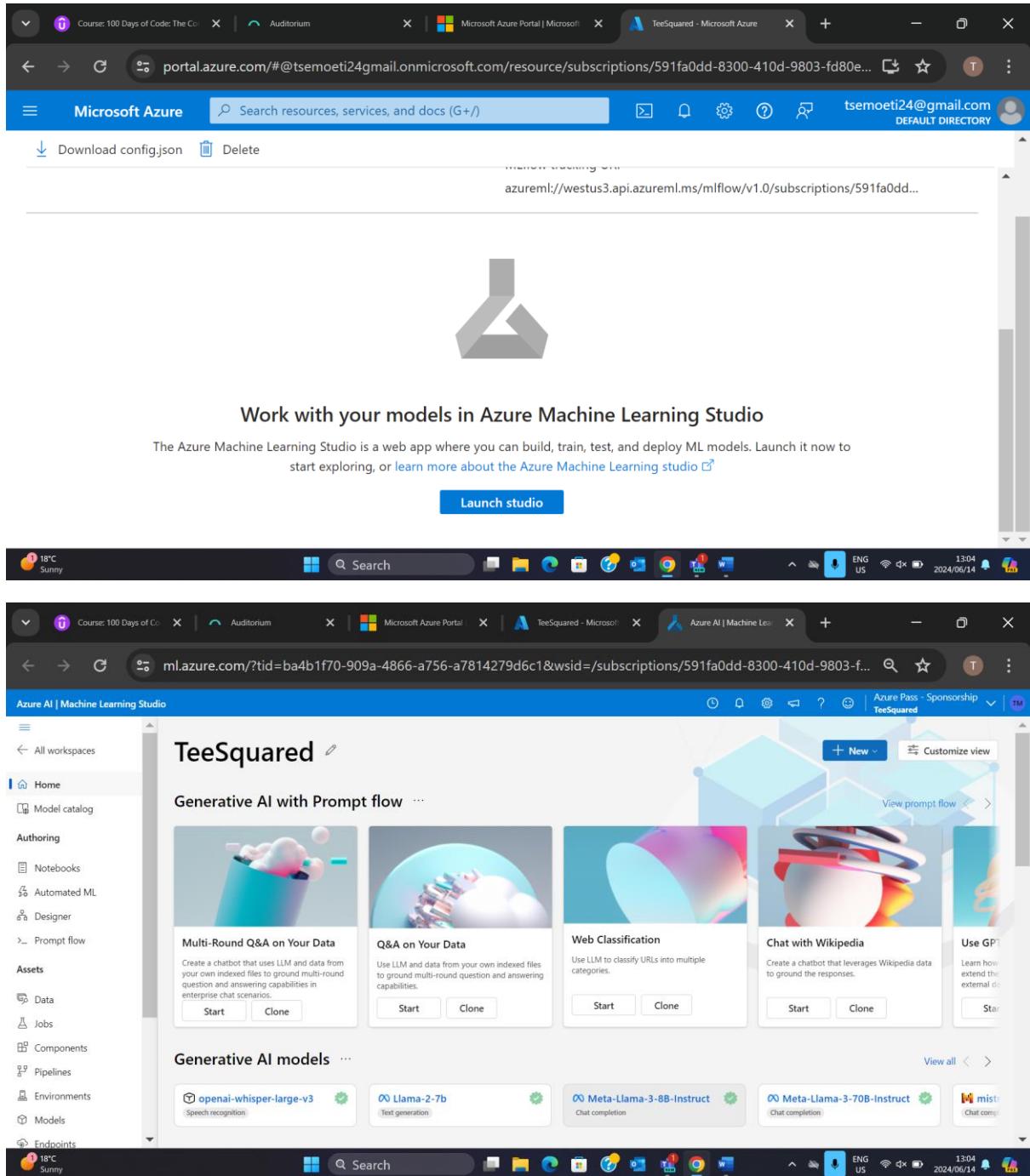
Setting up an Azure Machine Learning Workspace to organize and manage all the projects. Firstly, creating a resource group that will manage all my workspaces.

The screenshot shows the Microsoft Azure Portal interface. The user is navigating through the 'Resource groups' section under the 'Essentials' category. A new resource group named 'Lenzi' has been created, as indicated by the notification on the right side of the screen. The notification states: 'Resource group created' and 'Creating resource group 'Lenzi' in subscription 'Azure Pass - Sponsorship' succeeded.' The portal also displays various filtering and sorting options for the resource list.

The created Resource Group is Lenzi and in it, a workspace named TeeSquared is created.

The screenshot shows the Microsoft Azure Portal interface, specifically the 'TeeSquared' workspace page. The workspace was created in the 'Lenzi' resource group, located in 'West US 3' and associated with the 'Azure Pass - Sponsorship' subscription. The workspace configuration includes links to its 'Studio web URL' (<https://ml.azure.com?tid=ba4b1f70-909a-4866-a756-a7814279d6c1&...>), 'Container Registry' (link to teesquared1031018114), 'Key Vault' (link to teesquared1031018114), 'Application Insights' (link to teesquared1662002410), and 'MLflow tracking URI'. The workspace also provides options to 'Download config.json' or 'Delete' it.

## Navigating to Azure ML studio.



Now, one can access all the tools within the workspace.

## Data Preparation

The dataset is loaded into Azure Blob Storage for easy access using the Data tab in Azure Workspace.

**Create data asset**

**Data type**

Name: Diabetes

Description: Data asset description

Type: Tabular

**Use cases for data types**

**When should I use File type?**

The File type is recommended in most scenarios when you are working with a single data file of any type (including tabular data). This type allows you to specify a file location by URI in a storage location on your local computer, an attached Datastore, blob/ADLS storage, or a publicly available http(s) location. There are many types of supported URIs. In the Azure Machine Learning CLI v2 or Python SDK v2, this data type is called uri\_file. [Learn more about the uri\\_file type](#)

**When should I use Folder type?**

The Folder type has all the same capabilities and use cases as the File type, but is used when specifying a folder location. In the Azure Machine Learning CLI v2 or Python SDK v2, this data type is called uri\_folder. [Learn more about the uri\\_folder type](#)

**Next**

**Create data asset**

**Data source**

**From local files**

**From Azure storage**

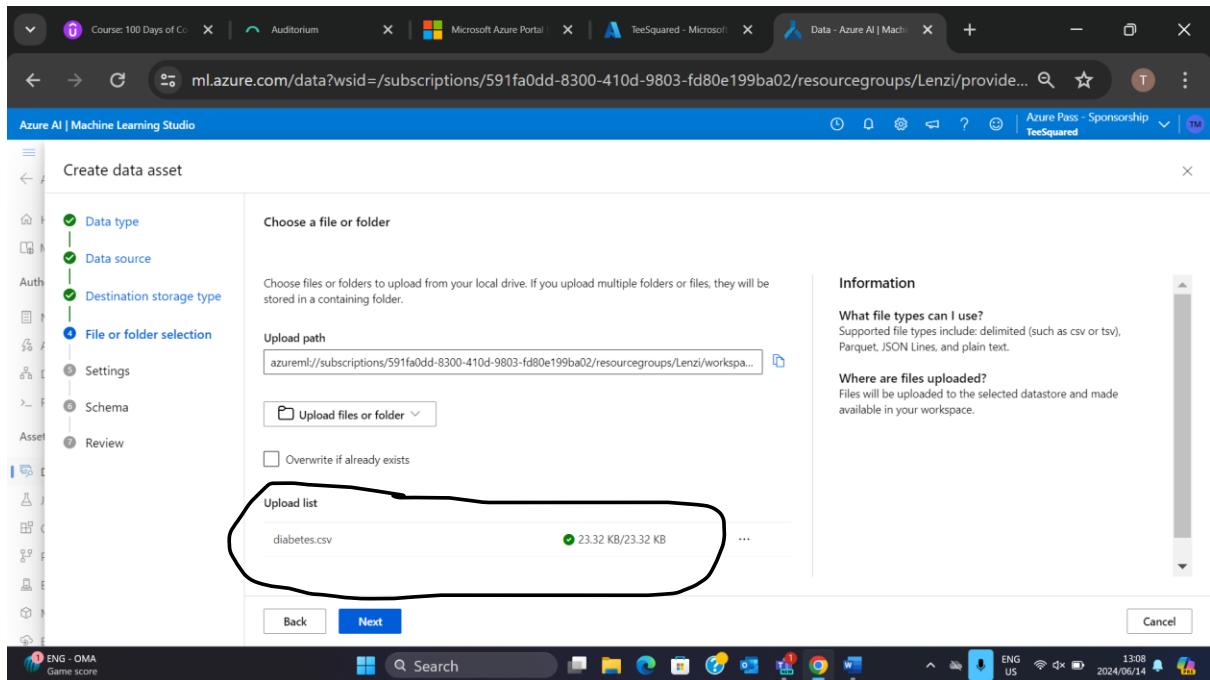
**From SQL databases**

**From web files**

**From Azure Open Datasets**

**Back**

**Next**



The screenshot shows the 'Data' assets page in the Azure AI | Machine Learning Studio. The left sidebar includes sections for All workspaces, Home, Model catalog, Authoring (Notebooks, Automated ML, Designer, Prompt flow), Assets (Data, Jobs, Components, Pipelines, Environments, Models, Endpoints), and a status bar showing 'FTSE jse 40 0.71%'. The main area displays a table of data assets. The table has columns: Name, Source, Version, Created on, Modified on, Type, Properties, and Created by. One row is highlighted with a black oval, showing 'Diabetes' as the name, 'This workspace' as the source, Version 1, and created on Jun 14, 2024 at 1:09 PM by Tselane Moeti. There are also buttons for '+ Create', 'Refresh', 'Archive', and 'Reset view'.

Now, the data is in storage.

## Compute Instance

For efficient model training, provision a Compute Instance. In the Compute Instance applications, the JupyterLab application is used to train the model using the Python 3.8-AzureML kernel.

The screenshot shows the Azure AI | Machine Learning Studio interface. On the left, there's a sidebar with sections like Automated ML, Assets, Pipelines, and Compute. The main area is titled 'Compute' and shows a table of 'Compute instances'. One instance, 'tsemoeti241', is highlighted and has a black oval drawn around its 'Applications' column entry.

## JupyterLab Environment

### Necessary libraries

The screenshot shows a JupyterLab interface with a code editor. The code cell contains the following Python code:

```
[1]: from azureml.core.workspace import Workspace
      from azureml.core.dataset import Dataset
      from azureml.core.experiment import Experiment
      from sklearn.model_selection import train_test_split
      from azureml.train.automl import AutoMLConfig
      from azureml.widgets import RunDetails
```

In order to read diabetes dataset in the JupyterLab environment, the workspace is called in the JupyterLab environment.

The screenshot shows a JupyterLab interface with a code editor. The code cell contains the following Python code:

```
[7]: ws = Workspace.from_config()
      print("Notebook connected to Workspace.")

Notebook connected to Workspace.
```

## Extracting the dataset from the workspace.

The screenshot shows a Jupyter Notebook interface with the title "Untitled.ipynb". The notebook is connected to a workspace, and the kernel is "Python 3.8 - AzureML". The code cell [8] retrieves details of the dataset from the workspace. Cells [9] through [12] demonstrate how to call the dataset by name, convert it to a pandas DataFrame, and view the first 5 rows. Cell [13] shows how to get a list of column titles for easier analysis.

```
[8]: # getting details of the dataset in the workspace
ws.datasets

[9]: {'Diabetes': DatasetRegistration(id='cd5ab4de-51cc-455e-990c-0de59e600457', name='Diabetes', version=1, description='', tags={})}

[10]: # Calling the dataset by name using the Dataset package
dia_ds = Dataset.get_by_name(workspace = ws, name = "Diabetes")

[11]: #the dataset as a dataframe
dia_df = dia_ds.to_pandas_dataframe()

[12]: # Calling the first 5 rows of the dataset
dia_df.head()

[12]: # Calling the first 5 rows of the dataset
dia_df.head()

[12]: Pregnancies Glucose BloodPressure SkinThickness Insulin BMI DiabetesPedigreeFunction Age Outcome
[12]: 0 6 148 72 35 0 33.6 0.627 50 1
[12]: 1 1 85 66 29 0 26.6 0.351 31 0
[12]: 2 8 183 64 0 0 23.3 0.672 32 1
[12]: 3 1 89 66 23 94 28.1 0.167 21 0
[12]: 4 0 137 40 35 168 43.1 2.288 33 1

[13]: # Getting a list of the column titles for easier analysis
dia_df.columns

[13]: Index(['Pregnancies', 'Glucose', 'BloodPressure', 'SkinThickness', 'Insulin', 'BMI', 'DiabetesPedigreeFunction', 'Age', 'Outcome'], dtype='object')
```

The above dataset is the clean version of the diabetes dataset, thus there is no need for the cleaning stage. So one can go ahead to split and train the data.

## Data Training

70% of the dataset will be used to train the model and 30% will be used to test it.

The screenshot shows a Jupyter Notebook interface with the title "Untitled.ipynb". The notebook is connected to a workspace, and the kernel is "Python 3.8 - AzureML". The code cell [14] splits the dataset into training and testing sets. Cells [15] and [16] show the shapes of the resulting training and testing datasets respectively.

```
[14]: x_train, x_test = train_test_split(dia_df, test_size = 0.3)

[15]: x_train.shape

[15]: (537, 9)

[16]: x_test.shape

[16]: (231, 9)
```

## AutoML and Algorithm specifications

Automl automates several important tasks in the machine learning workflow, which include feature engineering and algorithm selection. In the following snippet, by defining these parameters, automl efficiently manages the duration of training iterations and experiments, stopping the iterations early when the performance metrics are stable. The metric used for algorithm selection is “AUC\_weighted”. With the n\_cross\_validations parameter set to 5, ensures a thorough validation of models during training, improving their generalization capabilities.

```
[29]: (231, 9)

Automl and Experiment Settings

[30]: automl_settings = {"iteration_timeout_minutes":2,
                       "experiment_timeout_minutes":15,
                       "enable_early_stopping":True,
                       "primary_metric":'AUC_weighted',
                       "featurization": 'auto',
                       "n_cross_validations":5,
}
```

These configurations emphasise the role of AutoML in optimizing machine learning pipelines, making it easier to deploy accurate predictive models. In the case of this project, accurately predicting diabetes outcomes.

### Specifying task and algorithm

```
[32]: automl_config = AutoMLConfig(task='classification',
                                   debug_log='automl_errors.log',
                                   training_data = x_train,
                                   label_column_name = "Outcome",
                                   **automl_settings)
```

An experiment named Diabetes-Experiment is created in the workspace to track and manage all the model training runs

## Experiment for Deployment

```
[33]: experiment = Experiment(ws, "Diabetes-Experiment")
```

```
[*]: run = experiment.submit(automl_config, show_output=True)
```

No run\_configuration provided, running on local with default configuration  
Running in the active local environment.

Experiment	Id	Type	Status	Details Page	Docs Page
Diabetes-Experiment	AutoML_f595901b-e4a2-42c2-bda7-6e8469ec21e1	automl	Preparing	<a href="#">Link to Azure Machine Learning studio</a>	<a href="#">Link to Documentation</a>

Current status: DatasetEvaluation. Gathering dataset statistics.  
Current status: FeaturesGeneration. Generating features for the dataset.  
Current status: DatasetFeaturization. Beginning to fit featurizers and featurize the dataset.  
Current status: DatasetFeaturizationCompleted. Completed fit featurizers and featurizing the dataset.

Current status: DatasetEvaluation. Gathering dataset statistics.  
Current status: FeaturesGeneration. Generating features for the dataset.  
Current status: DatasetFeaturization. Beginning to fit featurizers and featurize the dataset.  
Current status: DatasetFeaturizationCompleted. Completed fit featurizers and featurizing the dataset.  
Current status: DatasetCrossValidationSplit. Generating individually featurized CV splits.

```
2024/06/14 12:07:02 WARNING mlflow.sklearn: Model was missing function: predict. Not logging python_function flavor!
```

\*\*\*\*\*  
DATA GUARDRAILS:

TYPE: Class balancing detection  
STATUS: PASSED  
DESCRIPTION: Your inputs were analyzed, and all classes are balanced in your training data.  
Learn more about imbalanced data: <https://aka.ms/AutomatedMLImbalancedData>

\*\*\*\*\*  
TYPE: Missing feature values imputation  
STATUS: PASSED

```

13 StandardScalerWrapper ExtremeRandomTrees      0:00:42      0.7554  0.8192
14 StandardScalerWrapper RandomForest           0:00:44      0.7983  0.8192
15 MaxAbsScaler LightGBM                      0:00:41      0.8195  0.8195
16 MaxAbsScaler LogisticRegression            0:00:41      0.7928  0.8195
17 StandardScalerWrapper ExtremeRandomTrees      0:00:46      0.8197  0.8197
18 VotingEnsemble                           0:00:46      0.8347  0.8347
19 StackEnsemble                            0:00:46      0.8303  0.8347
Stopping criteria reached at iteration 20. Ending experiment.
*****
Current status: BestRunExplainModel. Best run model explanations started
Current status: ModelExplanationDataSetSetup. Model explanations data setup completed
Current status: PickSurrogateModel. Choosing LightGBM as the surrogate model for explanations
Current status: EngineeredFeatureExplanations. Computation of engineered features started
2024-06-14:12:24:27,38 INFO [explanation_client.py:334] Using default datastore for uploads
Current status: EngineeredFeatureExplanations. Computation of engineered features completed
Current status: RawFeaturesExplanations. Computation of raw features started
Current status: RawFeaturesExplanations. Computation of raw features completed
Current status: BestRunExplainModel. Best run model explanations completed
*****

```

Experiment	Latest job	Last submitted	Created
Diabetes-Experiment	nifty_collar_32vqq797	Jun 14, 2024 2:06 PM	Jun 14, 2024 2:05 PM

The following shows all the jobs and child jobs under the created experiment job.

A screenshot of a Microsoft Edge browser window displaying the Azure Machine Learning Studio interface. The URL in the address bar is `ml.azure.com/experiments/id/be8ac735-5717-4a59-ab80-7bc830613ac7?wsid=/subscriptions/591fa0dd-8300-41...`. The page title is "Azure AI | Machine Learning Studio". The main content area shows a "Diabetes-Experiment" with one job listed:

Display name (1 visualized)	Parent job name	Status	Created on	Duration	Created by	Cor
nifty_colar_32vqq797 (20)		Completed	Jun 14, 2024 2:06 PM	16m 46s	Tselane Moeti	loca

The left sidebar navigation includes sections for All workspaces, Home, Model catalog, Authoring (Notebooks, Automated ML, Designer, Prompt flow), Assets (Data, Jobs, Components, Pipelines, Environments, Models, Endpoints), and a weather widget showing 19°C Sunny.

Screenshot of the Azure Machine Learning Studio interface showing a completed experiment named "nifty\_colar\_32vqq797".

**Experiment Overview:**

- Completed:** Yes
- Warning:** Experiment timeout reached, hence experiment stopped. Current experiment timeout: 0 hour(s) 15 minute(s).
- Created by:** Tselane Moeti
- Job type:** Automated ML
- Experiment:** Diabetes-Experiment
- Arguments:** None
- Compute duration:** 16m 45.94s
- Compute target:** local
- Name:** AutoML\_f595901b-e4a2-42c2-bda7-6e8469ec21e1

**VotingEnsemble**

- Ensemble details:** View ensemble details
- AUC weighted:** 0.83469 | View all other metrics
- Sampling:** 100.00 %
- Registered models:** No registration yet
- Deploy status:** No deployment yet

**Run summary:**

- Task type:** Classification | View configuration settings
- Featureization:** Auto

**Child jobs:**

Display name	Parent job name	Status	Created on	Duration	Created by	Compute target	Job type	Tags
tidy_apple_df25p9d3	AutoML_f595901b-e4...	Completed	Jun 14, 2024 2:07 PM	43s	Tselane Moeti	local		
honest_book_ll5t52dp	AutoML_f595901b-e4...	Completed	Jun 14, 2024 2:07 PM	1m 7s	Tselane Moeti	local		
nice_grass_j936cbx0	AutoML_f595901b-e4...	Completed	Jun 14, 2024 2:08 PM	42s	Tselane Moeti	local		
dreamy_atemoya_311bf3kt	AutoML_f595901b-e4...	Completed	Jun 14, 2024 2:09 PM	42s	Tselane Moeti	local		
lemon_sponge_2407vz0r	AutoML_f595901b-e4...	Completed	Jun 14, 2024 2:10 PM	44s	Tselane Moeti	local		
patient_frame_0pr57hfh	AutoML_f595901b-e4...	Completed	Jun 14, 2024 2:11 PM	58s	Tselane Moeti	local		
quirky_rhubarb_k2247ylf	AutoML_f595901b-e4...	Completed	Jun 14, 2024 2:12 PM	43s	Tselane Moeti	local		
orange_rice_91124lbn	AutoML_f595901b-e4...	Completed	Jun 14, 2024 2:12 PM	59s	Tselane Moeti	local		

The screenshot shows the Azure AI | Machine Learning Studio interface. The left sidebar has sections for All workspaces, Home, Model catalog, Authoring (Notebooks, Automated ML, Designer, Prompt flow), Assets (Data, Components, Pipelines, Environments, Models, Endpoints), and a selected Jobs section. The main content area shows a completed job named "tidy\_apple\_dfz5p9d3". The job details include its status as "Completed", creation date (Jun 14, 2024 2:07 PM), start time (Jun 14, 2024 2:07 PM), duration (42.76s), compute duration (42.76s), and compute target (local). The "Metrics" section displays performance scores: Accuracy (0.72816), AUC macro (0.77943), AUC micro (0.79805), AUC weighted (0.77943), Average precision score macro (0.75005), and Average precision score micro (0.80237). The top navigation bar shows the URL ml.azure.com/experiments/id/be8ac735-5717-4a59-ab80-7bc830613ac7/runs/AutoML\_f595901b-e4a2-42c2-bda... and the browser title tidy\_apple.

The outputs can be retrieved and reviewed from all the model training runs to identify the best-performing model. The artifacts that can be examined to evaluate model performance include the score files. The outputs can be graphically displayed as well.

## Getting the Run output

```
[35]: best_run, model = run.get_output()
```

```
[36]: RunDetails(run).show()
```

AutoML\_f595901b-e4a2-42c2-bda7-6e8469ec21e1:

Status: Completed



Iteration Pipeline	Iteration metric	Best metric	Status	Duration	Start
--------------------	------------------	-------------	--------	----------	-------

Screenshot of a Jupyter Notebook cell titled "Getting the Run output". The cell contains two code snippets:

```
[35]: best_run, model = run.get_output()
[36]: RunDetails(run).show()
```

The output shows:

AutoML\_f595901b-e4a2-42c2-bda7-6e8469ec21e1:  
Status: Completed

A horizontal bar chart titled "Status" showing the completion status of 15 iterations. The bars are solid green, indicating completed status. The x-axis is labeled with iteration numbers 0, 2, 4, 6, 8, 10, and 12.

Iteration	Pipeline	Iteration metric	Best metric	Status
18	VotingEnsemble	0.83468818	0.83468818	Completed
19	StackEnsemble	0.83033946	0.83468818	Completed
17	StandardScalerWrapper, ExtremeRandomTrees	0.81974585	0.81974585	Completed
15	MaxAbsScaler, LightGBM	0.81950978	0.81950978	Completed
4	StandardScalerWrapper, LightGBM	0.81921348	0.81921348	Completed

Pages: 1 2 3 4 Next Last 5 per page

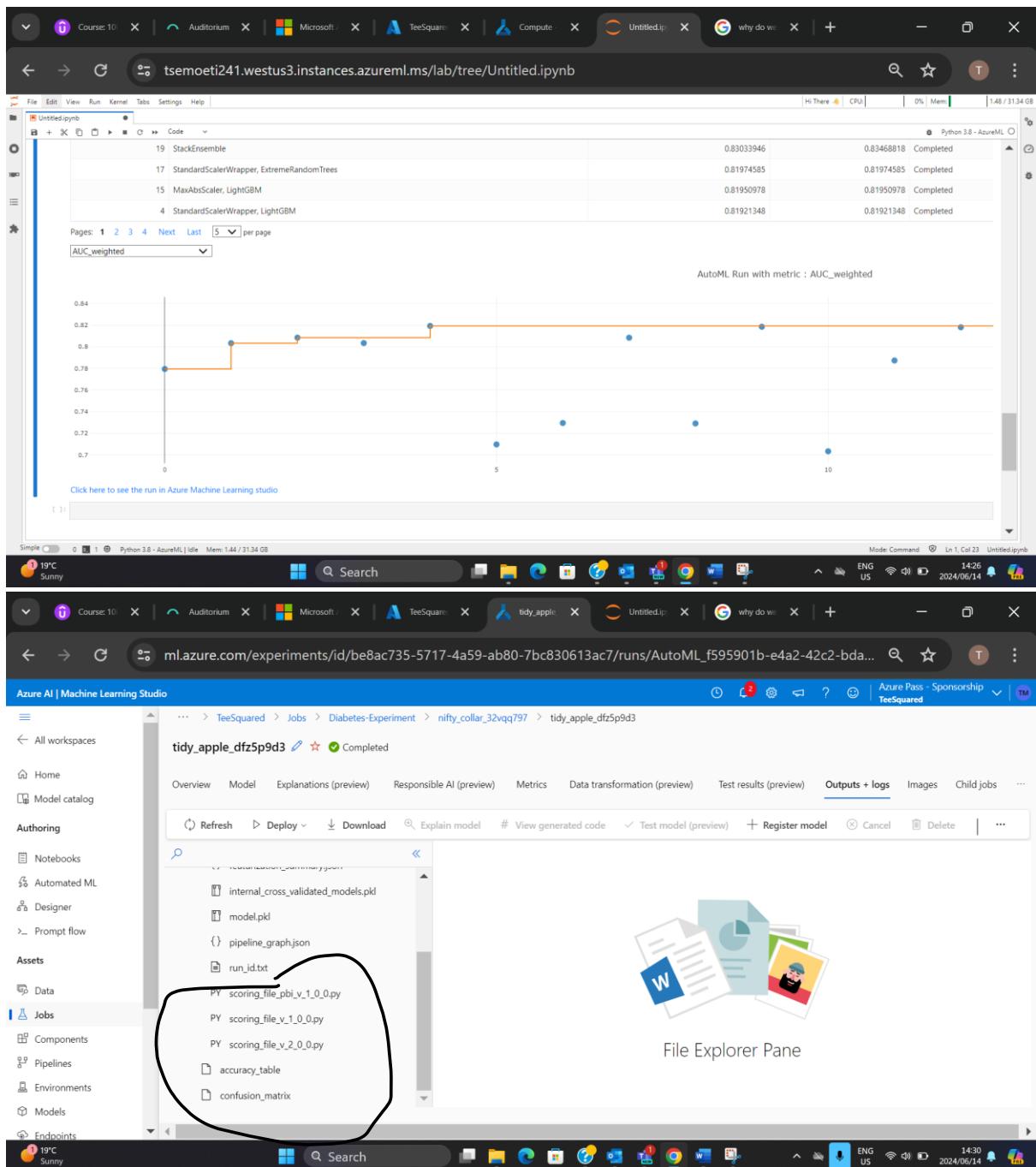
AUC\_weighted

AutoML Run with metric : AUC\_weighted

19°C Sunny

Python 3.8 - AzureML | Idle Mem: 1.44 / 31.34 GB

Mode: Command ENG US 14:26 2024/06/14 Untitled.ipynb



For version control and deployment, the trained model is registered in the Azure Machine Learning Model Registry.

[Click here to see the run in Azure Machine Learning studio](#)

## Registering the Model

```
[37]: model_name = best_run.properties["model_name"]
registered_name = run.register_model(model_name = model_name, description= "AutoML Diabetes", tags = None)
```

The screenshot shows the Azure AI | Machine Learning Studio interface. On the left, there's a sidebar with 'Automated ML' selected under 'Models'. The main area is titled 'Model List' and displays a table with one row:

Name	Version	Type	Source	Experiment	Job (Run ID)
AutoMLf595901be18	1	CUSTOM	This workspace	Diabetes-Experiment	AutoML_f595901b-e4a2-42c2-...

At the bottom of the browser window, there's a 'Share this window' button and a URL: <https://ml.azure.com/model/list?wsid=/subscriptions/591fa0dd-8300-410d-9803-fd80e199ba02/resourcegroups/Lenzi/providers/Microsoft.MachineLearningServices/workspaces/TeeSquared&tid=ba4b1170-909a-4866-a756-a78142796ec1>.

## Deployment

### Imports for Deployment

```
[39]: from azureml.core.model import InferenceConfig
from azureml.core.webservice import AciWebservice, Webservice
from azureml.core.model import Model
from azureml.core.environment import Environment
```

Creating and downloading the score file which contains the scoring script for handling predictions.

## Score.py File

```
40]: best_run.download_file("outputs/scoring_file_v_1_0_0.py", "Inference(score.py")
```

The screenshot shows a Jupyter Notebook interface in a browser window. A black oval highlights the 'Inference' folder in the left sidebar file tree, which contains a single file named 'score.py'. The main notebook area displays code for registering a model and imports for deployment. A 'Share this window' button is visible at the bottom right of the notebook area.

```
[37]: model_name = best_run.properties["model_name"]
registered_name = run.register_model(model_name = model_name, description= "AutoML Diabetes",tags = None)
```

```
[38]: from azureml.core.model import InferenceConfig
from azureml.core.webservice import AciWebservice, Webservice
from azureml.core.model import Model
from azureml.core.environment import Environment
```

### Score.py File

```
[40]: best_run.download_file("outputs/scoring_file_v_1_0_0.py", "Inference(score.py")
```

## Deploying

```
[42]: from azureml.automl.core.shared import constants

best_run.download_file(constants.CONDA_ENV_FILE_PATH, "myenv.yml")
env = Environment.from_conda_specification(name="myenv",file_path="myenv.yml")

inference_config = InferenceConfig(entry_script="inference(score.py", environment=env)
aciconfig = AciWebservice.deploy_configuration(cpu_cores=1, memory_gb=1, description="Diabetes classification")
service = Model.deploy(ws, "automl-diabetes", [registered_name], inference_config,aciconfig)
service.wait_for_deployment(True)

Tips: You can try get_logs(): https://aka.ms/debugimage#dockerlog or local deployment: https://aka.ms/debugimage#debug-locally to debug if deployment takes longer than 10 minutes.
Running
2024-06-14 12:42:32+00:00 Creating Container Registry if not exists..
2024-06-14 12:52:33+00:00 Registering the environment..
2024-06-14 12:52:39+00:00 Building image..
2024-06-14 13:11:10+00:00 Generating deployment configuration.
2024-06-14 13:11:11+00:00 Submitting deployment to compute..
2024-06-14 13:11:19+00:00 Checking the status of deployment automl-diabetes..
2024-06-14 13:13:17+00:00 Checking the status of inference endpoint automl-diabetes.
Succeeded
ACI service creation operation finished, operation "Succeeded"
```

## Establishing endpoints for real-time predictions.

The screenshot shows the Azure AI | Machine Learning Studio interface. The left sidebar has sections for Automated ML, Designer, Prompt flow, Assets (Data, Jobs, Components, Pipelines, Environments, Models), and Endpoints (selected). The main content area shows the 'Endpoints' page under 'Default Directory > TeeSquared > Endpoints'. It lists 'Real-time endpoints' (one entry: 'automl-diabetes'), 'Batch endpoints', 'Azure OpenAI', and 'Serverless endpoints'. A preview link is also present. The table details the endpoint's name, description, quota type, creation date, creator, update date, and compute type. The 'automl-diabetes' entry has a name, a red star icon, and a description of 'Diabetes classification'. It was created on Jun 14, 2024 at 2:42 PM by Tselane Moeti, last updated on Jun 14, 2024 at 2:42 PM, and is of type Container. The status bar at the bottom shows the URL, weather (19°C, sunny), search bar, taskbar icons, and system info (ENG US, 15:15, 2024/06/14).

Name	Description	Quota type	Created on	Created by	Updated on	Compute type
automl-diabetes	Diabetes classification		Jun 14, 2024 2:42 PM	Tselane Moeti	Jun 14, 2024 2:42 PM	Container insta...

The screenshot shows the Azure AI | Machine Learning Studio interface. The left sidebar is titled "Automated ML" and includes sections for Designer, Prompt flow, Assets (Data, Jobs, Components, Pipelines, Environments, Models), Endpoints, Manage (Compute, Monitoring, Data Labeling, Linked Services, Connections), and a preview section for Compute, Monitoring, Data Labeling, and Connections.

The main content area displays the "automl-diabetes" endpoint under "Endpoints". The "Details" tab is selected. The "Endpoint attributes" section shows:

- Service ID: automl-diabetes
- Description: Diabetes classification
- Deployment state: Healthy
- Operation state: Succeeded
- Compute type: Container instance
- Created by: Tselane Moeti
- Model ID: AutoMLf595901be18:1

The "Properties" section shows:

- hasInferenceSchema: True
- hasHttps: False
- authEnabled: False

The "Tags" section indicates "No data".

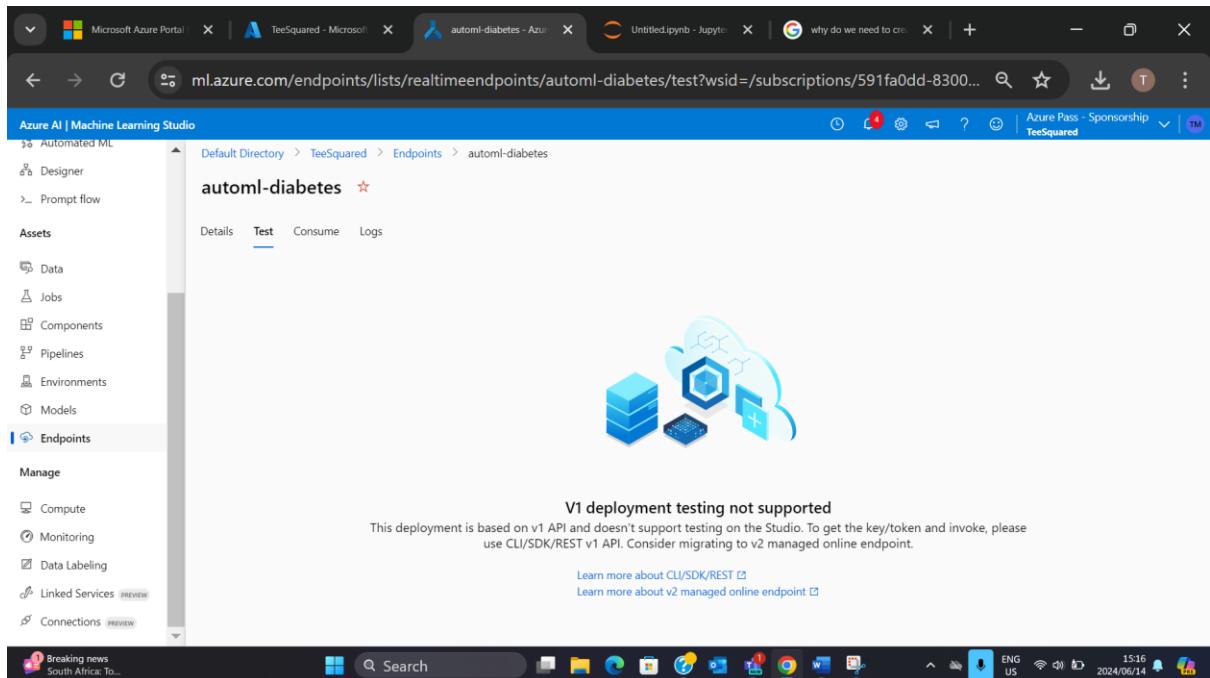
The second screenshot below it shows the same interface but with the "Test" tab selected. It displays the REST endpoint URL: <http://ba5a5897-c61f-4ae5-a1ce-26144e90ded7.westus3.azurecontainer.io/score>.

## The Deployment URL:

<http://ba5a5897-c61f-4ae5-a1ce-26144e90ded7.westus3.azurecontainer.io/score>

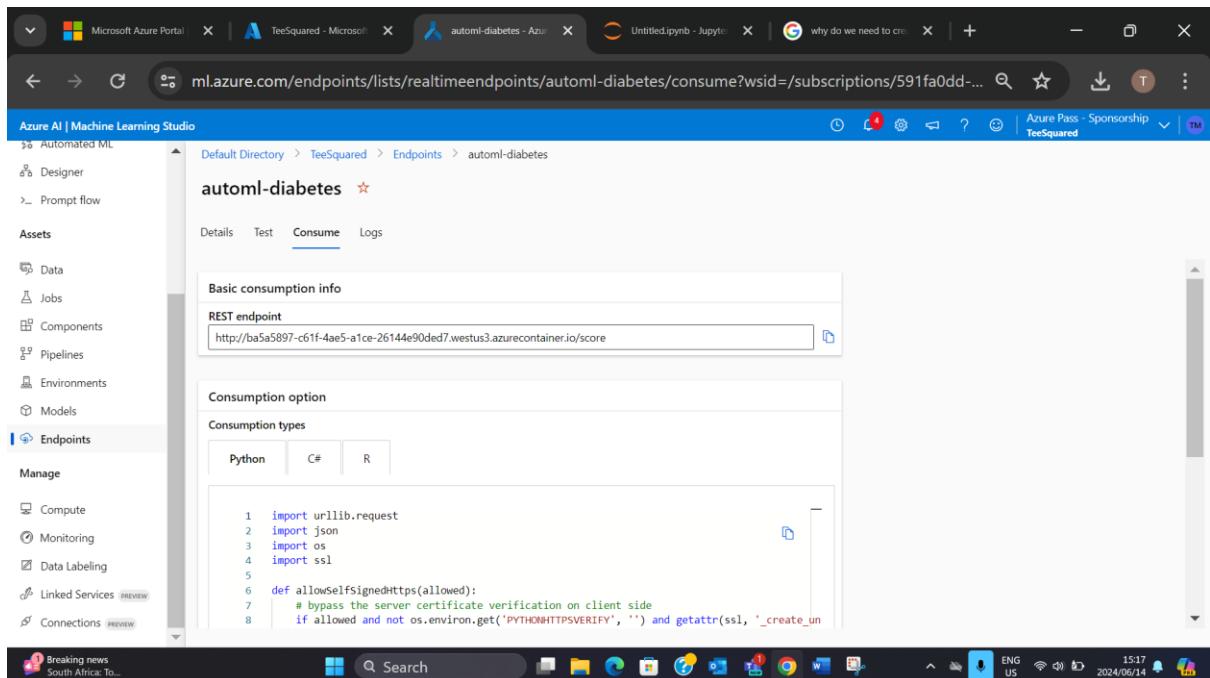
## Testing and Consuming

The model can be tested and



The screenshot shows the Azure AI | Machine Learning Studio interface. On the left, there's a sidebar with 'Automated ML' selected. Under 'Endpoints', 'automl-diabetes' is listed. Below it, there are tabs for 'Details', 'Test' (which is selected), 'Consume', and 'Logs'. A central area displays a cloud icon with data blocks. Below the icon, a message states 'V1 deployment testing not supported' and provides instructions for migrating to v2 managed online endpoint. At the bottom, there's a search bar and a taskbar.

The model can be tested using different codes in the endpoint>consume tab as shown in the snippet below.



The screenshot shows the same interface as above, but the 'Consume' tab is selected. In the 'Basic consumption info' section, the 'REST endpoint' field contains the URL 'http://ba5a5897-c61f-4ae5-a1ce-26144e90ded7.westus3.azurecontainer.io/score'. In the 'Consumption option' section, under 'Consumption types', the 'Python' tab is selected. Below it, a code editor shows a Python script:

```
1 import urllib.request
2 import json
3 import os
4 import ssl
5
6 def allowSelfSignedHttps(allowed):
7     # bypass the server certificate verification on client side
8     if allowed and not os.environ.get('PYTHONHTTPSVERIFY', '') and getattr(ssl, '_create_un
```

Microsoft Azure Portal | TeeSquared - Microsoft | automl-diabetes - Azure | Untitled.ipynb - Jupyter | why do we need to cre... | ml.azure.com/endpoints/lists/realtimeendpoints/automl-diabetes/consume?wsid=/subscriptions/591fa0dd-... | - | ☰ | ×

Azure AI | Machine Learning Studio

Default Directory > TeeSquared > Endpoints > automl-diabetes

## automl-diabetes

Details Test Consume Logs

Assets

- Data
- Jobs
- Components
- Pipelines
- Environments
- Models
- Endpoints**
- Manage
- Compute
- Monitoring
- Data Labeling
- Linked Services PREVIEW
- Connections PREVIEW

13 # Request data goes here  
14 # The example below assumes JSON formatting which may be updated  
15 # depending on the format your endpoint expects.  
16 # More information can be found here:  
17 # <https://docs.microsoft.com/azure/machine-learning/how-to-deploy-advanced-entry-script>  
18 data = (  
19 "data": [  
20 (  
21 "Age": 0,  
22 "BMI": 0.0,  
23 "BloodPressure": 0,  
24 "DiabetesPedigreeFunction": 0.0,  
25 "Glucose": 0,  
26 "Insulin": 0,  
27 "Pregnancies": "example\_value",  
28 "SkinThickness": 0  
29 ),  
30 ],  
31 "method": "predict"  
32 )  
33  
34 body = str.encode(json.dumps(data))

Breaking news South Africa: To... Search ENG US 15:17 2024/06/14