**Menyebarkan layanan pembelajaran mesin real time dengan Azure Machine Learning**

Pelajari cara mendaftar dan menyebarkan model ML dengan Azure Machine Learning service.

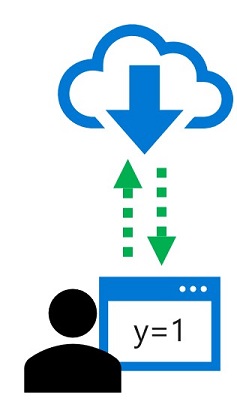
**Tujuan pembelajaran**

Dalam modul ini, Anda akan mempelajari cara:

* Menyebarkan model sebagai layanan inferensi real time.
* Menggunakan layanan inferensi real time.
* Memecahkan masalah penyebaran layanan

1. **Pengantar**

Dalam pembelajaran mesin, inferensi mengacu pada penggunaan model terlatih untuk memprediksi label untuk data baru yang belum digunakan dalam pelatihan model. Sering kali, model ini disebarkan sebagai bagian dari layanan yang memungkinkan aplikasi untuk meminta prediksi langsung atau real time untuk pengamatan data individual atau dalam jumlah kecil.



Di Azure Machine Learning, Anda dapat membuat solusi inferensi real time dengan menyebarkan model sebagai layanan, yang dihosting di platform dalam kontainer, sepert**i Azure Kubernetes Services (AKS).**

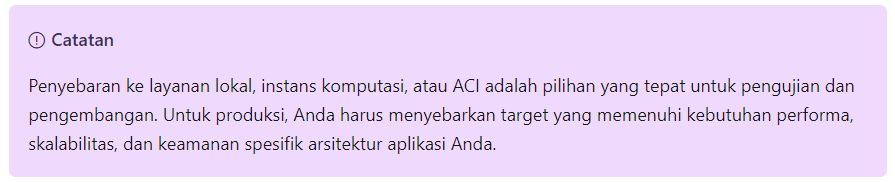
1. **Tujuan**

Dalam modul ini, Anda akan mempelajari cara:

* Menyebarkan model sebagai layanan inferensi real time.
* Menggunakan layanan inferensi real time.
* Memecahkan masalah penyebaran layanan

1. **Menyebarkan model sebagai layanan real time**

Anda dapat menyebarkan model sebagai layanan web real time ke beberapa jenis target komputasi, termasuk komputasi lokal, instans komputasi Azure Machine Learning, Azure Container Instance (ACI), kluster Azure Kubernetes Service (AKS), Azure Function, atau modul Internet of Things (IoT). Azure Machine Learning menggunakan kontainer sebagai mekanisme penyebaran, mengemas model dan kode untuk digunakan sebagai citra yang dapat disebarkan ke kontainer di target komputasi yang Anda pilih.



Untuk menyebarkan model sebagai layanan inferensi real time, Anda harus melakukan tugas berikut:

1. **Mendaftarkan model terlatih**

Setelah berhasil melatih model, Anda harus mendaftarkannya di ruang kerja Azure Machine Learning Anda. Layanan real time Anda kemudian akan dapat memuat model ketika diperlukan.

Untuk mendaftarkan model dari file lokal, Anda dapat menggunakan metode **daftar** objek **Model** seperti yang ditunjukkan di sini:



Selain itu, jika Anda memiliki referensi ke **Run** yang digunakan untuk melatih model, Anda dapat menggunakan metode **register\_model** seperti yang ditunjukkan di sini:



1. **Menentukan konfigurasi inferensi**

Model akan disebarkan sebagai layanan yang terdiri dari:

* Skrip untuk memuat model dan menampilkan prediksi untuk data yang dikirimkan.
* Lingkungan tempat skrip akan dijalankan.

Oleh karena itu, Anda harus menentukan skrip dan lingkungan untuk layanan ini.

1. **Membuat skrip entri**

Buat skrip entri (kadang-kadang disebut sebagai skrip penilaian) untuk layanan sebagai file Python (.py). Hal ini harus mencakup dua fungsi:

* **init()**: Dipanggil ketika layanan diinisialisasi.
* **run(raw\_data)**: Dipanggil ketika data baru dikirimkan ke layanan.

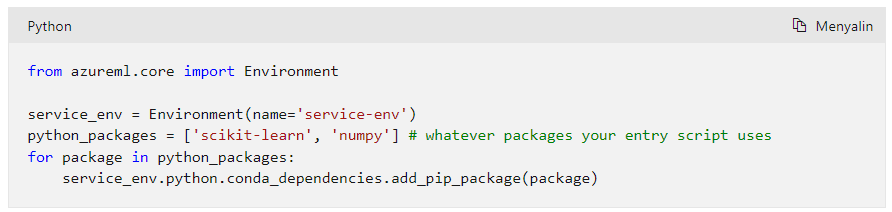
Biasanya, Anda menggunakan fungsi **init** untuk memuat model dari registri model dan menggunakan fungsi **run** untuk menghasilkan prediksi dari data input. Contoh skrip berikut menunjukkan pola ini:



Simpan skrip dalam folder sehingga Anda dapat dengan mudah mengidentifikasinya nanti. Misalnya, Anda dapat menyimpan skrip di atas sebagai score.py dalam folder bernama service\_files.

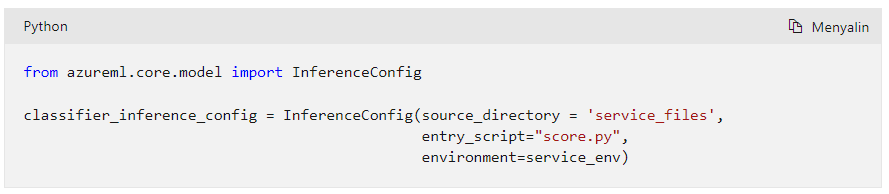
1. **Membuat lingkungan**

Layanan Anda memerlukan lingkungan Python untuk menjalankan skrip entri, yang dapat Anda tentukan dengan membuat Lingkungan yang berisi paket yang diperlukan:



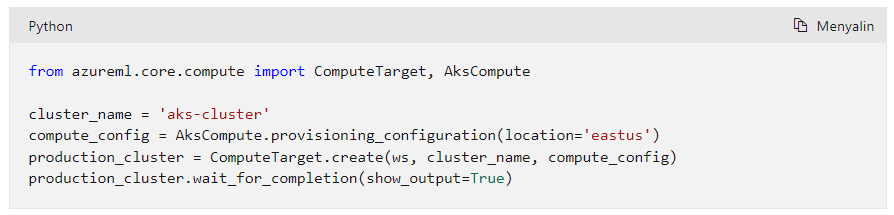
1. **Menggabungkan skrip dan lingkungan dalam InferenceConfig**

Setelah membuat lingkungan dan skrip entri, Anda dapat menggabungkannya dalam **InferenceConfig** untuk layanan seperti ini:



1. **Menentukan konfigurasi penyebaran**

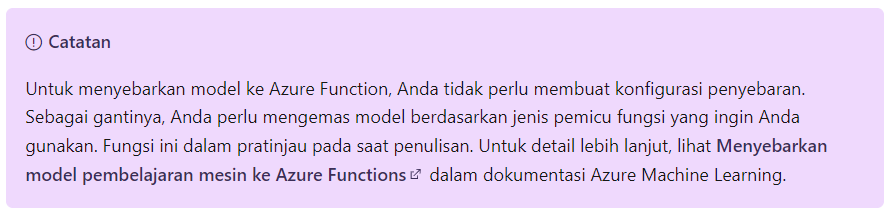
Sekarang setelah Anda memiliki skrip entri dan lingkungan, Anda perlu mengonfigurasi komputasi yang akan disebarkan layanan. Jika Anda menyebarkan ke kluster AKS, Anda harus membuat kluster dan target komputasi untuk kluster tersebut sebelum menyebarkan:



Dengan target komputasi yang dibuat, Anda sekarang dapat menentukan konfigurasi penyebaran, yang menetapkan spesifikasi komputasi khusus target untuk penyebaran yang dalam kontainer:

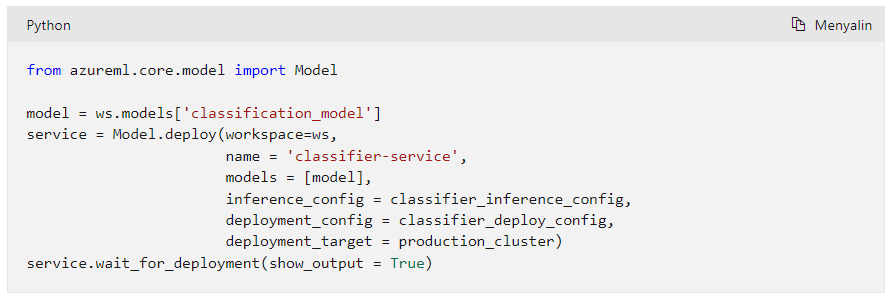


Penyebaran ACI dikonfigurasi dengan kode yang serupa. Hal yang berbeda adalah Anda tidak perlu secara eksplisit membuat target komputasi ACI dan Anda harus menggunakan kelas **deploy\_configuration** dari namespace layanan **azureml.core.webservice.AciWebservice.** Demikian pula, Anda dapat menggunakan namespace layanan **azureml.core.webservice.LocalWebservice** untuk mengonfigurasi layanan lokal berbasis Docker.



1. **Menyebarkan model**

Setelah semua konfigurasi disiapkan, Anda dapat menyebarkan model. Cara termudah untuk melakukan hal ini adalah dengan memanggil metode penyebaran dari kelas Model, seperti ini:



Untuk ACI atau layanan lokal, Anda dapat menghilangkan parameter **deployment\_target** (atau mengaturnya ke **Tidak Ada**).

Untuk informasi selengkapnya tentang cara menyebarkan model dengan Azure Machine Learning, **lihat Menyebarkan model pembelajaran mesin ke Azure** dalam dokumentasi.

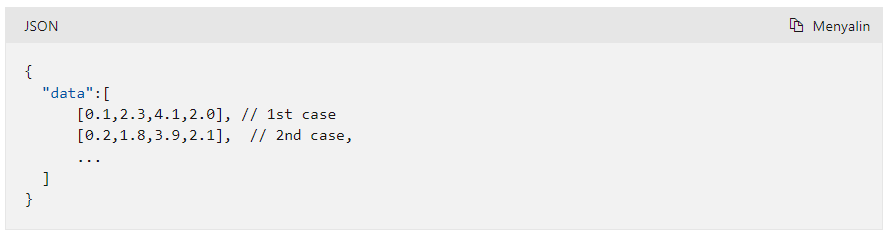
**Link** : <https://learn.microsoft.com/id-id/azure/machine-learning/how-to-deploy-online-endpoints?tabs=azure-cli>

**3. Menggunakan layanan inferensi real time**

Setelah menyebarkan layanan real time, Anda dapat menggunakannya dari aplikasi klien untuk memprediksi label untuk kasus data baru.

1. **Menggunakan Azure Machine Learning SDK**

Untuk pengujian, Anda dapat menggunakan Azure Machine Learning SDK untuk memanggil layanan web melalui metode **run** dari objek **WebService** yang mereferensikan layanan yang disebarkan. Biasanya, Anda mengirim data ke metode **run** dalam format JSON dengan struktur berikut:



Respons dari metode run adalah koleksi JSON dengan prediksi untuk setiap kasus yang dikirimkan dalam data. Contoh kode berikut memanggil layanan dan menampilkan respons:

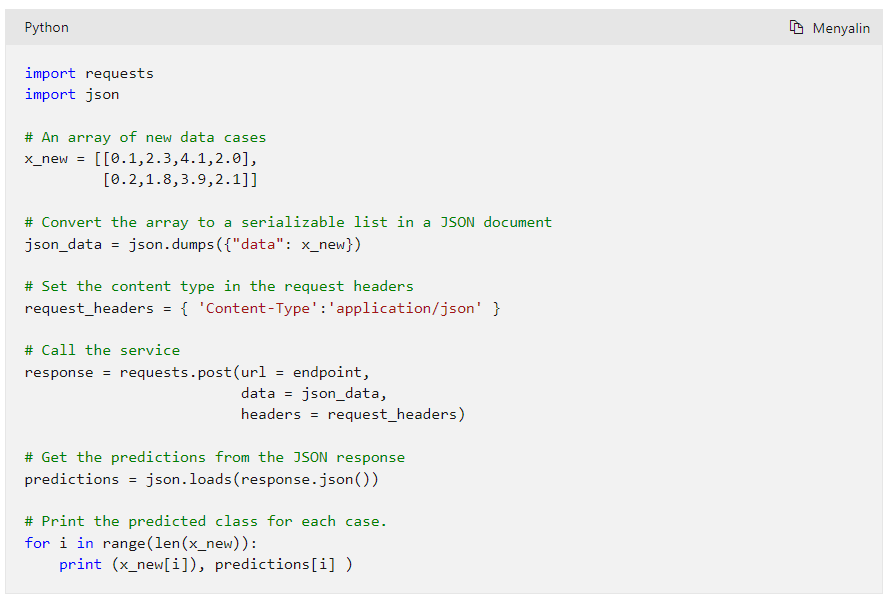


1. **Menggunakan titik akhir REST**

Dalam produksi, sebagian besar aplikasi klien tidak akan menyertakan Azure Machine Learning SDK dan akan menggunakan layanan melalui antarmuka REST-nya. Anda dapat menentukan titik akhir layanan yang disebarkan di studio Azure Machine Learning atau dengan mengambil properti **scoring\_uri** dari objek **Webservice** di SDK, seperti ini:



Dengan mengetahui titik akhirnya, Anda dapat menggunakan permintaan HTTP POST dengan data JSON untuk memanggil layanan. Contoh berikut menunjukkan cara untuk melakukan hal ini menggunakan Python:



**C, Autentikasi**

Dalam produksi, Anda mungkin ingin membatasi akses ke layanan Anda dengan menerapkan autentikasi. Ada dua jenis autentikasi yang dapat Anda gunakan:

* **Key**: Permintaan diautentikasi dengan menentukan kunci yang terkait dengan layanan.
* **Token**: Permintaan diautentikasi dengan menyediakan JSON Web Token (JWT).

Secara default, autentikasi dinonaktifkan untuk layanan ACI dan diatur ke autentikasi berbasis kunci untuk layanan AKS (tempat kunci utama dan sekunder secara otomatis dibuat). Anda dapat secara opsional mengonfigurasi layanan AKS untuk menggunakan autentikasi berbasis token (yang tidak didukung untuk layanan ACI).

Dengan asumsi Anda memiliki sesi terautentikasi yang dibuat dengan ruang kerja, Anda dapat mengambil kunci untuk layanan menggunakan metode **get\_keys** dari objek **WebService** yang terkait dengan layanan:



Untuk autentikasi berbasis token, aplikasi klien Anda perlu menggunakan autentikasi perwakilan layanan untuk memverifikasi identitasnya melalui Azure Active Directory (Azure AD) dan memanggil metode **get\_token** layanan untuk mengambil token dengan waktu terbatas.

Untuk melakukan panggilan terautentikasi ke titik akhir REST layanan, Anda harus menyertakan kunci atau token di header permintaan seperti ini:



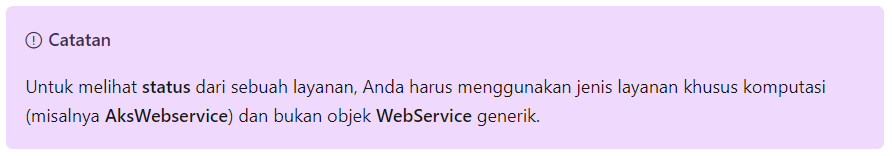
**4. Memecahkan masalah penyebaran layanan**

Ada banyak elemen untuk penyebaran layanan real time, termasuk model terlatih, konfigurasi lingkungan runtime, skrip penilaian, citra kontainer, dan host kontainer. Memecahkan masalah penyebaran yang gagal atau kesalahan saat menggunakan layanan yang disebarkan dapat menjadi proses yang rumit.

1. **Memeriksa status layanan**

Sebagai langkah pemecahan masalah awal, Anda dapat memeriksa status layanan dengan memeriksa **statusnya**:





Untuk layanan operasional, statusnya harus Sehat.

1. **Meninjau log layanan**

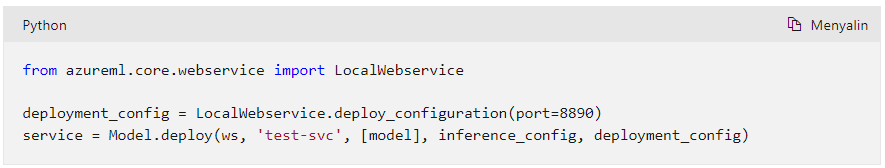
Jika sebuah layanan tidak sehat atau Anda mengalami kesalahan saat menggunakannya, Anda dapat meninjau log layanan:



Log tersebut mencakup informasi terperinci tentang provisi layanan dan permintaan yang telah diproses. Log ini sering kali dapat memberikan wawasan tentang penyebab kesalahan yang tidak terduga.

1. **Menyebarkan ke kontainer lokal**

Kesalahan penyebaran dan runtime dapat lebih mudah didiagnosis dengan menyebarkan layanan sebagai kontainer di instans Docker lokal, seperti ini:



Anda kemudian dapat menguji layanan yang disebarkan secara lokal menggunakan SDK:



Anda kemudian dapat memecahkan masalah runtime dengan membuat perubahan pada file penilaian yang direferensikan dalam konfigurasi inferensi dan memuat ulang layanan tanpa menyebarkannya ulang (sesuatu yang hanya dapat Anda lakukan dengan layanan lokal):



**5. Latihan - Menyebarkan model sebagai layanan real time**

Sekarang saatnya Anda menggunakan Azure Machine Learning untuk menyebarkan model pembelajaran mesin sebagai layanan real time.

Dalam latihan ini, Anda akan:

* Melatih dan mendaftarkan model.
* Menyebarkan model sebagai layanan real time.
* Menggunakan layanan yang disebarkan.

1. **Instruksi**

Ikuti petunjuk berikut untuk menyelesaikan latihan:

1. Jika Anda belum memiliki langganan Azure, daftar untuk uji coba gratis di <https://azure.microsoft.com.>
2. Menampilkan repo latihan di <https://aka.ms/mslearn-dp100.> \ <https://microsoftlearning.github.io/mslearn-dp100/>
3. Jika Anda belum melakukannya, selesaikan latihan **Membuat ruang kerja Azure Machine Learning** untuk menyediakan ruang kerja Azure Machine Learning, membuat instans komputasi, dan mengkloning file yang diperlukan.
4. Selesaikan latihan **Membuat layanan inferensi real time.**

**6. Ringkasan**

Dalam modul ini, Anda belajar cara:

* Menyebarkan model sebagai layanan inferensi real time.
* Menggunakan layanan inferensi real time.
* Memecahkan masalah penyebaran layanan.