Compute Resources

Deploy a trained model as a webservice

In previous lessons, we spent much time talking about training a machine learning model, which is a multi-step process involving data preparation, feature engineering, training, evaluation, and model selection. The model training process can be very compute-intensive, with training times spanning across many hours, days, or weeks depending on the amount of data, type of algorithm used, and other factors. A trained model, on the other hand, is used to make decisions on new data quickly. In other words, it infers things about new data it is given based on its training. Making these decisions on new data on-demand is called real-time inferencing.

Overview

In this lab, you learn how to deploy a trained model that can be used as a webservice, hosted on an Azure Kubernetes Service (AKS) cluster. This process is what enables you to use your model for real-time inferencing.

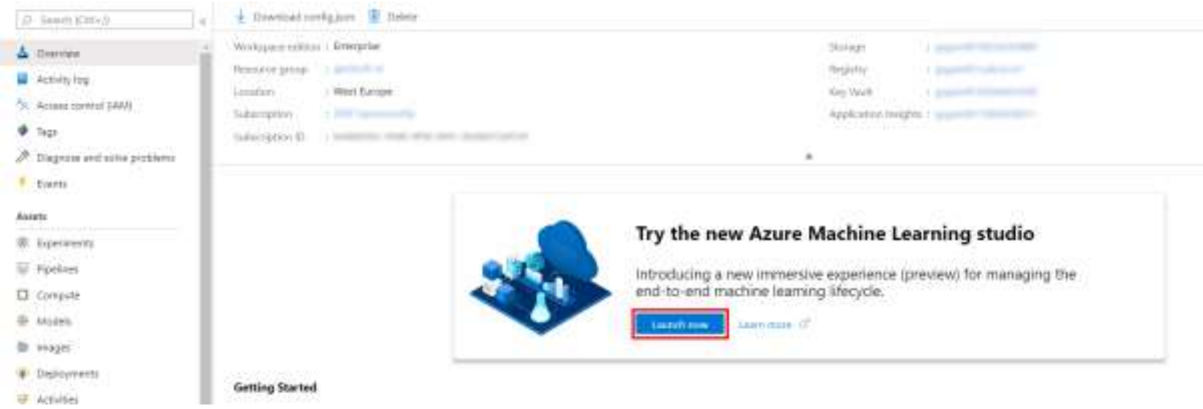
The Azure Machine Learning designer simplifies the process by enabling you to train and deploy your model without writing any code.

Exercise 1: Open a sample training pipeline

Task 1: Open the pipeline authoring editor

1. In Azure portal, open the available machine learning workspace.

2. Select **Launch now** under the **Try the new Azure Machine Learning studio** message.



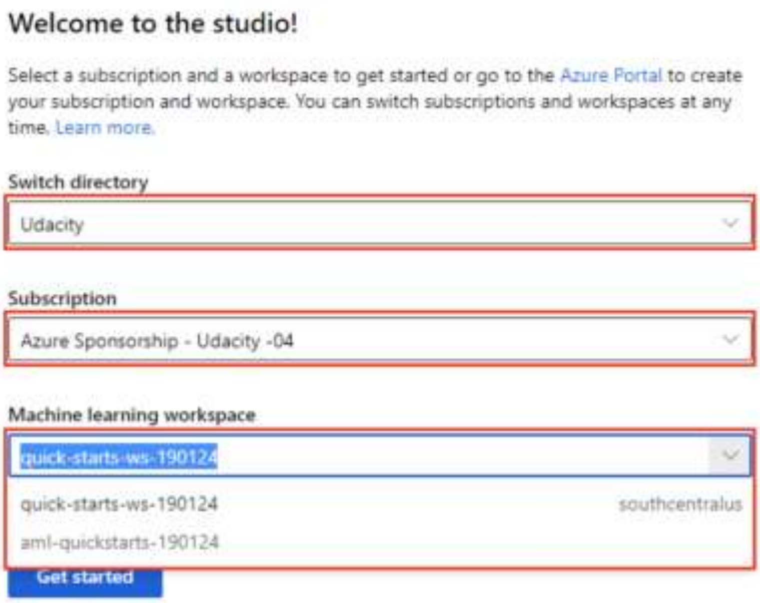
Workspace edition : Enterprise
Resource group :
Location : West Europe
Subscription :
Subscription ID :
Storage :
Registry :
Key Vault :
Application Insights :

Try the new Azure Machine Learning studio

Introducing a new immersive experience (preview) for managing the end-to-end machine learning lifecycle.

Getting Started

3. When you first launch the studio, you may need to set the directory and subscription. If so, you will see this screen:



Welcome to the studio!

Select a subscription and a workspace to get started or go to the Azure Portal to create your subscription and workspace. You can switch subscriptions and workspaces at any time. Learn more.

Switch directory: Udacity

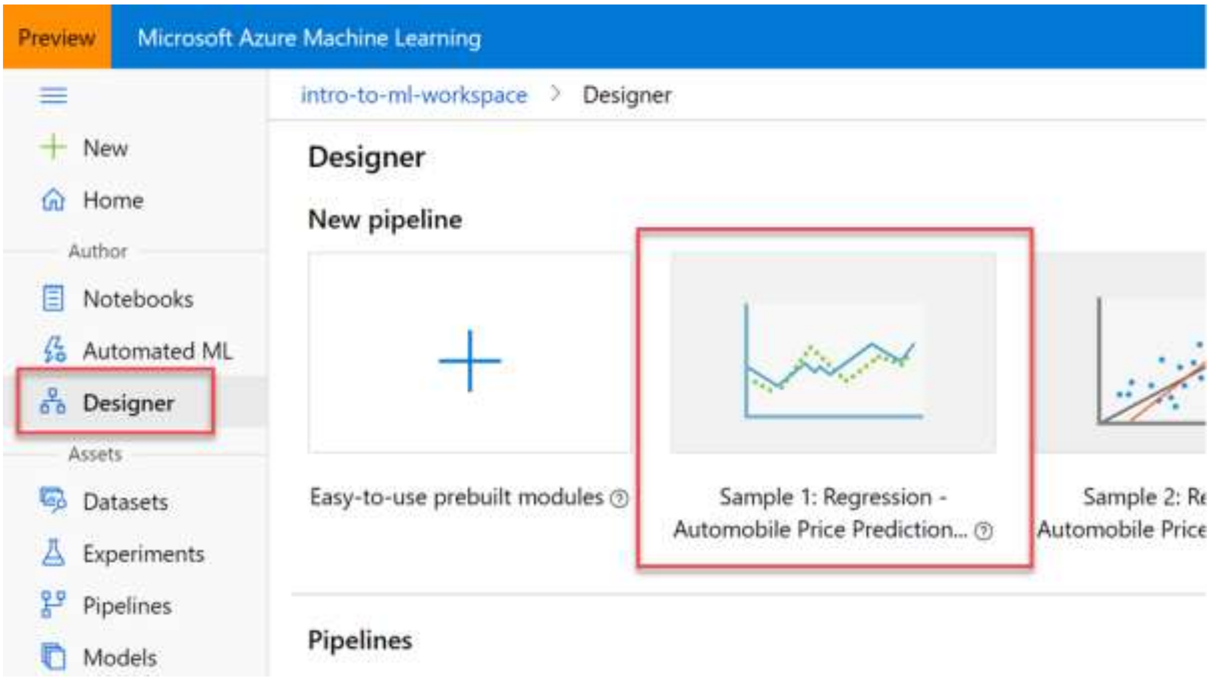
Subscription: Azure Sponsorship - Udacity -04

Machine learning workspace: quick-starts-ws-190124

quick-starts-ws-190124	southcentralus
aml-quickstarts-190124	

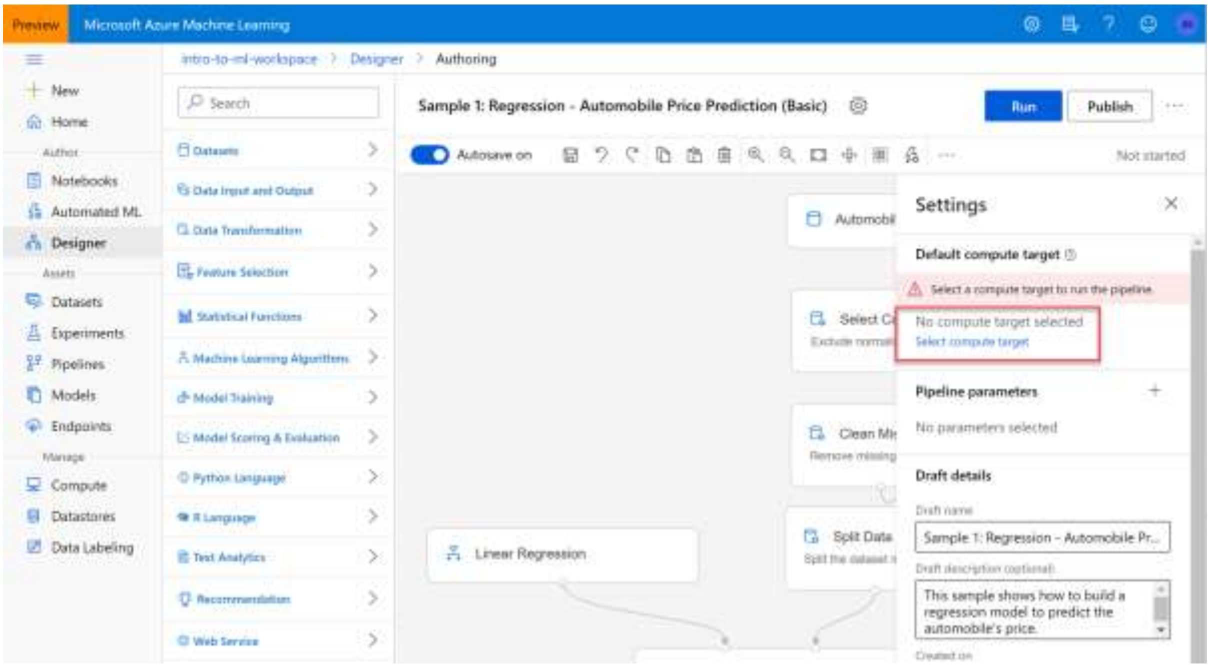
For the directory, select **Udacity** and for the subscription, select **Azure Sponsorship**. For the machine learning workspace, you may see multiple options listed. **Select any of these** (it doesn't matter which) and then click **Get started**.

4. From the studio, select **Designer** in the left-hand menu. Next, select **Sample 1: Regression - Automobile Price Prediction (Basic)** under the **New pipeline** section. This will open a `visual pipeline authoring editor`.


Preview
Microsoft Azure Machine Learning
New
Home
Author
Notebooks
Automated ML
Designer
Assets
Datasets
Experiments
Pipelines
Models
intro-to-ml-workspace > Designer
Designer
New pipeline
Easy-to-use prebuilt modules
Sample 1: Regression - Automobile Price Prediction...
Sample 2: Re Automobile Price
Pipelines

Task 2: Setup the compute target

1. In the settings panel on the right, select **Select compute target**.



Select a compute target to run the pipeline.

No compute target selected
Select compute target

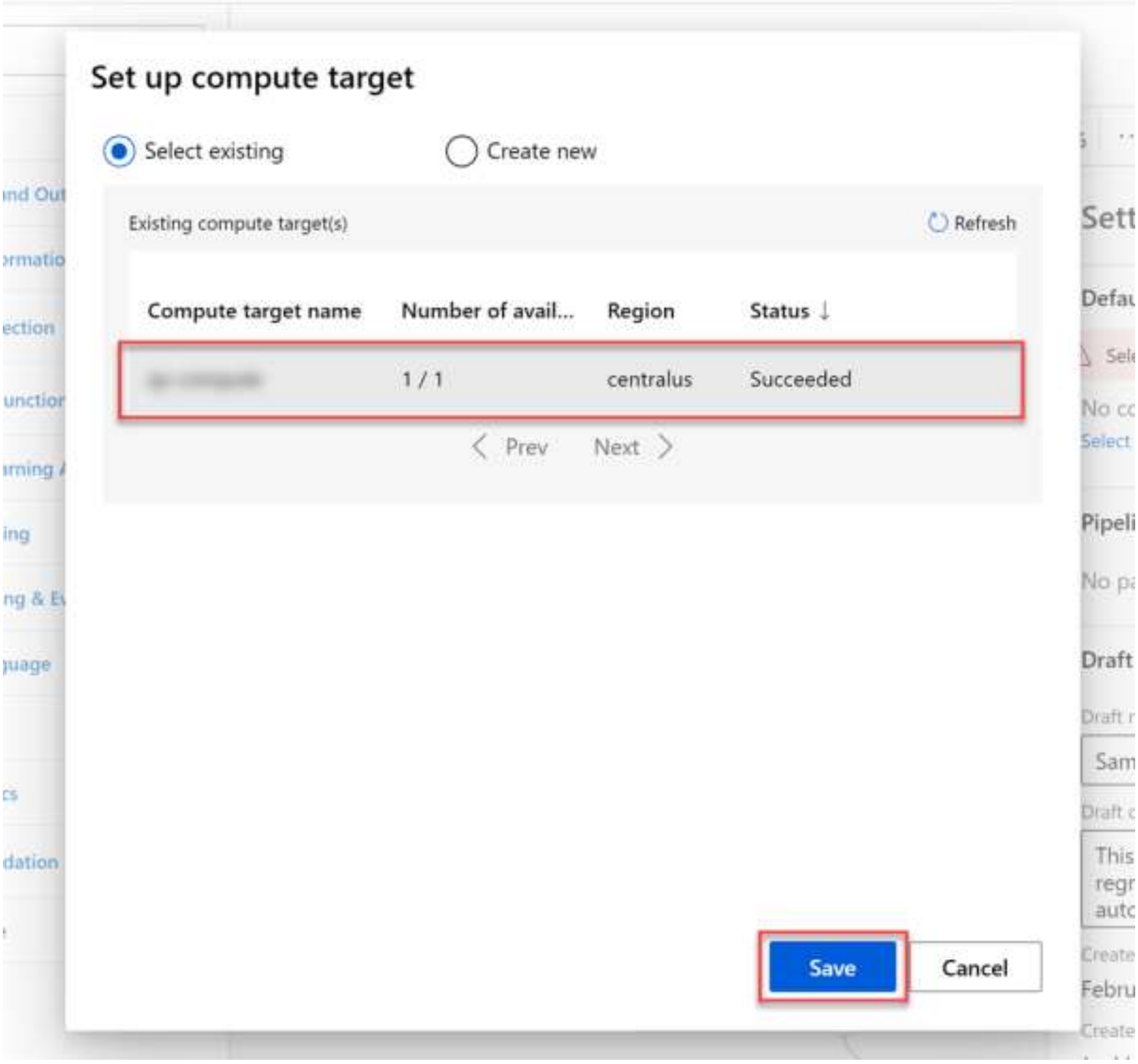
No parameters selected

Draft name: Sample 1: Regression – Automobile Pr...

Draft description (optional): This sample shows how to build a regression model to predict the automobile's price.

2. In the `Set up compute target` editor, select the existing compute target, then select **Save**.

Note: If you are facing difficulties in accessing pop-up windows or buttons in the user interface, please refer to the Help section in the lab environment.



Set up compute target

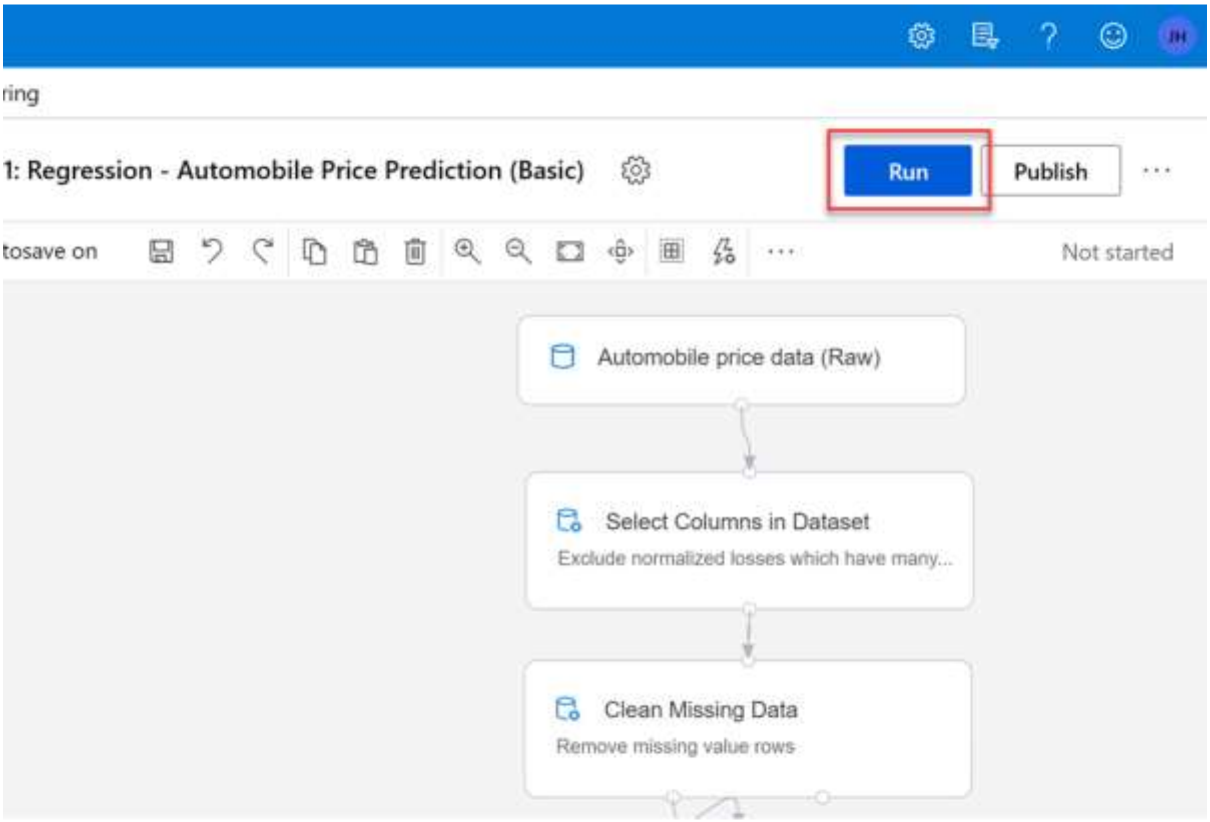
● Select existing ○ Create new

Existing compute target(s)

Compute target name	Number of avail...	Region	Status ↓
[redacted]	1 / 1	centralus	Succeeded

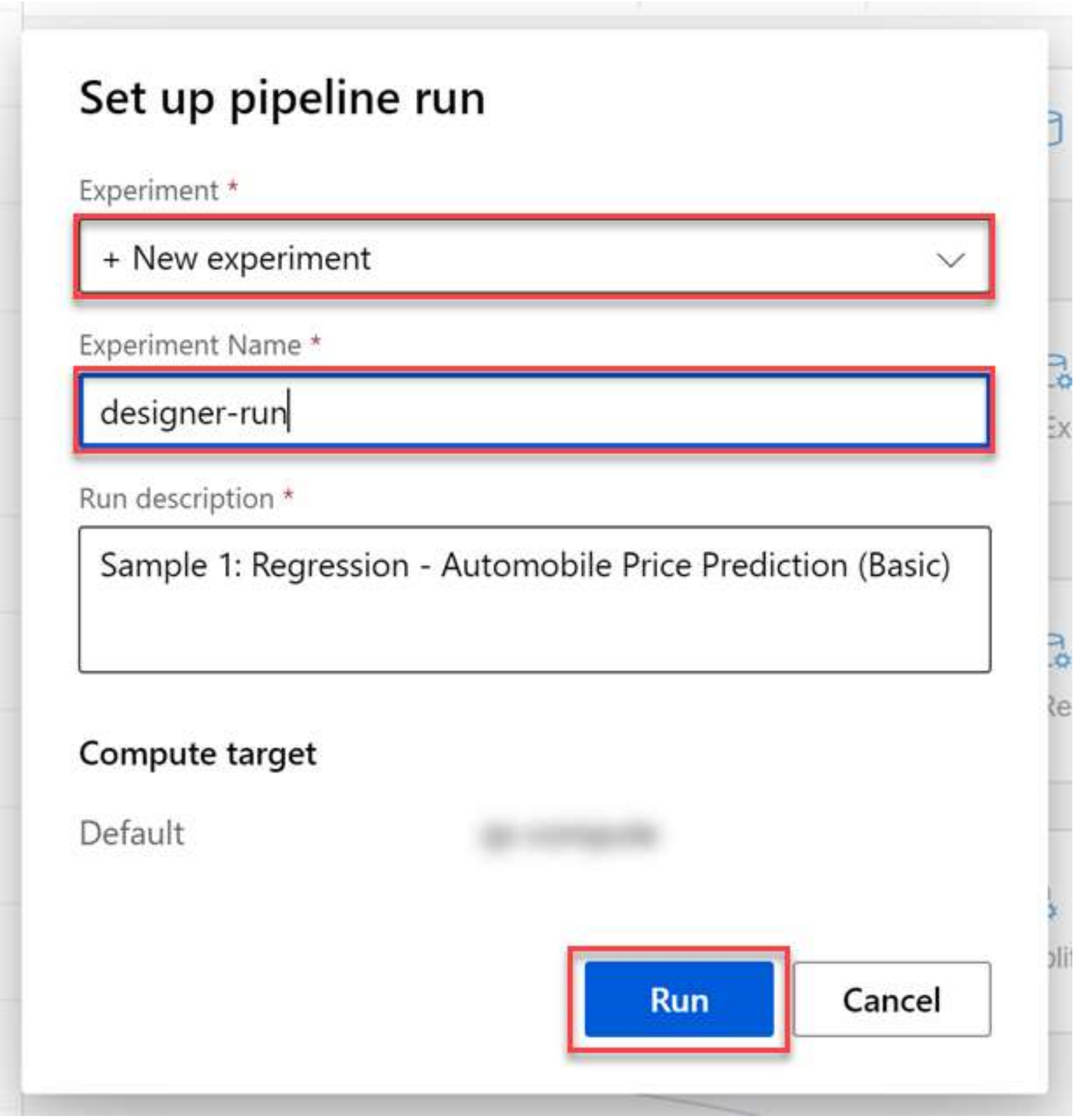
Task 3: Create a new experiment and submit the pipeline

1. Select **Submit** to open the `Set up pipline run` editor.


ring
1: Regression - Automobile Price Prediction (Basic)
Run
Publish
tosave on
Not started
Automobile price data (Raw)
Select Columns in Dataset
Exclude normalized losses which have many...
Clean Missing Data
Remove missing value rows

Please note that the button name in the UI is changed from **Run** to **Submit**.

2. In the `Setup pipeline run editor`, select **Experiment, Create new** and provide `New experiment name`: **designer-run**, and then select **Submit**.



Set up pipeline run

Experiment *

+ New experiment

Experiment Name *

designer-run

Run description *

Sample 1: Regression - Automobile Price Prediction (Basic)

Compute target

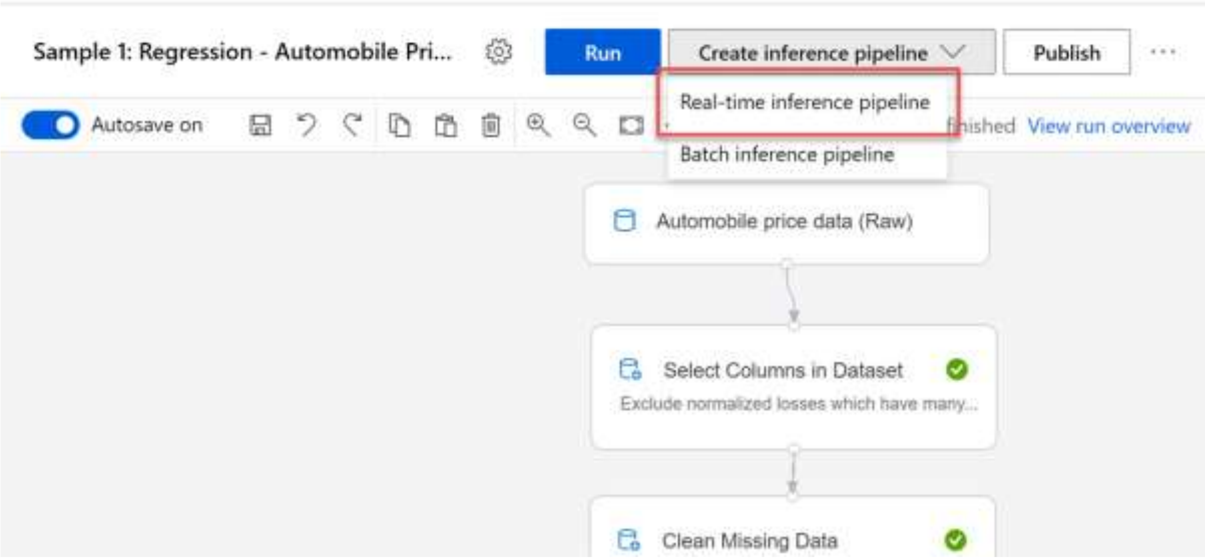
Default

3. Wait for the pipeline run to complete. It will take around **10 minutes** to complete the run.

Exercise 2: Real-time inference pipeline

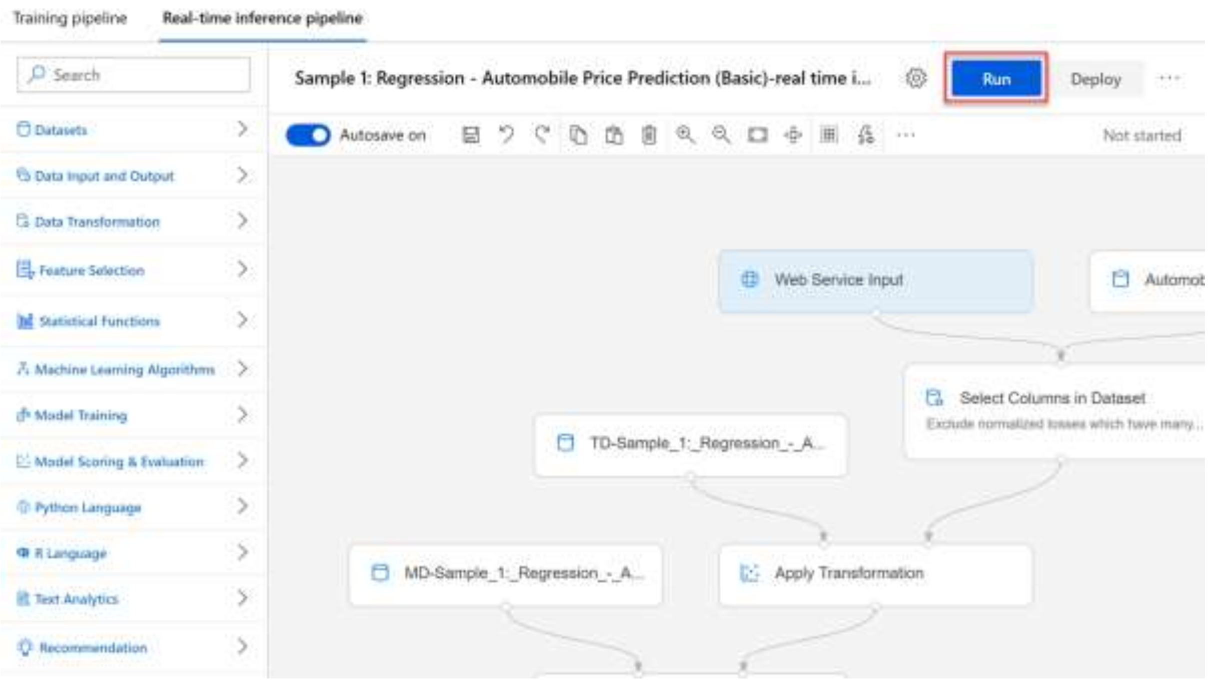
Task 1: Create pipeline

1. Select **Create inference pipeline**, then select **Real-time inference pipeline** from the list to create a new inference pipeline.


Sample 1: Regression - Automobile Pri...
Run
Create inference pipeline
Publish
Real-time inference pipeline
Batch inference pipeline
Autosave on
Finished View run overview
Automobile price data (Raw)
Select Columns in Dataset
Exclude normalized losses which have many...
Clean Missing Data

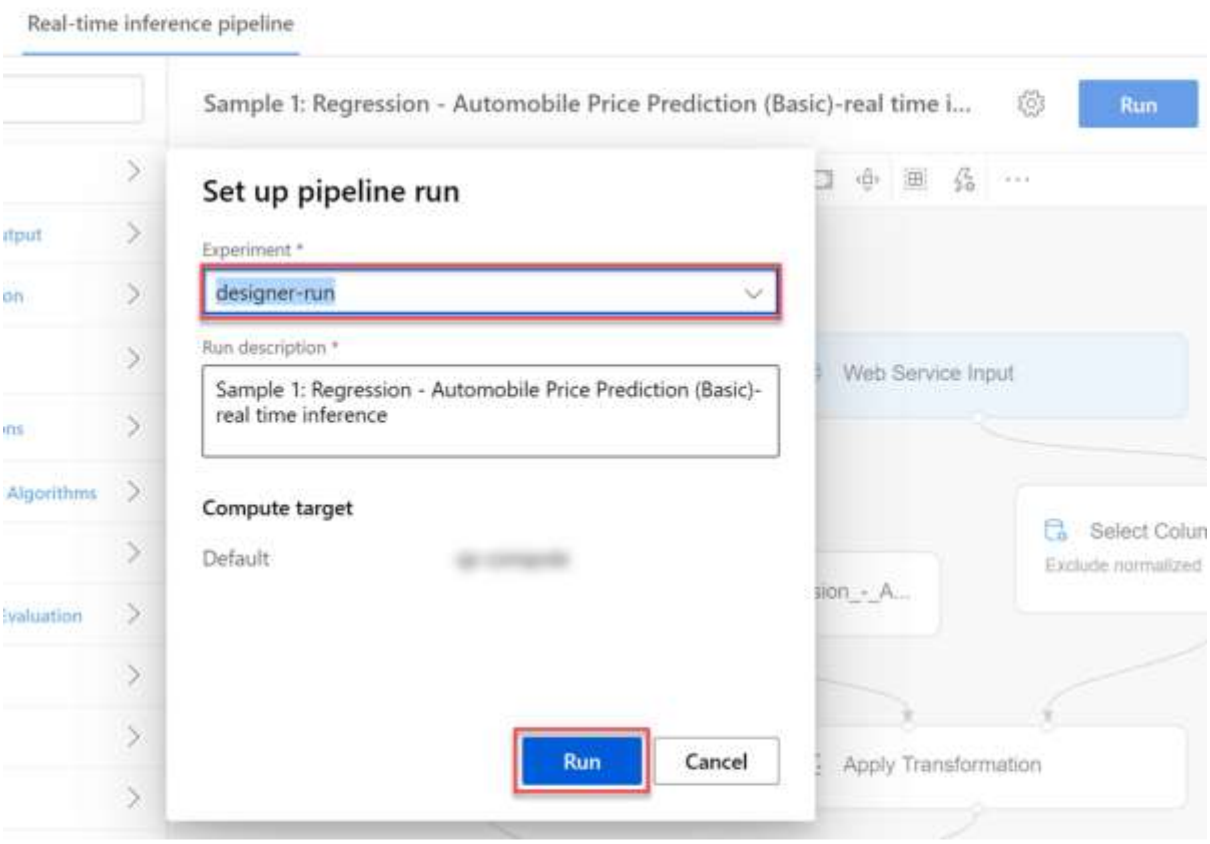
Task 2: Submit the pipeline

1. Select **Submit** to open the `Set up pipeline run` editor.


Training pipeline
Real-time inference pipeline
Search
Datasets
Data Input and Output
Data Transformation
Feature Selection
Statistical Functions
Machine Learning Algorithms
Model Training
Model Scoring & Evaluation
Python Language
R Language
Text Analytics
Recommendation
Sample 1: Regression - Automobile Price Prediction (Basic)-real time i...
Run
Deploy
Autosave on
Not started
Web Service Input
Automob
Select Columns in Dataset
Exclude normalized losses which have many...
TD-Sample_1:_Regression_-_A...
MD-Sample_1:_Regression_-_A...
Apply Transformation

Please note that the button name in the UI is changed from **Run** to **Submit**.

2. In the `Setup pipeline run` editor, select **Select existing**, then select the experiment you created in an earlier step: **designer-run**. Select **Submit** to start the pipeline.


Real-time inference pipeline
Sample 1: Regression - Automobile Price Prediction (Basic)-real time i...
Run
Web Service Input
Select Colun
Exclude normalized
Apply Transformation
Algorithms
Evaluation
Run
Cancel

Set up pipeline run

Experiment *: designer-run

Run description *: Sample 1: Regression - Automobile Price Prediction (Basic)-real time inference

Compute target

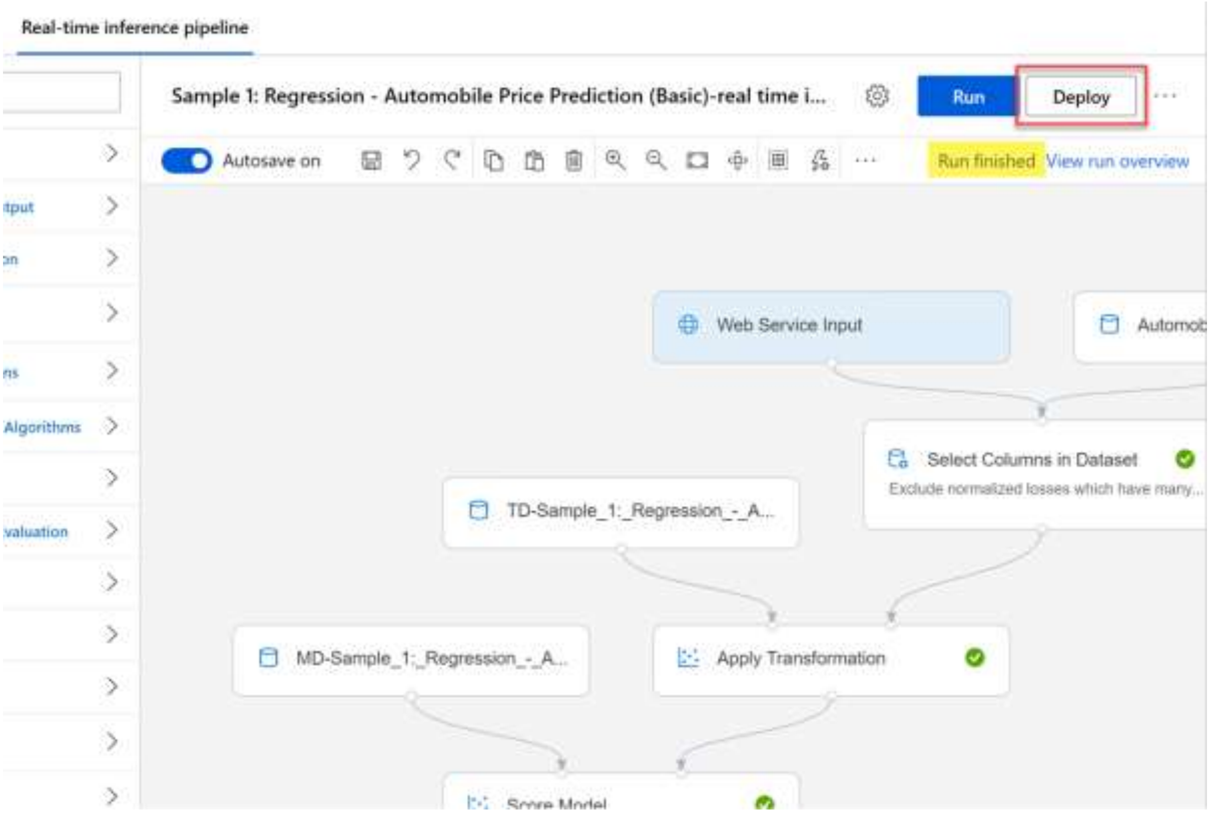
Default

3. Wait for pipeline run to complete. It will take around **7 minutes** to complete the run.

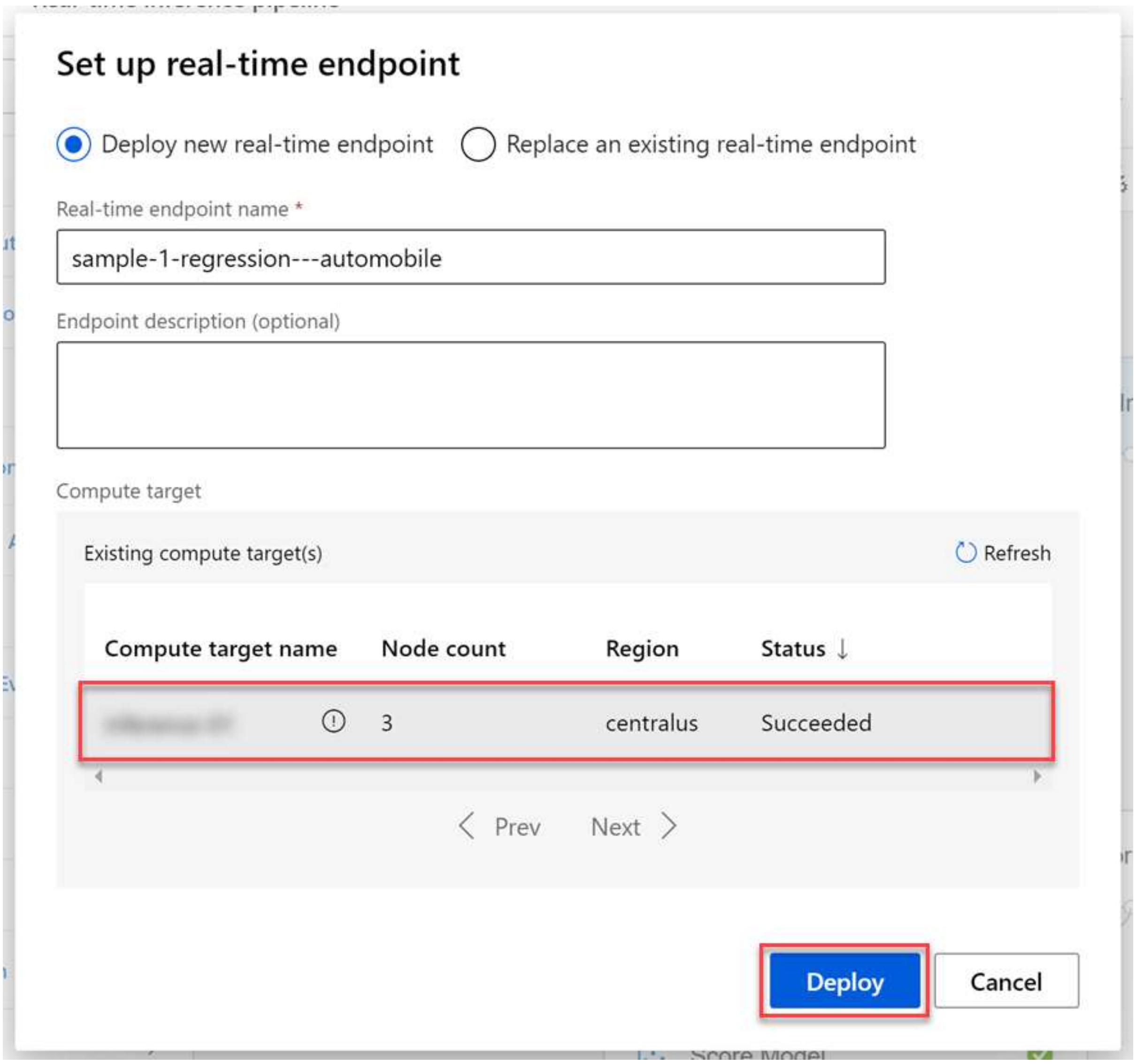
Exercise 3: Deploy web service on Azure Kubernetes Service compute

Task 1: Deploy the web service

1. After the inference pipeline run is finished, select **Deploy** to open the `Set up real-time endpoint` editor.


Real-time inference pipeline
Sample 1: Regression - Automobile Price Prediction (Basic)-real time i...
Run
Deploy
Autosave on
Run finished
View run overview
tput
on
ns
Algorithms
valuation
Web Service Input
Automob
Select Columns in Dataset
Exclude normalized losses which have many...
TD-Sample_1:_Regression_-_A...
MD-Sample_1:_Regression_-_A...
Apply Transformation
Score Model

2. In the `Set up real-time endpoint` editor, select your **existing compute target**, then select **Deploy**.



Set up real-time endpoint

● Deploy new real-time endpoint ○ Replace an existing real-time endpoint

Real-time endpoint name *

sample-1-regression---automobile

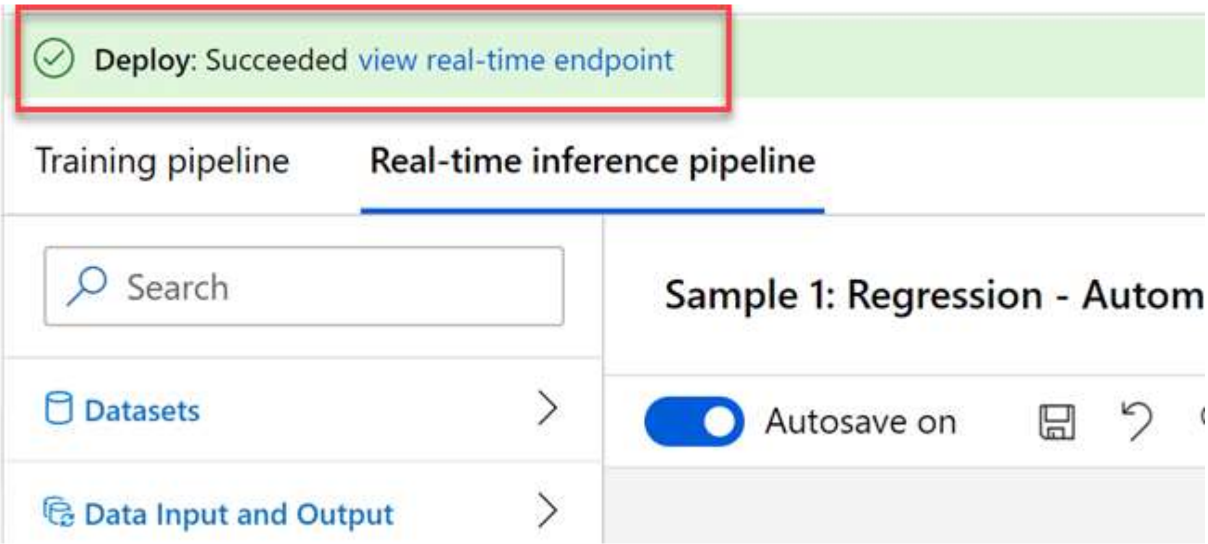
Endpoint description (optional)

Compute target

Existing compute target(s)

Compute target name	Node count	Region	Status ↓
[redacted] ⓘ	3	centralus	Succeeded

3. Wait for the deployment to complete. The status of the deployment can be observed above the `Pipeline Authoring Editor`.

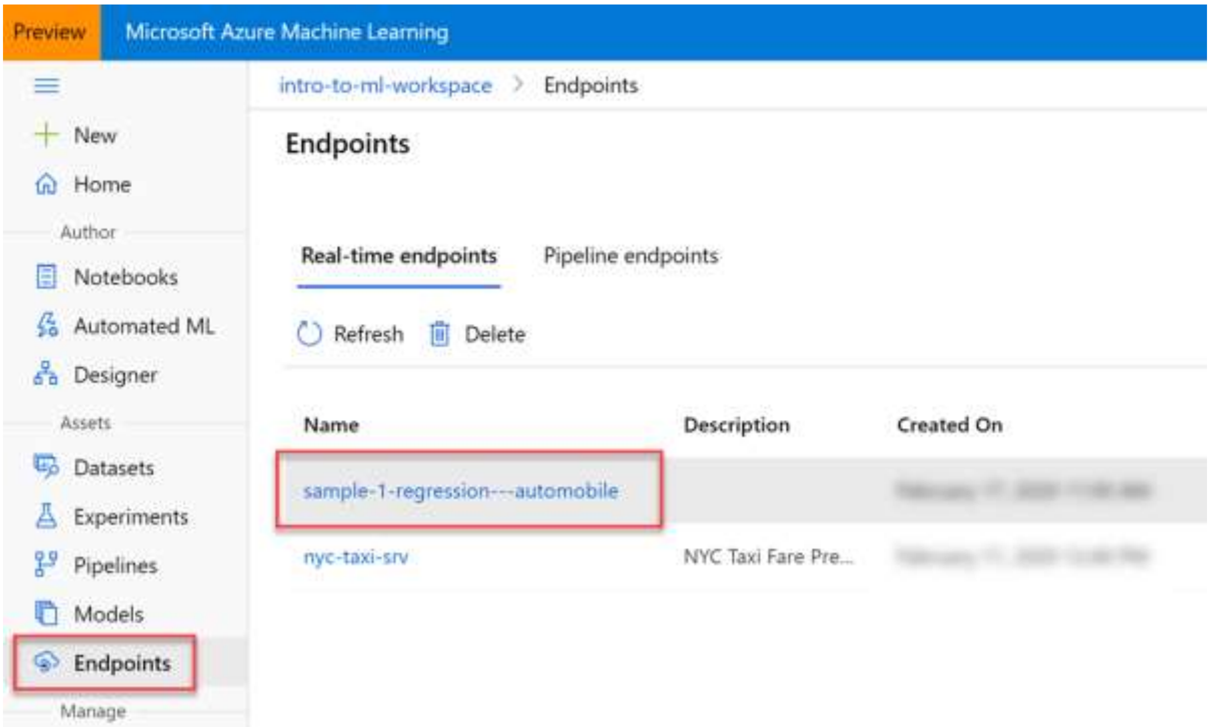


Deploy: Succeeded view real-time endpoint

Task 2: Review deployed web service

1. To view the deployed web service, select the **Endpoints** section in your Azure Portal Workspace.

2. Select the deployed web service: **sample-1-regression—-automobile** to open the deployment details page.



Name	Description	Created On
sample-1-regression---automobile		[redacted]
nyc-taxi-srv	NYC Taxi Fare Pre...	[redacted]

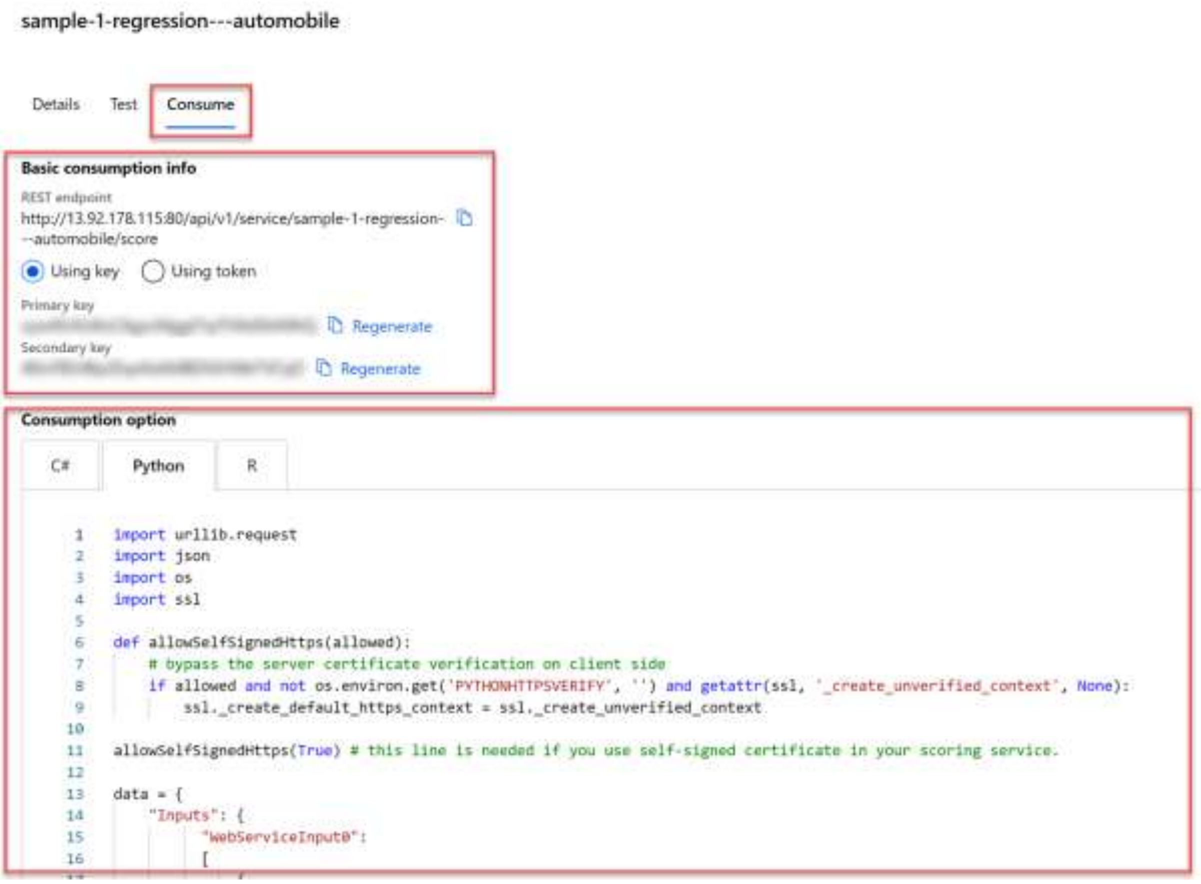
Note: you have to select the text of the service name to open the deployment details page

Task 3: Review how to consume the deployed web service

1. Select the **Consume** tab to observe the following information:

    1. `Basic consumption info` displays the **REST endpoint**, **Primary key**, and **Secondary key**.

    2. `Consumption option` shows code samples in **C#**, **Python**, and **R** on how to call the endpoint to consume the webservice.



sample-1-regression---automobile

Basic consumption info

REST endpoint
http://13.92.178.115:80/api/v1/service/sample-1-regression---automobile/score

● Using key ○ Using token

Primary key
[redacted]

Secondary key
[redacted]

Consumption option

```python
import urllib.request
import json
import os
import ssl

def allowSelfSignedHttps(allowed):
    # bypass the server certificate verification on client side
    if allowed and not os.environ.get('PYTHONHTTPSVERIFY', '') and getattr(ssl, '_create_unverified_context', None):
        ssl._create_default_https_context = ssl._create_unverified_context

allowSelfSignedHttps(True) # this line is needed if you use self-signed certificate in your scoring service.

data = {
    "Inputs": {
            "WebServiceInput0":
            [
```

Next Steps

Congratulations! You have just learned how to train and deploy a model to an Azure Kubernetes Service (AKS) cluster for real-time inferencing. You can now return to the Udacity portal to continue with the lesson.