**Melatih model pembelajaran mesin dengan Azure Databricks**

Pembelajaran mesin melibatkan penggunaan data untuk melatih model prediktif. Azure Databricks mendukung beberapa kerangka kerja pembelajaran mesin yang sering digunakan yang dapat Anda gunakan untuk melatih model.

**Tujuan pembelajaran**

Setelah menyelesaikan modul ini, Anda akan dapat:

* Memahami Spark ML
* Melatih dan memvalidasi model
* Menggunakan kerangka kerja pembelajaran mesin lainnya

1. **Pengantar**

Untuk melatih model pembelajaran mesin dengan Azure Databricks, ilmuwan data dapat menggunakan pustaka Spark ML. Dalam modul ini, Anda mempelajari cara melatih dan mengevaluasi model pembelajaran mesin menggunakan pustaka Spark ML serta kerangka kerja pembelajaran mesin lainnya.

1. **Tujuan pembelajaran**

Setelah menyelesaikan modul ini, Anda akan mampu:

* Menjelaskan Spark ML.
* Melatih dan memvalidasi model pembelajaran mesin.
* Menggunakan kerangka kerja pembelajaran mesin lainnya.

1. **Memahami Spark ML**

Azure Databricks mendukung beberapa pustaka untuk pembelajaran mesin. Ada satu pustaka utama, yang memiliki dua pendekatan dan bersifat native untuk Apache Spark: **MLLib** dan **Spark ML.**

1. **MLLib**

MLLib adalah pendekatan lama untuk pembelajaran mesin di Apache Spark. Pendekatan dibangun dari struktur data **Himpunan Data Terdistribusi yang Tangguh (RDD) Spark**. Struktur data ini membentuk dasar Apache Spark, tetapi struktur data tambahan di atas RDD, seperti DataFrames, telah mengurangi kebutuhan untuk bekerja secara langsung dengan RDD.

Pada Apache Spark 2.0, pustaka memasuki mode pemeliharaan. Ini berarti bahwa MLLib masih tersedia dan belum ditinggalkan, tetapi tidak akan ada fungsionalitas baru yang ditambahkan ke pustaka. Sebagai gantinya, pelanggan disarankan untuk pindah ke pustaka **org.apache.spark.ml**, yang biasa disebut sebagai Spark ML.

1. **Spark ML**

Spark ML adalah pustaka utama untuk pengembangan pembelajaran mesin di Apache Spark. Ini mendukung DataFrames di API-nya, versus pendekatan RDD klasik. Hal ini membuat Spark ML menjadi pustaka yang lebih mudah untuk digunakan oleh para ilmuwan data, karena Spark DataFrames berbagi banyak ide umum dengan DataFrames yang digunakan di Pandas dan R.

Bagian yang paling membingungkan tentang MLLib versus Spark ML adalah **keduanya merupakan pustaka yang sama.** Perbedaannya adalah namespace MLLib "klasik" adalah org.apache.spark.mllib sedangkan namespace Spark ML adalah org.apache.spark.ml. **Jika memungkinkan, gunakan namespace Spark ML saat melakukan aktivitas ilmu data baru.**

1. **Melatih dan memvalidasi model**

Proses pelatihan dan validasi model pembelajaran mesin menggunakan Spark ML cukup mudah. Langkah langkahnya adalah sebagai berikut:

1. Memisahkan data.
2. Melatih model.
3. Memvalidasi model.
4. **Memisahkan data**

Langkah pertama melibatkan pemisahan data antara kumpulan data **pelatihan** dan **validasi**. Melakukannya memungkinkan ilmuwan data untuk melatih model dengan bagian data yang representatif, sambil tetap mempertahankan beberapa persentase sebagai kumpulan data yang bertahan. Kumpulan data penangguhan ini dapat berguna untuk menentukan apakah model pelatihan **overfitting** - yaitu, mengaitkan kekhasan kumpulan data pelatihan daripada menemukan hubungan yang dapat diterapkan secara umum antar variabel.

DataFrames mendukung metode **randomSplit(),** yang membuat proses pemisahan data ini menjadi sederhana.

1. **Melatih model**

Melatih model bergantung pada tiga abstraksi utama**: transformator, estimator, dan pipeline.**

Transformator mengambil DataFrame sebagai input dan mengembalikan DataFrame baru sebagai output. Transformer sangat membantu untuk melakukan rekayasa fitur dan pemilihan fitur, karena transformator adalah DataFrame lain. Contohnya adalah membaca dalam kolom teks, memetakan kolom teks tersebut ke dalam kumpulan vektor fitur, dan menampilkan DataFrame dengan kolom yang baru dipetakan. Transformers akan menerapkan metode **.transform().**

Penaksir mengambil DataFrame sebagai input dan mengembalikan model. Dibutuhkan DataFrame sebagai input dan mengembalikan model, yang merupakan transformator itu sendiri. Contoh penaksir adalah **LinearRegression** algoritme pembelajaran mesin. Ia menerima DataFrame dan menghasilkan **Model**. Penaksir menerapkan metode **.fit().**

Pipeline menggabungkan estimator dan transformer dan menerapkan metode **.fit().** Dengan memecah proses pelatihan menjadi serangkaian tahapan, lebih mudah untuk menggabungkan beberapa algoritma.

1. **Memvalidasi model**

Setelah model dilatih, menjadi mungkin untuk memvalidasi hasilnya. Spark ML menyertakan statistik ringkasan bawaan untuk model berdasarkan algoritme pilihan. Menggunakan regresi linier misalnya, model berisi objek **summary**, yang mencakup skor seperti Root Mean Square Error (RMSE), Mean Absolute Error (MAE), dan koefisien determinasi (**Rsummary**, diucapkan kuadrat-R). Ini akan menjadi langkah-langkah ringkasan berdasarkan data **pelatihan**.

Dari sana, dengan kumpulan data **validasi**, dimungkinkan untuk menghitung statistik ringkasan pada kumpulan data yang belum pernah dilihat sebelumnya, menjalankan fungsi model terhadap kumpulan data validasi. Dari sana, gunakan evaluator seperti **RegressionEvaluator** untuk menghitung ukuran seperti RMSE, MAE, dan **RRegressionEvaluator**.

**4. Menggunakan kerangka kerja pembelajaran mesin lainnya**

Azure Databricks mendukung kerangka kerja pembelajaran mesin selain Spark ML dan MLLib. Misalnya, Azure Databricks menawarkan dukungan untuk pustaka populer seperti TensorFlow dan PyTorch.

Pustaka ini dapat diinstal secara langsung, namun sebaiknya gunakan **Waktu Proses Databricks untuk Pembelajaran Mesin.** \ <https://learn.microsoft.com/id-id/azure/databricks/runtime/mlruntime>

Runtime ini dilengkapi dengan berbagai pustaka pembelajaran mesin yang telah diinstal sebelumnya, termasuk TensorFlow, PyTorch, Keras, dan XGBoost. Ini juga mencakup pustaka yang penting untuk pelatihan terdistribusi, yang memungkinkan ilmuwan data memanfaatkan sifat terdistribusi dari Apache Spark.

Untuk pustaka, yang tidak mendukung pelatihan terdistribusi, Anda juga dapat menggunakan **kluster simpul tungga**l. \ https://learn.microsoft.com/id-id/azure/databricks/clusters/single-node

Misalnya, **PyTorch \** <https://learn.microsoft.com/id-id/azure/databricks/machine-learning/train-model/pytorch#use-pytorch-on-a-single-node>

dan **TensorFlow \** <https://learn.microsoft.com/id-id/azure/databricks/machine-learning/train-model/tensorflow#use-tensorflow-on-a-single-node>

keduanya mendukung penggunaan simpul tunggal.

**5. Latihan - Latih model pembelajaran mesin**

Sekarang, ini adalah kesempatan Anda untuk menggunakan Azure Databricks untuk melatih model regresi multivarian dan menginterpretasikan hasilnya.

Dalam latihan ini, Anda akan:

* Melatih Model.
* Memvalidasi Model.

1. **Instruksi**

Ikuti petunjuk berikut untuk menyelesaikan latihan:

* Buka petunjuk latihan di <https://aka.ms/mslearn-dp090.> \ <https://microsoftlearning.github.io/dp-090-databricks-ml/>
* Selesaikan latihan Melatih dan Memvalidasi Model Pembelajaran Mesin.

1. **Ringkasan**

Dalam modul ini, Anda mempelajari cara melatih dan mengevaluasi model pembelajaran mesin.

Sekarang setelah Anda menyelesaikan modul ini, Anda dapat:

* Menjelaskan Spark ML.
* Melatih dan memvalidasi model pembelajaran mesin.
* Menggunakan kerangka kerja pembelajaran mesin lainnya.