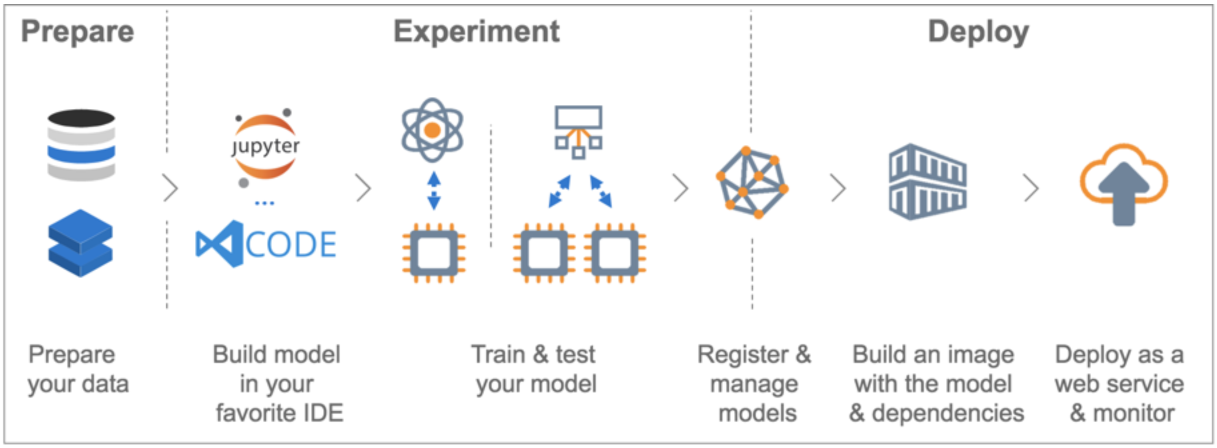
Day 2, Experience 3 - Better models made easy with Automated Machine Learning

Technology overview

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


Prepare
Experiment
Deploy
jupyter
CODE
Prepare your data
Build model in your favorite IDE
Train & test your model
Register & manage models
Build an image with the model & dependencies
Deploy as a web service & monitor

Azure Machine Learning overview

Azure Machine Learning service fully supports open-source technologies. So you can use tens of thousands of open-source Python packages with machine learning components. Examples are PyTorch, TensorFlow, and scikit-learn. Support for rich tools makes it easy to interactively explore and prepare data and then develop and test models. Examples are Jupyter notebooks or the Azure Machine Learning for Visual Studio Code extension.

By using Azure Machine Learning service, you can start training on your local machine and then scale out to the cloud. With many available compute targets, like Azure Machine Learning Compute and Azure Databricks, and with advanced hyperparameter tuning services, you can build better models faster by using the power of the cloud. When you have the right model, you can easily deploy it in a container such as Docker. So it's simple to deploy to Azure Container Instances or Azure Kubernetes Service. Or you can use the container in your own deployments, either on-premises or in the cloud. For more information, see the article on how to deploy and where.

You can manage the deployed models and track multiple runs as you experiment to find the best solution. After it's deployed, your model can return predictions in real time or asynchronously on large quantities of data. And with advanced machine learning pipelines, you can collaborate on all the steps of data preparation, model training and evaluation, and deployment.

Azure Machine Learning service also includes features that automate model generation and tuning to help you create models with ease, efficiency, and accuracy. Automated machine learning is the process of taking training data with a defined target feature, and iterating through combinations of algorithms and feature selections to automatically select the best model for your data based on the training scores. The traditional machine learning model development process is highly resource-intensive, and requires significant domain knowledge and time investment to run and compare the results of dozens of models. Automated machine learning simplifies this process by generating models tuned from the goals and constraints you defined for your experiment, such as the time for the experiment to run or which models to blacklist.

Scenario overview

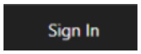
In this experience you will learn how the automated machine learning capability in Azure Machine Learning (AML) can be used for the life cycle management of the manufactured vehicles and how AML helps in creation of better vehicle maintenance plans. To accomplish this, you will train a Linear Regression model to predict the number of days until battery failure using Automated Machine Learning in Jupyter Notebooks.

Task 1: Train and evaluate a model using Azure Machine Learning

In this exercise, you will use compute resources provided by Azure Machine Learning to remotely train a set of models using Automated Machine Learning, evaluate performance of each model and pick the best performing model to deploy as a web service. You will perform this lab using Azure Notebooks.

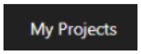
1. To start, open a new web browser window and navigate to https://notebooks.azure.com.

2. Select **Sign In** and then use your Microsoft Account to complete the sign in process.

   
   Sign In

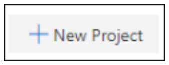
   The Sign In button

3. Dismiss the dialog to create the user ID (you will not need this). Within the Microsoft Azure Notebooks portal, select **My Projects** from the menu at the top.

   
   My Projects

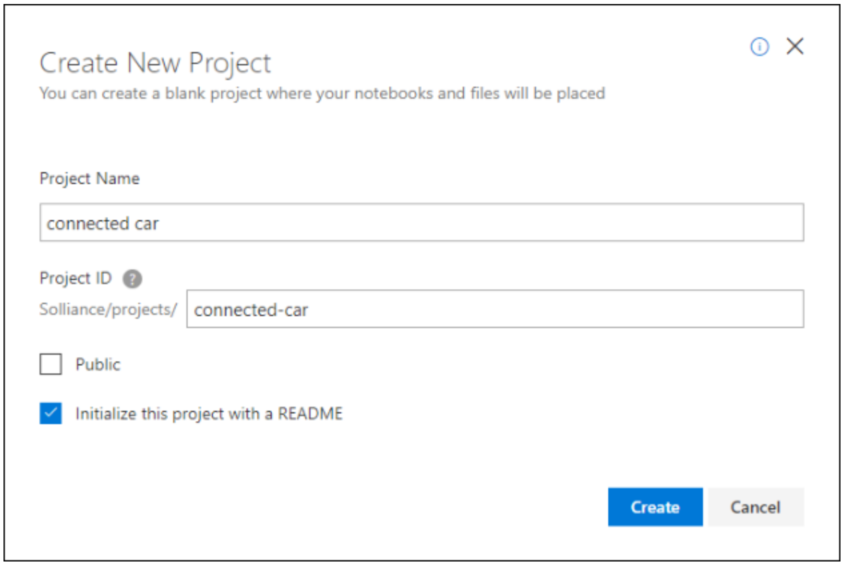
   The My Projects button

4. Then select **New Project**.



The New Project button

5. On the Create New Project dialog, provide a Project Name (this should be a user friendly description) and Project ID (this will form a part of the URL used to access this project in the browser) and uncheck Public. Select **Create**.



Create New Project

You can create a blank project where your notebooks and files will be placed

Project Name: connected car

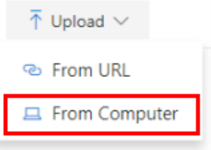
Project ID: Solliance/projects/ connected-car

□ Public

☑ Initialize this project with a README

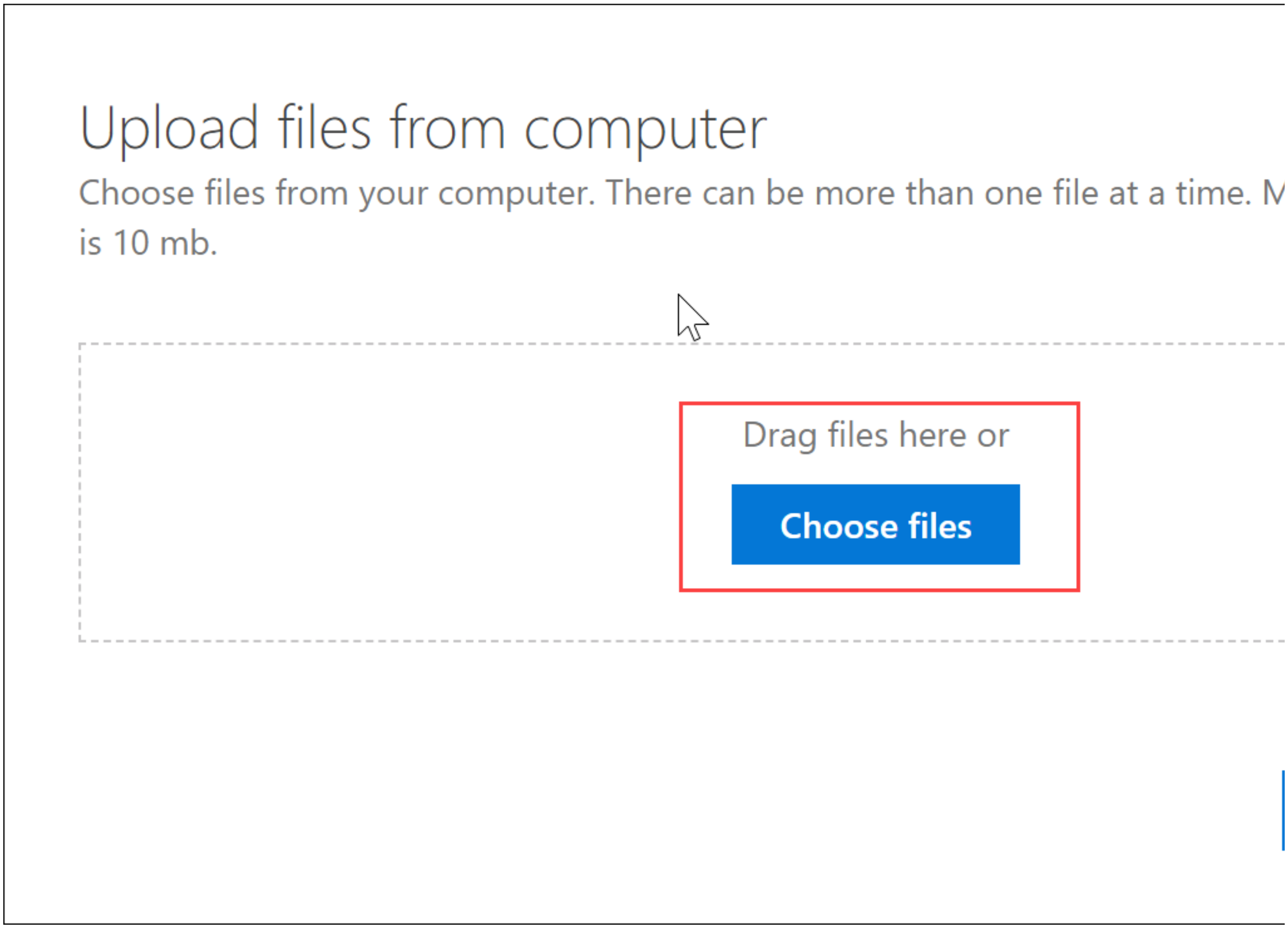
The Create New Project dialog

6. Select the **Upload** menu and then choose **From Computer**.



The Upload menu

7. In the Upload files from Computer dialog, select **Choose Files** and then select the `predict-battery-life-with-AML.ipynb` from lab starter files (C:-files\3). Select **Upload** to upload the notebook and then select **Done** to dismiss the dialog.

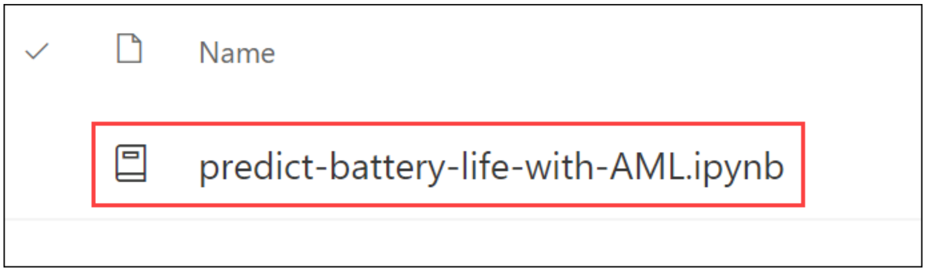


Upload files from computer

Choose files from your computer. There can be more than one file at a time. M
is 10 mb.

The Upload files from Computer dialog

8. In the listing, select the Notebook you just uploaded (predict-battery-life-with-AML.ipynb) to open it.



Select the notebook

9. Follow the instructions within the notebook to complete the experience.

Wrap-up

Congratulations on completing the Auto ML experience. In this experience you completed an end-to-end process for training a model and deploying the model into production, all from within a Jupyter Notebook hosted in Microsoft Azure Notebooks.

To recap, you experienced:

1. How to use Azure Machine Learning Auto ML to simplify the processs of getting to a performant model.
2. Using Auto ML to train multiple models locally as well by using remote capabilities provided by compute targets.
3. Capturing and querying the telemetry of training runs using an Experiment.
4. Retrieving the best model created from an Auto ML session.
5. Registering the best model with the Model Registry, which enables versioning and makes the model file available for deployment to a web service.
6. Creating and deploying a web service to Azure Kubernetes Service that uses the best model for scoring.

Additional resources and more information

To learn more about the Azure Machine Learning service, visit the documentation