aws

Panduan Developer

Amazon Machine Learning

Versi Latest

Amazon Machine Learning: Panduan Developer

Table of Contents

Kami tidak lagi memperbarui layanan Amazon Machine Learning atau menerima pengguna baru untuk itu. Dokumentasi ini tersedia untuk pengguna yang sudah ada, tetapi kami tidak lagi memperbaruinya. Untuk informasi selengkapnya, lihatApa itu Amazon Machine Learning.

Terjemahan disediakan oleh mesin penerjemah. Jika konten terjemahan yang diberikan bertentangan dengan versi bahasa Inggris aslinya, utamakan versi bahasa Inggris.

Apa itu Amazon Machine Learning?

Kami tidak lagi memperbarui layanan Amazon Machine Learning (Amazon ML) atau menerima pengguna baru untuk layanan tersebut. Dokumentasi ini tersedia untuk pengguna yang sudah ada, tetapi kami tidak lagi memperbaruinya.

AWSsekarang menyediakan layanan berbasis cloud yang kuat - Amazon SageMaker - sehingga pengembang dari semua tingkat keterampilan dapat menggunakan teknologi machine learning. SageMaker adalah layanan machine learning yang dikelola sepenuhnya yang membantu Anda menciptakan model machine learning yang kuat. Dengan SageMaker, ilmuwan dan developer data dapat membangun dan melatih model machine learning, dan kemudian langsung men-deploynya ke lingkungan yang di-host dan siap produksi.

Untuk informasi lebih lanjut, lihatSageMaker.

Topik

Konsep Kunci Amazon Machine Learning

Bagian ini merangkum konsep-konsep kunci berikut dan menjelaskan secara lebih rinci bagaimana mereka digunakan dalam Amazon ML-nya:

- Sumber datamengandung metadata yang terkait dengan input data ke Amazon ML-nya
- Model MLmenghasilkan prediksi menggunakan pola yang diekstrak dari data input
- evaluasimengukur kualitas model ML-nya
- Prediksi Batch asinkronmenghasilkan prediksi untuk beberapa pengamatan data masukan
- Prediksi waktu nyata serentakmenghasilkan prediksi untuk pengamatan data individual

Sumber data

Sumber data adalah objek yang berisi metadata tentang data input Anda. Amazon IL membaca data input Anda, menghitung statistik deskriptif pada atributnya, dan menyimpan statistik—bersama

dengan skema dan informasi lainnya—sebagai bagian dari objek sumber data. Selanjutnya, Amazon MLnya menggunakan sumber data untuk melatih dan mengevaluasi model ML. dan menghasilkan prediksi batch.

> **Important**
>
> Sebuah datasource tidak menyimpan salinan data masukan Anda. Sebagai gantinya, referensi ke lokasi Amazon S3 tempat data input Anda berada. Jika Anda memindahkan atau mengubah file Amazon S3, Amazon IL tidak dapat mengakses atau menggunakannya untuk membuat model ML-nya, menghasilkan evaluasi, atau menghasilkan prediksi.

Tabel berikut mendefinisikan istilah yang terkait dengan sumber data.

Jangka waktu	Definisi
Atribut	Properti unik dan dinamakan dalam pengamatan. Dalam data tabular seperti spreadsheet atau file dengan nilai yang dipisahkan koma (CSV), judul kolom mewakili atribut, dan barisnya berisi nilai-nilai untuk setiap atribut. Sinonim: variabel, nama variabel, bidang, kolom
Nama sumber data	(Opsional) Memungkinkan Anda untuk menentukan nama yang dapat dibaca manusia untuk sumber data. Nama-nama ini memungkinkan Anda menemukan dan mengelola sumber data Anda di konsol Amazon ML-nya.
Data input	Nama kolektif untuk semua pengamatan yang disebut oleh sumber data.
Lokasi	Lokasi data input. Saat ini, Amazon IL dapat menggunakan data yang disimpan dalam bucket Amazon S3, database Amazon Redshift, atau database MySQL di Amazon Relational Database Service (RDS).
observasi	Sebuah unit data input tunggal. Misalnya, jika Anda membuat model ML untuk mendeteksi transaksi penipuan, data input Anda akan terdiri dari banyak pengamatan, masing-masing mewakili transaksi individual. Sinonim: catatan, contoh, contoh, baris

Jangka waktu	Definisi
ID baris	(Opsional) Sebuah bendera yang, jika ditentukan, mengidentifikasi atribut dalam data input untuk dimasukkan dalam output prediksi. Atribut ini membuatnya lebih mudah untuk mengasosiasikan prediksi mana yang sesuai dengan pengamatan mana. Sinonim: pengidentifikasi baris
Skema	Informasi yang diperlukan untuk menafsirkan data input, termasuk nama atribut dan jenis data yang ditugaskannya, dan nama-nama atribut khusus.
Statistik	Ringkasan statistik untuk setiap atribut dalam data input. Statistik ini melayani dua tujuan: Konsol Amazon XML menampilkannya dalam grafik untuk membantu Anda memahami data Anda secara sekilas dan mengidentifikasi penyimpangan atau kesalahan. Amazon ML-nya menggunakannya selama proses pelatihan untuk meningkat kan kualitas model ML-nya.
Status	Menunjukkan status sumber data saat ini, sepertiDalam Progres,Completed (Lengkap), atauGagal.
Atribut target	Dalam konteks pelatihan model L, atribut target mengidentifikasi nama atribut dalam data input yang berisi jawaban “benar”. Amazon IL menggunakan ini untuk menemukan pola dalam data input dan menghasilkan model ML-nya. Dalam konteks mengevaluasi dan menghasilkan prediksi, atribut target adalah atribut yang nilainya akan diprediksi oleh model ML-terlatih. Sinonim: target

Model ML

Model MLadalah model matematika yang menghasilkan prediksi dengan menemukan pola dalam data Anda. Amazon ML-model: klasifikasi biner, klasifikasi multikelas, klasifikasi multikelas, dan regresi.

Tabel berikut mendefinisikan istilah yang terkait dengan model ML.

Jangka waktu	Definisi
Regresi	Tujuan dari pelatihan model ML-regresi adalah untuk memprediksi nilai numerik.
Multiclass	Tujuan pelatihan model multikelas adalah untuk memprediksi nilai-nilai yang termasuk dalam rangkaian nilai yang diizinkan yang terbatas dan telah ditentukan.
Biner	Tujuan pelatihan model biner ML adalah untuk memprediksi nilai-nilai yang hanya dapat memiliki satu dari dua negara, seperti true atau false.
Ukuran model	Model ML-menangkap dan menyimpan pola. Semakin banyak pola model ML-toko, semakin besar akan. Ukuran model L dijelaskan dalam Mbytes.
Jumlah Pass	Saat Anda melatih model ML-nya, Anda menggunakan data dari sumber data. Kadang-kadang bermanfaat untuk menggunakan setiap catatan data dalam proses pembelajaran lebih dari satu kali. Jumlah berapa kali Anda membiarkan Amazon menggunakan catatan data yang sama disebut jumlah pass.
Regularisasi	Regularisasi adalah teknik machine learning yang dapat Anda gunakan untuk mendapatkan model berkualitas tinggi. Amazon ML-menawarkan pengaturan default yang berfungsi dengan baik untuk sebagian besar kasus.

evaluasi

Evaluasi mengukur kualitas model ML Anda dan menentukan apakah itu berkinerja baik.

Tabel berikut mendefinisikan istilah yang terkait dengan evaluasi.

Jangka waktu	Definisi
Wawasan Model	Amazon IL memberi Anda metrik dan sejumlah wawasan yang dapat Anda gunakan untuk mengevaluasi kinerja prediktif model Anda.

Jangka waktu	Definisi
AUC	Area Under the ROC Curve (AUC) mengukur kemampuan model biner untuk memprediksi skor yang lebih tinggi untuk contoh-contoh positif dibandingkan dengan contoh-contoh negatif.
Makro-rata-rata F1 skor	Skor F1-rata-rata makro digunakan untuk mengevaluasi kinerja prediktif model ML multiclass.
RMSE	Root Mean Square Error (RMSE) adalah metrik yang digunakan untuk mengevaluasi kinerja prediktif model regresi ML-nya.
Memotong	Model L bekerja dengan menghasilkan skor prediksi numerik. Dengan menerapkan nilai cut-off, sistem mengubah skor ini menjadi 0 dan 1 label.
Akurasi	Akurasi mengukur persentase prediksi yang benar.
Presisi	Presisi menunjukkan persentase contoh positif aktual (sebagai lawan positif palsu) di antara contoh-contoh yang telah diambil (yang diperkirakan positif). Dengan kata lain, berapa banyak item yang dipilih yang positif?
Recall	Ingat menunjukkan persentase positif aktual di antara jumlah total kasus yang relevan (aktual positif). Dengan kata lain, berapa banyak item positif yang dipilih?

Prediksi Batch

Prediksi Batch adalah untuk satu set pengamatan yang dapat dijalankan sekaligus. Ini sangat ideal untuk analisis prediktif yang tidak memiliki persyaratan real-time.

Tabel berikut mendefinisikan istilah yang terkait dengan prediksi batch.

Jangka waktu	Definisi
Lokasi output	Hasil prediksi batch disimpan di lokasi output bucket S3.
File Manifes	File ini terkait setiap file input dengan hasil prediksi batch yang terkait. Ini disimpan di lokasi keluaran bucket S3.

Prediksi waktu nyata

Prediksi real-time adalah untuk aplikasi dengan persyaratan latensi rendah, seperti aplikasi web interaktif, seluler, atau desktop. Setiap model ML-nya dapat dipertanyakan untuk prediksi dengan menggunakan API prediksi real-time latensi rendah.

Tabel berikut mendefinisikan istilah yang terkait dengan prediksi real-time.

Jangka waktu	Definisi
API prediksi waktu nyata	Real-time Prediction API menerima observasi masukan tunggal dalam payload permintaan dan mengembalikan prediksi dalam respon.
Titik akhir prediksi	Untuk menggunakan model L dengan API prediksi real-time, Anda perlu membuat titik akhir prediksi real-time. Setelah dibuat, endpoint berisi URL yang dapat Anda gunakan untuk meminta prediksi real-time.

Mengakses Amazon Machine Learning

Anda dapat mengakses Amazon MLI dengan menggunakan salah satu hal berikut:

Konsol Amazon MLI

Anda dapat mengakses konsol Amazon ML-dengan masuk ke AWS Management Console, dan membuka konsol Amazon ML-nyahttps://console.aws.amazon.com/machinelearning/.

AWS CLI

Untuk informasi tentang cara menginstal dan mengonfigurasi AWS CLI, lihat Menyiapkan dengan Antarmuka Baris Perintah AWS dalamAWS Command Line InterfacePanduan Pengguna.

API Amazon

Untuk informasi selengkapnya tentang API Amazon MLI, lihatReferensi Amazon API.

AWS SDK

Untuk informasi selengkapnya tentang AWS SDKs, lihatAlat untuk Amazon Web Services.

Wilayah dan titik akhir

Amazon Machine Learning (Amazon ML) mendukung titik akhir prediksi real-time di dua wilayah berikut:

Nama wilayah	Wilayah	Titik akhir	Protokol
US East (N. Virginia)	us-east-1	machinelearning.us-east-1.amazonaws.com	HTTPS
Europe (Ireland)	eu-west-1	machinelearning.eu-west-1.amazonaws.com	HTTPS

Anda dapat meng-host set data, melatih dan mengevaluasi model, dan memicu prediksi di wilayah mana pun.

Kami menyarankan agar Anda menyimpan semua sumber daya Anda di wilayah yang sama. Jika data masukan Anda berada di wilayah yang berbeda dari sumber daya Amazon ML-mu, Anda akan memperoleh biaya transfer data lintas regional. Anda dapat memanggil titik akhir prediksi real-time dari wilayah mana pun, tetapi memanggil titik akhir dari wilayah yang tidak memiliki titik akhir yang Anda panggil dapat memengaruhi latensi prediksi real-time.

Harga Amazon MLS

DenganAWSlayanan, Anda hanya membayar atas apa yang Anda gunakan. Tidak ada biaya minimum dan tidak ada komitmen di muka.

Amazon Machine Learning (Amazon ML) mengenakan tarif per jam untuk waktu komputasi yang digunakan untuk menghitung statistik data dan melatih dan mengevaluasi model, dan kemudian Anda membayar jumlah prediksi yang dihasilkan untuk aplikasi Anda. Untuk prediksi real-time, Anda juga membayar biaya kapasitas cadangan per jam berdasarkan ukuran model Anda.

Amazon ML-memperkirakan biaya untuk prediksi hanya diKonsol Amazon MLS.

Untuk informasi selengkapnya tentang harga Amazon MLI, lihatHarga Amazon Machine Learning.

Topik

Memperkirakan Batch Prediksi Biaya

Saat Anda meminta prediksi batch dari model Amazon ML-menggunakan wizard Create Batch Prediction, Amazon MLakan memperkirakan biaya prediksi ini. Metode untuk menghitung perkiraan bervariasi berdasarkan jenis data yang tersedia.

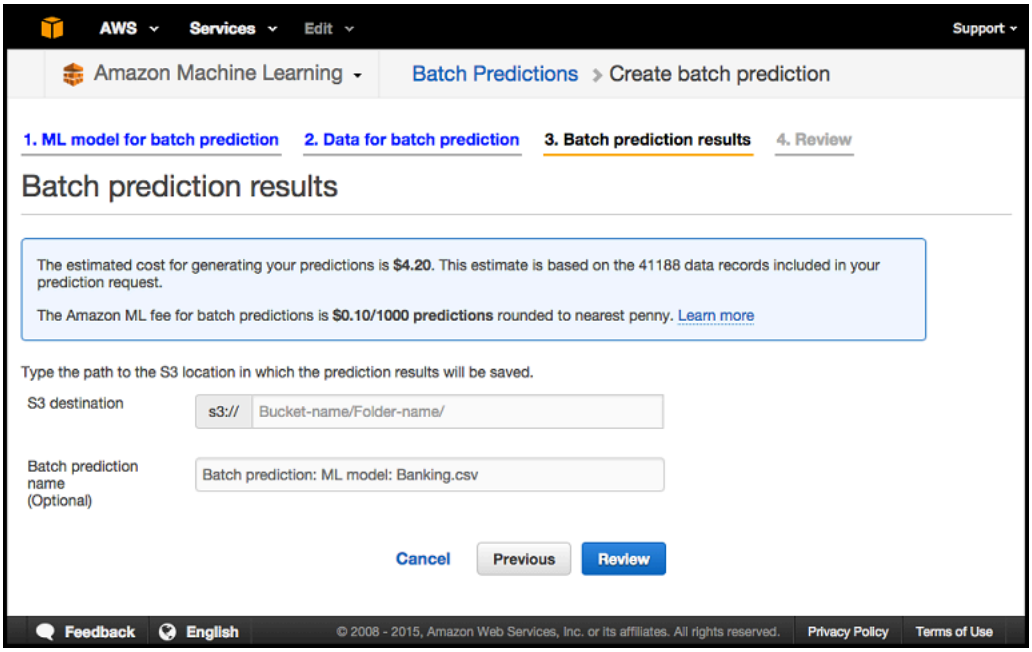
Memperkirakan Batch Prediksi Biaya Ketika Statistik Data Tersedia

Perkiraan biaya yang paling akurat diperoleh ketika Amazon L telah menghitung statistik ringkasan pada sumber data yang digunakan untuk meminta prediksi. Statistik ini selalu dihitung untuk sumber data yang telah dibuat menggunakan konsol Amazon ML-nya. Pengguna API harus mengaturComputeStatisticsbenderaTruesaat membuat sumber data secara terprogram menggunakandibuatasourcefroms3,CreateDataSourceFromRedshift, atauCreateDataSourceFromRDSAPI. Sumber data harus berada diREADYnegara untuk statistik yang akan tersedia.

Salah satu statistik yang dihitung Amazon ML-nya adalah jumlah data record. Bila jumlah catatan data tersedia, wizard Amazon ML-Create Batch Prediction memperkirakan jumlah prediksi dengan mengalikan jumlah catatan data denganbiaya untuk prediksi batch.

Biaya aktual Anda dapat bervariasi dari perkiraan ini karena alasan berikut:

- Beberapa catatan data mungkin gagal diproses. Anda tidak ditagih untuk prediksi dari catatan data yang gagal.
- Perkiraan tidak memperhitungkan kredit yang sudah ada sebelumnya atau penyesuaian lain yang diterapkan oleh AWS.



Memperkirakan Batch Prediksi Biaya Ketika Hanya Ukuran Data yang Tersedia

Ketika Anda meminta prediksi batch dan statistik data untuk sumber data permintaan tidak tersedia, Amazon ML-memperkirakan biaya berdasarkan hal-hal berikut:

- Total ukuran data yang dihitung dan bertahan selama validasi datasource
- Ukuran catatan data rata-rata, yang diperkirakan oleh Amazon ML-nya dengan membaca dan mengurai 100 MB pertama file data Anda

Untuk memperkirakan biaya prediksi batch Anda, Amazon ML-membagi total ukuran data dengan ukuran catatan data rata-rata. Metode prediksi biaya ini kurang tepat daripada metode yang digunakan ketika jumlah catatan data tersedia karena catatan pertama dari file data Anda mungkin tidak akurat mewakili ukuran catatan rata-rata.

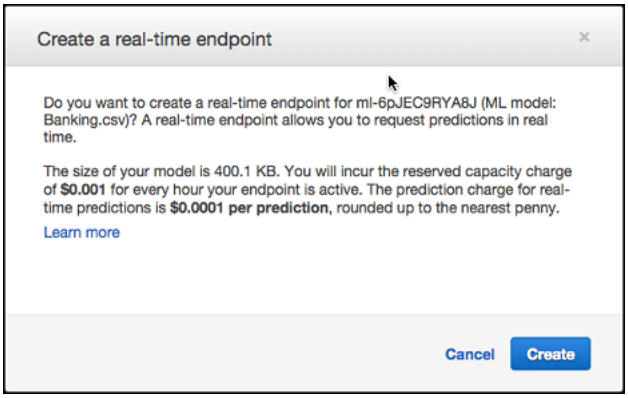
Memperkirakan Batch Prediksi Biaya Ketika Baik Statistik Data maupun Ukuran Data yang Tersedia

Ketika tidak ada statistik data maupun ukuran data yang tersedia, Amazon MLM tidak dapat memperkirakan biaya prediksi batch Anda. Hal ini biasanya terjadi ketika sumber data yang Anda gunakan untuk meminta prediksi batch belum divalidasi oleh Amazon ML-nya. Hal ini dapat terjadi ketika Anda telah membuat sumber data yang didasarkan pada kueri Amazon Redshift (Amazon Redshift) atau Amazon Relational Database Service (Amazon RDS), dan transfer data belum selesai, atau ketika pembuatan sumber data diantri di belakang operasi lain di akun Anda. Dalam hal ini,

konsol Amazon XML memberi tahu Anda tentang biaya prediksi batch. Anda dapat memilih untuk melanjutkan dengan permintaan prediksi batch tanpa perkiraan, atau untuk membatalkan wizard dan kembali setelah sumber data yang digunakan untuk prediksi dalam keadaan INPROGRESS atau READY.

Memperkirakan Biaya Prediksi Waktu Nyata

Ketika Anda membuat titik akhir prediksi real-time menggunakan konsol Amazon XML, Anda akan ditampilkan estimasi biaya kapasitas cadangan, yang merupakan biaya berkelanjutan untuk memesan titik akhir untuk pemrosesan prediksi. Biaya ini bervariasi berdasarkan ukuran model, seperti yang dijelaskan padahalaman harga layanan. Anda juga akan diberitahu tentang biaya prediksi real-time Amazon ML-nya.



Machine Learning

Machine learning (ML) dapat membantu Anda menggunakan data historis untuk membuat keputusan bisnis yang lebih baik. Algoritma XML menemukan pola dalam data, dan membangun model matematika menggunakan penemuan ini. Kemudian Anda dapat menggunakan model untuk membuat prediksi pada data masa depan. Misalnya, satu kemungkinan penerapan model pembelajaran mesin adalah memprediksi seberapa besar kemungkinan pelanggan membeli produk tertentu berdasarkan perilaku masa lalu mereka.

Topik

Memecahkan Masalah Bisnis dengan Amazon Machine Learning

Anda dapat menggunakan Amazon Machine Learning untuk menerapkan machine learning ke masalah yang Anda miliki contoh jawaban aktual yang ada. Misalnya, jika Anda ingin menggunakan Amazon Machine Learning untuk memprediksi apakah email adalah spam, Anda harus mengumpulkan contoh email yang diberi label dengan benar sebagai spam atau bukan spam. Anda kemudian dapat menggunakan machine learning untuk menggeneralisasi dari contoh email ini untuk memprediksi seberapa mungkin email baru spam atau tidak. Pendekatan pembelajaran dari data yang telah diberi label dengan jawaban sebenarnya dikenal sebagai pembelajaran mesin yang diawasi.

Anda dapat menggunakan pendekatan ML yang diawasi untuk tugas pembelajaran mesin spesifik ini: klasifikasi biner (memprediksi salah satu dari dua hasil yang mungkin), klasifikasi multiclass (memprediksi salah satu dari lebih dari dua hasil) dan regresi (memprediksi nilai numerik).

Contoh masalah klasifikasi biner:

- Akankah pelanggan membeli produk ini atau tidak membeli produk ini?
- Apakah email ini spam atau bukan spam?
- Apakah produk ini buku atau hewan ternak?

- Apakah ulasan ini ditulis oleh pelanggan atau robot?

Contoh masalah klasifikasi multiclass:

- Apakah produk ini buku, film, atau pakaian?
- Apakah film ini komedi romantis, dokumenter, atau film thriller?
- Kategori produk mana yang paling menarik bagi pelanggan ini?

Contoh masalah klasifikasi regresi:

- Apa yang akan suhu di Seattle besok?
- Untuk produk ini, berapa banyak unit yang akan menjual?
- Berapa hari sebelum pelanggan ini berhenti menggunakan aplikasi?
- Berapa harga rumah ini akan menjual?

Kapan Menggunakan Machine Learning

Penting untuk diingat bahwa L bukanlah solusi untuk setiap jenis masalah. Ada kasus-kasus tertentu di mana solusi yang kuat dapat dikembangkan tanpa menggunakan teknik ML-nya. Misalnya, Anda tidak memerlukan ML jika Anda dapat menentukan nilai target dengan menggunakan aturan sederhana, perhitungan, atau langkah-langkah yang telah ditentukan yang dapat diprogram tanpa memerlukan pembelajaran berbasis data.

Gunakan pembelajaran mesin untuk situasi berikut:

- Anda tidak dapat membuat kode aturan: Banyak tugas manusia (seperti mengenali apakah email adalah spam atau bukan spam) tidak dapat diselesaikan secara memadai menggunakan solusi sederhana (deterministik), berbasis aturan. Sejumlah besar faktor dapat mempengaruhi jawabannya. Ketika aturan tergantung pada terlalu banyak faktor dan banyak dari aturan ini tumpang tindih atau perlu disetel dengan sangat halus, segera menjadi sulit bagi manusia untuk secara akurat kode aturan. Anda dapat menggunakan ML-nya untuk mengatasi masalah ini secara efektif.
- Anda tidak dapat skala: Anda mungkin dapat mengenali beberapa ratus email secara manual dan memutuskan apakah mereka spam atau tidak. Namun, tugas ini menjadi membosankan bagi jutaan email. Solusi ML-efektif dalam menangani masalah berskala besar.

Membangun Aplikasi Machine Learning

Membangun aplikasi L adalah proses berulang yang melibatkan urutan langkah. Untuk membuat aplikasi ML-nya, ikuti langkah-langkah umum berikut:

1. Bingkai masalah inti inti (s) dalam hal apa yang diamati dan jawaban apa yang Anda inginkan model untuk memprediksi.
2. Mengumpulkan, membersihkan, dan mempersiapkan data untuk membuatnya cocok untuk konsumsi oleh algoritma pelatihan model L. Visualisasikan dan analisis data untuk menjalankan pemeriksaan kewarasan untuk memvalidasi kualitas data dan untuk memahami data.
3. Seringkali, data mentah (variabel input) dan jawaban (target) tidak diwakili dengan cara yang dapat digunakan untuk melatih model yang sangat prediktif. Oleh karena itu, Anda biasanya harus mencoba untuk membangun representasi input yang lebih prediktif atau fitur dari variabel mentah.
4. Umpan fitur yang dihasilkan ke algoritma pembelajaran untuk membangun model dan mengevaluasi kualitas model pada data yang dipegang dari bangunan model.
5. Gunakan model untuk menghasilkan prediksi jawaban target untuk instance data baru.

Merumuskan Masalah

Langkah pertama dalam pembelajaran mesin adalah memutuskan apa yang ingin Anda prediksi, yang dikenal sebagai label atau jawaban target. Bayangkan sebuah skenario di mana Anda ingin memproduksi produk, tetapi keputusan Anda untuk memproduksi setiap produk tergantung pada jumlah penjualan potensial. Dalam skenario ini, Anda ingin memprediksi berapa kali setiap produk akan dibeli (memprediksi jumlah penjualan). Ada beberapa cara untuk menentukan masalah ini dengan menggunakan machine learning. Memilih cara menentukan masalah tergantung pada kasus penggunaan atau kebutuhan bisnis Anda.

Apakah Anda ingin memprediksi jumlah pembelian pelanggan Anda akan membuat untuk setiap produk (dalam hal target numerik dan Anda memecahkan masalah regresi)? Atau apakah Anda ingin memprediksi produk mana yang akan mendapatkan lebih dari 10 pembelian (dalam hal ini targetnya adalah biner dan Anda memecahkan masalah klasifikasi biner)?

Penting untuk menghindari masalah yang terlalu rumit dan membingkai solusi paling sederhana yang memenuhi kebutuhan Anda. Namun, penting juga untuk menghindari kehilangan informasi, terutama informasi dalam jawaban historis. Di sini, mengubah nomor penjualan masa lalu yang sebenarnya menjadi variabel biner “lebih dari 10" versus “lebih sedikit” akan kehilangan informasi berharga.

Menginvestasikan waktu dalam menentukan target mana yang paling masuk akal bagi Anda untuk memprediksi akan menyelamatkan Anda dari membangun model yang tidak menjawab pertanyaan Anda.

Data Berlabel

Masalah ML dimulai dengan data—sebaiknya, banyak data (contoh atau pengamatan) yang sudah Anda ketahui jawaban target. Data yang sudah Anda ketahui jawaban target disebutdata berlabel. Dalam ML yang diawasi, algoritma mengajarkan dirinya sendiri untuk belajar dari contoh berlabel yang kami sediakan.

Setiap contoh/pengamatan dalam data Anda harus berisi dua elemen:

- Target — Jawaban yang ingin Anda prediksi. Anda memberikan data yang diberi label dengan target (jawaban yang benar) ke algoritma ML-nya untuk dipelajari. Kemudian, Anda akan menggunakan model ML-terlatih untuk memprediksi jawaban ini pada data yang Anda tidak tahu jawaban target.
- Variabel/fitur - Ini adalah atribut dari contoh yang dapat digunakan untuk mengidentifikasi pola untuk memprediksi jawaban target.

Misalnya, untuk masalah klasifikasi email, target adalah label yang menunjukkan apakah email spam atau bukan spam. Contoh variabel adalah pengirim email, teks di badan email, teks di baris subjek, waktu email dikirim, dan adanya korespondensi sebelumnya antara pengirim dan penerima.

Seringkali, data tidak tersedia dalam bentuk berlabel. Mengumpulkan dan mempersiapkan variabel dan target sering merupakan langkah yang paling penting dalam memecahkan masalah ML-nya. Contoh data harus mewakili data yang akan Anda miliki ketika Anda menggunakan model untuk membuat prediksi. Misalnya, jika Anda ingin memprediksi apakah email spam atau tidak, Anda harus mengumpulkan baik positif (email spam) dan negatif (email non-spam) untuk algoritma pembelajaran mesin untuk dapat menemukan pola yang akan membedakan antara dua jenis email.

Setelah Anda memiliki data berlabel, Anda mungkin perlu mengubahnya menjadi format yang dapat diterima oleh algoritma atau perangkat lunak Anda. Misalnya, untuk menggunakan Amazon MLAnda perlu mengonversi data ke format yang dipisahkan koma (CSV) dengan setiap contoh yang membentuk satu baris file CSV, setiap kolom yang berisi satu variabel input, dan satu kolom yang berisi jawaban target.

Menganalisis Data Anda

Sebelum memasukkan data berlabel Anda ke algoritma ML-nya, adalah praktik yang baik untuk memeriksa data Anda untuk mengidentifikasi masalah dan mendapatkan wawasan tentang data yang Anda gunakan. Kekuatan prediktif model Anda hanya akan sama baiknya dengan data yang Anda beri makan.

Saat menganalisis data Anda, Anda harus mengingat hal berikut:

- Ringkasan data variabel dan target - Hal ini berguna untuk memahami nilai-nilai yang diambil variabel Anda dan nilai mana yang dominan dalam data Anda. Anda dapat menjalankan ringkasan ini oleh ahli materi pelajaran untuk masalah yang ingin Anda selesaikan. Tanyakan pada diri sendiri atau ahli materi pelajaran: Apakah data sesuai dengan harapan Anda? Apakah terlihat seperti Anda memiliki masalah pengumpulan data? Apakah satu kelas dalam target Anda lebih sering daripada kelas lain? Apakah ada lebih banyak nilai yang hilang atau data tidak valid dari yang Anda harapkan?
- Korelasi variabel-target - Mengetahui korelasi antara setiap variabel dan kelas target sangat membantu karena korelasi tinggi menyiratkan bahwa ada hubungan antara variabel dan kelas target. Secara umum, Anda ingin memasukkan variabel dengan korelasi tinggi karena mereka adalah orang-orang dengan daya prediktif yang lebih tinggi (sinyal), dan meninggalkan variabel dengan korelasi rendah karena mereka mungkin tidak relevan.

Di Amazon IL, Anda dapat menganalisis data Anda dengan membuat sumber data dan dengan meninjau laporan data yang dihasilkan.

Pengolahan Fitur

Setelah mengenal data Anda melalui ringkasan data dan visualisasi, Anda mungkin ingin mengubah variabel Anda lebih jauh untuk membuatnya lebih bermakna. Hal ini dikenal sebagaipengolahan fitur. Misalnya, katakanlah Anda memiliki variabel yang menangkap tanggal dan waktu di mana suatu peristiwa terjadi. Tanggal dan waktu ini tidak akan pernah terjadi lagi dan karenanya tidak akan berguna untuk memprediksi target Anda. Namun, jika variabel ini diubah menjadi fitur yang mewakili jam hari, hari dalam seminggu, dan bulan, variabel ini bisa berguna untuk mengetahui apakah acara cenderung terjadi pada jam tertentu, hari kerja, atau bulan. Pemrosesan fitur tersebut untuk membentuk titik data yang lebih umum untuk dipelajari dapat memberikan perbaikan yang signifikan pada model prediktif.

Contoh lain dari pemrosesan fitur umum:

- Mengganti data yang hilang atau tidak valid dengan nilai yang lebih berarti (misalnya, jika Anda tahu bahwa nilai yang hilang untuk variabel jenis produk sebenarnya berarti itu adalah buku, Anda kemudian dapat mengganti semua nilai yang hilang dalam jenis produk dengan nilai buku). Strategi umum yang digunakan untuk impute nilai yang hilang adalah mengganti nilai yang hilang dengan nilai rata-rata atau median. Penting untuk memahami data Anda sebelum memilih strategi untuk mengganti nilai yang hilang.
- Membentuk produk Cartesian dari satu variabel dengan yang lain. Misalnya, jika Anda memiliki dua variabel, seperti kepadatan populasi (urban, suburban, rural) dan state (Washington, Oregon, California), mungkin ada informasi yang berguna dalam fitur yang dibentuk oleh produk Cartesian dari dua variabel ini menghasilkan fitur (Urban_Washington, Suburban_Washington, rural_ Washington, Urban_Oregon, Suburban_Oregon, Rural_oregon, Urban_California, Suburban_California, Rural_California).
- Transformasi non-linear seperti binning variabel numerik ke kategori. Dalam banyak kasus, hubungan antara fitur numerik dan target tidak linear (nilai fitur tidak meningkat atau menurun secara monoton dengan target). Dalam kasus seperti itu, mungkin berguna untuk bin fitur numerik ke dalam fitur kategoris yang mewakili rentang yang berbeda dari fitur numerik. Setiap fitur kategoris (bin) kemudian dapat dimodelkan sebagai memiliki hubungan linier sendiri dengan target. Misalnya, katakanlah Anda tahu bahwa usia fitur numerik terus menerus tidak berkorelasi linear dengan kemungkinan untuk membeli buku. Anda dapat bin usia ke fitur kategoris yang mungkin dapat menangkap hubungan dengan target lebih akurat. Jumlah optimum sampah untuk variabel numerik tergantung pada karakteristik variabel dan hubungannya dengan target, dan ini paling baik ditentukan melalui eksperimen. Amazon IL menyarankan nomor bin optimal untuk fitur numerik berdasarkan statistik data dalam resep yang disarankan. Lihat Panduan Pengembang untuk rincian tentangresep yang disarankan.
- Fitur khusus domain (misalnya, Anda memiliki panjang, lebar, dan tinggi sebagai variabel terpisah; Anda dapat membuat fitur volume baru untuk menjadi produk dari ketiga variabel ini).
- Fitur variabel-spesifik. Beberapa tipe variabel seperti fitur teks, fitur yang menangkap struktur halaman web, atau struktur kalimat memiliki cara pemrosesan generik yang membantu mengekstrak struktur dan konteks. Misalnya, membentukgram ndari teks “rubah melompati pagar” dapat direpresentasikan denganunigram: rubah, melompat, lebih, pagar ataubigram: rubah, rubah melompat, melompati, di atas, pagar.

Termasuk fitur yang lebih relevan membantu meningkatkan daya prediksi. Jelas, tidak selalu mungkin untuk mengetahui fitur dengan “sinyal” atau pengaruh prediktif terlebih dahulu. Jadi ada baiknya untuk memasukkan semua fitur yang berpotensi terkait dengan label target dan membiarkan

algoritma pelatihan model memilih fitur dengan korelasi terkuat. Di Amazon ML-nya, pemrosesan fitur dapat ditentukan dalam resep saat membuat model. Lihat Panduan Pengembang untuk daftar prosesor fitur yang tersedia.

Memisahkan Data menjadi Data Pelatihan dan Evaluasi

Tujuan mendasar dari ML adalah untukmenggeneralisasidi luar contoh data yang digunakan untuk melatih model. Kami ingin mengevaluasi model untuk memperkirakan kualitas generalisasi pola untuk data model belum dilatih. Namun, karena instance di masa depan memiliki nilai target yang tidak diketahui dan kami tidak dapat memeriksa keakuratan prediksi kami untuk instance mendatang sekarang, kita perlu menggunakan beberapa data yang sudah kita ketahui jawabannya sebagai proxy untuk data masa depan. Mengevaluasi model dengan data yang sama yang digunakan untuk pelatihan tidak berguna, karena memberi penghargaan kepada model yang dapat “mengingat” data pelatihan, sebagai lawan generalisasi darinya.

Strategi umum adalah mengambil semua data berlabel yang tersedia, dan membaginya menjadi subset pelatihan dan evaluasi, biasanya dengan rasio 70-80 persen untuk pelatihan dan 20-30 persen untuk evaluasi. Sistem L menggunakan data pelatihan untuk melatih model untuk melihat pola, dan menggunakan data evaluasi untuk mengevaluasi kualitas prediktif model yang terlatih. Sistem L mengevaluasi kinerja prediktif dengan membandingkan prediksi pada data evaluasi yang ditetapkan dengan nilai sebenarnya (dikenal sebagai ground truth) menggunakan berbagai metrik. Biasanya, Anda menggunakan model “terbaik” pada bagian evaluasi untuk membuat prediksi pada contoh masa depan yang Anda tidak tahu jawaban target.

Amazon ML-membagi data yang dikirim untuk melatih model melalui konsol Amazon ML-70 persen untuk pelatihan dan 30 persen untuk evaluasi. Secara default, Amazon IL menggunakan 70 persen pertama dari data masukan dalam urutan yang muncul dalam data sumber untuk sumber data pelatihan dan 30 persen sisanya dari data untuk sumber data evaluasi. Amazon IL juga memungkinkan Anda untuk memilih 70 persen data sumber acak untuk pelatihan alih-alih menggunakan 70 persen pertama, dan menggunakan pelengkap subset acak ini untuk evaluasi. Anda dapat menggunakan API Amazon IL untuk menentukan rasio split khusus dan untuk menyediakan data pelatihan dan evaluasi yang terbagi di luar Amazon IL. Amazon IL juga menyediakan strategi untuk membagi data Anda. Untuk informasi selengkapnya tentang strategi pemisahan, lihatMemisahkan Data Anda.

Pelatihan Model

Anda sekarang siap untuk menyediakan algoritma ML (yaitu,algoritme pembelajaran) dengan data pelatihan. Algoritma akan belajar dari pola data pelatihan yang memetakan variabel ke target, dan itu

akan menghasilkan model yang menangkap hubungan ini. Model L kemudian dapat digunakan untuk mendapatkan prediksi pada data baru yang Anda tidak tahu jawaban target.

Model linier

Ada sejumlah besar model ML-nya yang tersedia. Amazon IL mempelajari satu jenis model ML: model linier. Istilah model linier menyiratkan bahwa model ditentukan sebagai kombinasi linear fitur. Berdasarkan data pelatihan, proses pembelajaran menghitung satu bobot untuk setiap fitur untuk membentuk model yang dapat memprediksi atau memperkirakan nilai target. Misalnya, jika target Anda adalah jumlah asuransi yang akan dibeli pelanggan dan variabel Anda adalah usia dan pendapatan, model linier sederhana adalah sebagai berikut:

```
Estimated target = 0.2 + 5·age + 0.0003·income
```

Algoritma Pembelajaran

Tugas algoritma pembelajaran adalah mempelajari bobot untuk model. Bobot menggambarkan kemungkinan bahwa pola yang dipelajari model mencerminkan hubungan aktual dalam data. Algoritma pembelajaran terdiri dari fungsi kerugian dan teknik optimasi. Kerugian adalah hukuman yang terjadi ketika perkiraan target yang diberikan oleh model MLnya tidak sama persis dengan target. Sebuah fungsi kerugian mengukur hukuman ini sebagai nilai tunggal. Teknik optimasi berusaha meminimalkan kerugian. Di Amazon Machine Learning, kami menggunakan tiga fungsi kerugian, satu untuk masing-masing dari tiga jenis masalah prediksi. Teknik optimasi yang digunakan di Amazon IL adalah Stochastic Gradient Descent (SGD) online. SGD membuat umpan berurutan atas data pelatihan, dan selama setiap lulus, update fitur bobot satu contoh pada satu waktu dengan tujuan mendekati bobot optimal yang meminimalkan kerugian.

Amazon IL menggunakan algoritme pembelajaran berikut:

- Untuk klasifikasi biner, Amazon IL menggunakan regresi logistik (fungsi kerugian logistik + SGD).
- Untuk klasifikasi multiclass, Amazon IL menggunakan regresi logistik multinomial (kerugian logistik multinomial+SGD).
- Untuk regresi, Amazon IL menggunakan regresi linier (fungsi kerugian kuadrat + SGD).

Parameter Pelatihan

Algoritma pembelajaran Amazon ML-menerima parameter, yang disebut hiperparameter atau parameter pelatihan, yang memungkinkan Anda untuk mengontrol kualitas model yang dihasilkan. Bergantung pada hyperparameter, Amazon ML-memilih pengaturan secara otomatis atau

menyediakan default statis untuk hyperparameters. Meskipun pengaturan hyperparameter default umumnya menghasilkan model yang berguna, Anda mungkin dapat meningkatkan kinerja prediktif model Anda dengan mengubah nilai hyperparameter. Bagian berikut menjelaskan hiper umumparameter yang terkait dengan algoritma pembelajaran untuk model linier, seperti yang dibuat oleh Amazon IL.

Tingkat Pembelajaran

Tingkat pembelajaran adalah nilai konstan yang digunakan dalam algoritma Stochastic Gradient Descent (SGD). Tingkat belajar mempengaruhi kecepatan di mana algoritma mencapai (menyatu ke) bobot optimal. Algoritma SGD membuat update bobot model linier untuk setiap contoh data yang dilihatnya. Ukuran pembaruan ini dikendalikan oleh tingkat pembelajaran. Tingkat pembelajaran yang terlalu besar mungkin mencegah bobot mendekati solusi optimal. Hasil nilai yang terlalu kecil dalam algoritma yang membutuhkan banyak lintasan untuk mendekati bobot optimal.

Di Amazon ML-nya, tingkat pembelajaran dipilih secara otomatis berdasarkan data Anda.

Ukuran model

Jika Anda memiliki banyak fitur input, jumlah pola yang mungkin dalam data dapat menghasilkan model yang besar. Model besar memiliki implikasi praktis, seperti membutuhkan lebih banyak RAM untuk memegang model saat berlatih dan saat menghasilkan prediksi. Di Amazon IL, Anda dapat mengurangi ukuran model dengan menggunakan regularisasi L1 atau dengan secara khusus membatasi ukuran model dengan menentukan ukuran maksimum. Perhatikan bahwa jika Anda mengurangi ukuran model terlalu banyak, Anda dapat mengurangi daya prediktif model Anda.

Untuk informasi tentang ukuran model default, lihatParameter Pelatihan: Jenis dan Nilai Default. Untuk informasi selengkapnya tentang regularisasi, lihatRegularisasi.

Jumlah Pass

Algoritma SGD membuat berurutan melewati data pelatihan. Parameter`Number of passes`parameter mengontrol jumlah pass bahwa algoritma membuat lebih dari data pelatihan. Lebih melewati menghasilkan model yangpasdata yang lebih baik (jika tingkat pembelajaran tidak terlalu besar), namun manfaatnya berkurang dengan meningkatnya jumlah pass. Untuk set data yang lebih kecil, Anda dapat secara signifikan meningkatkan jumlah pass, yang memungkinkan algoritma belajar menyesuaikan data secara efektif lebih dekat. Untuk dataset yang sangat besar, satu pass mungkin cukup.

Untuk informasi tentang jumlah default pass, lihatParameter Pelatihan: Jenis dan Nilai Default.

Menyeret Data

Di Amazon IL, Anda harus mengacak data Anda karena algoritma SGD dipengaruhi oleh urutan baris dalam data pelatihan. Mengocokkan data pelatihan Anda menghasilkan model ML-nya lebih baik karena membantu algoritma SGD menghindari solusi yang optimal untuk jenis data pertama yang dilihatnya, tetapi tidak untuk berbagai data lengkap. Shuffling mencampur urutan data Anda sehingga algoritma SGD tidak menemukan satu jenis data untuk terlalu banyak pengamatan berturut-turut. Jika hanya melihat satu jenis data untuk banyak pembaruan bobot berturut-turut, algoritma mungkin tidak dapat memperbaiki bobot model untuk tipe data baru karena pembaruan mungkin terlalu besar. Selain itu, ketika data tidak disajikan secara acak, sulit bagi algoritma untuk menemukan solusi optimal untuk semua tipe data dengan cepat; dalam beberapa kasus, algoritma mungkin tidak pernah menemukan solusi optimal. Mengoyak data pelatihan membantu algoritma untuk bertemu pada solusi optimal lebih cepat.

Misalnya, Anda ingin melatih model ML-nya untuk memprediksi jenis produk, dan data pelatihan Anda mencakup jenis produk film, mainan, dan video game. Jika Anda mengurutkan data berdasarkan kolom jenis produk sebelum mengunggah data ke Amazon S3, maka algoritma akan melihat data berdasarkan abjad berdasarkan jenis produk. Algoritma ini melihat semua data Anda untuk film terlebih dahulu, dan model ML-mu mulai mempelajari pola film. Kemudian, ketika model Anda menemukan data tentang mainan, setiap pembaruan yang dibuat algoritma akan sesuai dengan model dengan jenis produk mainan, bahkan jika pembaruan tersebut menurunkan pola yang sesuai dengan film. Ini tiba-tiba beralih dari film ke jenis mainan dapat menghasilkan model yang tidak belajar bagaimana untuk memprediksi jenis produk secara akurat.

Untuk informasi tentang jenis pengocokan default, lihatParameter Pelatihan: Jenis dan Nilai Default.

Regularisasi

Regularisasi membantu mencegah model linier dari contoh data pelatihan yang terlalu pas (yaitu, menghafal pola alih-alih menggeneralisasikannya) dengan menghukum nilai berat ekstrem. L1 regularisasi memiliki efek mengurangi jumlah fitur yang digunakan dalam model dengan mendorong ke nol bobot fitur yang jika tidak akan memiliki bobot kecil. Akibatnya, hasil regularisasi L1 dalam model yang jarang dan mengurangi jumlah kebisingan dalam model. L2 regularisasi menghasilkan nilai bobot keseluruhan yang lebih kecil, dan menstabilkan bobot ketika ada korelasi tinggi antara fitur input. Anda mengontrol jumlah regularisasi L1 atau L2 yang diterapkan dengan menggunakan`Regularization type`dan`Regularization amount`parameter. Nilai regularisasi yang sangat besar dapat menghasilkan semua fitur yang memiliki bobot nol, mencegah model dari pola belajar.

Untuk informasi tentang nilai regularisasi default, lihatParameter Pelatihan: Jenis dan Nilai Default.

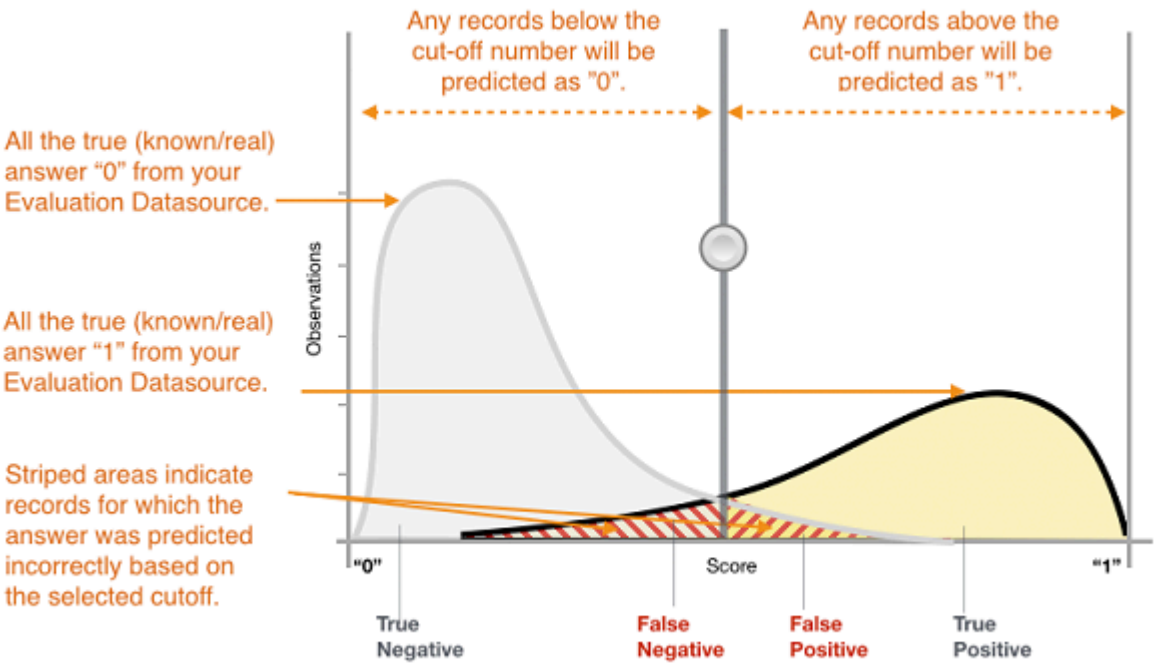
Mengevaluasi Akurasi Model

Tujuan dari model MLnya adalah mempelajari pola yang menggeneralisasi dengan baik untuk data yang tak terlihat, bukan hanya menghafal data yang ditunjukkan selama pelatihan. Setelah Anda memiliki model, penting untuk memeriksa apakah model Anda berkinerja baik pada contoh tak terlihat yang belum Anda gunakan untuk melatih model. Untuk melakukan ini, Anda menggunakan model untuk memprediksi jawaban pada dataset evaluasi (diadakan data) dan kemudian membandingkan target yang diprediksi dengan jawaban yang sebenarnya (ground truth).

Sejumlah metrik digunakan dalam ML untuk mengukur keakuratan prediktif model. Pilihan metrik akurasi tergantung pada tugas ML-nya. Penting untuk meninjau metrik ini untuk memutuskan apakah model Anda berkinerja baik.

Klasifikasi Biner

Output aktual dari banyak algoritma klasifikasi biner adalah skor prediksi. Skor menunjukkan kepastian sistem bahwa pengamatan yang diberikan milik kelas positif. Untuk membuat keputusan tentang apakah pengamatan harus diklasifikasikan sebagai positif atau negatif, sebagai konsumen dari skor ini, Anda akan menafsirkan skor dengan memilih ambang klasifikasi (cut-off) dan membandingkan skor terhadapnya. Setiap pengamatan dengan skor yang lebih tinggi dari ambang batas kemudian diprediksi sebagai kelas positif dan skor lebih rendah dari ambang yang diprediksi sebagai kelas negatif.



Gambar 1: Distribusi Skor untuk Model Klasifikasi Biner

Prediksi sekarang jatuh ke dalam empat kelompok berdasarkan jawaban yang sebenarnya diketahui dan jawaban yang diprediksi: prediksi positif yang benar (benar positif), prediksi negatif yang benar (negatif sejati), prediksi positif yang salah (false positive) dan prediksi negatif yang salah (negatif palsu).

Metrik akurasi klasifikasi biner mengukur dua jenis prediksi yang benar dan dua jenis kesalahan. Metrik tipikal adalah akurasi (ACC), presisi, recall, false positive rate, F1-measure. Setiap metrik mengukur aspek yang berbeda dari model prediktif. Akurasi (ACC) mengukur fraksi prediksi yang benar. Presisi mengukur fraksi positif aktual di antara contoh-contoh yang diprediksi positif. Ingat mengukur berapa banyak positif aktual yang diprediksi positif. F1-measure adalah mean harmonik presisi dan recall.

AUC adalah jenis metrik yang berbeda. Ini mengukur kemampuan model untuk memprediksi skor yang lebih tinggi untuk contoh-contoh positif dibandingkan dengan contoh-contoh negatif. Karena AUC tidak tergantung pada ambang batas yang dipilih, Anda bisa merasakan kinerja prediksi model Anda dari metrik AUC tanpa memilih ambang batas.

Tergantung pada masalah bisnis Anda, Anda mungkin lebih tertarik pada model yang berkinerja baik untuk subset tertentu dari metrik ini. Misalnya, dua aplikasi bisnis mungkin memiliki persyaratan yang sangat berbeda untuk model ML-nya:

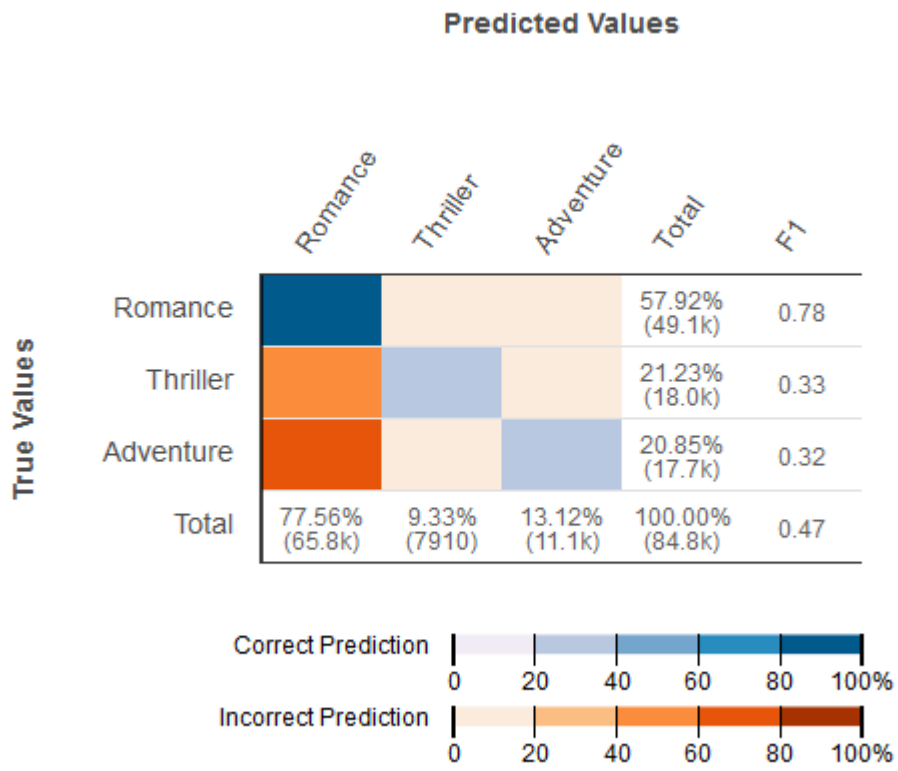
- Satu aplikasi mungkin perlu sangat yakin tentang prediksi positif yang sebenarnya positif (presisi tinggi) dan mampu mengklasifikasikan beberapa contoh positif sebagai negatif (recall moderat).
- Aplikasi lain mungkin perlu memprediksi dengan benar sebanyak mungkin contoh positif (recall tinggi) dan akan menerima beberapa contoh negatif yang salah diklasifikasikan sebagai positif (presisi moderat).

Di Amazon IL, pengamatan mendapatkan skor yang diprediksi dalam kisaran [0,1]. Ambang batas skor untuk membuat keputusan mengklasifikasikan contoh sebagai 0 atau 1 diatur secara default menjadi 0,5. Amazon IL memungkinkan Anda untuk meninjau implikasi memilih ambang skor yang berbeda dan memungkinkan Anda untuk memilih ambang batas yang sesuai dengan kebutuhan bisnis Anda.

Klasifikasi Multiclass

Berbeda dengan proses untuk masalah klasifikasi biner, Anda tidak perlu memilih ambang skor untuk membuat prediksi. Jawaban yang diprediksi adalah kelas (yaitu, label) dengan skor prediksi tertinggi. Dalam beberapa kasus, Anda mungkin ingin menggunakan jawaban yang diprediksi hanya jika diprediksi dengan skor tinggi. Dalam hal ini, Anda dapat memilih ambang batas pada skor yang diprediksi berdasarkan mana Anda akan menerima jawaban yang diprediksi atau tidak.

Metrik tipikal yang digunakan dalam multiclass sama dengan metrik yang digunakan dalam kasus klasifikasi biner. Metrik dihitung untuk setiap kelas dengan memperlakukannya sebagai masalah klasifikasi biner setelah mengelompokkan semua kelas lain sebagai milik kelas kedua. Kemudian metrik biner rata-rata atas semua kelas untuk mendapatkan rata-rata makro (memperlakukan setiap kelas sama) atau rata-rata tertimbang (tertimbang dengan frekuensi kelas) metrik. Di Amazon ML-rata-rata makro digunakan untuk mengevaluasi keberhasilan prediktif dari classifier multiclass.

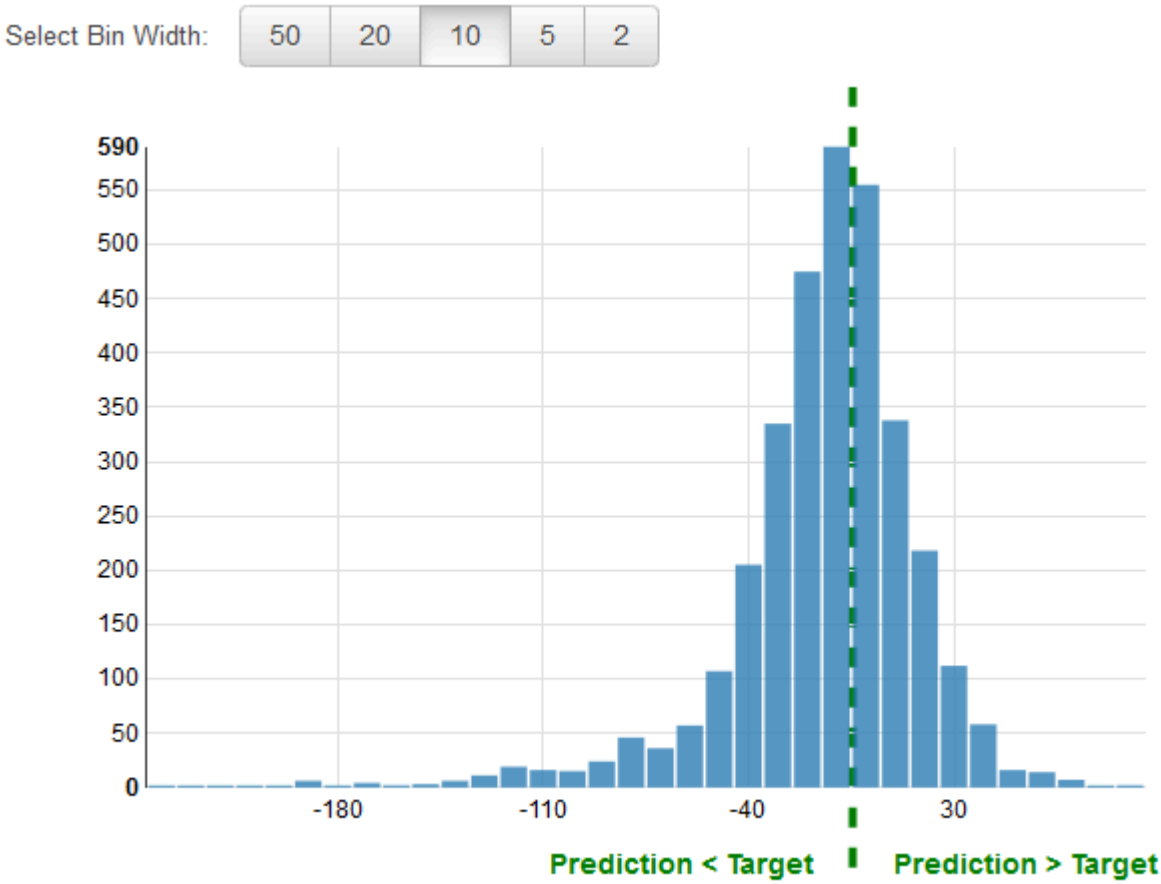


Gambar 2: Kebingungan Matrix untuk model klasifikasi multiclass

Hal ini berguna untuk meninjaumatriks kebingunganuntuk masalah multiclass. Matriks kebingungan adalah tabel yang menunjukkan setiap kelas dalam data evaluasi dan jumlah atau persentase prediksi yang benar dan prediksi yang salah.

Regresi

Untuk tugas regresi, metrik akurasi khas adalah root mean square error (RMSE) dan mean absolute percentage error (MAPE). Metrik ini mengukur jarak antara target numerik yang diprediksi dan jawaban numerik aktual (ground truth). Di Amazon MLE, metrik RMSE digunakan untuk mengevaluasi akurasi prediktif model regresi.



Gambar 3: Distribusi residu untuk model Regresi

Ini adalah praktik umum untuk meninjauresiduuntuk masalah regresi. Sisa untuk pengamatan dalam data evaluasi adalah perbedaan antara target sebenarnya dan target yang diprediksi. Residu mewakili bagian dari target bahwa model tidak dapat memprediksi. Sisa positif menunjukkan bahwa model meremehkan target (target sebenarnya lebih besar dari target yang diprediksi). Residual negatif menunjukkan overestimation (target sebenarnya lebih kecil dari target yang diprediksi). Histogram residu pada data evaluasi ketika didistribusikan dalam bentuk lonceng dan berpusat pada nol menunjukkan bahwa model membuat kesalahan secara acak dan tidak secara sistematis atas atau di bawah memprediksi kisaran tertentu nilai target. Jika residu tidak membentuk bentuk lonceng berpusat nol, ada beberapa struktur dalam kesalahan prediksi model. Menambahkan lebih banyak variabel ke model mungkin membantu model menangkap pola yang tidak ditangkap oleh model saat ini.

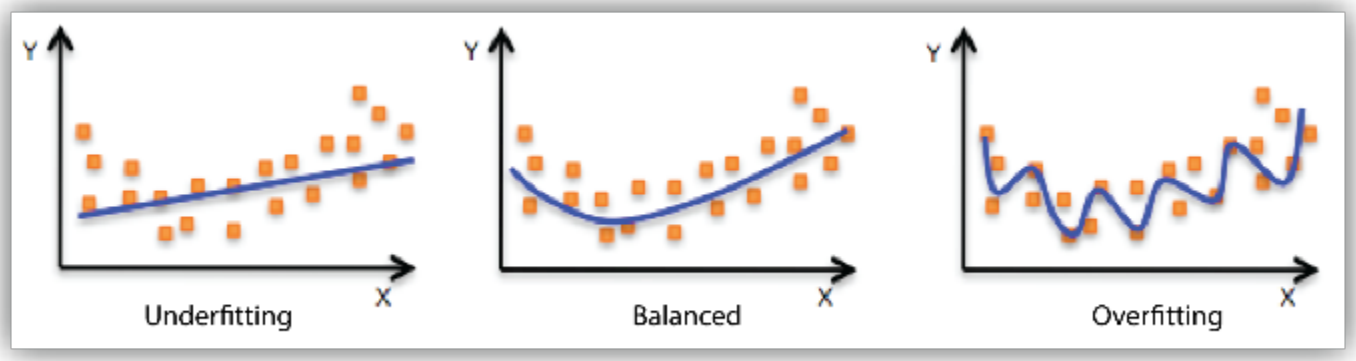
Meningkatkan Akurasi Model

Mendapatkan model yang sesuai dengan kebutuhan Anda biasanya melibatkan iterasi melalui proses ML-nya dan mencoba beberapa variasi. Anda mungkin tidak mendapatkan model yang sangat prediktif dalam iterasi pertama, atau Anda mungkin ingin meningkatkan model Anda untuk mendapatkan prediksi yang lebih baik. Untuk meningkatkan kinerja, Anda dapat melakukan iterasi melalui langkah-langkah berikut:

1. Mengumpulkan data: Meningkatkan jumlah contoh pelatihan
2. Pengolahan fitur: Tambahkan lebih banyak variabel dan pemrosesan fitur yang lebih baik
3. Penyetelan parameter model: Pertimbangkan nilai alternatif untuk parameter pelatihan yang digunakan oleh algoritma pembelajaran Anda

Cocok model: Overfitting vs. Overfitting

Memahami model fit penting untuk memahami akar penyebab akurasi model yang buruk. Pemahaman ini akan memandu Anda untuk mengambil langkah-langkah korektif. Kita dapat menentukan apakah model prediktif adalah underfitting atau overfitting data pelatihan dengan melihat kesalahan prediksi pada data pelatihan dan data evaluasi.



Model Andaunderfittingdata pelatihan ketika model melakukan buruk pada data pelatihan. Hal ini karena model tidak dapat menangkap hubungan antara contoh input (sering disebut X) dan nilai target (sering disebut Y). Model AndaoverfittingData pelatihan Anda ketika Anda melihat bahwa model bekerja dengan baik pada data pelatihan tetapi tidak berkinerja baik pada data evaluasi. Hal ini karena model menghafal data yang telah dilihat dan tidak dapat menggeneralisasi contoh yang tak terlihat.

Kinerja yang buruk pada data pelatihan bisa jadi karena modelnya terlalu sederhana (fitur input tidak cukup ekspresif) untuk menggambarkan target dengan baik. Kinerja dapat ditingkatkan dengan meningkatkan fleksibilitas model. Untuk meningkatkan fleksibilitas model, coba hal berikut:

- Tambahkan fitur khusus domain baru dan lebih banyak fitur produk Cartesian, dan ubah jenis pemrosesan fitur yang digunakan (misalnya, meningkatkan ukuran n-gram)
- Mengurangi jumlah regularisasi yang digunakan

Jika model Anda kelebihan data pelatihan, masuk akal untuk mengambil tindakan yang mengurangi fleksibilitas model. Untuk mengurangi fleksibilitas model, coba hal berikut:

- Pilihan fitur: pertimbangkan untuk menggunakan kombinasi fitur yang lebih sedikit, kurangi ukuran n-gram, dan kurangi jumlah tempat sampah atribut numerik.
- Meningkatkan jumlah regularisasi yang digunakan.

Akurasi pada data pelatihan dan uji bisa menjadi buruk karena algoritma pembelajaran tidak memiliki cukup data untuk dipelajari. Anda dapat meningkatkan kinerja dengan melakukan hal berikut:

- Meningkatkan jumlah contoh data pelatihan.
- Tingkatkan jumlah pass pada data pelatihan yang ada.

Menggunakan Model untuk Membuat Prediksi

Sekarang setelah Anda memiliki model L yang berkinerja baik, Anda akan menggunakannya untuk membuat prediksi. Di Amazon Machine Learning, ada dua cara untuk menggunakan model untuk membuat prediksi:

Prediksi Batch

Prediksi Batch berguna ketika Anda ingin menghasilkan prediksi untuk satu set pengamatan sekaligus, dan kemudian mengambil tindakan pada persentase tertentu atau jumlah pengamatan. Biasanya, Anda tidak memiliki persyaratan latensi rendah untuk aplikasi semacam itu. Misalnya, ketika Anda ingin memutuskan mana pelanggan untuk menargetkan sebagai bagian dari kampanye iklan untuk produk, Anda akan mendapatkan skor prediksi untuk semua pelanggan, mengurutkan prediksi model Anda untuk mengidentifikasi pelanggan mana yang paling mungkin untuk membeli, dan kemudian menargetkan mungkin atas 5% pelanggan yang paling mungkin untuk membeli.

Prediksi online

Skenario prediksi online adalah untuk kasus ketika Anda ingin menghasilkan prediksi secara satu-per satu untuk setiap contoh independen dari contoh lain, dalam lingkungan latensi rendah. Misalnya, Anda dapat menggunakan prediksi untuk membuat keputusan segera tentang apakah transaksi tertentu mungkin merupakan transaksi penipuan.

Model Pelatihan Ulang pada Data Baru

Agar model memprediksi secara akurat, data yang dibuat prediksi harus memiliki distribusi yang sama dengan data dimana model dilatih. Karena distribusi data dapat diharapkan melayang dari waktu ke waktu, menyebarkan model bukanlah latihan satu kali melainkan proses yang berkelanjutan. Ini adalah praktik yang baik untuk terus memantau data yang masuk dan melatih kembali model Anda pada data yang lebih baru jika Anda menemukan bahwa distribusi data telah menyimpang secara signifikan dari distribusi data pelatihan asli. Jika data pemantauan untuk mendeteksi perubahan dalam distribusi data memiliki overhead yang tinggi, maka strategi yang lebih sederhana adalah melatih model secara berkala, misalnya harian, mingguan, atau bulanan. Untuk melatih kembali model di Amazon IL, Anda perlu membuat model baru berdasarkan data pelatihan baru Anda.

Proses Amazon Machine Learning

Tabel berikut menjelaskan cara menggunakan konsol Amazon XML untuk melakukan proses ML-nya yang diuraikan dalam dokumen ini.

Proses MLS	Tugas Amazon
Menganalisis data Anda	Untuk menganalisis data Anda di Amazon, buat sumber data dan tinjau halaman wawasan data.
Membagi data menjadi sumber data pelatihan dan evaluasi	Amazon L dapat membagi sumber data untuk menggunakan 70% data untuk pelatihan model dan 30% untuk mengevaluasi kinerja prediktif model Anda. Saat Anda menggunakan wizard Create ML-Model dengan pengaturan default, Amazon ML-membagi data untuk Anda. Jika Anda menggunakan wizard Create ML-Model dengan pengatura n khusus, dan memilih untuk mengevaluasi model MLnya, Anda akan

Proses MLS	Tugas Amazon
	melihat opsi untuk mengizinkan Amazon ML-membagi data untuk Anda dan menjalankan evaluasi pada 30% data.
Rentang data pelatihan	Saat Anda menggunakan wizard Create ML-Model dengan pengatura n default, Amazon ML-shuffle data Anda untuk Anda. Anda juga dapat mengacak data Anda sebelum mengimpornya ke Amazon ML-nya.
Fitur Proses	Proses menyusun data pelatihan dalam format yang optimal untuk pembelajaran dan generalisasi dikenal sebagai transformasi fitur. Saat Anda menggunakan wizard Create ML-Model dengan pengaturan default, Amazon MLG menyarankan pengaturan pemrosesan fitur untuk data Anda. Untuk menentukan pengaturan pemrosesan fitur, gunakan wizard Buat Model ML'sKhususpilihan dan memberikan resep pengolahan fitur.
Melatih modelnya	Saat Anda menggunakan wizard Create ML-Model untuk membuat model di Amazon ML-nya, Amazon ML-melatih model Anda.
Pilih parameter model	Di Amazon XML, Anda dapat menyetel empat parameter yang memengaruhi kinerja prediktif model Anda: ukuran model, jumlah pass, jenis pengocokan, dan regularisasi. Anda dapat mengatur parameter ini saat Anda menggunakan wizard Buat Model L untuk membuat model ML-nya dan memilihKhususPilihan.
Evaluasi performa model	Gunakan wizard Create Evaluation untuk menilai kinerja prediktif model Anda.
Pilihan Fitur	Algoritma pembelajaran Amazon XML dapat menurunkan fitur yang tidak banyak berkontribusi pada proses pembelajaran. Untuk menunjukkan bahwa Anda ingin menjatuhkan fitur-fitur tersebut, pilih`L1 regulariz ation` parameter saat Anda membuat model ML-nya.

Proses MLS	Tugas Amazon
Tetapkan ambang skor untuk akurasi prediksi	Tinjau kinerja prediktif model dalam laporan evaluasi pada ambang skor yang berbeda, dan kemudian atur ambang skor berdasarkan aplikasi bisnis Anda. Ambang batas skor menentukan bagaimana model mendefinisikan kecocokan prediksi. Sesuaikan nomor untuk mengontrol positif palsu dan negatif palsu.
Gunakan model	Gunakan model Anda untuk mendapatkan prediksi untuk batch pengamatan dengan menggunakan wizard Create Batch Prediction. Atau, dapatkan prediksi untuk pengamatan individu sesuai permintaa n dengan memungkinkan model ML-memproses prediksi real-time menggunakan`Predict`API

Menyiapkan Amazon Machine Learning

Anda perlu akun AWS sebelum Anda dapat menggunakan Amazon Machine Learning untuk pertama kalinya. Jika Anda belum memiliki akun, lihat Daftar AWS.

Mendaftar ke AWS

Saat Anda mendaftar ke Amazon Web Services (AWS), akun AWS Anda secara otomatis terdaftar untuk semua layanan di AWS, termasuk Amazon ML. Anda hanya dikenakan biaya untuk layanan yang Anda gunakan. Jika Anda sudah memiliki akun AWS, lewati langkah ini. Jika Anda tidak memiliki akun AWS, gunakan prosedur berikut untuk membuatnya.

Untuk mendaftar akun AWS

1. Pergi kehttp://aws.amazon.comdan pilihDaftar.
2. Ikuti petunjuk di layar.

Bagian dari prosedur pendaftaran melibatkan menerima panggilan telepon dan memasukkan PIN menggunakan keypad telepon.

Tutorial: Menggunakan Amazon XML untuk Memprediksi Tanggapan terhadap Penawaran Pemasaran

Dengan Amazon Machine Learning (Amazon ML), Anda dapat membangun dan melatih model prediktif dan meng-host aplikasi Anda dalam solusi cloud yang dapat diskalakan. Dalam tutorial ini, kami menunjukkan kepada Anda bagaimana menggunakan konsol Amazon MLuntuk membuat sumber data, membangun model machine learning (L), dan menggunakan model untuk menghasilkan prediksi yang dapat Anda gunakan dalam aplikasi Anda.

Latihan sampel kami menunjukkan cara mengidentifikasi calon pelanggan untuk kampanye pemasaran yang ditargetkan, tetapi Anda dapat menerapkan prinsip yang sama untuk membuat dan menggunakan berbagai model ML-nya. Untuk menyelesaikan latihan sampel, Anda akan menggunakan dataset perbankan dan pemasaran yang tersedia untuk umum dariUniversity of California di Irvine (UCI) Repositori Machine Learning. Dataset ini berisi informasi umum tentang pelanggan, dan informasi tentang bagaimana mereka menanggapi kontak pemasaran sebelumnya. Anda akan menggunakan data ini untuk mengidentifikasi pelanggan mana yang paling mungkin untuk berlangganan produk baru Anda, deposit jangka bank, juga dikenal sebagai sertifikat deposit (CD).

> **Warning**
>
> Tutorial ini tidak disertakan dalam tingkat gratis AWS. Untuk informasi selengkapnya tentang harga Amazon, lihatAmazon Machine Learning.

Prasyarat

Untuk melakukan tutorial, Anda harus memiliki akun AWS. Jika Anda belum memiliki akun AWS, lihatMenyiapkan Amazon Machine Learning.

Langkah-langkah

- Langkah 1: Mempersiapkan Data Anda
- Langkah 2: Membuat Pelatihan Datasource
- Langkah 3: Membuat Model ML-nya
- Langkah 4: Tinjau Kinerja Prediktif Model L dan Tetapkan Ambang Skor

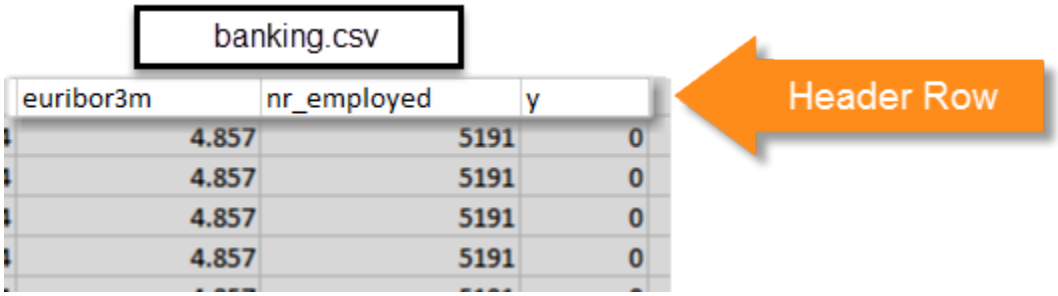
Langkah 1: Mempersiapkan Data Anda

Dalam pembelajaran mesin, Anda biasanya mendapatkan data dan memastikan bahwa itu diformat dengan baik sebelum memulai proses pelatihan. Untuk tujuan tutorial ini, kami memperoleh dataset sampel dariRepositori Machine Learning, diformat agar sesuai dengan pedoman Amazon IL, dan membuatnya tersedia untuk Anda unduh. Unduh dataset dari lokasi penyimpanan Amazon Simple Storage Service (Amazon S3) kami dan unggah ke bucket S3 Anda sendiri dengan mengikuti prosedur dalam topik ini.

Untuk persyaratan pemformatan Amazon IL, lihatMemahami Format Data untuk Amazon.

Untuk mengunduh dataset

1. Download file yang berisi data historis bagi nasabah yang telah membeli produk serupa dengan deposit jangka bank Anda dengan mengklikbanking.zip. Unzip folder dan simpan file banking.csv ke komputer Anda.
2. Unduh file yang akan Anda gunakan untuk memprediksi apakah calon pelanggan akan menanggapi penawaran Anda dengan mengklikbanking-batch.zip. Unzip folder dan simpan file banking-batch.csv ke komputer Anda.
3. Buka `banking.csv`. Anda akan melihat baris dan kolom data. Parameterbaris sundulanberisi nama atribut untuk setiap kolom. Sesitambahanadalah unik, bernama properti yang menggambarkan karakteristik tertentu dari setiap pelanggan; misalnya, nr_employed menunjukkan status kerja pelanggan. Setiap baris mewakili koleksi pengamatan tentang satu pelanggan.

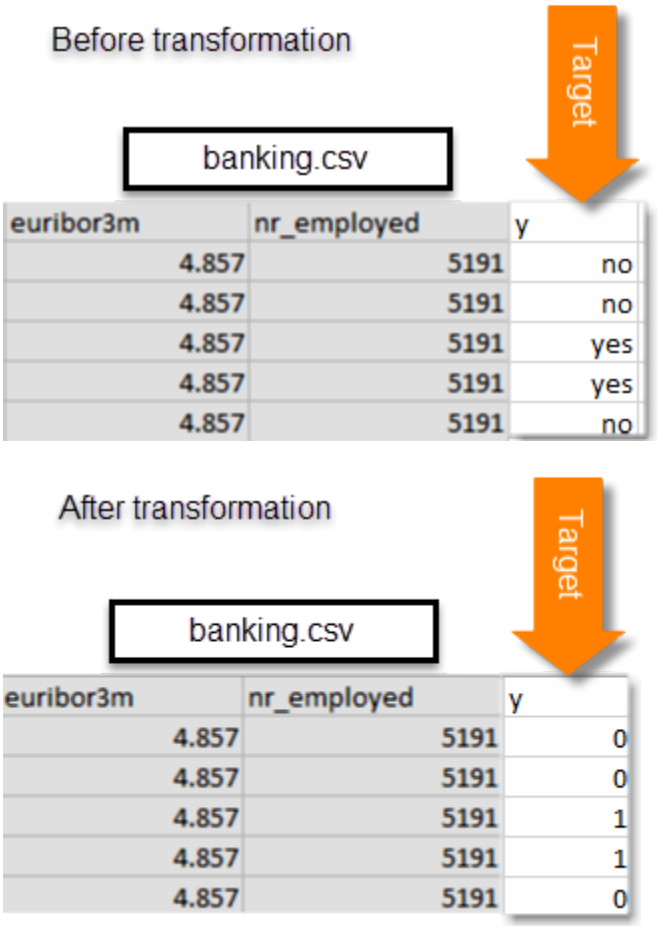


Anda ingin model ML-mu menjawab pertanyaan “Akankah pelanggan ini berlangganan produk baru saya?”. Dibanking.csvdataset, jawaban atas pertanyaan ini adalah atributy, yang berisi

nilai 1 (untuk ya) atau 0 (untuk tidak). Atribut yang Anda inginkan Amazon ML-belajar bagaimana memprediksi dikenal sebagaiatribut target.

> **Note**
>
> Atributyadalah atribut biner. Ini hanya dapat berisi satu dari dua nilai, dalam hal ini 0 atau 1. Dalam dataset UCI asli,yatribut adalah baik Ya atau Tidak. Kami telah mengedit dataset asli untuk Anda. Semua nilai atributyitu berarti ya sekarang 1, dan semua nilai yang berarti tidak ada sekarang 0. Jika Anda menggunakan data Anda sendiri, Anda dapat menggunakan nilai lain untuk atribut biner. Untuk informasi selengkapnya tentang nilai yang valid, lihatMenggunakan Field AttributeType.

Contoh berikut menunjukkan data sebelum dan sesudah kita mengubah nilai-nilai dalam atributyke atribut biner 0 dan 1.



Parameter`banking-batch.csv`berkas tidak berisiyatribut. Setelah Anda membuat model ML-nya, Anda akan menggunakan model untuk memprediksiyuntuk setiap catatan dalam file itu.

Selanjutnya, unggah`banking.csv` dan`banking-batch.csv`file ke Amazon S3.

Mengunggah file ke lokasi Amazon S3

1. Masuk ke AWS Management Console dan buka konsol Amazon S3 di https://console.aws.amazon.com/s3/.
2. DiSemua Bucketdaftar, membuat bucket atau memilih lokasi di mana Anda ingin mengunggah file.
3. Di bilah navigasi, pilihUnggah.
4. PilihTambahkan File.
5. Di kotak dialog, arahkan ke desktop, pilih`banking.csv`dan`banking-batch.csv`, dan kemudian pilihBuka.

Sekarang, Anda siap untukbuat sumber data pelatihan.

Langkah 2: Membuat Pelatihan Datasource

Setelah Anda meng-upload`banking.csv`lokasi Amazon Simple Storage Service (Amazon S3), Anda menggunakannya untuk membuat sumber data pelatihan. Sumber data adalah objek Amazon Machine Learning (Amazon ML) yang berisi lokasi data input dan metadata penting tentang data input Anda. Amazon IL menggunakan sumber data untuk operasi seperti pelatihan dan evaluasi model ML-nya.

Untuk membuat sumber data, berikan yang berikut ini:

- Lokasi Amazon S3 dari data Anda untuk mengakses data
- Skema, yang mencakup nama-nama atribut dalam data dan jenis masing-masing atribut (Numeric, Text, Categical, or Binary)
- Nama atribut yang berisi jawaban yang Anda inginkan Amazon MLnya untuk memprediksi, atribut target

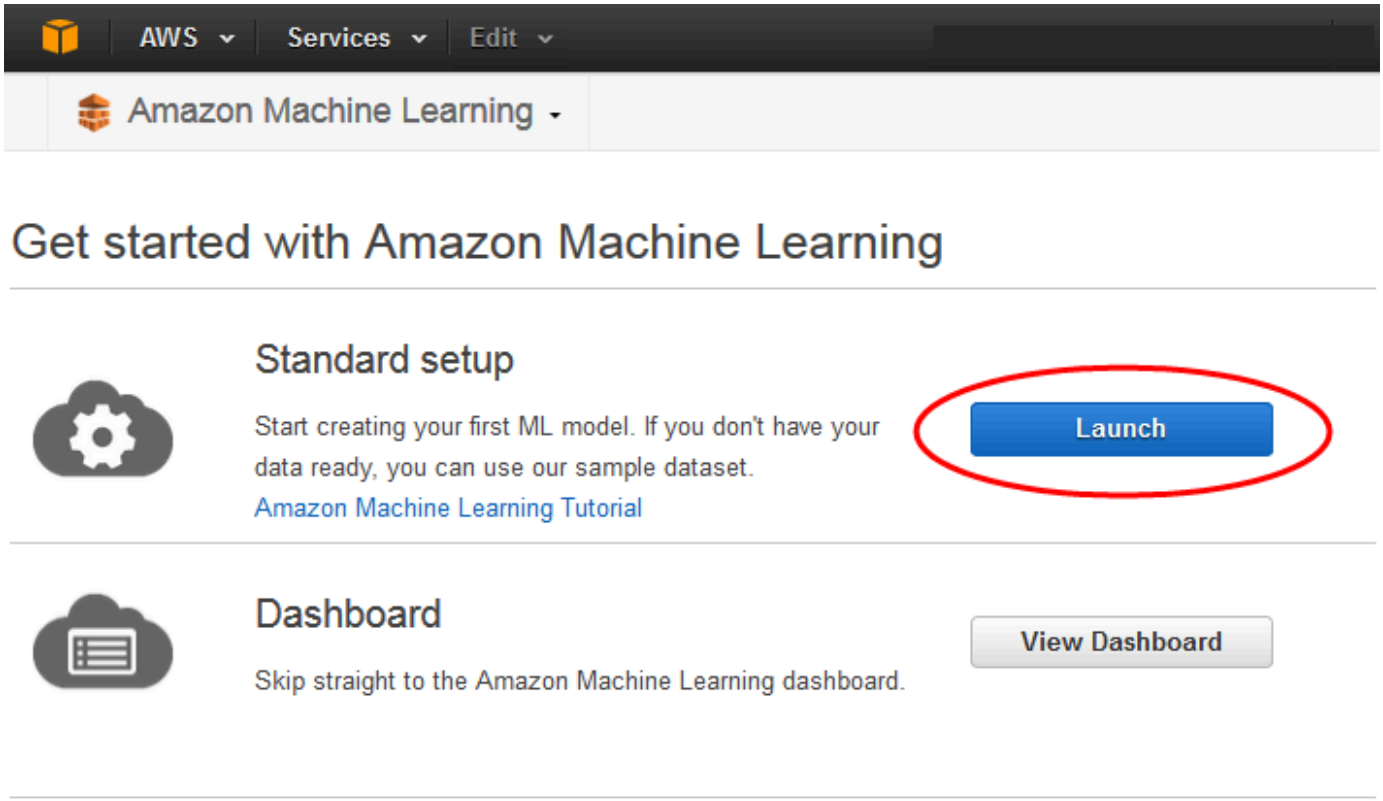
> **Note**
>
> Sumber data tidak benar-benar menyimpan data Anda, itu hanya mereferensikannya. Hindari memindahkan atau mengubah file yang tersimpan di Amazon S3. Jika Anda memindahkan atau mengubahnya, Amazon XML tidak dapat mengaksesnya untuk membuat model ML-nya, menghasilkan evaluasi, atau menghasilkan prediksi.

Untuk membuat sumber data pelatihan

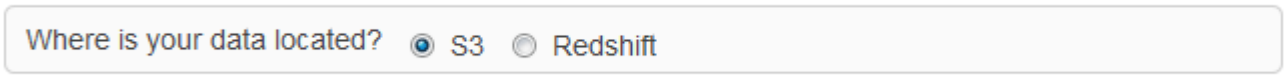
1. Buka konsol Amazon Machine Learning dihttps://console.aws.amazon.com/machinelearning/.
2. Pilih Mulai.

   > **Note**
   >
   > Tutorial ini mengasumsikan bahwa ini adalah pertama kalinya Anda menggunakan Amazon ML-nya. Jika Anda telah menggunakan Amazon ML-nya sebelumnya, Anda dapat menggunakanBuat...daftar drop down di dasbor Amazon MLuntuk membuat datasource baru.

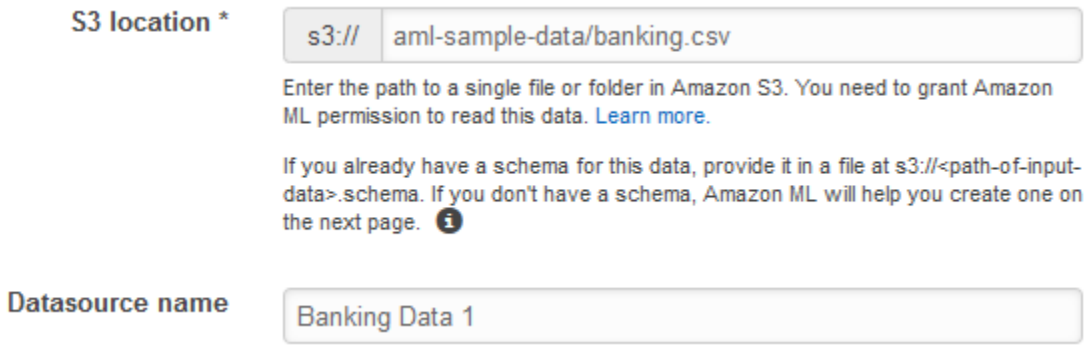
3. PadaMemulai dengan Amazon Machine Learninghalaman, pilihLuncurkan.



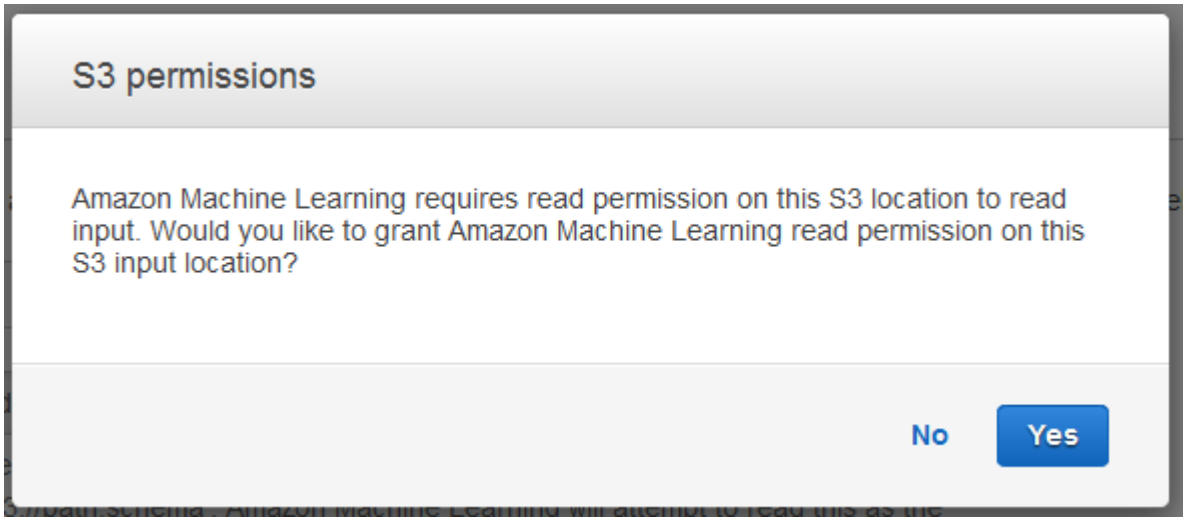
4. PadaData Masukanhalaman, untukDimana data Anda berada? , pastikan bahwaS3dipilih.



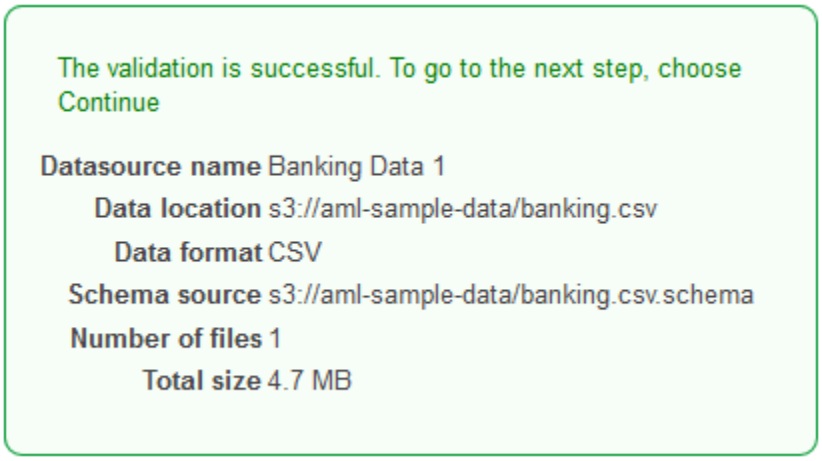
5. UntukLokasi S3, ketik lokasi lengkap`banking.csv` File dari Langkah 1: Siapkan Data Anda. Misalnya:*`bucket Anda`***`/banking.csv`**. Amazon IL menambahkan s3://ke nama bucket Anda untuk Anda.
6. UntukNama sumber data, jenis**`Banking Data 1`**.



7. MemiilihVerifikasi.
8. DiIzin S3kotak dialog, pilihya.



9. Jika Amazon L dapat mengakses dan membaca file data di lokasi S3, Anda akan melihat halaman yang mirip dengan berikut ini. Tinjau properti, lalu pilihLanjutkan.



Selanjutnya, Anda membuat skema. SEBUAHskemaadalah informasi yang diperlukan Amazon ML-nya untuk menafsirkan data input untuk model MLnya, termasuk nama atribut dan jenis data yang ditugaskannya, dan nama-nama atribut khusus. Ada dua cara untuk menyediakan Amazon MLnya dengan skema:

- Berikan file skema terpisah saat Anda mengunggah data Amazon S3 Anda.
- Izinkan Amazon ML-menyimpulkan jenis atribut dan membuat skema untuk Anda.

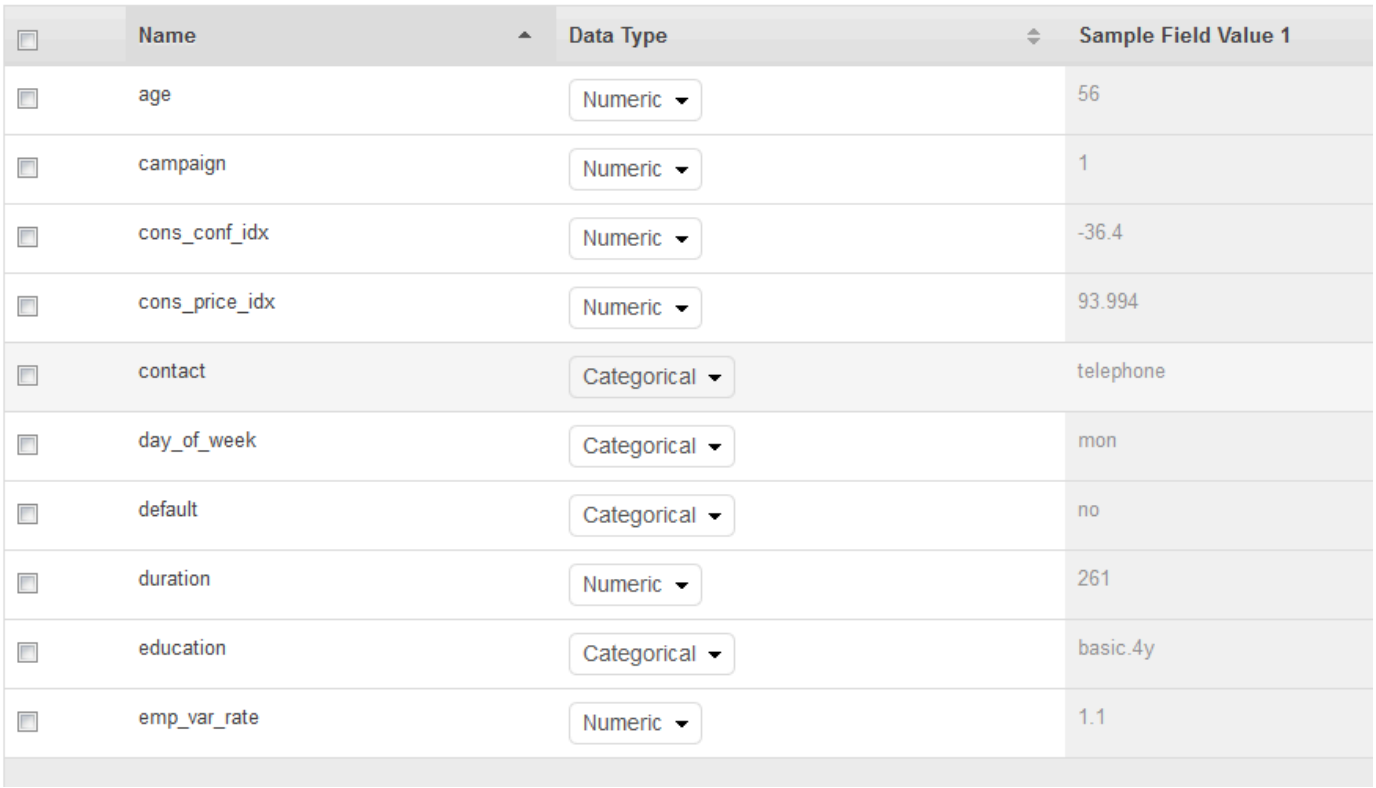
Dalam tutorial ini, kita akan meminta Amazon ML-nya untuk menyimpulkan skema.

Untuk informasi selengkapnya tentang membuat file skema terpisah, lihatMembuat Skema Data untuk Amazon.

Untuk memungkinkan Amazon ML-menyimpulkan skema

1. PadaSkemahalaman, Amazon IL menunjukkan skema yang disimpulkan. Tinjau jenis data yang disimpulkan Amazon ML-nya untuk atribut. Adalah penting bahwa atribut ditetapkan jenis data yang benar untuk membantu Amazon ML-menelan data dengan benar dan untuk mengaktifkan pemrosesan fitur yang benar pada atribut.

   - Atribut yang hanya memiliki dua negara yang mungkin, seperti ya atau tidak, harus ditandai sebagaiBiner.
   - Atribut yang merupakan angka atau string yang digunakan untuk menunjukkan kategori harus ditandai sebagaiKategoris.
   - Atribut yang jumlah numerik yang urutannya berarti harus ditandai sebagaiNumerik.
   - Atribut yang string yang ingin Anda perlakukan sebagai kata-kata yang dibatasi oleh spasi harus ditandai sebagaiText.



Name	Data Type	Sample Field Value 1
age	Numeric	56
campaign	Numeric	1
cons_conf_idx	Numeric	-36.4
cons_price_idx	Numeric	93.994
contact	Categorical	telephone
day_of_week	Categorical	mon
default	Categorical	no
duration	Numeric	261
education	Categorical	basic.4y
emp_var_rate	Numeric	1.1

2. Dalam tutorial ini, Amazon L telah mengidentifikasi jenis data dengan benar untuk semua atribut, jadi pilihlahLanjutkan.

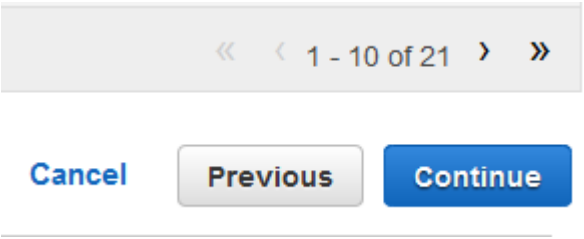
Selanjutnya, pilih atribut target.

Ingat bahwa target adalah atribut yang harus dipelajari oleh model ML-nya. Atributymenunjukkan apakah seseorang telah berlangganan kampanye di masa lalu: 1 (ya) atau 0 (tidak).

> **Note**
>
> Pilih atribut target hanya jika Anda akan menggunakan sumber data untuk pelatihan dan mengevaluasi model ML-nya.

Untuk memilih y sebagai atribut target

1. Di kanan bawah tabel, pilih panah tunggal untuk maju ke halaman terakhir dari tabel, di mana atribut bernamaymuncul.



2. DiTargetkolom, pilihy.

   Amazon MLnya mengkonfirmasikanydipilih sebagai target Anda.

3. Pilih Continue (Lanjutkan).

4. PadaID barishalaman, untukApakah data Anda berisi pengenal?, pastikan bahwaTidak, default, dipilih.
5. MemiilihTinjau, dan kemudian pilihLanjutkan.

Setelah Anda memiliki sumber data pelatihan, Anda siapmembuat model Anda.

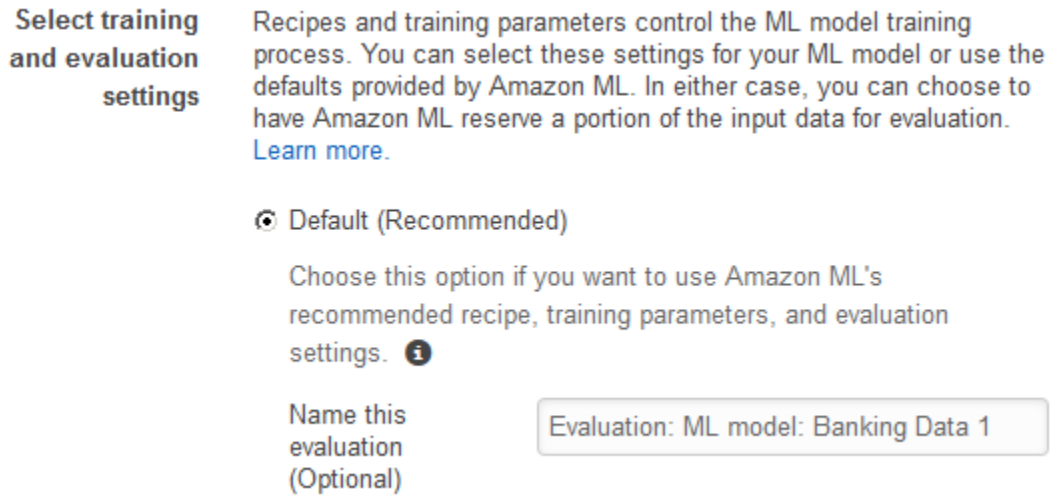
Langkah 3: Membuat Model ML-nya

Setelah membuat sumber data pelatihan, Anda menggunakannya untuk membuat model ML-nya, melatih model, dan kemudian mengevaluasi hasilnya. Model MLnya adalah kumpulan pola yang ditemukan Amazon ML-nya dalam data Anda selama latihan. Anda menggunakan model untuk membuat prediksi.

Untuk membuat model ML-nya

1. Karena wizard Memulai membuat sumber data pelatihan dan model, Amazon Machine Learning (Amazon ML) secara otomatis menggunakan sumber data pelatihan yang baru saja Anda buat, dan membawa Anda langsung kePengaturan model Lhalaman. PadaPengaturan model Llaman, untukNama model L, pastikan bahwa default,**`ML model: Banking Data 1`**, ditampilkan.

   Menggunakan nama ramah, seperti default, membantu Anda dengan mudah mengidentifikasi dan mengelola model ML-nya.
2. UntukPengaturan pelatihan dan evaluasi, memastikan bahwaDefaultdipilih.



3. UntukBeri nama evaluasi ini, terima default,**`Evaluation: ML model: Banking Data 1`**.
4. PilihTinjauan, tinjau pengaturan Anda, dan kemudian pilihSelesai.

Setelah Anda memilihSelesai, Amazon ML-menambahkan model Anda ke antrian pemrosesan. Saat Amazon IL membuat model Anda, model tersebut akan menerapkan default dan melakukan tindakan berikut:

- Membagipelatihan datasource menjadi dua bagian, satu berisi 70% dari data dan satu berisi sisa 30%
- Melatih model ML pada bagian yang berisi 70% dari data input
- Mengevaluasi model menggunakan sisa 30% dari data input

Saat model Anda berada dalam antrian, Amazon ML-melaporkan statusnya sebagaiTertunda. Sementara Amazon ML-membuat model Anda, itu melaporkan status sebagaiDalam Progres. Ketika telah menyelesaikan semua tindakan, ia melaporkan status sebagaiCompleted (Lengkap). Tunggu evaluasi selesai sebelum melanjutkan.

Sekarang Anda siap untukmeninjau kinerja model Anda dan menetapkan skor cut-off.

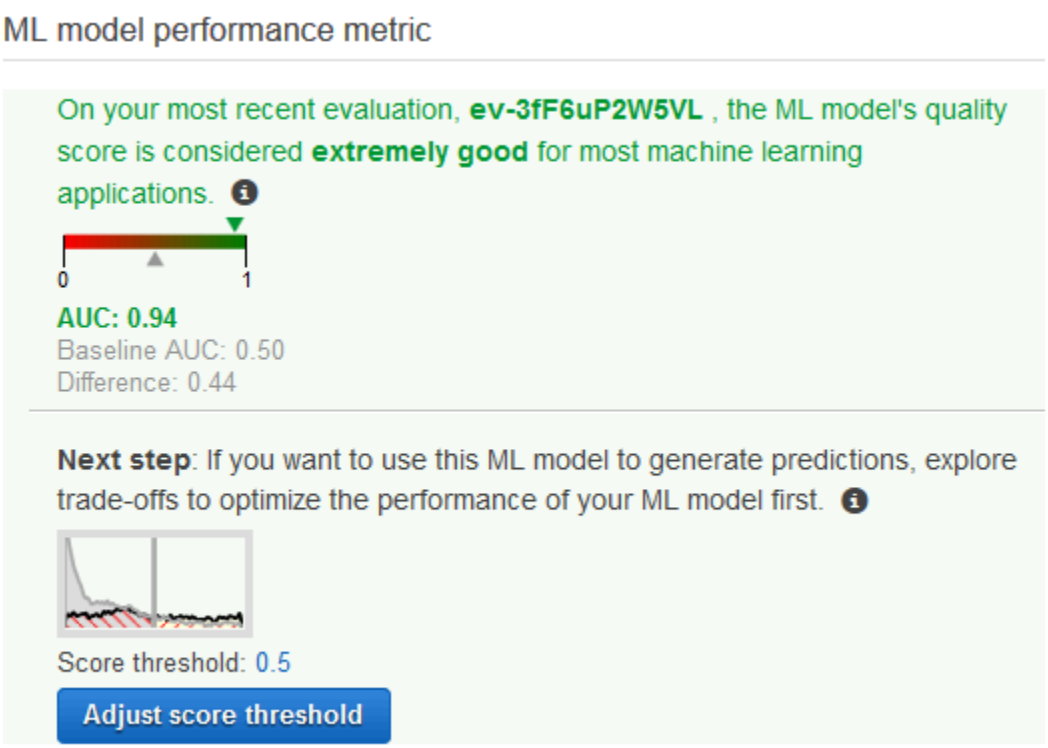
Untuk informasi selengkapnya tentang model pelatihan dan evaluasi, lihatModel ML-Pelatihandanevaluate an ML model.

Langkah 4: Tinjau Kinerja Prediktif Model L dan Tetapkan Ambang Skor

Sekarang setelah Anda membuat model ML-mu dan Amazon Machine Learning (Amazon ML) telah mengevaluasinya, mari kita lihat apakah itu cukup baik untuk digunakan. Selama evaluasi, Amazon ML-menghitung metrik kualitas standar industri, yang disebut metrik Area Under a Curve (AUC), yang mengekspresikan kualitas kinerja model ML-mu. Amazon XML juga menafsirkan metrik AUC untuk memberi tahu Anda apakah kualitas model MLnya memadai untuk sebagian besar aplikasi machine learning. (Pelajari selengkapnya tentang AUCMengukur Akurasi Model.) Mari kita tinjau metrik AUC, dan kemudian sesuaikan ambang skor atau cut-off untuk mengoptimalkan kinerja prediktif model Anda.

Untuk meninjau metrik AUC untuk model ML-mu

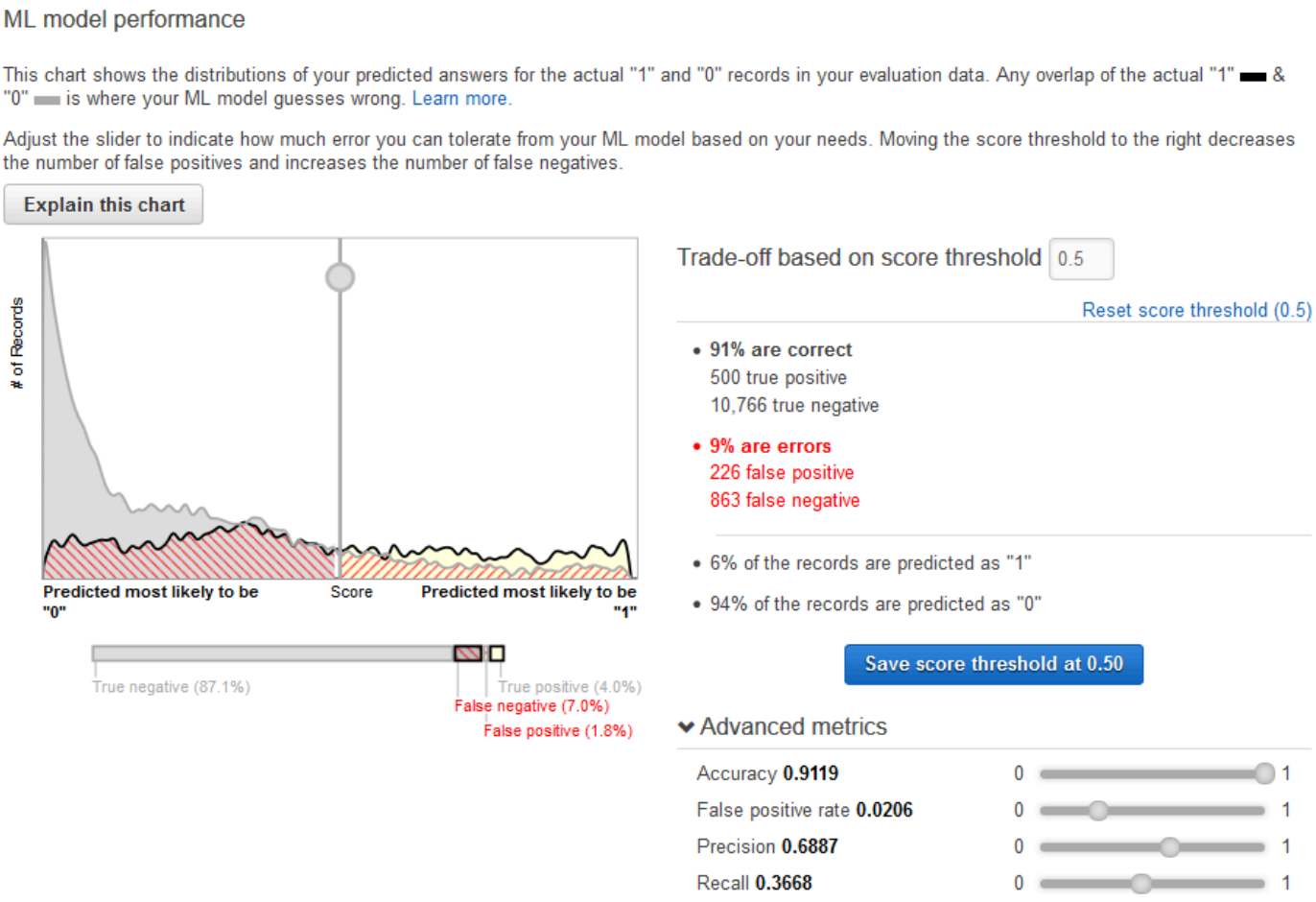
1. PadaRingkasan model Lhalaman, diLaporan model Lpanel navigasi, pilihEvaluasi, pilihEvaluasi: Model L: Model perbankan 1, dan kemudian pilihRingkasan.
2. PadaRingkasan evaluasihalaman, tinjau ringkasan evaluasi, termasuk metrik kinerja AUC model.



Model L menghasilkan skor prediksi numerik untuk setiap catatan dalam sumber data prediksi, dan kemudian menerapkan ambang batas untuk mengubah skor ini menjadi label biner 0 (untuk tidak) atau 1 (untuk ya). Dengan mengubahAmbang batas nilai, Anda dapat menyesuaikan bagaimana model ML-memberikan label ini. Sekarang, atur batas skor.

Untuk menetapkan ambang skor untuk model ML-mu

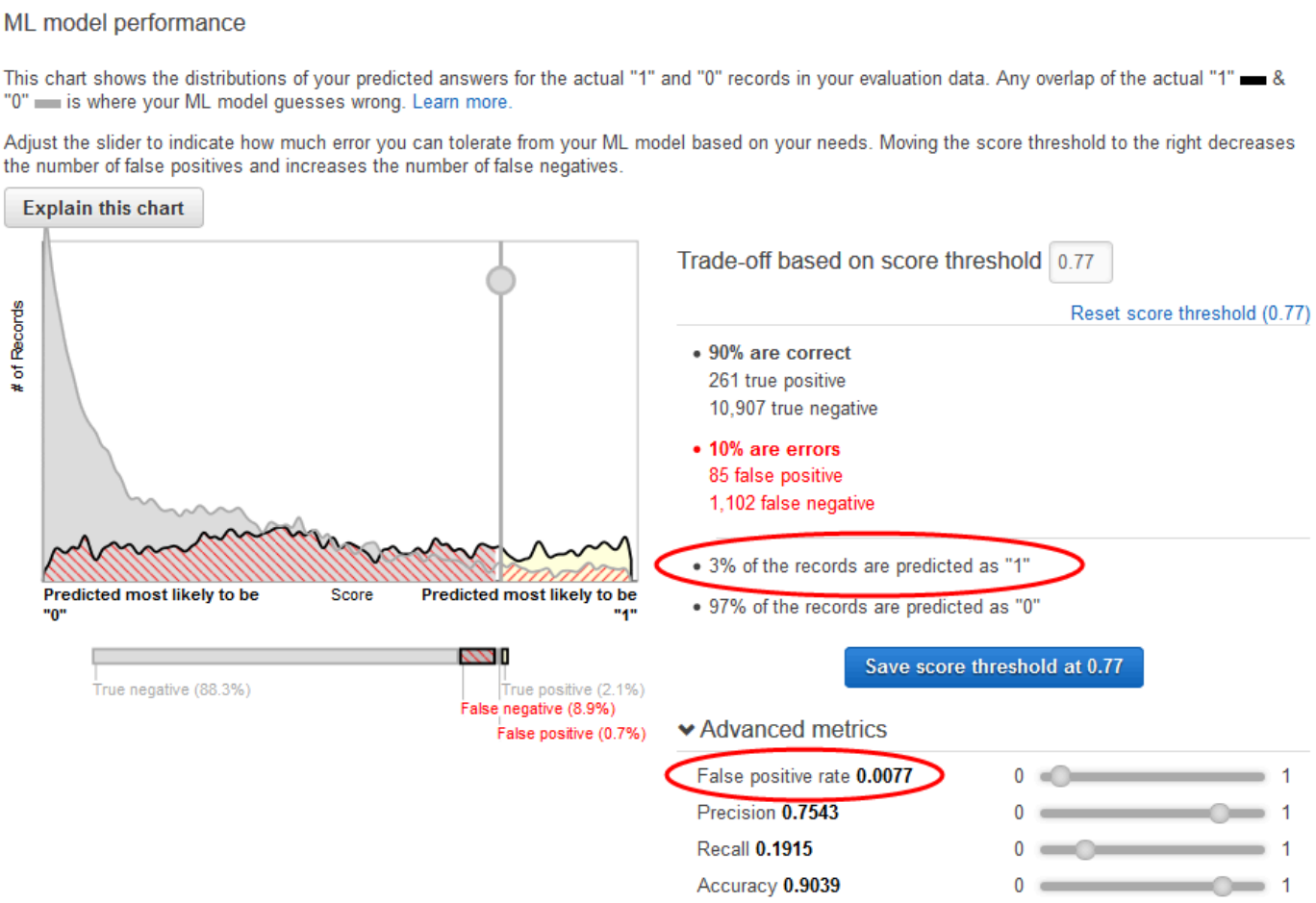
1. PadaRingkasan evaluasihalaman, pilihSesuaikan Threshold Skor.



Anda dapat menyempurnakan metrik kinerja model ML-mu dengan menyesuaikan ambang batas skor. Menyesuaikan nilai ini mengubah tingkat kepercayaan bahwa model harus memiliki dalam prediksi sebelum menganggap prediksi menjadi positif. Hal ini juga mengubah berapa banyak negatif palsu dan positif palsu Anda bersedia untuk mentolerir dalam prediksi Anda.

Anda dapat mengontrol cutoff untuk apa model menganggap prediksi positif dengan meningkatkan ambang skor sampai menganggap hanya prediksi dengan kemungkinan tertinggi menjadi positif sejati. Anda juga dapat mengurangi ambang skor sampai Anda tidak lagi memiliki negatif palsu. Pilih cutoff Anda untuk mencerminkan kebutuhan bisnis Anda. Untuk tutorial ini, setiap biaya positif palsu kampanye uang, jadi kami ingin rasio positif sejati yang tinggi terhadap positif palsu.

2. Katakanlah Anda ingin menargetkan 3% teratas dari pelanggan yang akan berlangganan produk. Geser pemilih vertikal untuk mengatur ambang skor ke nilai yang sesuai dengan3% dari catatan diprediksi sebagai “1".



Perhatikan dampak ambang skor ini pada kinerja model ML: tingkat positif palsu adalah 0,007. Mari kita asumsikan bahwa tingkat positif palsu dapat diterima.

3. PilihSimpan ambang batas skor di 0,77.

Setiap kali Anda menggunakan model L ini untuk membuat prediksi, itu akan memprediksi catatan dengan skor lebih dari 0,77 sebagai “1", dan sisa catatan sebagai “0".

Untuk mempelajari selengkapnya tentang ambang batas skor, lihatKlasifikasi Biner.

Sekarang Anda siap untukmembuat prediksi menggunakan model Anda.

Langkah 5: Gunakan Model ML untuk Menghasilkan Prediksi

Amazon Machine Learning (Amazon IL) dapat menghasilkan dua jenis prediksi — batch dan real-time.

SEBUAHPrediksi real-timeadalah prediksi untuk observasi tunggal yang dihasilkan Amazon ML-sesuai permintaan. Prediksi real-time sangat ideal untuk aplikasi seluler, situs web, dan aplikasi lain yang perlu menggunakan hasil secara interaktif.

SEBUAHprediksi batchadalah seperangkat prediksi untuk sekelompok pengamatan. Amazon ML-memproses catatan dalam prediksi batch bersama-sama, sehingga pemrosesan dapat memakan waktu lama. Gunakan prediksi batch untuk aplikasi yang memerlukan prediksi untuk serangkaian pengamatan atau prediksi yang tidak menggunakan hasil secara interaktif.

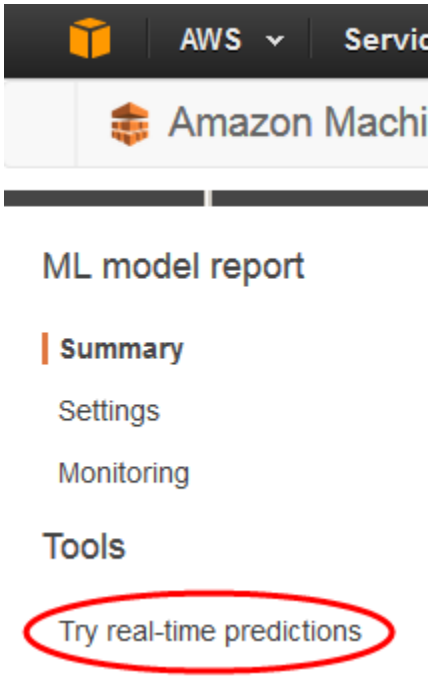
Untuk tutorial ini, Anda akan menghasilkan prediksi real-time yang memprediksi apakah satu calon pelanggan akan berlangganan produk baru. Anda juga akan menghasilkan prediksi untuk sejumlah besar pelanggan potensial. Untuk prediksi batch, Anda akan menggunakan`banking-batch.csv`file yang Anda uploadLangkah 1: Mempersiapkan Data Anda.

Mari kita mulai dengan prediksi real-time.

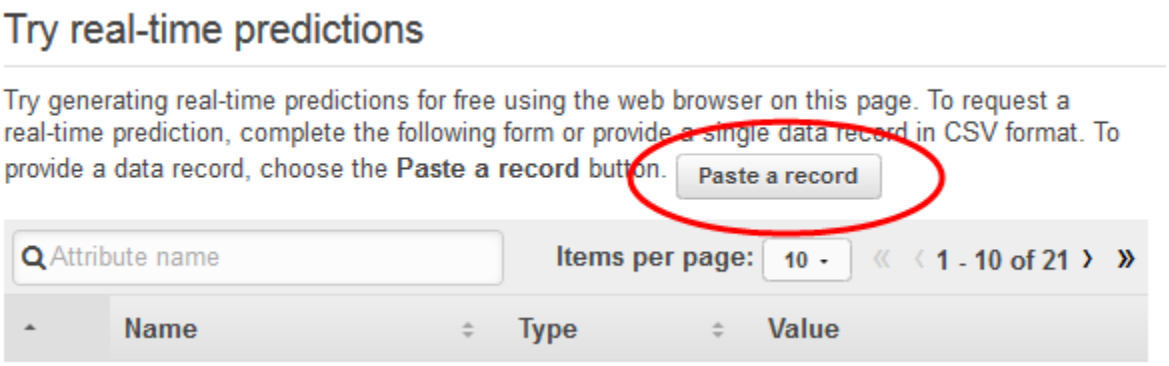
> **Note**
>
> Untuk aplikasi yang memerlukan prediksi real-time, Anda harus membuat titik akhir real-time untuk model ML-nya. Anda akan dikenakan biaya sementara titik akhir real-time tersedia. Sebelum Anda berkomitmen untuk menggunakan prediksi real-time dan mulai menimbulkan biaya yang terkait dengannya, Anda dapat mencoba menggunakan fitur prediksi real-time di browser web Anda, tanpa membuat titik akhir real-time. Itulah yang akan kita lakukan untuk tutorial ini.

Untuk mencoba prediksi waktu nyata

1. DiLaporan modelpanel navigasi, pilihCoba prediksi waktu nyata.



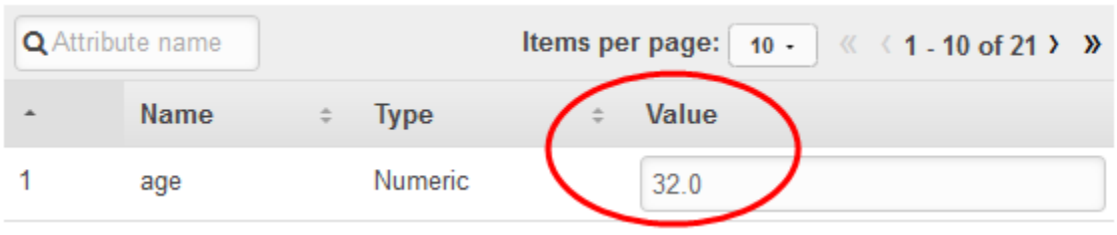
2. PilihTempel catatan.



3. DiTempel catatankotak dialog, tempel pengamatan berikut ini:

```
32,services,divorced,basic.9y,no,unknown,yes,cellular,dec,mon,110,1,11,0,nonexistent,-1.8,9
```

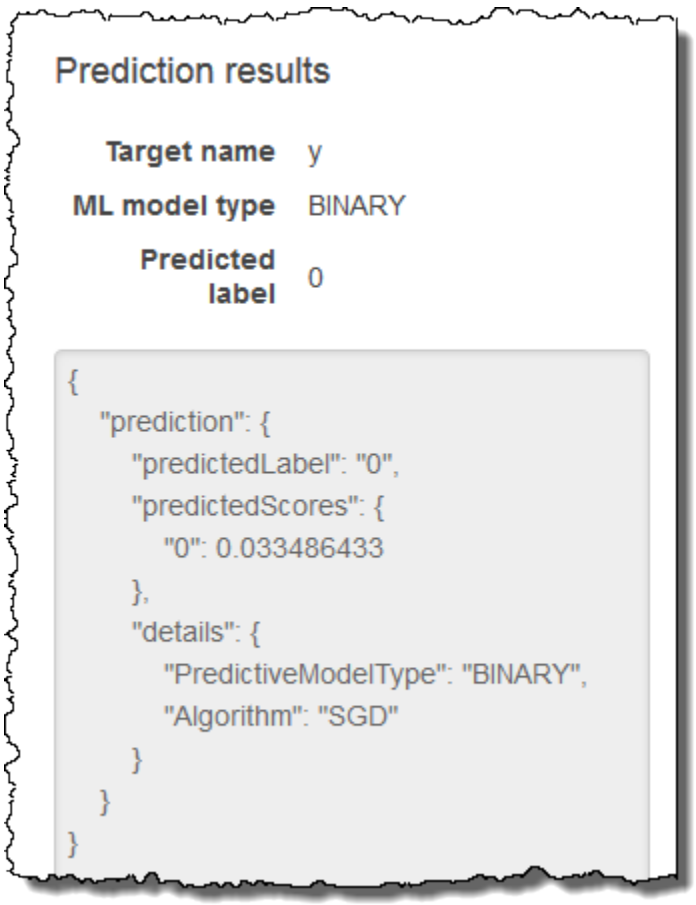
4. DiTempel catatankotak dialog, pilihKirimuntuk mengonfirmasi bahwa Anda ingin menghasilkan prediksi untuk pengamatan ini. Amazon ML-mengisi nilai dalam bentuk prediksi real-time.



> **Note**
>
> Anda juga dapat mengisiNilaibidang dengan mengetikkan nilai-nilai individu. Terlepas dari metode yang Anda pilih, Anda harus memberikan pengamatan yang tidak digunakan untuk melatih model.

5. Di bagian bawah halaman, pilihPrediksi.

   Prediksi muncul diHasil prediksipanel di sebelah kanan. Prediksi ini memilikiLabel yang diprediksidari0, yang berarti bahwa calon pelanggan ini tidak mungkin untuk menanggapi kampanye. SEBUAHLabel yang diprediksidari1berarti bahwa pelanggan cenderung menanggapi kampanye.

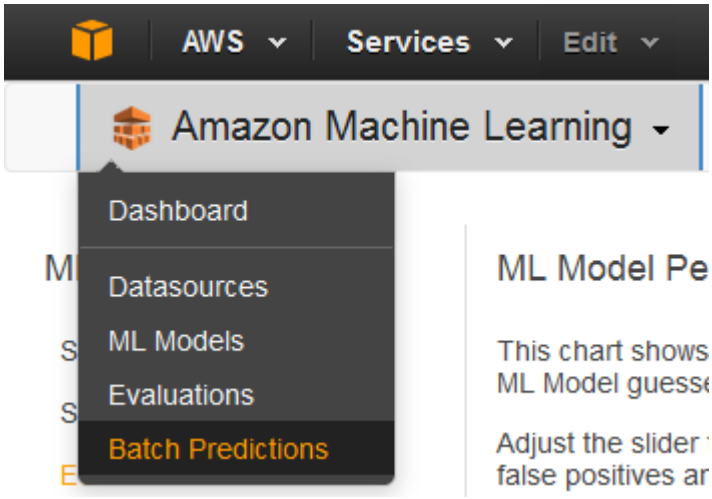


Sekarang, buat prediksi batch. Anda akan memberikan nama model ML-Amazon yang Anda gunakan; lokasi Amazon Simple Storage Service (Amazon S3) dari data input yang ingin Anda

hasilkan prediksi (Amazon IL akan membuat sumber data prediksi batch dari data ini); dan lokasi Amazon S3 untuk menyimpan hasilnya.

Untuk membuat prediksi batch

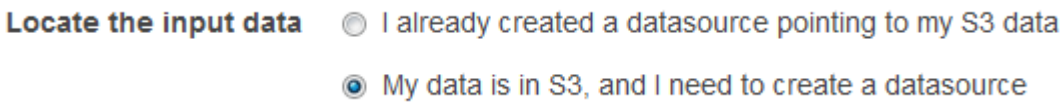
1. PilihAmazon Machine Learning, dan kemudian pilihBatchPrediksi.



2. PilihPrediksi batch baru.
3. PadaModel L untuk prediksi batchhalaman, pilihModel L: Data Perbankan 1.

   Amazon IL menampilkan nama model, ID, waktu pembuatan, dan ID sumber data terkait.
4. Pilih Continue (Lanjutkan).
5. Untuk menghasilkan prediksi, Anda harus memberikan Amazon MLdata yang Anda butuhkan prediksi. Ini disebutdata input. Pertama, masukkan data input ke dalam sumber data sehingga Amazon IL dapat mengaksesnya.

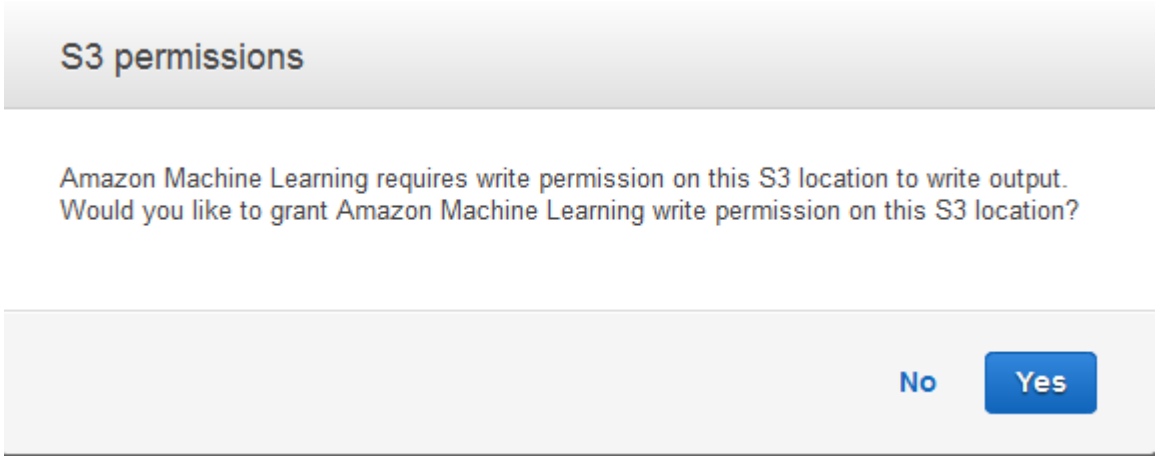
   UntukMenemukan data input, pilihData saya ada di S3, dan saya perlu membuat sumber data.



6. UntukNama sumber data, tipe**`Banking Data 2`**.
7. UntukLokasi S3, ketik lokasi lengkap`banking-batch.csv`berkas:*`bucket Anda`***`/banking-batch.csv`**.
8. UntukApakah baris pertama di CSV Anda berisi nama kolom?, pilihya.
9. PilihVerifikasi.

Amazon ML-memvalidasi lokasi data Anda.

10. Pilih Continue (Lanjutkan).
11. UntukTujuan S3, ketik nama lokasi Amazon S3 tempat Anda mengunggah file di Langkah 1: Siapkan Data Anda. Amazon ML-upload hasil prediksi di sana.
12. UntukNama prediksi Batch, terima default,**`Batch prediction: ML model: Banking Data 1`**. Amazon IL memilih nama default berdasarkan model yang akan digunakan untuk membuat prediksi. Dalam tutorial ini, model dan prediksi dinamai berdasarkan sumber data pelatihan,`Banking Data 1`.
13. Pilih Tinjau.
14. DiIzin S3kotak dialog, pilihya.

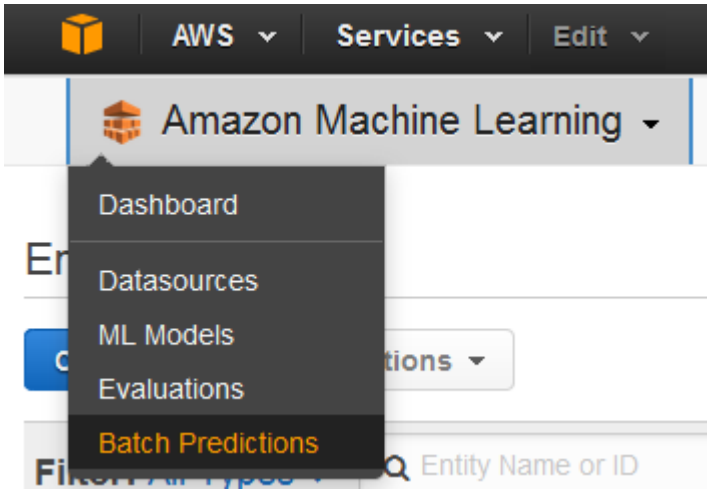


15. PadaTinjauhalaman, pilihSelesai.

    Permintaan prediksi batch dikirim ke Amazon IL dan dimasukkan ke dalam antrian. Waktu yang dibutuhkan Amazon untuk memproses prediksi batch tergantung pada ukuran sumber data Anda dan kompleksitas model ML-mu. Sementara Amazon ML-memproses permintaan tersebut, aplikasi ini melaporkan statusDalam Progres. Setelah prediksi batch selesai, status permintaan berubah menjadiCompleted (Lengkap). Sekarang Anda dapat melihat hasilnya.

Untuk melihat prediksi

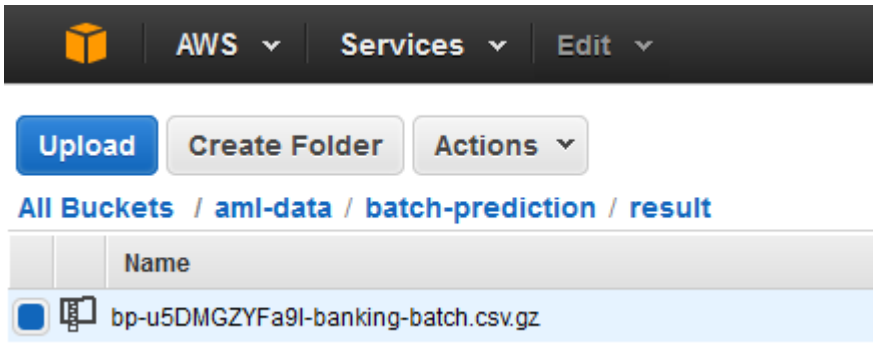
1. PilihAmazon Machine Learning, dan kemudian pilihPrediksi Batch.



2. Dalam daftar prediksi, pilihprediksi Batch: Model L: Data Perbankan 1. ParameterInfo prediksi BatchHalaman akan muncul.

Name	Subscription propensity Predictions
ID	bp-u5DMGZYFa9l
Creation Time	Mar 5, 2015 3:28:33 PM
Status	Completed
Log	Download Log
Datasource ID	ds-33Rqgz9w3ee
ML Model ID	ml-u7ljoShX2kX
Input S3 URL	s3://aml-data/banking-batch.csv
Output S3 URL	s3://aml-data/

3. Untuk melihat hasil prediksi batch, buka Konsol Amazon S3 dihttps://console.aws.amazon.com/s3/dan navigasikan ke lokasi Amazon S3 yang direferensikan dalamURL Outputbidang. Dari sana, arahkan ke folder hasil, yang akan memiliki nama yang mirip dengan`s3://aml-data/batch-prediction/result`.



Prediksi disimpan dalam file.gzip terkompresi dengan ekstensi.gz.

4. Download file prediksi ke desktop Anda, uncompress, dan membukanya.

bestAnswer	score
0	0.06046
0	0.00507
0	0.01410
0	0.00170
0	0.00184
0	0.07133
0	0.30811

File ini memiliki dua kolom,Jawaban terbaikdanSkor, dan baris untuk setiap pengamatan di datasource Anda. Hasil dalamJawaban terbaikkolom didasarkan pada ambang skor 0.77 yang Anda tetapkanLangkah 4: Tinjau Kinerja Prediktif Model L dan Tetapkan Ambang Skor. SEBUAHSkorLebih besar dari 0,77 hasil dalamJawaban terbaikdari 1, yang merupakan respon positif atau prediksi, danSkorkurang dari 0,77 hasilJawaban terbaikdari 0, yang merupakan respon negatif atau prediksi.

Contoh berikut menunjukkan prediksi positif dan negatif berdasarkan ambang skor 0.77.

Prediksi positif:

bestAnswer	score
1	0.8228876

Dalam contoh ini, nilai untukJawaban terbaikadalah 1, dan nilaiSkoradalah 0.8228876. Nilai untukJawaban terbaikadalah 1 karenaSkorlebih besar dari ambang skor 0,77. SEBUAHJawaban terbaikdari 1 menunjukkan bahwa pelanggan kemungkinan untuk membeli produk Anda, dan, oleh karena itu, dianggap sebagai prediksi positif.

Prediksi negatif:

bestAnswer	score
0	0.7695356

Dalam contoh ini, nilaiJawaban terbaikadalah 0 karenaSkoradalah 0,7695356, yang kurang dari ambang skor 0,77. ParameterJawaban terbaikdari 0 menunjukkan bahwa pelanggan tidak mungkin untuk membeli produk Anda, dan, karena itu, dianggap sebagai prediksi negatif.

Setiap baris hasil batch sesuai dengan baris dalam input batch Anda (observasi di sumber data Anda).

Setelah menganalisis prediksi, Anda dapat menjalankan kampanye pemasaran yang ditargetkan; misalnya, dengan mengirimkan selebaran kepada semua orang dengan skor prediksi1.

Sekarang Anda telah membuat, meninjau, dan menggunakan model Anda,membersihkan data dan sumber daya AWS yang Anda buatuntuk menghindari biaya yang tidak perlu dan untuk menjaga ruang kerja Anda rapi.

Langkah 6: Pembersihan

Untuk menghindari biaya Amazon Simple Storage Service (Amazon S3), hapus data yang tersimpan di Amazon S3. Anda tidak dikenakan biaya untuk sumber daya Amazon ML-nya yang tidak digunakan, namun kami menyarankan agar Anda menghapusnya agar ruang kerja tetap bersih.

Untuk menghapus data input yang disimpan di Amazon S3

1. Buka konsol Amazon S3 di https://console.aws.amazon.com/s3/.
2. Arahkan ke lokasi Amazon S3 tempat Anda menyimpan`banking.csv`dan`banking-batch.csv`file.
3. Pilih`banking.csv`,`banking-batch.csv`, dan`.writePermissionCheck.tmp`file.
4. Pilih Actions (Tindakan), lalu pilih Delete (Hapus).
5. Saat diminta konfirmasi, pilihOKE.

Meskipun Anda tidak dikenakan biaya untuk menyimpan catatan prediksi batch yang dijalankan Amazon MLatau sumber data, model, dan evaluasi yang Anda buat selama tutorial, sebaiknya Anda menghapusnya untuk mencegah mengacaukan ruang kerja Anda.

Untuk menghapus prediksi batch

1. Arahkan ke lokasi Amazon S3 tempat Anda menyimpan output prediksi batch.
2. Pilih`batch-prediction`folder.
3. Pilih Actions (Tindakan), lalu pilih Delete (Hapus).
4. Saat diminta konfirmasi, pilihOKE.

Untuk menghapus sumber daya Amazon ML-nya

1. Di dasbor Amazon XML, pilih sumber daya berikut.

   - Parameter`Banking Data 1`sumber data
   - Parameter`Banking Data 1_[percentBegin=0, percentEnd=70, strategy=sequential]`sumber data
   - Parameter`Banking Data 1_[percentBegin=70, percentEnd=100, strategy=sequential]`sumber data
   - Parameter`Banking Data 2`sumber data
   - Parameter`ML model: Banking Data 1`Model ML
   - Parameter`Evaluation: ML model: Banking Data 1`evaluasi
2. Pilih Actions (Tindakan), lalu pilih Delete (Hapus).
3. Di kotak dialog, pilihHapusuntuk menghapus semua sumber daya yang dipilih.

Anda telah berhasil menyelesaikan tutorial ini. Untuk terus menggunakan konsol untuk membuat sumber data, model, dan prediksi melihatPanduan Pengembang Amazon Machine Learning. Untuk mempelajari cara menggunakan API, lihatReferensi API Amazon Machine Learning.

Membuat dan Menggunakan Sumber Data

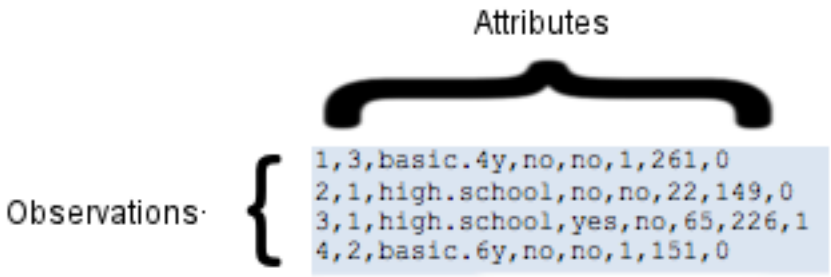
Anda dapat menggunakan sumber data Amazon ML-untuk melatih model ML-nya, mengevaluasi model ML-nya, dan menghasilkan prediksi batch menggunakan model ML-nya. Objek sumber data berisi metadata tentang data input Anda. Saat membuat sumber data, Amazon MLS membaca data input Anda, menghitung statistik deskriptif pada atributnya, dan menyimpan statistik, skema, dan informasi lainnya sebagai bagian dari objek sumber data. Setelah membuat sumber data, Anda dapat menggunakanWawasan data Amazonuntuk mengeksplorasi properti statistik data masukan Anda, dan Anda dapat menggunakan sumber data untukmelatih model MLnya.

> **Note**
>
> Bagian ini mengasumsikan bahwa Anda sudah memahamiAmazon Machine Learning.

Topik

Memahami Format Data untuk Amazon

Data input adalah data yang Anda gunakan untuk membuat sumber data. Anda harus menyimpan data input dalam format nilai yang dipisahkan koma (.csv). Setiap baris dalam file.csv adalah catatan data tunggal atau observasi. Setiap kolom dalam file.csv berisi atribut observasi. Misalnya, gambar berikut menunjukkan isi file.csv yang memiliki empat pengamatan, masing-masing dalam barisnya sendiri. Setiap pengamatan berisi delapan atribut, dipisahkan dengan koma. Atribut mewakili informasi berikut tentang setiap individu yang diwakili oleh pengamatan: customerId, jobId, pendidikan, perumahan, pinjaman, kampanye, durasi, WillRespondToCampaign.



Atribut

Amazon IL memerlukan nama untuk setiap atribut. Anda dapat menentukan nama atribut dengan:

- Termasuk nama atribut di baris pertama (juga dikenal sebagai baris header) dari file.csv yang Anda gunakan sebagai data input Anda
- Termasuk nama atribut dalam file skema terpisah yang terletak di bucket S3 yang sama dengan data input Anda

Untuk informasi selengkapnya tentang penggunaan file skema, lihatMembuat Skema Data.

Contoh berikut dari file.csv mencakup nama-nama atribut di baris header.

```
customerId,jobId,education,housing,loan,campaign,duration,willRespondToCampaign

1,3,basic.4y,no,no,1,261,0

2,1,high.school,no,no,22,149,0

3,1,high.school,yes,no,65,226,1

4,2,basic.6y,no,no,1,151,0
```

Persyaratan Format File

File .csv yang berisi data input harus memenuhi persyaratan berikut:

- Harus dalam teks biasa menggunakan set karakter seperti ASCII, Unicode, atau EBCDIC.
- Terdiri dari pengamatan, satu pengamatan per baris.
- Untuk setiap pengamatan, nilai atribut harus dipisahkan dengan koma.

- Jika nilai atribut berisi koma (pembatas), seluruh nilai atribut harus tertutup dalam tanda kutip ganda.
- Setiap pengamatan harus dihentikan dengan karakter end-of-line, yang merupakan karakter khusus atau urutan karakter yang menunjukkan akhir baris.
- Nilai atribut tidak dapat menyertakan karakter end-of-line, bahkan jika nilai atribut tertutup dalam tanda kutip ganda.
- Setiap pengamatan harus memiliki jumlah atribut dan urutan atribut yang sama.
- Setiap pengamatan harus tidak lebih besar dari 100 KB. Amazon ML-menolak pengamatan yang lebih besar dari 100 KB selama pemrosesan. Jika Amazon ML-menolak lebih dari 10.000 observasi, itu menolak seluruh file.csv.

Menggunakan Beberapa File Sebagai Input Data ke Amazon IL

Anda dapat memberikan masukan Anda ke Amazon ML-nya sebagai satu file, atau sebagai kumpulan file. Koleksi harus memenuhi kondisi ini:

- Semua file harus memiliki skema data yang sama.
- Semua file harus berada dalam awalan Amazon Simple Storage Service (Amazon S3) yang sama, dan jalur yang Anda berikan untuk koleksi harus diakhiri dengan karakter garis miring ('/') yang sama.

Misalnya, jika file data Anda diberi nama input1.csv, input2.csv, dan input3.csv, dan nama bucket S3 Anda adalah s3: //examplebucket, path file Anda mungkin terlihat seperti ini:

s3: //examplebucket/path/to/data/input1.csv

s3: //examplebucket/path/to/data/input2.csv

s3: //examplebucket/path/to/data/input3.csv

Anda akan memberikan lokasi S3 berikut sebagai masukan ke Amazon ML-nya:

's3: //examplebucket/path/ke/data/'

Karakter Akhir-of-line dalam Format CSV

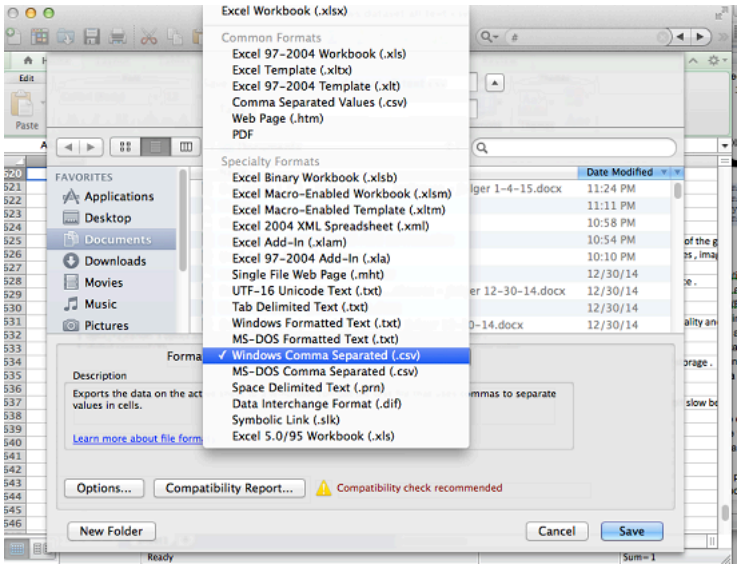
Ketika Anda membuat file.csv Anda, setiap observasi akan dihentikan oleh karakter end-of-line khusus. Karakter ini tidak terlihat, tetapi secara otomatis disertakan pada akhir setiap pengamatan ketika Anda menekan AndaENTERatauKembalikunci. Karakter khusus yang mewakili end-of-

line bervariasi tergantung pada sistem operasi Anda. Sistem Unix, seperti Linux atau OS X, menggunakanUmpan bariskarakter yang ditunjukkan oleh “\ n” (kode ASCII 10 dalam desimal atau 0x0a dalam heksadesimal). Microsoft Windows menggunakan dua karakter yang disebutkereta kembalidanUmpan barisyang ditunjukkan oleh “\ r\ n” (kode ASCII 13 dan 10 dalam desimal atau 0x0d dan 0x0a dalam heksadesimal).

Jika Anda ingin menggunakan OS X dan Microsoft Excel untuk membuat file .csv, lakukan prosedur berikut ini. Pastikan untuk memilih format yang benar.

Untuk menyimpan file.csv jika Anda menggunakan OS X dan Excel

1. Saat menyimpan file .csv, pilihFormat,dan kemudian pilihJendela Koma Terpisah (.csv).
2. Pilih Save (Simpan).



> **Important**
>
> Jangan simpan file.csv dengan menggunakanNilai Terpisah Koma (.csv)atauMS-DOS Koma Terpisah (.csv)format karena Amazon IL tidak dapat membacanya.

Membuat Skema Data untuk Amazon

SEBUAHskematerdiri dari semua atribut dalam input data dan jenis data yang sesuai mereka. Hal ini memungkinkan Amazon IL untuk memahami data dalam sumber data. Amazon IL menggunakan informasi dalam skema untuk membaca dan menafsirkan data input, statistik komputasi, menerapkan

transformasi atribut yang benar, dan menyempurnakan algoritma pembelajarannya. Jika Anda tidak menyediakan skema, Amazon IL menyimpulkan satu dari data.

Skema contoh

Agar Amazon IL dapat membaca data input dengan benar dan menghasilkan prediksi yang akurat, setiap atribut harus diberi tipe data yang benar. Mari kita telusuri contoh untuk melihat bagaimana tipe data ditugaskan ke atribut, dan bagaimana atribut dan tipe data disertakan dalam skema. Kami akan memanggil contoh “Kampanye Pelanggan” karena kami ingin memprediksi pelanggan mana yang akan menanggapi kampanye email kami. File input kami adalah file.csv dengan sembilan kolom:

```
1,3,web developer,basic.4y,no,no,1,261,0

2,1,car repair,high.school,no,no,22,149,0

3,1,car mechanic,high.school,yes,no,65,226,1

4,2,software developer,basic.6y,no,no,1,151,0
```

Ini skema untuk data ini:

```
{
    "version": "1.0",
    "rowId": "customerId",
    "targetAttributeName": "willRespondToCampaign",
    "dataFormat": "CSV",
    "dataFileContainsHeader": false,
    "attributes": [
        {
            "attributeName": "customerId",
            "attributeType": "CATEGORICAL"
        },
        {
            "attributeName": "jobId",
            "attributeType": "CATEGORICAL"
        },
        {
            "attributeName": "jobDescription",
            "attributeType": "TEXT"
        },
        {
```

```
            "attributeName": "education",
            "attributeType": "CATEGORICAL"
        },
        {
            "attributeName": "housing",
            "attributeType": "CATEGORICAL"

        },
        {
            "attributeName": "loan",
            "attributeType": "CATEGORICAL"

        },
        {
            "attributeName": "campaign",
            "attributeType": "NUMERIC"
        },
        {
            "attributeName": "duration",
            "attributeType": "NUMERIC"
        },

        {
            "attributeName": "willRespondToCampaign",
            "attributeType": "BINARY"
        }

    ]
}
```

Dalam file skema untuk contoh ini, nilai untuk`rowId`adalah`customerId`:

```
"rowId": "customerId",
```

Atribut`willRespondToCampaign`didefinisikan sebagai atribut target:

```
"targetAttributeName": "willRespondToCampaign ",
```

Parameter`customerId`atribut dan`CATEGORICAL`tipe data yang terkait dengan kolom pertama,`jobId`atribut dan`CATEGORICAL`tipe data yang terkait dengan kolom kedua, yang`jobDescription`atribut dan`TEXT`tipe data yang terkait dengan kolom ketiga,`education`atribut dan`CATEGORICAL`tipe data yang terkait dengan kolom keempat, dan sebagainya. Kolom kesembilan

dikaitkan dengan`willRespondToCampaign`atribut dengan`BINARY`tipe data, dan atribut ini juga didefinisikan sebagai atribut target.

Menggunakan TargetAttributeName Field

Parameter`targetAttributeName`value adalah nama atribut yang ingin Anda prediksi. Anda harus menetapkan`targetAttributeName`saat membuat atau mengevaluasi model.

Ketika Anda melatih atau mengevaluasi model ML-nya,`targetAttributeName`mengidentifikasi nama atribut dalam data input yang berisi jawaban “benar” untuk atribut target. Amazon IL menggunakan target, yang mencakup jawaban yang benar, untuk menemukan pola dan menghasilkan model ML-nya.

Saat mengevaluasi model, Amazon ML-nya menggunakan target untuk memeriksa keakuratan prediksi Anda. Setelah Anda membuat dan mengevaluasi model ML, Anda dapat menggunakan data dengan unassigned`targetAttributeName`untuk menghasilkan prediksi dengan model ML-mu.

Anda menentukan atribut target di konsol Amazon MLsaat Anda membuat sumber data, atau dalam file skema. Jika Anda membuat file skema Anda sendiri, gunakan sintaks berikut untuk menentukan atribut target:

```
"targetAttributeName": "exampleAttributeTarget",
```

Dalam contoh ini,`exampleAttributeTarget`adalah nama atribut dalam file input Anda yang merupakan atribut target.

Menggunakan Bidang RowID

Parameter`row ID`adalah bendera opsional yang terkait dengan atribut dalam data input. Jika ditentukan, atribut ditandai sebagai`row ID`termasuk dalam output prediksi. Atribut ini membuatnya lebih mudah untuk mengasosiasikan prediksi mana yang sesuai dengan pengamatan mana. Contoh yang baik`row ID`adalah ID pelanggan atau atribut unik yang serupa.

> **Note**
>
> ID baris hanya untuk referensi Anda. Amazon ML-nya tidak menggunakannya saat melatih model ML-nya. Memilih atribut sebagai ID baris tidak termasuk dari yang digunakan untuk melatih model ML-nya.

Anda menentukan`row ID`di konsol Amazon XML saat Anda membuat sumber data, atau dalam file skema. Jika Anda membuat file skema Anda sendiri, gunakan sintaksis berikut untuk menentukan`row ID`:

```
"rowId": "exampleRow",
```

Dalam contoh sebelumnya,`exampleRow`adalah nama atribut dalam file input Anda yang didefinisikan sebagai ID baris.

Ketika menghasilkan prediksi batch, Anda mungkin mendapatkan output sebagai berikut:

```
tag,bestAnswer,score
55,0,0.46317
102,1,0.89625
```

Dalam contoh ini,`RowID`mewakili atribut`customerId`. Misalnya, customerId`55`diperkirakan akan menanggapi kampanye email kami dengan keyakinan rendah (0.46317), sementara`customerId` `102`diperkirakan akan menanggapi kampanye email kami dengan keyakinan tinggi (0.89625).

Menggunakan Field AttributeType

Di Amazon IL, ada empat tipe data untuk atribut:

Biner

PilihBINARYuntuk atribut yang hanya memiliki dua negara yang mungkin, seperti`yes`atau`no`.

Misalnya, atribut`isNew`, untuk melacak apakah seseorang adalah pelanggan baru, akan memiliki`true`nilai untuk menunjukkan bahwa individu adalah pelanggan baru, dan`false`nilai untuk menunjukkan bahwa ia bukan pelanggan baru.

Nilai negatif yang valid adalah`0`,`n`,`no`,`f`, dan`false`.

Nilai positif yang valid adalah`1`,`y`,`yes`,`t`, dan`true`.

Amazon ML-mengabaikan kasus input biner dan strip ruang putih di sekitarnya. Misalnya,`" FaLSe "`adalah nilai biner yang valid. Anda dapat mencampur nilai biner yang Anda gunakan dalam sumber data yang sama, seperti menggunakan`true`,`no`, dan`1`. Hanya output Amazon`0`dan`1`atribut biner.

Kategoris

PilihCATEGORICALuntuk atribut yang mengambil sejumlah nilai string yang unik. Misalnya, ID pengguna, bulan, dan kode pos adalah nilai kategoris. Atribut kategoris diperlakukan sebagai string tunggal, dan tidak tokenized lebih lanjut.

Numerik

PilihNUMERICuntuk atribut yang mengambil kuantitas sebagai nilai.

Misalnya, suhu, berat, dan tingkat klik adalah nilai numerik.

Tidak semua atribut yang memegang angka adalah numerik. Atribut kategoris, seperti hari dalam sebulan dan ID, sering direpresentasikan sebagai angka. Untuk dianggap numerik, angka harus sebanding dengan nomor lain. Misalnya, ID pelanggan`664727`tidak memberitahu Anda tentang ID pelanggan`124552`, tapi berat`10`memberitahu Anda bahwa atribut yang lebih berat daripada atribut dengan berat`5`. Hari dalam sebulan tidak numerik, karena yang pertama dari satu bulan bisa terjadi sebelum atau sesudah kedua bulan lagi.

> **Note**
>
> Bila Anda menggunakan Amazon XML untuk membuat skema Anda, itu menetapkan`Numeric`tipe data untuk semua atribut yang menggunakan angka. Jika Amazon ML-membuat skema Anda, periksa tugas yang salah dan tetapkan atribut tersebut ke`CATEGORICAL`.

Text

PilihTEXTuntuk atribut yang merupakan serangkaian kata. Saat membaca dalam atribut teks, Amazon MLnya mengubahnya menjadi token, dibatasi oleh spasi putih.

Misalnya,`email subject`menjadi`email`dan`subject`, dan`email-subject here`menjadi`email-subject`dan`here`.

Jika tipe data untuk variabel dalam skema pelatihan tidak sesuai dengan tipe data untuk variabel tersebut dalam skema evaluasi, Amazon IL mengubah jenis data evaluasi agar sesuai dengan tipe data pelatihan. Misalnya, jika skema data pelatihan memberikan tipe data dari`TEXT`ke

variabelage, tapi skema evaluasi memberikan tipe dataNUMERICkepadaage, kemudian Amazon ML-memperlakukan usia dalam data evaluasi sebagaiTEXTvariabel bukanNUMERIC.

Untuk informasi tentang statistik yang terkait dengan setiap jenis data, lihatStatistik deskriptif.

Menyediakan Skema ke Amazon ML-nya

Setiap sumber data membutuhkan skema. Anda dapat memilih dari dua cara untuk menyediakan Amazon MLdengan skema:

- Izinkan Amazon XML untuk menyimpulkan jenis data dari setiap atribut dalam file data input dan secara otomatis membuat skema untuk Anda.
- Menyediakan file skema saat Anda mengunggah data Amazon Simple Storage Service (Amazon S3).

Memungkinkan Amazon ML-Membuat Skema Anda

Saat Anda menggunakan konsol Amazon MLuntuk membuat sumber data, Amazon MLnya menggunakan aturan sederhana, berdasarkan nilai variabel Anda, untuk membuat skema Anda. Kami sangat menyarankan Anda meninjau skema Amazon ML-dibuat, dan memperbaiki jenis data jika tidak akurat.

Menyediakan Skema

Setelah membuat file skema, Anda harus membuatnya tersedia untuk Amazon ML-nya. Anda memiliki dua opsi:

1. Sediakan skema dengan menggunakan konsol Amazon ML-nya.

   Gunakan konsol untuk membuat sumber data Anda, dan sertakan file skema dengan menambahkan ekstensi.schema ke nama file file file data input Anda. Misalnya, jika URI Amazon Simple Storage Service (Amazon S3) pada data input Anda adalah s3: //my-bucket-name/data/input.csv, URI ke skema Anda adalah s3: //my-bucket-name/data/input.csv.schema. Amazon IL secara otomatis menempatkan file skema yang Anda berikan alih-alih mencoba menyimpulkan skema dari data Anda.

   Untuk menggunakan direktori file sebagai masukan data ke Amazon IL, tambahkan ekstensi.schema ke jalur direktori Anda. Misalnya, jika file data Anda berada di lokasi s3: //examplebucket/path/to/data/, URI ke skema Anda akan s3: //examplebucket/path/to/data/.schema.
2. Sediakan skema dengan menggunakan API Amazon ML-nya.

Jika Anda berencana untuk memanggil API Amazon XML untuk membuat sumber data Anda, Anda dapat mengunggah file skema ke Amazon S3, dan kemudian memberikan URI ke file tersebut di`DataSchemaLocationS3`atribut`CreateDataSourceFromS3`API Untuk informasi selengkapnya, lihatdibuatasourcefroms3.

Anda dapat memberikan skema langsung di payload`CreateDataSource*API`salih-alih menyimpannya terlebih dahulu ke Amazon S3. Anda melakukan ini dengan menempatkan string skema penuh di`DataSchema`atribut`CreateDataSourceFromS3`,`CreateDataSourceFromRDS`, atau`CreateDataSourceFromRedshift`API. Untuk informasi selengkapnya, lihatReferensi API Amazon Machine Learning.

Memisahkan Data Anda

Tujuan mendasar dari model ML adalah untuk membuat prediksi akurat pada instance data masa depan di luar yang digunakan untuk melatih model. Sebelum menggunakan model L untuk membuat prediksi, kita perlu mengevaluasi kinerja prediktif model. Untuk memperkirakan kualitas prediksi model L dengan data yang belum terlihat, kita dapat memesan, atau membagi, sebagian dari data yang kita sudah tahu jawabannya sebagai proxy untuk data masa depan dan mengevaluasi seberapa baik model L memprediksi jawaban yang benar untuk data tersebut. Anda membagi sumber data menjadi bagian untuk sumber data pelatihan dan sebagian untuk sumber data evaluasi.

Amazon IL menyediakan tiga opsi untuk membagi data Anda:

- Pra-split data- Anda dapat membagi data menjadi dua lokasi input data, sebelum mengunggahnya ke Amazon Simple Storage Service (Amazon S3) dan membuat dua sumber data terpisah dengan mereka.
- Amazon ML-split- Anda dapat memberi tahu Amazon IL untuk membagi data Anda secara berurutan saat membuat sumber data pelatihan dan evaluasi.
- Amazon ML-split- Anda dapat memberi tahu Amazon XML untuk membagi data Anda menggunakan metode acak unggulan saat membuat sumber data pelatihan dan evaluasi.

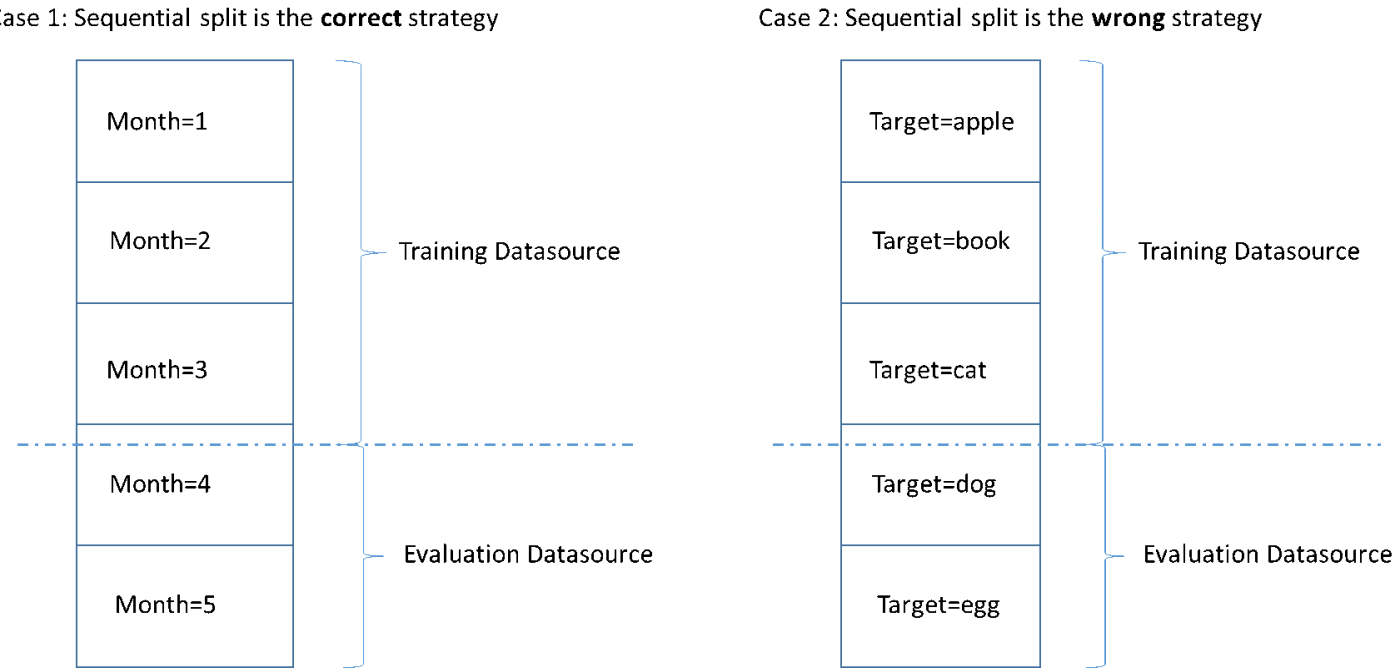
Pra-membelah Data Anda

Jika Anda menginginkan kontrol eksplisit atas data dalam sumber data pelatihan dan evaluasi Anda, pisahkan data Anda menjadi lokasi data terpisah, dan buat sumber data terpisah untuk lokasi input dan evaluasi.

Berurutan Memisahkan Data Anda

Cara sederhana untuk membagi data input Anda untuk pelatihan dan evaluasi adalah dengan memilih subset data yang tidak tumpang tindih sambil mempertahankan urutan catatan data. Pendekatan ini berguna jika Anda ingin mengevaluasi model ML-mu pada data untuk tanggal tertentu atau dalam rentang waktu tertentu. Misalnya, katakan bahwa Anda memiliki data keterlibatan pelanggan selama lima bulan terakhir, dan Anda ingin menggunakan data historis ini untuk memprediksi keterlibatan pelanggan di bulan depan. Menggunakan awal rentang untuk pelatihan, dan data dari akhir rentang untuk evaluasi mungkin menghasilkan perkiraan kualitas model yang lebih akurat daripada menggunakan data catatan yang diambil dari seluruh rentang data.

Gambar berikut menunjukkan contoh kapan Anda harus menggunakan strategi pemisahan berurutan versus kapan Anda harus menggunakan strategi acak.

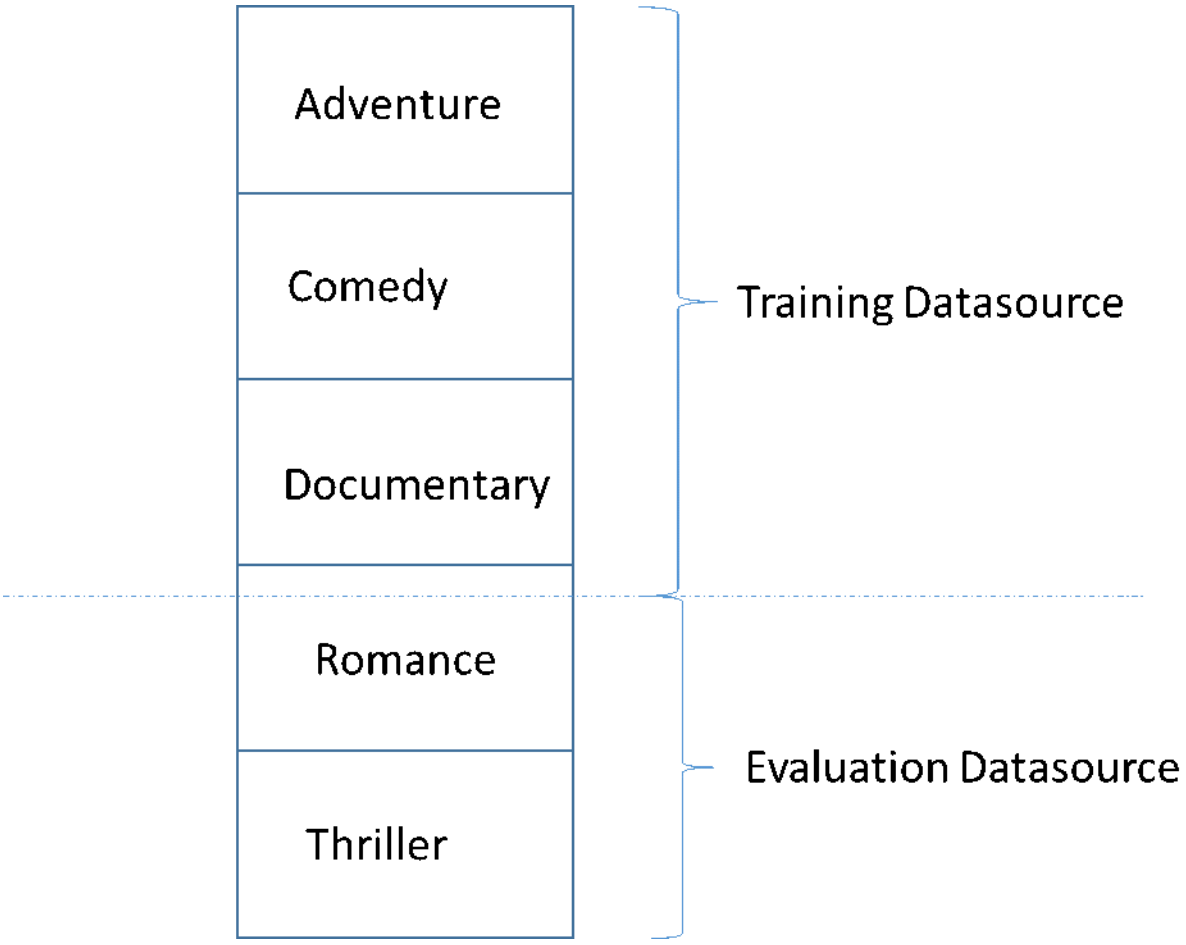


Ketika Anda membuat sumber data, Anda dapat memilih untuk membagi sumber data secara berurutan, dan Amazon IL menggunakan 70 persen pertama data Anda untuk pelatihan dan 30 persen sisanya dari data untuk evaluasi. Ini adalah pendekatan default saat Anda menggunakan konsol Amazon MLuntuk membagi data Anda.

Memisahkan Data Anda secara acak

Secara acak membagi data input ke dalam pelatihan dan evaluasi sumber data memastikan bahwa distribusi data serupa dalam pelatihan dan evaluasi sumber data. Pilih opsi ini ketika Anda tidak perlu mempertahankan urutan data input Anda.

Amazon IL menggunakan metode pembuatan nomor pseudo-acak unggulan untuk membagi data Anda. Benih didasarkan sebagian pada nilai string input dan sebagian pada isi data itu sendiri. Secara default, konsol Amazon ML-menggunakan lokasi S3 data input sebagai string. Pengguna API dapat memberikan string kustom. Ini berarti bahwa mengingat bucket dan data S3 yang sama, Amazon ML-membagi data dengan cara yang sama setiap saat. Untuk mengubah cara Amazon ML-membagi data, Anda dapat menggunakan`CreateDatasourceFromS3`,`CreateDatasourceFromRedshift`, atau`CreateDatasourceFromRDS`API dan memberikan nilai untuk string benih. Saat menggunakan API ini untuk membuat sumber data terpisah untuk pelatihan dan evaluasi, penting untuk menggunakan nilai string benih yang sama untuk sumber data dan bendera pelengkap untuk satu sumber data, untuk memastikan bahwa tidak ada tumpang tindih antara data pelatihan dan evaluasi.



Perangkap umum dalam mengembangkan model ML-berkualitas tinggi adalah mengevaluasi model ML pada data yang tidak mirip dengan data yang digunakan untuk pelatihan. Misalnya, misalnya Anda menggunakan L untuk memprediksi genre film, dan data pelatihan Anda berisi film dari genre Adventure, Comedy, dan Documentary. Namun, data evaluasi Anda hanya berisi data dari genre Romantis dan Thriller. Dalam hal ini, model L tidak mempelajari informasi tentang genre Romantis

dan Thriller, dan evaluasi tidak mengevaluasi seberapa baik model tersebut mempelajari pola untuk genre Adventure, Comedy, dan Documentary. Akibatnya, informasi genre tidak berguna, dan kualitas prediksi model L untuk semua genre dikompromikan. Model dan evaluasi terlalu berbeda (memiliki statistik deskriptif yang sangat berbeda) untuk menjadi berguna. Hal ini dapat terjadi ketika data input diurutkan berdasarkan salah satu kolom dalam dataset, dan kemudian dibagi secara berurutan.

Jika sumber data pelatihan dan evaluasi Anda memiliki distribusi data yang berbeda, Anda akan melihat peringatan evaluasi dalam evaluasi model Anda. Untuk informasi selengkapnya tentang pemberitahuan evaluasi, lihatPeringatan Evaluasi.

Anda tidak perlu menggunakan pemisahan acak di Amazon MLjika Anda telah mengacak data input Anda, misalnya, dengan menyeret data input secara acak di Amazon S3, atau dengan menggunakan kueri Amazon Redshift SQL`random()`fungsi atau MySQL SQL query`rand()`berfungsi saat membuat sumber data. Dalam kasus ini, Anda dapat mengandalkan opsi split sekuensial untuk membuat sumber data pelatihan dan evaluasi dengan distribusi serupa.

Wawasan Data

Amazon ML-menghitung statistik deskriptif pada data masukan yang dapat Anda gunakan untuk memahami data Anda.

Statistik deskriptif

Amazon ML-menghitung statistik deskriptif berikut untuk jenis atribut yang berbeda:

Numerik:

- Histogram distribusi
- Jumlah nilai tidak valid
- Nilai minimum, median, mean, dan maksimum

Biner dan kategoris:

- Hitung (dari nilai yang berbeda per kategori)
- Histogram distribusi nilai
- Nilai yang paling sering
- Nilai unik dihitung

- Persentase nilai sebenarnya (hanya biner)
- Kata yang paling menonjol
- Kata yang paling sering

Teks:

- Nama atribut
- Korelasi terhadap target (jika target ditetapkan)
- Kata Total
- Kata unik
- Rentang jumlah kata berturut-turut
- Rentang panjang kata
- Kata yang paling menonjol

Mengakses Wawasan Data di konsol Amazon ML-nya

Pada konsol Amazon XML, Anda dapat memilih nama atau ID dari sumber data apa pun untuk melihatnyaWawasan Datahalaman. Halaman ini menyediakan metrik dan visualisasi yang memungkinkan Anda mempelajari data input yang terkait dengan sumber data, termasuk informasi berikut:

- Ringkasan data
- Distribusi target
- Nilai yang Hilang
- Nilai tidak valid
- Ringkasan statistik variabel berdasarkan tipe data
- Distribusi variabel berdasarkan tipe data

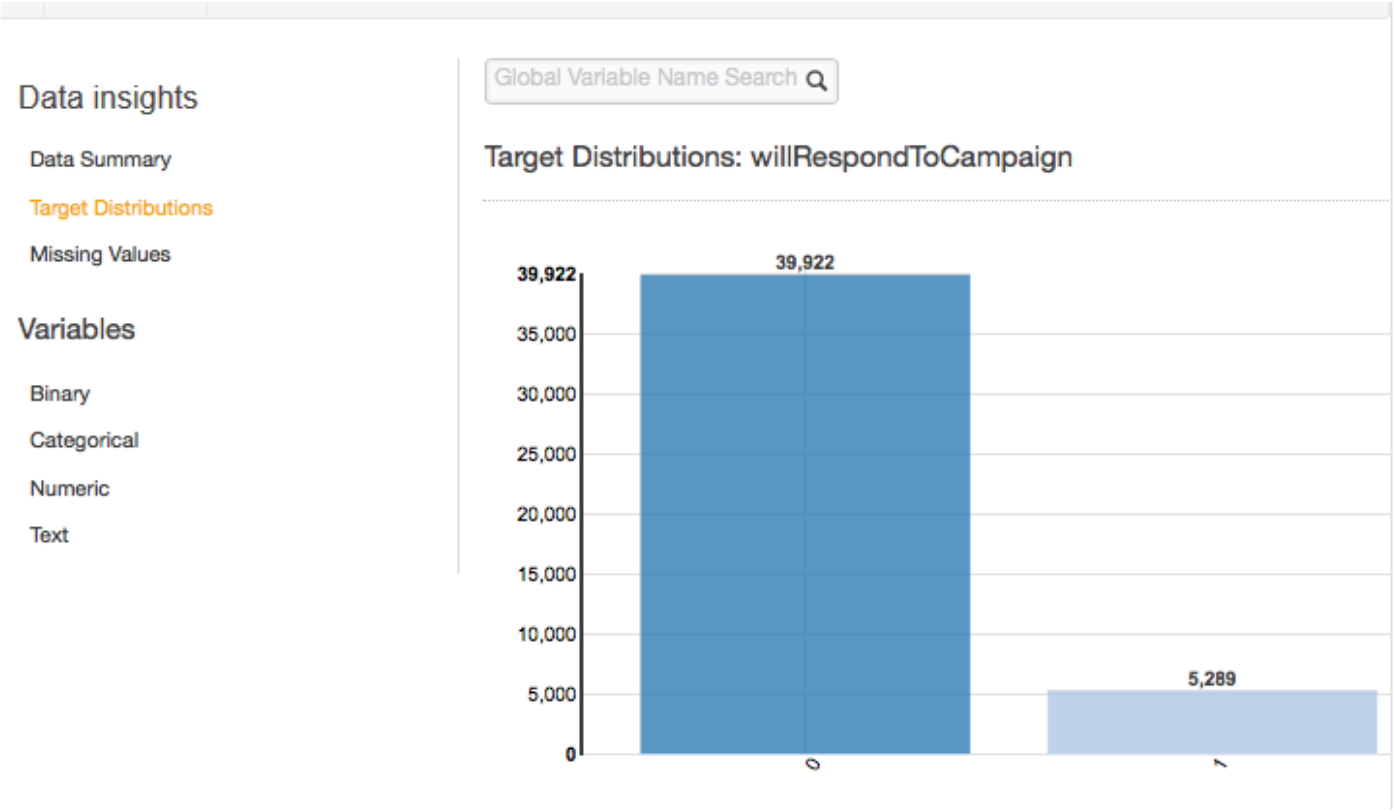
Bagian berikut menjelaskan metrik dan visualisasi secara lebih detail.

Ringkasan data

Laporan ringkasan data dari sumber data menampilkan informasi ringkasan, termasuk ID sumber data, nama, tempat selesai, status saat ini, atribut target, informasi data input (lokasi bucket S3, format data, jumlah catatan yang diproses dan jumlah catatan buruk yang dihadapi selama pemrosesan) juga sebagai jumlah variabel dengan tipe data.

Distribusi target

Laporan distribusi target menunjukkan distribusi atribut target dari sumber data. Pada contoh berikut, ada 39,922 observasi di mana atribut target willRespondToCampaign sama dengan 0. Ini adalah jumlah pelanggan yang tidak menanggapi kampanye email. Ada 5,289 pengamatan di mana WillRespondToCampaign sama dengan 1. Ini adalah jumlah pelanggan yang menanggapi kampanye email.



Nilai yang Hilang

Laporan nilai yang hilang mencantumkan atribut dalam data input yang nilainya hilang. Hanya atribut dengan tipe data numerik yang dapat memiliki nilai yang hilang. Karena nilai yang hilang dapat memengaruhi kualitas pelatihan model ML-nya, kami menyarankan agar nilai yang hilang diberikan, jika memungkinkan.

Selama pelatihan model ML, jika atribut target hilang, Amazon ML-menolak rekaman yang sesuai. Jika atribut target ada dalam catatan, namun nilai untuk atribut numerik lain hilang, maka Amazon MLnya akan mengabaikan nilai yang hilang. Dalam hal ini, Amazon XML membuat atribut pengganti dan menetapkannya ke 1 untuk menunjukkan bahwa atribut ini hilang. Hal ini memungkinkan Amazon ML-mempelajari pola dari terjadinya nilai yang hilang.

Nilai Tidak Valid

Nilai tidak valid hanya dapat terjadi dengan tipe data Numerik dan Biner. Anda dapat menemukan nilai yang tidak valid dengan melihat statistik ringkasan variabel dalam laporan tipe data. Dalam contoh berikut, ada satu nilai yang tidak valid dalam durasi atribut numerik dan dua nilai tidak valid dalam tipe data Biner (satu di atribut perumahan dan satu di atribut pinjaman).

Numeric Variables

Variables	Correlations to Target	Missing Values	Invalid Values	Range	Mean	Median	Preview
duration	0.05165	2 (0%)	1 (0%)	0 - 4918	258.1618	180	

Binary Variables

Variables	Correlations to Target	Percent True	Invalid Values	Preview
campaign	NA	100%	27667 (61%)	
housing	0.01842	56%	1 (0%)	
loan	0.00656	16%	1 (0%)	
willRespondToCampaign	NA	12%	0 (0%)	

Korelasi Variabel-Target

Setelah Anda membuat sumber data, Amazon L dapat mengevaluasi sumber data dan mengidentifikasi korelasi, atau dampak, antara variabel dan target. Misalnya, harga produk mungkin memiliki dampak yang signifikan pada apakah atau tidak itu adalah penjual terbaik, sedangkan dimensi produk mungkin memiliki sedikit daya prediktif.

Ini umumnya merupakan praktik terbaik untuk memasukkan sebanyak mungkin variabel dalam data pelatihan Anda. Namun, kebisingan yang diperkenalkan dengan memasukkan banyak variabel dengan sedikit daya prediktif mungkin berdampak negatif pada kualitas dan keakuratan model ML-mu.

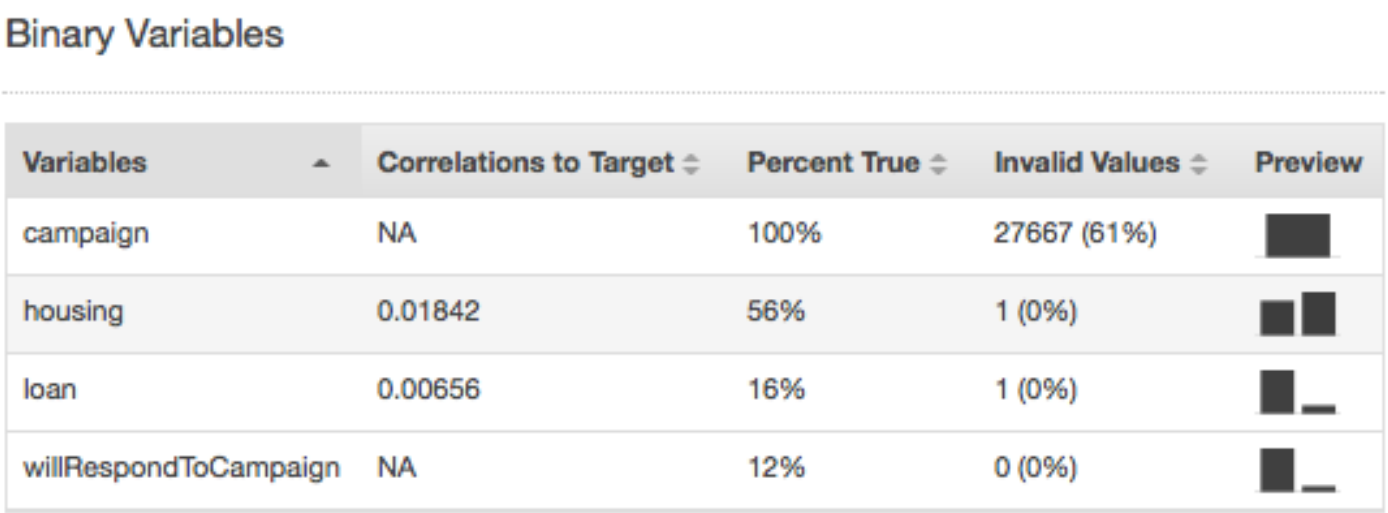
Anda mungkin dapat meningkatkan kinerja prediktif model Anda dengan menghapus variabel yang memiliki dampak kecil ketika Anda melatih model Anda. Anda dapat menentukan variabel mana yang tersedia untuk proses pembelajaran mesin dalamresep, yang merupakan mekanisme transformasi Amazon ML-nya. Untuk mempelajari lebih lanjut tentang resep, lihatTransformasi Data untuk Machine Learning.

Ringkasan Statistik Atribut berdasarkan Tipe Data

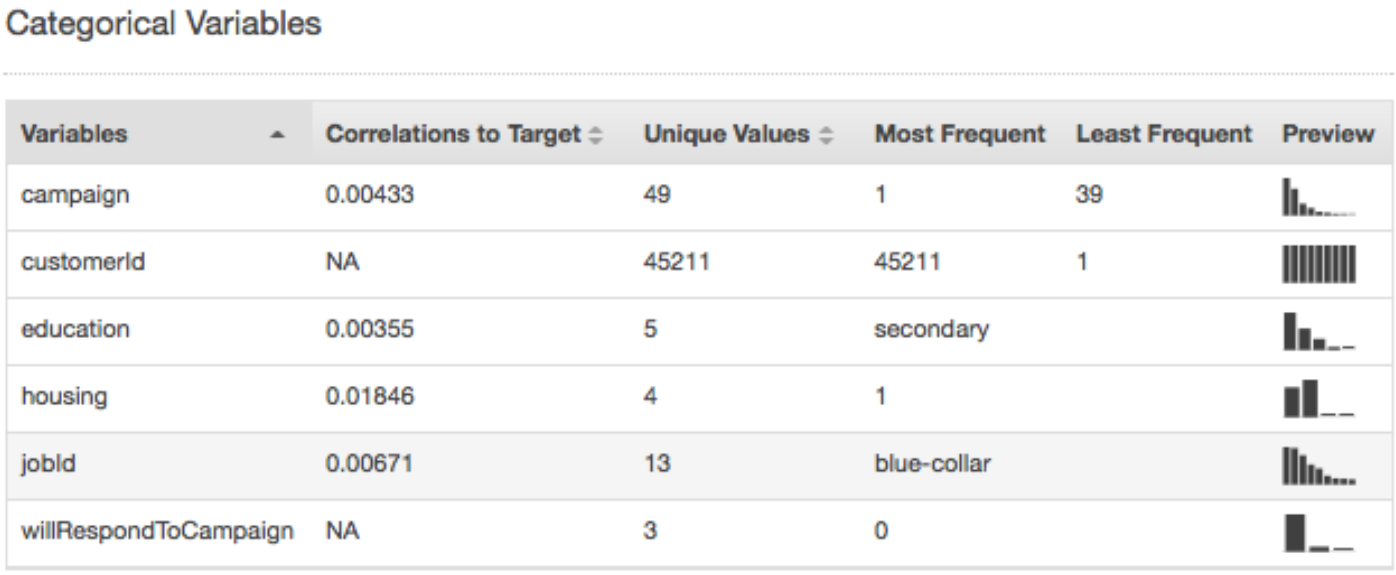
Dalam laporan wawasan data, Anda dapat melihat statistik ringkasan atribut berdasarkan tipe data berikut:

- Biner
- Kategorik
- Numerik
- Teks

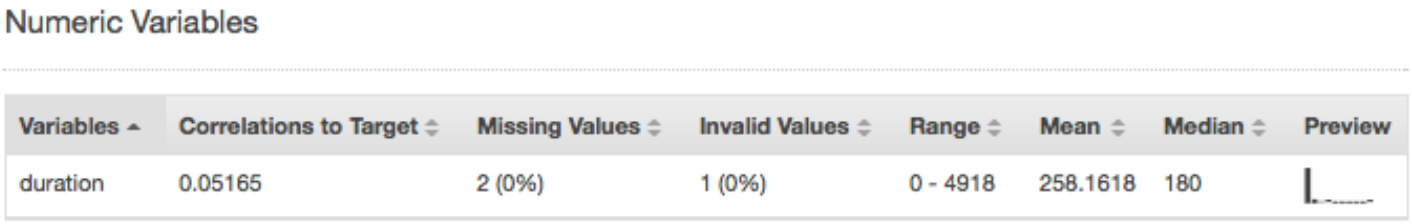
Ringkasan statistik untuk tipe data Biner menunjukkan semua atribut biner. ParameterKorelasi untuk menargetkankolom menunjukkan informasi yang dibagikan antara kolom target dan kolom atribut. ParameterPersen benarkolom menunjukkan persentase pengamatan yang memiliki nilai 1. ParameterNilai tidak validkolom menunjukkan jumlah nilai yang tidak valid serta persentase nilai yang tidak valid untuk setiap atribut. ParameterPratinjaukolom menyediakan link ke distribusi grafis untuk setiap atribut.



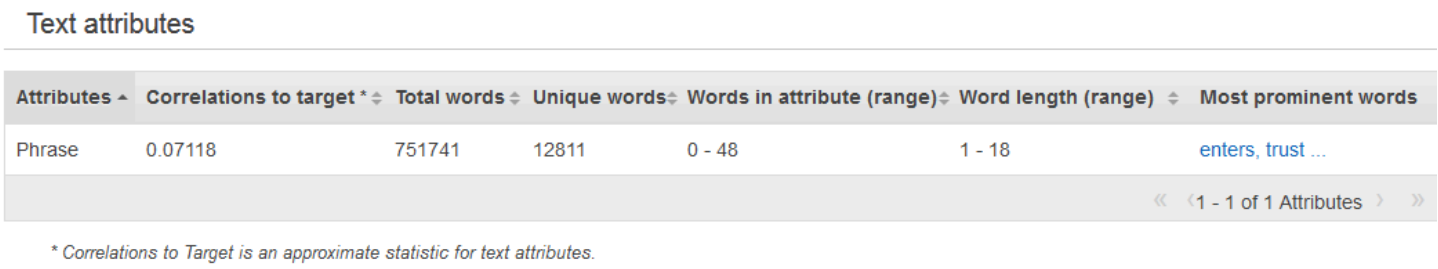
Ringkasan statistik untuk tipe data kategoris menunjukkan semua atribut kategoris dengan jumlah nilai unik, nilai yang paling sering, dan nilai yang paling sering. ParameterPratinjaukolom menyediakan link ke distribusi grafis untuk setiap atribut.



Statistik ringkasan untuk tipe data Numerik menunjukkan semua atribut Numerik dengan jumlah nilai yang hilang, nilai tidak valid, rentang nilai, mean, dan median. ParameterPratinjaukolom menyediakan link ke distribusi grafis untuk setiap atribut.



Statistik ringkasan untuk tipe data Teks menunjukkan semua atribut Teks, jumlah kata dalam atribut itu, jumlah kata unik dalam atribut itu, rentang kata dalam atribut, rentang panjang kata, dan kata-kata yang paling menonjol. ParameterPratinjaukolom menyediakan link ke distribusi grafis untuk setiap atribut.



Text attributes

Attributes	Correlations to target *	Total words	Unique words	Words in attribute (range)	Word length (range)	Most prominent words
Phrase	0.07118	751741	12811	0 - 48	1 - 18	enters, trust ...

« ‹ 1 - 1 of 1 Attributes › »

* Correlations to Target is an approximate statistic for text attributes.

Contoh berikutnya menunjukkan statistik tipe data Teks untuk variabel teks yang disebut review, dengan empat catatan.

```
1. The fox jumped over the fence.
2. This movie is intriguing.
3.
4. Fascinating movie.
```

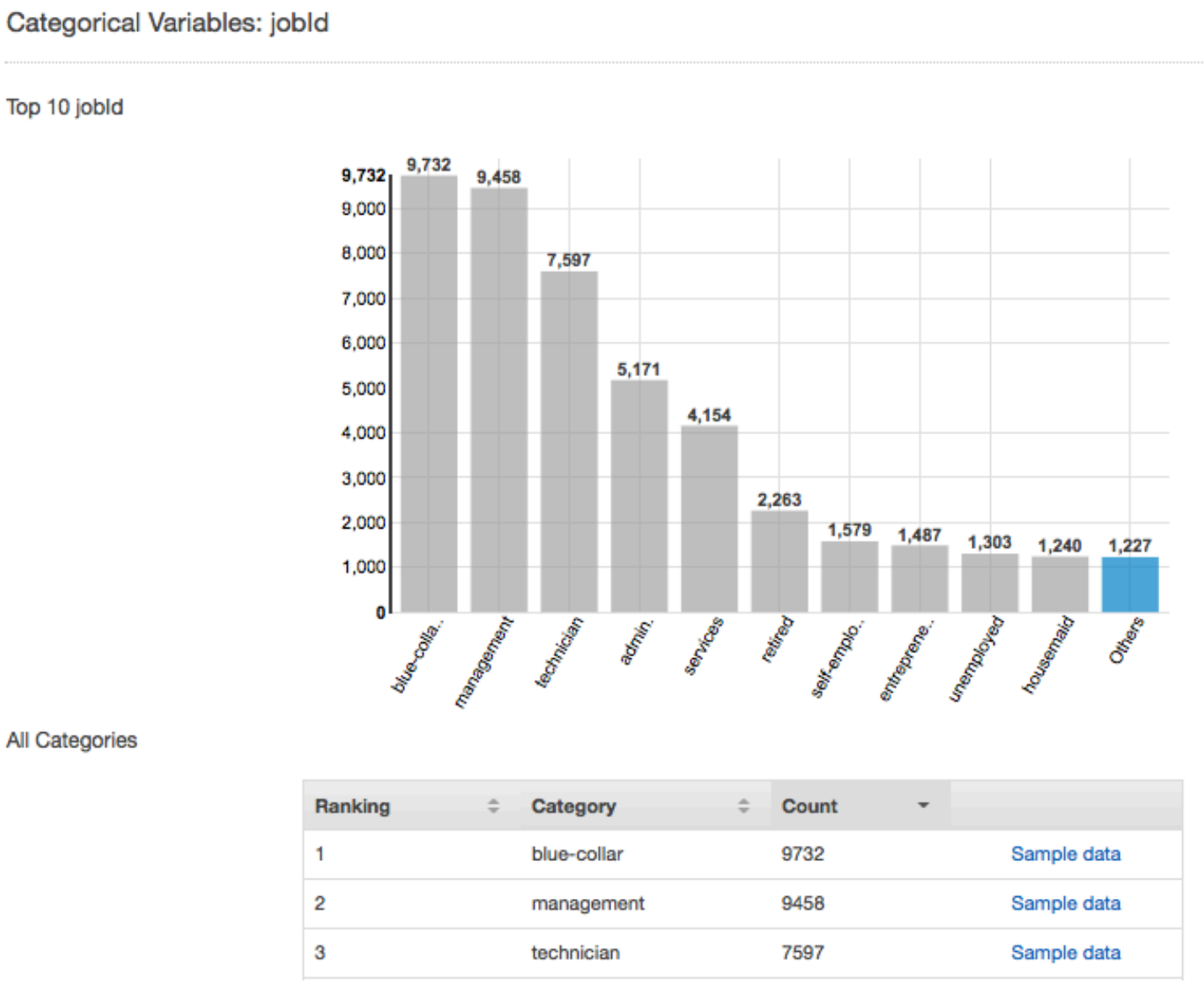
Kolom untuk contoh ini akan menampilkan informasi berikut.

- ParameterAtributkolom menunjukkan nama variabel. Dalam contoh ini, kolom ini akan mengatakan “review.”
- ParameterKorelasi untuk menargetkankolom hanya ada jika target ditentukan. Korelasi mengukur jumlah informasi yang diberikan atribut ini tentang target. Semakin tinggi korelasi, semakin banyak atribut ini memberitahu Anda tentang target. Korelasi diukur dalam hal informasi bersama antara representasi disederhanakan dari atribut teks dan target.
- ParameterKata Totalkolom menunjukkan jumlah kata yang dihasilkan dari tokenizing setiap record, membatasi kata-kata dengan spasi putih. Dalam contoh ini, kolom ini akan mengatakan “12".
- ParameterKata unikkolom menunjukkan jumlah kata unik untuk atribut. Dalam contoh ini, kolom ini akan mengatakan “10.”
- ParameterKata-kata dalam atribut (range)kolom menunjukkan jumlah kata dalam satu baris dalam atribut. Dalam contoh ini, kolom ini akan mengatakan “0-6.”
- ParameterPanjang kata (rentang)kolom menunjukkan kisaran berapa banyak karakter dalam kata-kata. Dalam contoh ini, kolom ini akan mengatakan “2-11.”
- ParameterKata yang paling menonjolkolom menunjukkan daftar peringkat kata-kata yang muncul dalam atribut. Jika ada atribut target, kata-kata diberi peringkat berdasarkan korelasi mereka dengan target, yang berarti bahwa kata-kata yang memiliki korelasi tertinggi tercantum terlebih dahulu. Jika tidak ada target yang ada dalam data, maka kata-kata tersebut diberi peringkat oleh entropi mereka.

Memahami Distribusi Atribut Kategoris dan Biner

Dengan mengklikPratinjaulink yang terkait dengan atribut kategoris atau biner, Anda dapat melihat distribusi atribut itu serta data sampel dari file input untuk setiap nilai kategoris atribut.

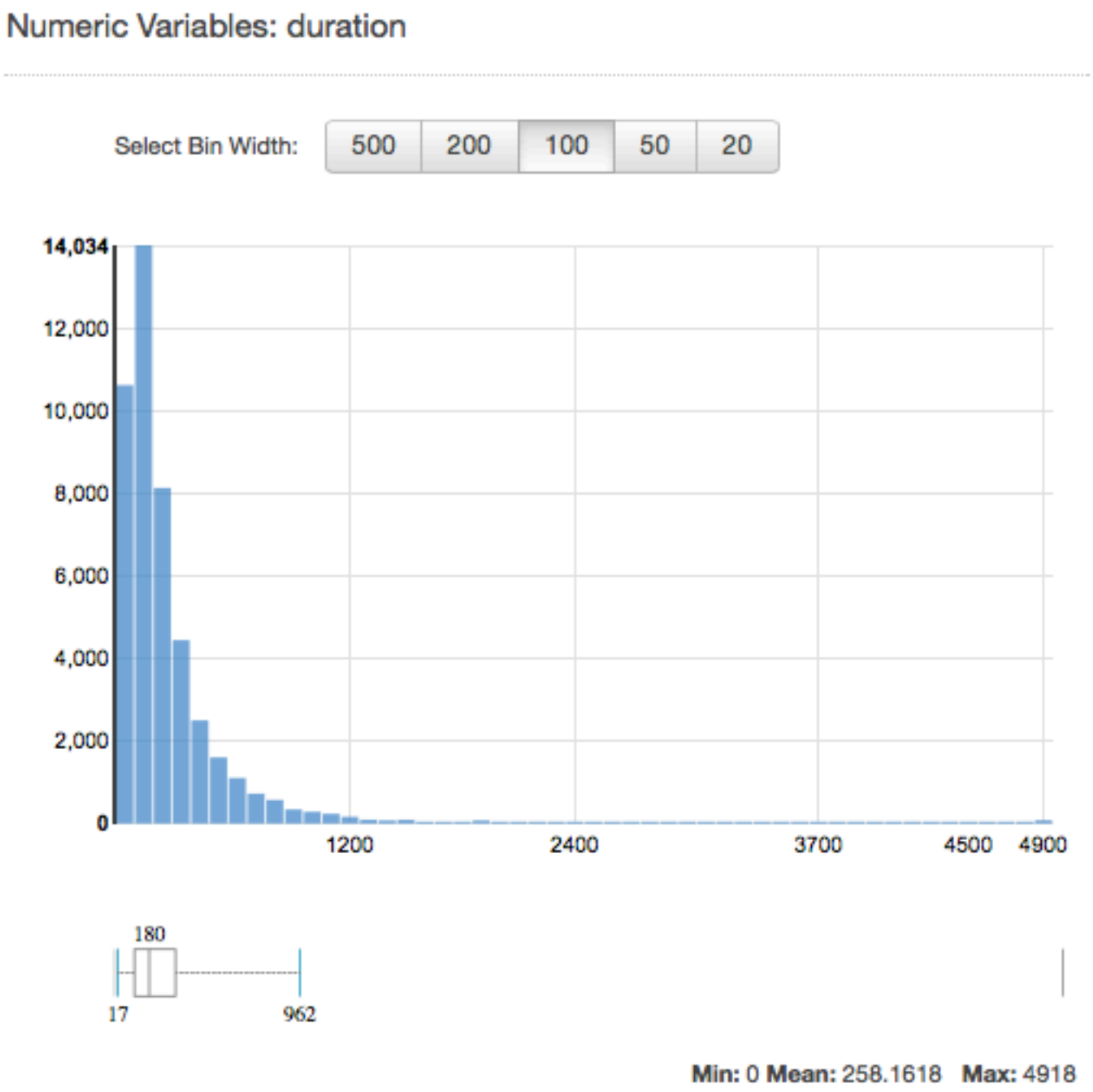
Misalnya, tangkapan layar berikut menunjukkan distribusi untuk atribut kategoris jobId. Distribusi menampilkan 10 nilai kategoris teratas, dengan semua nilai lainnya dikelompokkan sebagai “lainnya”. Ini peringkat masing-masing dari 10 nilai kategoris teratas dengan jumlah pengamatan dalam file input yang berisi nilai itu, serta link untuk melihat pengamatan sampel dari file data input.



Memahami Distribusi Atribut Numerik

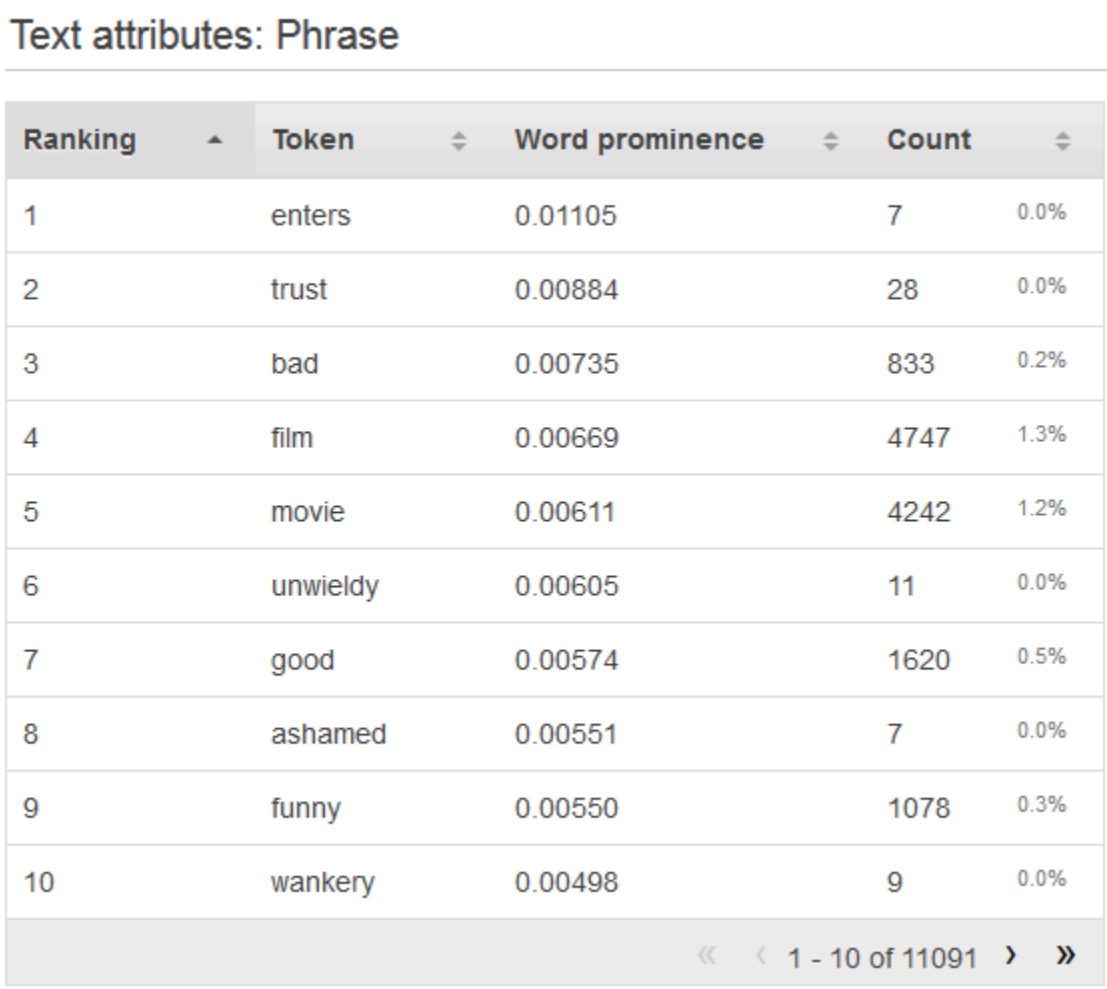
Untuk melihat distribusi atribut numerik, klikPratinjaulink dari atribut. Saat melihat distribusi atribut numerik, Anda dapat memilih ukuran bin 500, 200, 100, 50, atau 20. Semakin besar ukuran bin, jumlah yang lebih kecil dari grafik batang yang akan ditampilkan. Selain itu, resolusi distribusi akan kasar untuk ukuran bin besar. Sebaliknya, pengaturan ukuran bucket ke 20 meningkatkan resolusi distribusi yang ditampilkan.

Nilai minimum, dan maksimum juga ditampilkan, seperti yang ditunjukkan pada tangkapan layar berikut.



Memahami Distribusi Atribut Teks

Untuk melihat distribusi atribut teks, klikPratinjaulink dari atribut. Saat melihat distribusi atribut teks, Anda akan melihat informasi berikut.



Peringkat

Token teks diberi peringkat berdasarkan jumlah informasi yang mereka sampaikan, paling informatif hingga paling informatif.

Token

Token menunjukkan kata dari teks masukan bahwa deretan statistik adalah tentang.

Kata menonjol

Jika ada atribut target, kata-kata diberi peringkat berdasarkan korelasi mereka dengan target, sehingga kata-kata yang memiliki korelasi tertinggi tercantum terlebih dahulu. Jika tidak ada target hadir dalam data, maka kata-kata peringkat oleh entropi mereka, yaitu, jumlah informasi yang mereka dapat berkomunikasi.

Jumlah hitungan

Jumlah hitung menunjukkan jumlah catatan masukan bahwa token muncul di.

Persentase

Persentase hitungan menunjukkan persentase baris data input token muncul di.

Menggunakan Amazon S3 dengan Amazon MLS

Amazon Simple Storage Service (Amazon S3) adalah penyimpanan untuk Internet. Anda dapat menggunakan Amazon S3 untuk menyimpan dan mengambil data sebanyak apa pun kapan pun, dari mana pun di web. Amazon S3 menggunakan Amazon S3 sebagai repositori data utama untuk tugas berikut:

- Untuk mengakses file input Anda untuk membuat objek sumber data untuk melatih dan mengevaluasi model ML-mu.
- Untuk mengakses file input Anda untuk menghasilkan prediksi batch.
- Ketika Anda menghasilkan prediksi batch dengan menggunakan model ML-mu, untuk menampilkan file prediksi ke bucket S3 yang Anda tentukan.
- Untuk menyalin data yang telah Anda simpan di Amazon Redshift atau Amazon Relational Database Service (Amazon RDS) ke dalam file.csv dan unggah ke Amazon S3.

Untuk mengaktifkan Amazon ML-tugas ini, Anda harus memberikan izin kepada Amazon ML-nya untuk mengakses data Amazon S3 Anda.

> **Note**
>
> Anda tidak dapat menampilkan file prediksi batch ke bucket S3 yang hanya menerima file terenkripsi sisi server. Pastikan bahwa kebijakan bucket Anda memungkinkan mengunggah file yang tidak terenkripsi dengan mengonfirmasi bahwa kebijakan tersebut tidak termasuk`Deny`efek untuk`s3:PutObject`tindakan ketika tidak ada`s3:x-amz-server-side-encryption`header dalam permintaan. Untuk informasi selengkapnya tentang kebijakan bucket enkripsi sisi server S3, lihatMelindungi Data Menggunakan Enkripsi Sisi ServerdiPanduan Pengguna Amazon Simple Storage Service.

Mengunggah Data ke Amazon S3

Anda harus mengunggah data input ke Amazon Simple Storage Service (Amazon S3) karena Amazon ML-nya membaca data dari lokasi Amazon S3. Anda dapat mengunggah data secara langsung ke Amazon S3 (misalnya, dari komputer Anda), atau Amazon IL dapat menyalin data yang telah disimpan di Amazon Redshift atau Amazon Relational Database Service (RDS) ke dalam file.csv dan mengunggahnya ke Amazon S3.

Untuk informasi lebih lanjut tentang menyalin data dari Amazon Redshift atau Amazon RDS, lihatMenggunakan Amazon Redshift dengan AmazonatauMenggunakan Amazon RDS dengan Amazon, masing-masing.

Bagian ini menjelaskan cara mengunggah data input Anda secara langsung dari komputer ke Amazon S3. Sebelum Anda memulai prosedur di bagian ini, Anda harus memiliki data dalam file.csv. Untuk informasi tentang cara memformat file.csv Anda dengan benar sehingga Amazon IL dapat menggunakannya, lihatMemahami Format Data untuk Amazon.

Untuk mengunggah data dari komputer ke Amazon S3

1. Masuk ke Konsol Manajemen AWS dan buka konsol Amazon S3 di https://console.aws.amazon.com/s3.
2. Buat bucket atau pilih bucket yang ada.
    a. Untuk membuat bucket, pilihBuat Bucket. Beri nama bucket Anda, pilih wilayah (Anda dapat memilih wilayah yang tersedia), lalu pilihBuat. Untuk informasi selengkapnya, lihatBuat BucketdiPanduan Memulai Amazon Simple Storage.
    b. Untuk menggunakan bucket yang ada, cari bucket dengan memilih bucketSemua Bucketdaftar. Ketika nama bucket muncul, pilih itu, lalu pilihUnggah.
3. DiUnggahkotak dialog, pilihTambahkan File.
4. Arahkan ke folder yang berisi file input data .csv Anda, lalu pilihBuka.

Izin

Untuk memberikan izin untuk Amazon ML-nya mengakses salah satu bucket S3 Anda, Anda harus mengedit kebijakan bucket.

Untuk informasi tentang memberikan izin Amazon XML untuk membaca data dari bucket Anda di Amazon S3, lihatMemberikan Izin Amazon untuk Membaca Data Anda dari Amazon S3.

Untuk informasi tentang pemberian izin Amazon ML-output hasil prediksi batch ke bucket Anda di Amazon S3, lihatMemberikan Izin Amazon ML-Prediksi Output ke Amazon S3.

Untuk informasi tentang mengelola izin akses ke sumber daya Amazon S3, lihatPanduan Developer Amazon S3.

Membuat Sumber Data Amazon ML-dari Data di Amazon Redshift

Jika Anda memiliki data yang disimpan di Amazon Redshift, Anda dapat menggunakanBuat Sumber datawizard di konsol Amazon Machine Learning (Amazon ML) untuk membuat objek datasource. Saat membuat sumber data dari data Amazon Redshift, Anda menentukan klaster yang berisi data dan kueri SQL untuk mengambil data Anda. Amazon ML-mengeksekusi kueri dengan menerapkan Amazon Redshift`Unload`perintah pada klaster. Amazon ML-nya menyimpan hasil di lokasi Amazon Simple Storage Service (Amazon S3) pilihan Anda, lalu menggunakan data yang disimpan di Amazon S3 untuk membuat datasource. Klaster data, klaster Amazon Redshift, dan bucket S3 harus berada di wilayah yang sama.

> **Note**
>
> Amazon IL tidak mendukung pembuatan sumber data dari klaster Amazon Redshift di VPC pribadi. Klaster harus memiliki alamat IP publik.

Topik

- Parameter yang Diperlukan untuk Wizard Buat Datasource
- Membuat Sumber Data dengan Amazon Redshift Data (Konsol)
- Memecahkan Masalah Amazon Redshift

Parameter yang Diperlukan untuk Wizard Buat Datasource

Untuk memungkinkan Amazon ML-nya terhubung ke database Amazon Redshift Anda dan membaca data atas nama Anda, Anda harus menyediakan yang berikut ini:

- Amazon Redshift`ClusterIdentifier`
- Nama basis data Amazon Redshift
- Kredensi database Amazon Redshift (nama pengguna dan kata sandi)

- Amazon Amazon RedshiftAWS Identity and Access Management(IAM) peran
- Kueri SQL Amazon Redshift
- (Opsional) Lokasi skema Amazon ML-nya
- Lokasi pementasan Amazon S3 (tempat Amazon ML-menempatkan data sebelum membuat sumber data)

Selain itu, Anda perlu memastikan bahwa pengguna IAM atau peran yang membuat sumber data Amazon Redshift (baik melalui konsol atau dengan menggunakan`CreateDataSourceFromRedshift`tindakan) memiliki`iam:PassRole`izin.

Amazon Redshift**`ClusterIdentifier`**

Gunakan parameter peka huruf ini untuk memungkinkan Amazon IL menemukan dan terhubung ke klaster Anda. Anda dapat memperoleh pengenal klaster (nama) dari konsol Amazon Redshift. Untuk informasi selengkapnya tentang klaster, lihatKlaster Amazon Redshift.

Nama Basis Amazon Redshift

Gunakan parameter ini untuk memberi tahu Amazon ML-database mana di klaster Amazon Redshift berisi data yang ingin Anda gunakan sebagai sumber data Anda.

Kredensi Basis Amazon Redshift

Gunakan parameter ini untuk menentukan nama pengguna dan kata sandi pengguna database Amazon Redshift yang konteksnya kueri keamanan akan dieksekusi.

> **Note**
>
> Amazon IL memerlukan nama pengguna dan kata sandi Amazon Redshift untuk terhubung ke database Amazon Redshift Anda. Setelah membongkar data ke Amazon S3, Amazon IL tidak pernah menggunakan kembali kata sandi Anda, juga tidak menyimpannya.

Amazon ML-Peran Redshift Amazon

Gunakan parameter ini untuk menentukan nama peran IAM yang harus digunakan Amazon MLuntuk mengonfigurasi grup keamanan untuk klaster Amazon Redshift dan kebijakan bucket untuk lokasi pementasan Amazon S3.

Jika Anda tidak memiliki peran IAM yang dapat mengakses Amazon Redshift, Amazon ML-nya dapat membuat peran untuk Anda. Ketika Amazon ML-nya membuat peran, itu akan membuat dan melampirkan kebijakan terkelola pelanggan ke peran IAM. Kebijakan yang dibuat Amazon ML-memberikan izin Amazon ML-hanya untuk mengakses klaster yang Anda tetapkan.

Jika Anda sudah memiliki peran IAM untuk mengakses Amazon Redshift, Anda dapat mengetikkan ARN peran, atau memilih peran dari daftar drop-down. Peran IAM dengan akses Amazon Redshift tercantum di bagian atas drop down.

Peran IAM harus memiliki konten sebagai berikut:

```
{
    "Version": "2012-10-17",
    "Statement": [
    {
        "Effect": "Allow",
        "Principal": {
            "Service": "machinelearning.amazonaws.com"
        },
        "Action": "sts:AssumeRole",
        "Condition": {
            "StringEquals": { "aws:SourceAccount": "123456789012" },
           "ArnLike": { "aws:SourceArn": "arn:aws:machinelearning:us-
east-1:123456789012:datasource/*" }
        }
    }]
}
```

Untuk informasi selengkapnya tentang Kebijakan yang Dikelola Pelanggan, lihatKebijakan Terkelola PelanggandiPanduan Pengguna IAM.

Kueri Amazon Redshift

Gunakan parameter ini untuk menentukan kueri SQL SELECT yang dijalankan Amazon ML-nya di database Amazon Redshift Anda untuk memilih data Anda. Amazon RedshiftMEMBONGKARtindakan untuk menyalin hasil kueri Anda dengan aman ke lokasi Amazon S3.

> **Note**
>
> Amazon ML-bekerja paling baik ketika catatan masukan berada dalam urutan acak (dikocokkan). Anda dapat dengan mudah mengacak hasil kueri Amazon Redshift SQL

Anda dengan menggunakan Amazon Redshiftacak ()fungsi. Sebagai contoh, katakanlah bahwa ini adalah kueri asli:

```
"SELECT col1, col2, … FROM training_table"
```

Anda dapat menyematkan pengocokan acak dengan memperbarui kueri seperti ini:

```
"SELECT col1, col2, … FROM training_table ORDER BY random()"
```

Skema Lokasi (Opsional)

Gunakan parameter ini untuk menentukan jalur Amazon S3 ke skema Anda untuk data Amazon Redshift yang akan diekspor oleh Amazon ML-nya.

Jika Anda tidak menyediakan skema untuk sumber data Anda, konsol Amazon ML-otomatis membuat skema Amazon MLberdasarkan skema data kueri Amazon Redshift SQL. Skema Amazon IL memiliki tipe data yang lebih sedikit daripada skema Amazon Redshift, jadi ini bukan konversi satu-ke-satu. Konsol Amazon ML-mengubah jenis data Amazon Redshift ke jenis data Amazon ML-menggunakan skema konversi berikut.

Tipe Data Amazon Redshift	Alias Amazon Redshift	Tipe Data Amazon
SMALLINT	INT2	NUMERIK
BILANGAN BULAT	INT4	NUMERIK
BIGINT	INT8	NUMERIK
DESIMAL	NUMERIK	NUMERIK
NYATA	FLOAT4	NUMERIK
DOUBLE PRECISION	FLOAT8, MENGAPUNG	NUMERIK
BOOLEAN	BOOL	BINER
CHAR	KARAKTER, NCHAR, BPCHAR	KATEGORIS

Tipe Data Amazon Redshift	Alias Amazon Redshift	Tipe Data Amazon
VARCHAR	KARAKTER BERVARIASI, NVARCHAR, TEKS	TEXT
TANGGAL		TEXT
TIMESTAMP	TIMESTAMP TANPA ZONA WAKTU	TEXT

Untuk dikonversi ke Amazon`Binary`jenis data, nilai Amazon Redshift Boolean dalam data Anda harus didukung nilai Amazon L Binary. Jika tipe data Boolean Anda memiliki nilai yang tidak didukung, Amazon MLnya akan mengonversinya ke tipe data paling spesifik yang dapat dilakukan. Misalnya, jika Amazon Redshift Boolean memiliki nilai`0`,`1`, dan`2`, Amazon ML-mengkonversi Boolean ke`Numeric`tipe data. Untuk informasi selengkapnya tentang nilai biner yang didukung, lihatMenggunakan Field AttributeType.

Jika Amazon IL tidak dapat mengetahui jenis data, maka akan menjadi default`Text`.

Setelah Amazon IL mengubah skema, Anda dapat meninjau dan memperbaiki jenis data Amazon ML-yang ditetapkan di wizard Create Datasource, dan merevisi skema sebelum Amazon IL membuat sumber data.

Lokasi Penahapan Amazon S3

Gunakan parameter ini untuk menentukan nama lokasi pementasan Amazon S3 tempat Amazon LL menyimpan hasil kueri Amazon Redshift SQL. Setelah membuat sumber data, Amazon IL menggunakan data di lokasi pementasan alih-alih kembali ke Amazon Redshift.

> **Note**
>
> Karena Amazon ML-mengasumsikan peran IAM yang didefinisikan oleh peran Amazon ML-Amazon Redshift, Amazon IL memiliki izin untuk mengakses objek apa pun di lokasi pementasan Amazon S3 yang ditentukan. Karena itu, sebaiknya Anda menyimpan file yang tidak berisi informasi sensitif di lokasi pementasan Amazon S3. Misalnya, jika bucket root Anda`s3://mybucket/`, kami sarankan Anda membuat lokasi untuk menyimpan hanya file yang ingin Anda akses Amazon ML-nya, seperti`s3://mybucket/AmazonMLInput/`.

Membuat Sumber Data dengan Amazon Redshift Data (Konsol)

Konsol Amazon XML menyediakan dua cara untuk membuat sumber data menggunakan data Amazon Redshift. Anda dapat membuat sumber data dengan menyelesaikan wizard Create Datasource, atau, jika Anda sudah memiliki sumber data yang dibuat dari data Amazon Redshift, Anda dapat menyalin sumber data asli dan memodifikasi pengaturannya. Menyalin sumber data memungkinkan Anda untuk dengan mudah membuat beberapa sumber data serupa.

Untuk informasi tentang membuat datasource menggunakan API, lihatCreateDataSourceFromRedshift.

Untuk informasi selengkapnya tentang parameter dalam prosedur berikut ini, lihatParameter yang Diperlukan untuk Wizard Buat Datasource.

Topik

- Membuat Datasource (Konsol)
- Menyalin Datasource (Konsol)

Membuat Datasource (Konsol)

Untuk membongkar data dari Amazon Redshift ke dalam sumber data Amazon, gunakan wizard Create Datasource.

Untuk membuat sumber data dari data di Amazon Redshift

1. Buka konsol Amazon Machine Learning dihttps://console.aws.amazon.com/machinelearning/.
2. Di dasbor Amazon IL, di bawahEntitas, pilihBuat..., dan kemudian pilihSumber data.
3. PadaData inputhalaman, pilihAmazon Redshift.
4. Dalam wizard Buat Datasource, untukPengidentifikasi klaster, ketikkan nama klaster Anda.
5. UntukNama basis data, ketik nama database Amazon Redshift.
6. UntukNama pengguna basis data, ketik nama pengguna database Anda.
7. UntukKata sandi basis data, ketikkan kata sandi basis data Anda.
8. UntukPeran IAM, pilih IAM role. Jika Anda belum memilikinya, pilihBuat peran baru. Amazon IL menciptakan peran IAM Amazon Redshift untuk Anda.
9. Untuk menguji setelan Amazon Redshift Anda, pilihAkses uji(di sampingPeran IAM). Jika Amazon IL tidak dapat terhubung ke Amazon Redshift dengan pengaturan yang disediakan,

Anda tidak dapat terus membuat sumber data. Untuk bantuan penyelesaian masalah, lihat Menyelesaikan Masalah Kesalahan.

10. UntukKueri SQL, ketik kueri SQL Anda.
11. UntukLokasi skema, pilih apakah Anda ingin Amazon ML-membuat skema untuk Anda. Jika Anda telah membuat skema sendiri, ketik path Amazon S3 ke file skema Anda.
12. UntukLokasi pementasan Amazon S3, ketik jalur Amazon S3 ke bucket tempat Anda ingin Amazon ML-meletakkan data yang dibongkar dari Amazon Redshift.
13. (Opsional) UntukNama sumber data, ketikkan nama untuk sumber data Anda.
14. PilihVerifikasi. Amazon ML-nya memverifikasi dapat terhubung ke database Amazon Redshift Anda.
15. PadaSkemahalaman, meninjau jenis data untuk semua atribut dan memperbaikinya, yang diperlukan.
16. Pilih Continue (Lanjutkan).
17. Jika Anda ingin menggunakan sumber data ini untuk membuat atau mengevaluasi model MLnya, untukApakah Anda berencana untuk menggunakan dataset ini untuk membuat atau mengevaluasi model ML-nya?, pilihya. Jika Anda memilihya, pilih baris target Anda. Untuk informasi tentang target, lihatMenggunakan TargetAttributeName Field.

    Jika Anda ingin menggunakan sumber data ini bersama dengan model yang telah Anda buat untuk membuat prediksi, pilihTidak.
18. Pilih Continue (Lanjutkan).
19. UntukApakah data Anda berisi pengenal?, jika data Anda tidak berisi pengenal baris, pilihTidak.

    Jika data Anda mengandung pengenal baris, pilihya. Untuk informasi tentang pengidentifikasi baris, lihatMenggunakan Bidang RowID.
20. Pilih Tinjau.
21. PadaTinjauhalaman, tinjau pengaturan Anda, lalu pilihSelesai.

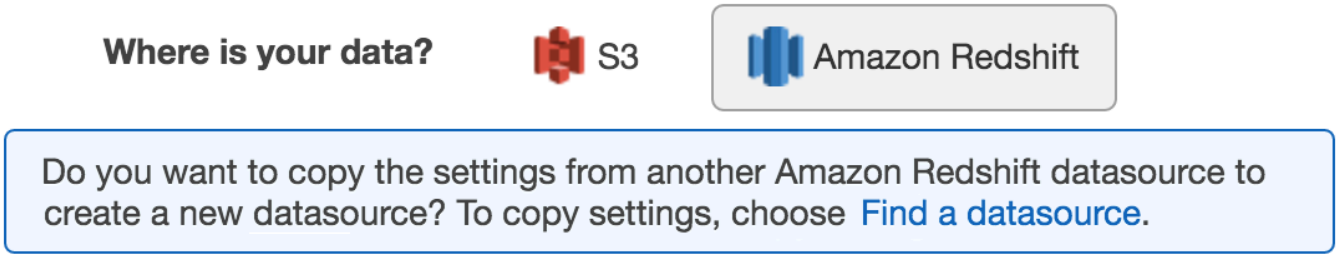
Setelah membuat datasource, Anda dapat menggunakannya untukcreate an ML model. Jika Anda telah membuat model, Anda dapat menggunakan datasource untukevaluate an ML modelataugenerate predictions.

Menyalin Datasource (Konsol)

Bila Anda ingin membuat sumber data yang mirip dengan sumber data yang ada, Anda dapat menggunakan konsol Amazon MLuntuk menyalin sumber data asli dan memodifikasi pengaturannya. Misalnya, Anda dapat memilih untuk memulai dengan sumber data yang ada, dan kemudian memodifikasi skema data agar sesuai dengan data Anda lebih dekat; mengubah kueri SQL yang digunakan untuk membongkar data dari Amazon Redshift; atau menentukan yang berbedaAWS Identity and Access Management(IAM) pengguna untuk mengakses klaster Amazon Redshift.

Untuk menyalin dan memodifikasi sumber data Amazon Redshift

1. Buka konsol Amazon Machine Learning dihttps://console.aws.amazon.com/machinelearning/.
2. Di dasbor Amazon IL, di bawahEntitas, pilihBuat..., dan kemudian pilihSumber data.
3. PadaData inputhalaman, untukDimana datamu?, pilihAmazon Redshift. Jika Anda sudah memiliki sumber data yang dibuat dari data Amazon Redshift, Anda memiliki opsi untuk menyalin pengaturan dari sumber data lain.



   Jika Anda belum memiliki datasource yang dibuat dari data Amazon Redshift, opsi ini tidak muncul.
4. PilihTemukan sumber data.
5. Pilih datasource yang ingin Anda salin, lalu pilihPengaturan Salin. Amazon ML-mengisi sebagian besar pengaturan sumber data dengan pengaturan dari sumber data asli. Itu tidak menyalin kata sandi database, lokasi skema, atau nama sumber data dari sumber data asli.
6. Memodifikasi pengaturan otomatis yang ingin Anda ubah. Misalnya, jika Anda ingin mengubah data yang dibongkar Amazon ML-nya dari Amazon Redshift, ubah kueri SQL.
7. UntukKata sandi basis data, ketikkan kata sandi basis data Anda. Amazon IL tidak menyimpan atau menggunakan kembali kata sandi Anda, jadi Anda harus selalu menyediakannya.
8. (Opsional) UntukLokasi skema, Amazon ML-pra-memilihSaya ingin Amazon ML-menghasilkan skema yang direkomendasikanuntuk Anda. Jika Anda telah membuat skema, pilihSaya ingin

menggunakan skema yang saya buat dan disimpan di Amazon S3dan ketik path ke file skema Anda di Amazon S3.

9. (Opsional) UntukNama sumber data, ketikkan nama untuk sumber data Anda. Jika tidak, Amazon IL menghasilkan nama sumber data baru untuk Anda.
10. PilihVerifikasi. Amazon ML-nya memverifikasi dapat terhubung ke database Amazon Redshift Anda.
11. (Opsional) Jika Amazon IL menyimpulkan skema untuk Anda, padaSkemahalaman, meninjau jenis data untuk semua atribut dan memperbaikinya, yang diperlukan.
12. Pilih Continue (Lanjutkan).
13. Jika Anda ingin menggunakan sumber data ini untuk membuat atau mengevaluasi model MLnya, untukApakah Anda berencana untuk menggunakan dataset ini untuk membuat atau mengevaluasi model ML-nya?, pilihya. Jika Anda memilihya, pilih baris target Anda. Untuk informasi tentang target, lihatMenggunakan TargetAttributeName Field.

    Jika Anda ingin menggunakan sumber data ini bersama dengan model yang telah Anda buat untuk membuat prediksi, pilihTidak.
14. Pilih Continue (Lanjutkan).
15. UntukApakah data Anda berisi pengenal?, jika data Anda tidak berisi pengenal baris, pilihTidak.

    Jika data Anda berisi pengenal baris, pilihya, lalu pilih baris yang ingin Anda gunakan sebagai pengenal. Untuk informasi tentang pengidentifikasi baris, lihatMenggunakan Bidang RowID.
16. Pilih Tinjau.
17. Tinjau pengaturan Anda, lalu pilihSelesai.

Setelah membuat datasource, Anda dapat menggunakannya untukcreate an ML model. Jika Anda telah membuat model, Anda dapat menggunakan datasource untukevaluate an ML modelataugenerate predictions.

Memecahkan Masalah Amazon Redshift

Saat Anda membuat sumber data Amazon Redshift, model, dan evaluasi, Amazon Machine Learning (Amazon ML) melaporkan status objek Amazon ML-Anda di konsol Amazon ML.dll. Jika Amazon ML mengembalikan pesan kesalahan, gunakan informasi dan sumber daya berikut untuk memecahkan masalah.

Untuk jawaban atas pertanyaan umum tentang Amazon ML, lihatFAQ Amazon Machine Learning. Anda juga dapat mencari jawaban dan memposting pertanyaan diAmazon Machine Learning.

Topik

- Menyelesaikan Masalah Kesalahan
- Menghubungi Support AWS

Menyelesaikan Masalah Kesalahan

Format peran tidak valid. Memberikan peran IAM yang valid. Misalnya, arn:n:aws:aws:iam::::::YourAccountID: peran/YourRedshiftRole.

Penyebab

Format Amazon Resource Name (ARN) dari IAM role IAM Anda tidak benar.

Solusi

Di wizard Create Datasource, perbaiki ARN untuk peran Anda. Untuk informasi tentang memformat ARN peran, lihatIAM ARNsdi dalamPanduan Pengguna IAM. Wilayah ini opsional untuk ARN peran IAM.

Perannya tidak valid. Amazon MLtidak dapat mengambil <role ARN>peran IAM. Menyediakan peran IAM yang valid dan membuatnya dapat diakses oleh Amazon ML.*

Penyebab

Peran Anda tidak diatur untuk mengizinkan Amazon MLnya menganggapnya.

Solusi

DiIAM konsol, edit peran Anda sehingga memiliki kebijakan kepercayaan yang memungkinkan Amazon MLuntuk mengambil peran yang melekat padanya.

<user ARN>Pengguna ini tidak berwenang untuk lulus <role ARN>peran IAM.

Penyebab

Pengguna IAM Anda tidak memiliki kebijakan izin yang memungkinkannya meneruskan peran ke Amazon ML.

Solusi

Lampirkan kebijakan izin ke pengguna IAM Anda yang memungkinkan Anda meneruskan peran ke Amazon ML. Anda dapat melampirkan kebijakan izin untuk pengguna IAM Anda diIAM konsol.

Melewati peran IAM di seluruh akun tidak diperbolehkan. IAM role harus menjadi milik akun ini.

Penyebab

Anda tidak dapat meneruskan peran yang dimiliki oleh akun IAM lain.

Solusi

Masuk ke akun AWS yang Anda gunakan untuk membuat peran. Anda dapat melihat peran IAM Anda diIAM konsol.

Aturan yang ditentukan tidak memiliki izin untuk melakukan operasi. Berikan peran yang memiliki kebijakan yang memberikan izin yang diperlukan kepada Amazon ML.../.

Penyebab

Peran IAM Anda tidak memiliki izin untuk melakukan operasi yang diminta.

Solusi

Edit kebijakan izin yang dilampirkan pada peran Anda diIAM konsoluntuk memberikan izin yang diperlukan.

Amazon MLtidak dapat mengonfigurasi grup keamanan pada klaster Amazon Redshift tersebut dengan peran IAM yang ditentukan.

Penyebab

Peran IAM Anda tidak memiliki izin yang diperlukan untuk mengonfigurasi klaster keamanan Amazon Redshift.

Solusi

Edit kebijakan izin yang dilampirkan pada peran Anda diIAM konsoluntuk memberikan izin yang diperlukan.

Terjadi kesalahan saat Amazon MLmencoba mengonfigurasi grup keamanan di klaster Anda. Coba lagi nanti.

Penyebab

Saat Amazon MLmencoba menyambung ke klaster Amazon Redshift Anda, Amazon MLnya mengalami masalah.

Solusi

Pastikan peran IAM yang Anda berikan di wizard Create Datasource memiliki semua izin yang diperlukan.

Format ID klaster tidak valid. Klaster ID harus dimulai dengan huruf dan harus berisi karakter alfanumerik dan tanda hubung saja. Pengidentifikasi tidak dapat berisi dua tanda hubung berturut-turut atau diakhiri dengan tanda hubung.

Penyebab

Format ID klaster Amazon Redshift Anda salah.

Solusi

Di wizard Create Datasource, perbaiki ID klaster Anda sehingga hanya berisi karakter alfanumerik dan tanda hubung dan tidak berisi dua tanda hubung berturut-turut atau diakhiri dengan tanda hubung.

Tidak ada <Amazon Redshift cluster name>klaster, atau klaster tidak berada di wilayah yang sama dengan layanan Amazon ML-mu. Tentukan klaster di Wilayah yang sama dengan Amazon MLM ini.

Penyebab

Amazon MLtidak dapat menemukan klaster Amazon Redshift Anda karena tidak terletak di wilayah tempat Anda membuat sumber data Amazon ML.dll.

Solusi

Verifikasi bahwa klaster Anda ada di konsol Amazon Redshiftklasterpage, bahwa Anda membuat sumber data di wilayah yang sama di mana klaster Amazon Redshift Anda berada, dan ID klaster yang ditentukan dalam wizard Create Datasource sudah benar.

Amazon ML' t dapat membaca data di klaster Amazon Redshift Anda. Berikan ID klaster Amazon Redshift yang benar.

Penyebab

Amazon MLtidak dapat membaca data di klaster Amazon Redshift yang Anda tentukan.

Solusi

Di wizard Create Datasource, tentukan ID klaster Amazon Redshift yang benar, verifikasi bahwa Anda membuat sumber data di wilayah yang sama dengan klaster Amazon Redshift Anda, dan klaster Anda terdaftar di Amazon Redshiftklasterhalaman.

<Amazon Redshift cluster name>Kluster tidak dapat diakses publik.

Penyebab

Amazon MLtidak dapat mengakses klaster Anda karena klaster tidak dapat diakses publik dan tidak memiliki alamat IP publik.

Solusi

Buat klaster dapat diakses publik dan berikan alamat IP publik. Untuk informasi tentang membuat klaster dapat diakses publik, lihatMemodifikasi klasterdi dalamPanduan Manajemen Amazon Redshift.

<Redshift>Status klaster tidak tersedia untuk Amazon ML. Gunakan konsol Amazon Redshift untuk melihat dan menyelesaikan masalah status klaster ini. Status cluster harus “tersedia.”

Penyebab

Amazon MLtidak dapat melihat status klaster.

Solusi

Pastikan klaster Anda tersedia. Untuk informasi tentang memeriksa status klaster Anda, lihatMendapatkan Ikhtisar Status Clusterdi dalamPanduan Manajemen Amazon Redshift. Untuk informasi tentang me-reboot cluster sehingga tersedia, lihatMelakukan boot ulang Klasterdi dalamPanduan Manajemen Amazon Redshift.

Tidak ada <database name>database di klaster ini. Verifikasi bahwa nama database benar atau tentukan cluster dan database lain.

Penyebab

Amazon MLtidak dapat menemukan database yang ditentukan dalam klaster yang ditentukan.

Solusi

Verifikasi bahwa nama database yang dimasukkan dalam wizard Create Datasource sudah benar, atau tentukan nama cluster dan database yang benar.

Amazon MLtidak dapat mengakses database Anda. Berikan kata sandi yang valid untuk pengguna database<user name>.

Penyebab

Kata sandi yang Anda berikan di wizard Create Datasource untuk memungkinkan Amazon MLmengakses database Amazon Redshift Anda tidak benar.

Solusi

Berikan kata sandi yang benar untuk pengguna database Amazon Redshift Anda.

Terjadi kesalahan saat Amazon MLmencoba memvalidasi kueri.

Penyebab

Ada masalah dengan kueri SQL Anda.

Solusi

Verifikasi bahwa kueri Anda adalah SQL yang valid.

Terjadi kesalahan saat menjalankan kueri SQL Anda. Verifikasi nama database dan query yang disediakan. Akar penyebab: {ServerMessage}.

Penyebab

Amazon Redshift tidak dapat menjalankan kueri Anda.

Solusi

Verifikasi bahwa Anda menentukan nama database yang benar di wizard Create Datasource, dan bahwa kueri Anda adalah SQL yang valid.

Terjadi kesalahan saat menjalankan kueri SQL Anda. Akar penyebab: {ServerMessage}.

Penyebab

Amazon Redshift tidak dapat menemukan tabel yang ditentukan.

Solusi

Pastikan tabel yang Anda tentukan di wizard Create Datasource ada di database klaster Amazon Redshift Anda, dan bahwa Anda memasukkan ID klaster, nama database, dan kueri SQL yang benar.

Menghubungi Support AWS

Jika Anda memiliki Support Premium AWS, Anda dapat membuat kasus dukungan teknis diPusat Dukungan AWS.

Menggunakan Data dari Database Amazon RDS untuk Membuat Amazon MI Datasource

Amazon IL memungkinkan Anda membuat objek datasource dari data yang disimpan dalam basis data MySQL di Amazon Relational Database Service (Amazon RDS). Saat Anda melakukan tindakan ini, Amazon IL membuat objek AWS Data Pipeline yang mengeksekusi kueri SQL yang Anda tetapkan, dan menempatkan output ke bucket S3 pilihan Anda. Amazon IL menggunakan data tersebut untuk membuat sumber data.

> **Note**
>
> Amazon ML-hanya mendukung database MySQL di VPC.

Sebelum Amazon IL dapat membaca data masukan Anda, Anda harus mengekspor data tersebut ke Amazon Simple Storage Service (Amazon S3). Anda dapat mengatur Amazon IL untuk melakukan ekspor untuk Anda dengan menggunakan API. (RDS terbatas pada API, dan tidak tersedia dari konsol.)

Agar Amazon IL dapat terhubung ke database MySQL Anda di Amazon RDS dan membaca data atas nama Anda, Anda harus memberikan yang berikut:

- Pengidentifikasi instans DB RDS
- Nama basis data MySQL
- ParameterAWS Identity and Access Management(IAM) peran yang digunakan untuk membuat, mengaktifkan, dan menjalankan pipa data
- Kredenal pengguna basis data:
  - Nama pengguna

  - Kata sandi
- Informasi keamanan AWS Data Pipeline:
  - Peran sumber daya IAM
  - Peran layanan IAM
- Informasi keamanan Amazon RDS:
  - ID subnet
  - ID grup keamanan
- Query SQL yang menentukan data yang ingin Anda gunakan untuk membuat sumber data
- S3 lokasi output (bucket) yang digunakan untuk menyimpan hasil query
- (Opsional) Lokasi file skema data

Selain itu, Anda perlu memastikan bahwa pengguna IAM atau peran yang membuat sumber data Amazon RDS dengan menggunakanCreateDataSourceFromRDSoperasi memilikiiam:PassRoleizin. Untuk informasi selengkapnya, lihat Mengontrol Akses ke Sumber Daya Amazon ML-dengan IAM.

Topik

- Pengidentifikasi instans Basis Data RDS
- Nama Basis Data MySQL
- Kredenal Pengguna Basis Data
- Informasi Keamanan AWS Data Pipeline
- Informasi Keamanan Amazon
- Kueri MySQL
- Lokasi Output S3

Pengidentifikasi instans Basis Data RDS

Pengenal instans DB RDS adalah nama unik yang Anda suplai yang mengidentifikasi instans database yang harus digunakan Amazon ML-nya saat berinteraksi dengan Amazon RDS. Anda dapat menemukan pengidentifikasi instans DB RDS di konsol Amazon RDS.

Nama Basis Data MySQL

Nama Database MySQL menentukan nama database MySQL di instans DB RDS.

Kredenal Pengguna Basis Data

Untuk menyambung ke instans DB RDS, Anda harus menyediakan nama pengguna dan kata sandi pengguna database yang memiliki izin yang cukup untuk mengeksekusi query SQL yang Anda berikan.

Informasi Keamanan AWS Data Pipeline

Untuk mengaktifkan akses AWS Data Pipeline yang aman, Anda harus memberikan nama peran sumber daya IAM dan peran layanan IAM.

Instans EC2 mengasumsikan peran sumber daya untuk menyalin data dari Amazon RDS ke Amazon S3. Cara termudah untuk membuat peran sumber daya ini adalah dengan menggunakan`DataPipelineDefaultResourceRole`Template, dan daftar**`machinelearning.aws.com`**sebagai layanan terpercaya. Untuk informasi lebih lanjut tentang templat, lihatMenyiapkan Peran IAMdiPanduan Pengembang AWS Data Pipeline.

Jika Anda membuat peran Anda sendiri, itu harus memiliki konten berikut:

```
{
    "Version": "2012-10-17",
    "Statement": [
    {
        "Effect": "Allow",
        "Principal": {
            "Service": "machinelearning.amazonaws.com"
        },
        "Action": "sts:AssumeRole",
        "Condition": {
            "StringEquals": { "aws:SourceAccount": "123456789012" },
            "ArnLike": { "aws:SourceArn": "arn:aws:machinelearning:us-
east-1:123456789012:datasource/*" }
        }
    }]
}
```

AWS Data Pipeline mengasumsikan peran layanan untuk memantau kemajuan menyalin data dari Amazon RDS ke Amazon S3. Cara termudah untuk membuat peran sumber daya ini adalah dengan menggunakan`DataPipelineDefaultRole`Template, dan daftar`machinelearning.aws.com`sebagai layanan terpercaya. Untuk informasi lebih lanjut tentang templat, lihatMenyiapkan Peran IAMdiPanduan Pengembang AWS Data Pipeline.

Informasi Keamanan Amazon

Untuk mengaktifkan akses Amazon RDS yang aman, Anda harus menyediakan`VPC Subnet IDs`dan`RDS Security Group IDs`. Anda juga perlu mengatur aturan masuknya yang sesuai untuk subnet VPC yang ditunjukkan oleh`Subnet ID`parameter, dan memberikan ID dari grup keamanan yang memiliki izin ini.

Kueri MySQL

Parameter`MySQL SQL Query`parameter menentukan query SQL SELECT yang ingin Anda jalankan pada database MySQL Anda. Hasil kueri disalin ke lokasi keluaran S3 (bucket) yang Anda tentukan.

> **Note**
>
> Teknologi pembelajaran mesin bekerja paling baik ketika catatan masukan disajikan dalam urutan acak (dikoyak). Anda dapat dengan mudah mengocokkan hasil kueri MySQL SQL Anda dengan menggunakan`rand()`fungsi. Sebagai contoh, katakanlah bahwa ini adalah kueri asli:
> “SELECT col1, col2,... DARI training_table”
> Anda dapat menambahkan menyeret acak dengan memperbarui kueri seperti ini:
> “SELECT col1, col2,... DARI training_table ORDER OLEH rand ()”

Lokasi Output S3

Parameter`S3 Output Location`parameter menentukan nama lokasi “pementasan” Amazon S3 di mana hasil kueri MySQL SQL adalah output.

> **Note**
>
> Anda perlu memastikan bahwa Amazon IL memiliki izin untuk membaca data dari lokasi ini setelah data diekspor dari Amazon RDS. Untuk informasi tentang menyetel izin ini, lihat Memberikan Izin Amazon ML-Membaca Data Anda dari Amazon S3.

Model ML-Pelatihan

Proses pelatihan model ML-melibatkan penyediaan algoritma ML (yaitu,Algoritma pembelajaran) dengan data pelatihan untuk belajar dari. IstilahModel MLmengacu pada model artefak yang diciptakan oleh proses pelatihan.

Data pelatihan harus berisi jawaban yang benar, yang dikenal sebagaitargetatauatribut target. Algoritma pembelajaran menemukan pola dalam data pelatihan yang memetakan atribut data input ke target (jawaban yang ingin Anda prediksi), dan menghasilkan model ML-nya yang menangkap pola-pola ini.

Anda dapat menggunakan model L untuk mendapatkan prediksi pada data baru yang Anda tidak tahu targetnya. Misalnya, katakanlah Anda ingin melatih model ML untuk memprediksi apakah email adalah spam atau bukan spam. Anda akan memberikan Amazon MLdengan data pelatihan yang berisi email yang Anda tahu target (yaitu, label yang memberitahu apakah email spam atau bukan spam). Amazon ML-nya akan melatih model ML-nya dengan menggunakan data ini, menghasilkan model yang mencoba memprediksi apakah email baru akan menjadi spam atau bukan spam.

Untuk informasi umum tentang model dan algoritma ML, lihatMachine Learning.

Topik

Jenis Model ML/Model

Amazon MLnya mendukung tiga jenis model MLnya: klasifikasi biner, klasifikasi multikelas, dan regresi. Jenis model yang harus Anda pilih tergantung dari jenis target yang ingin Anda prediksi.

Model klasifikasi

Model L untuk masalah klasifikasi biner memprediksi hasil biner (salah satu dari dua kelas yang mungkin). Untuk melatih model klasifikasi biner, Amazon IL menggunakan algoritma pembelajaran standar industri yang dikenal sebagai regresi logistik.

Contoh Masalah Klasifikasi Biner

- “Apakah email ini spam atau bukan spam?”
- “Akankah pelanggan membeli produk ini?”
- “Apakah produk ini buku atau hewan ternak?”
- “Apakah ulasan ini ditulis oleh pelanggan atau robot?”

Model klasifikasi

Model L untuk masalah klasifikasi multiclass memungkinkan Anda menghasilkan prediksi untuk beberapa kelas (memprediksi satu dari lebih dari dua hasil). Untuk melatih model multiclass, Amazon IL menggunakan algoritma pembelajaran standar industri yang dikenal sebagai regresi logistik multinomial.

Contoh Masalah Multiclass

- “Apakah produk ini buku, film, atau pakaian?”
- “Apakah film ini komedi romantis, dokumenter, atau film thriller?”
- “Kategori produk mana yang paling menarik bagi pelanggan ini?”

Model regresi

Model L untuk masalah regresi memprediksi nilai numerik. Untuk model regresi pelatihan, Amazon IL menggunakan algoritma pembelajaran standar industri yang dikenal sebagai regresi linier.

Contoh Masalah Regresi

- “Apa yang akan suhu di Seattle besok?”
- “Untuk produk ini, berapa banyak unit yang akan menjual?”
- “Berapa harga rumah ini akan menjual?”

Proses Pelatihan

Untuk melatih model ML-nya, Anda harus menentukan hal berikut:

- Sumber data pelatihan masukan
- Nama atribut data yang berisi target yang akan diprediksi

- Instruksi transformasi data yang diperlukan
- Parameter pelatihan untuk mengontrol algoritma pembelajaran

Selama proses pelatihan, Amazon XML secara otomatis memilih algoritma pembelajaran yang benar untuk Anda, berdasarkan jenis target yang Anda tentukan dalam sumber data pelatihan.

Parameter Pelatihan

Biasanya, algoritma pembelajaran mesin menerima parameter yang dapat digunakan untuk mengontrol sifat tertentu dari proses pelatihan dan model ML-nya. Dalam Amazon Machine Learning, ini disebutparameter pelatihan. Anda dapat mengatur parameter ini menggunakan konsol Amazon, API, atau antarmuka baris perintah (CLI). Jika Anda tidak menetapkan parameter apa pun, Amazon IL akan menggunakan nilai default yang diketahui berfungsi dengan baik untuk berbagai macam tugas machine learning.

Anda dapat menentukan nilai untuk parameter pelatihan berikut:

- Ukuran model maksimum
- Jumlah maksimum lintasan melalui data pelatihan
- Jenis Shuffle
- Jenis regularisasi
- Jumlah regularisasi

Di konsol Amazon ML-nya, parameter pelatihan ditetapkan secara default. Pengaturan default memadai untuk sebagian besar masalah ML-nya, tetapi Anda dapat memilih nilai lain untuk menyempurnakan kinerja. Beberapa parameter pelatihan lainnya, seperti tingkat pembelajaran, dikonfigurasi untuk Anda berdasarkan data Anda.

Bagian berikut menyediakan informasi lebih lanjut tentang parameter pelatihan.

Ukuran Model Maksimum

Ukuran model maksimum adalah ukuran total, dalam satuan byte, dari pola yang dibuat Amazon ML-nya selama pelatihan model ML-nya.

Secara default, Amazon IL membuat model 100 MB. Anda dapat menginstruksikan Amazon IL untuk membuat model yang lebih kecil atau lebih besar dengan menentukan ukuran yang berbeda. Untuk berbagai ukuran yang tersedia, lihatJenis Model ML/Model

Jika Amazon IL tidak dapat menemukan pola yang cukup untuk mengisi ukuran model, itu akan menciptakan model yang lebih kecil. Misalnya, jika Anda menentukan ukuran model maksimum 100 MB, tetapi Amazon IL menemukan pola yang total hanya 50 MB, model yang dihasilkan akan menjadi 50 MB. Jika Amazon IL menemukan lebih banyak pola daripada yang sesuai dengan ukuran yang ditentukan, Amazon MLM akan memberlakukan cut-off maksimum dengan memangkas pola yang paling tidak memengaruhi kualitas model yang dipelajari.

Memilih ukuran model memungkinkan Anda untuk mengontrol trade-off antara kualitas prediktif model dan biaya penggunaan. Model yang lebih kecil dapat menyebabkan Amazon ML-menghapus banyak pola agar sesuai dengan batas ukuran maksimum, yang memengaruhi kualitas prediksi. Model yang lebih besar, di sisi lain, biaya lebih untuk query untuk prediksi real-time.

> **Note**
>
> Jika Anda menggunakan model ML untuk menghasilkan prediksi real-time, Anda akan dikenakan biaya reservasi kapasitas kecil yang ditentukan oleh ukuran model. Untuk informasi selengkapnya, lihat Harga Amazon MLS.

Set data input yang lebih besar tidak selalu menghasilkan model yang lebih besar karena model menyimpan pola, bukan input data; jika polanya sedikit dan sederhana, model yang dihasilkan akan kecil. Masukan data yang memiliki sejumlah besar atribut mentah (kolom input) atau fitur turunan (output transformasi data Amazon XML) kemungkinan akan memiliki lebih banyak pola yang ditemukan dan disimpan selama proses pelatihan. Memilih ukuran model yang benar untuk data dan masalah Anda paling baik didekati dengan beberapa eksperimen. Log pelatihan model Amazon XML (yang dapat Anda unduh dari konsol atau melalui API) berisi pesan tentang berapa banyak pemangkasan model (jika ada) terjadi selama proses pelatihan, memungkinkan Anda memperkirakan potensi kualitas hit-to-prediction.

Jumlah Maksimum Pass atas Data

Untuk hasil terbaik, Amazon IL mungkin perlu melakukan beberapa pass atas data Anda untuk menemukan pola. Secara default, Amazon IL membuat 10 pass, tetapi Anda dapat mengubah default dengan menetapkan angka hingga 100. Amazon IL melacak kualitas pola (konvergensi model) saat berjalan, dan secara otomatis menghentikan pelatihan ketika tidak ada lagi titik data atau pola untuk ditemukan. Misalnya, jika Anda menetapkan jumlah pass ke 20, tetapi Amazon IL menemukan bahwa tidak ada pola baru yang dapat ditemukan pada akhir 15 pass, maka itu akan menghentikan pelatihan pada 15 pass.

Secara umum, set data dengan hanya beberapa pengamatan biasanya memerlukan lebih banyak melewati data untuk mendapatkan kualitas model yang lebih tinggi. Kumpulan data yang lebih besar sering mengandung banyak titik data serupa, yang menghilangkan kebutuhan akan sejumlah besar lintasan. Dampak memilih lebih banyak data melewati data Anda adalah dua kali lipat: pelatihan model membutuhkan waktu lebih lama, dan harganya lebih mahal.

Tipe Shuffle untuk Data Pelatihan

Di Amazon IL, Anda harus mengacak data pelatihan Anda. Shuffling mencampur urutan data Anda sehingga algoritma SGD tidak menemukan satu jenis data untuk terlalu banyak pengamatan berturut-turut. Misalnya, jika Anda melatih model L untuk memprediksi jenis produk, dan data latihan Anda mencakup jenis produk film, mainan, dan video game, jika Anda menyortir data berdasarkan kolom jenis produk sebelum mengunggahnya, algoritma akan melihat data berdasarkan abjad berdasarkan jenis produk. Algoritma ini melihat semua data Anda untuk film terlebih dahulu, dan model ML-mu mulai mempelajari pola film. Kemudian, ketika model Anda menemukan data tentang mainan, setiap pembaruan yang dibuat algoritma akan sesuai dengan model dengan jenis produk mainan, bahkan jika pembaruan tersebut menurunkan pola yang sesuai dengan film. Ini tiba-tiba beralih dari film ke jenis mainan dapat menghasilkan model yang tidak belajar bagaimana untuk memprediksi jenis produk secara akurat.

Anda harus mengacak data pelatihan Anda bahkan jika Anda memilih opsi split acak ketika Anda membagi sumber data input menjadi bagian pelatihan dan evaluasi. Strategi split acak memilih subset acak dari data untuk setiap sumber data, tetapi tidak mengubah urutan baris dalam sumber data. Untuk informasi selengkapnya tentang membagi data Anda, lihatMemisahkan Data Anda.

Saat Anda membuat model ML-nya menggunakan konsol, Amazon ML-nya akan menyeret data dengan teknik shuffling pseudo-random. Terlepas dari jumlah pass yang diminta, Amazon ML-mengacak data hanya satu kali sebelum melatih model ML-nya. Jika Anda mengacak data sebelum menyediakannya ke Amazon ML. dan tidak ingin Amazon ML-shuffle lagi data Anda, Anda dapat menyetelJenis Shufflekepadanone. Misalnya, jika Anda secara acak mengacak rekaman di file.csv sebelum mengunggahnya ke Amazon S3, menggunakanrand()berfungsi dalam kueri MySQL SQL Anda saat membuat sumber data Anda dari Amazon RDS, atau menggunakanrandom()berfungsi dalam kueri SQL Amazon Redshift Anda saat membuat sumber data Anda dari Amazon Redshift, menyetelJenis Shufflekepadanonetidak akan memengaruhi keakuratan prediktif model ML-mu. Shuffling data Anda hanya sekali mengurangi run-time dan biaya untuk membuat model ML-nya.

> **Important**
>
> Saat Anda membuat model ML-nya menggunakan API Amazon ML-nya, Amazon ML-nya tidak akan mengacak data secara default. Jika Anda menggunakan API alih-alih konsol untuk membuat model ML-mu, kami sangat menyarankan agar Anda mengocokkan data Anda dengan menyetelsgd.shuffleTypeparameter untukauto.

Jenis dan Jumlah Regularisasi

Kinerja prediktif model ML-kompleks (mereka yang memiliki banyak atribut input) menderita ketika data berisi terlalu banyak pola. Sebagai jumlah pola meningkat, begitu juga kemungkinan bahwa model belajar artefak data yang tidak disengaja, bukan pola data yang benar. Dalam kasus seperti itu, model sangat baik pada data pelatihan, tetapi tidak dapat menggeneralisasi data baru. Fenomena ini dikenal sebagaioverfittingdata pelatihan.

Regularisasi membantu mencegah model linier dari contoh data pelatihan yang terlalu pas dengan menghukum nilai berat ekstrem. L1 regularisasi mengurangi jumlah fitur yang digunakan dalam model dengan mendorong berat fitur yang jika tidak akan memiliki bobot yang sangat kecil menjadi nol. L1 regularisasi menghasilkan model jarang dan mengurangi jumlah kebisingan dalam model. L2 regularisasi menghasilkan nilai bobot keseluruhan yang lebih kecil, yang menstabilkan bobot ketika ada korelasi tinggi antara fitur. Anda dapat mengontrol jumlah L1 atau L2 regularisasi dengan menggunakan`Regularization amount`parameter. Menentukan sangat besar`Regularization amount`dapat menyebabkan semua fitur memiliki berat nol.

Memilih dan menyetel nilai regularisasi optimal adalah subjek aktif dalam penelitian pembelajaran mesin. Anda mungkin akan mendapatkan keuntungan dari memilih regularisasi L2 dalam jumlah moderat, yang merupakan default di konsol Amazon ML-nya. Pengguna tingkat lanjut dapat memilih antara tiga jenis regularisasi (none, L1, atau L2) dan jumlah. Untuk informasi lebih lanjut tentang regularisasi, kunjungiRegularisasi (matematika).

Parameter Pelatihan: Jenis dan Nilai Default

Tabel berikut mencantumkan parameter pelatihan Amazon ML, bersama dengan nilai default dan rentang yang diijinkan untuk masing-masing.

Parameter Pelatihan	Jenis	Nilai Default	Deskripsi
MaxMLModelSizeinBytes	Bulat	100.000.000 byte (100 MiB)	Rentang yang diizinkan: 100,000 (100 KiB) untuk 2,147,483,648 (2 GiB) Tergantung pada data input, ukuran model dapat mempengaruhi kinerja.
SGD.maxPasses	Bulat	10	Rentang yang diizinkan: 1-100
SGD.shuffletype	String	mobil	Nilai yang diizinkan:`auto`ataunone
SGD.l1RegularizationAmount	Double	0 (Secara default, L1 tidak digunakan)	Rentang yang diijinkan: 0 ke MAX_DOUBL Nilai L1 antara 1E-4 dan 1E-8 telah ditemukan untuk menghasilkan hasil yang baik. Nilai yang lebih besar cenderung menghasilkan model yang tidak terlalu berguna. Anda tidak dapat mengatur L1 dan L2. Anda harus memilih satu atau yang lainnya.
SGD.l2RegularizationAmount	Double	1E-6 (Secara default, L2 digunakan dengan jumlah regularisasi ini)	Rentang yang diijinkan: 0 ke MAX_DOUBL Nilai L2 antara 1E-2 dan 1E-6 telah ditemukan untuk menghasilkan hasil yang baik. Nilai yang lebih besar cenderung menghasilkan model yang tidak terlalu berguna. Anda tidak dapat mengatur L1 dan L2. Anda harus memilih satu atau yang lainnya.

Membuat Model ML-nya

Setelah membuat sumber data, Anda siap untuk membuat model ML. Jika Anda menggunakan konsol Amazon Machine Learning untuk membuat model, Anda dapat memilih untuk menggunakan pengaturan default atau menyesuaikan model dengan menerapkan opsi kustom.

Pilihan kustom meliputi:

- Pengaturan evaluasi: Anda dapat memilih untuk memiliki Amazon ML-cadangan sebagian dari data input untuk mengevaluasi kualitas prediktif model ML-nya. Untuk informasi tentang evaluasi, lihatMengevaluasi Model ML.
- Resep: Resep memberi tahu Amazon ML-atribut dan transformasi atribut yang tersedia untuk pelatihan model. Untuk informasi tentang resep Amazon, lihatFitur Transformasi dengan Data Recipes.
- Parameter pelatihan: Parameter mengontrol sifat tertentu dari proses pelatihan dan model ML-nya. Untuk informasi selengkapnya tentang parameter pelatihan, lihatParameter Pelatihan.

Untuk memilih atau menentukan nilai untuk pengaturan ini, pilihKhususpilihan ketika Anda menggunakan Wisaya Buat Model L. Jika Anda ingin Amazon ML-menerapkan pengaturan default, pilihDefault.

Saat Anda membuat model ML-nya, Amazon MLakan memilih jenis algoritma pembelajaran yang akan digunakan berdasarkan jenis atribut atribut target Anda. (Atribut target adalah atribut yang berisi jawaban “benar”.) Jika atribut target Anda adalah Binary, Amazon ML-membuat model klasifikasi biner, yang menggunakan algoritma regresi logistik. Jika atribut target Anda adalah Kategoris, Amazon ML-membuat model multiclass, yang menggunakan algoritma regresi logistik multinomial. Jika atribut target Anda adalah Numeric, Amazon ML-membuat model regresi, yang menggunakan algoritma regresi linier.

Topik

Prasyarat

Sebelum menggunakan konsol Amazon ML-untuk membuat model, Anda perlu membuat dua sumber data, satu untuk melatih model dan satu untuk mengevaluasi model. Jika Anda belum membuat dua sumber data, lihatLangkah 2: Membuat Pelatihan Datasourcedalam tutorial.

Membuat Model ML-nya dengan Opsi Default

PilihDefaultpilihan, jika Anda ingin Amazon ML-nya:

- Membagi data input untuk menggunakan 70 persen pertama untuk pelatihan dan menggunakan 30 persen sisanya untuk evaluasi
- Sarankan resep berdasarkan statistik yang dikumpulkan pada sumber data pelatihan, yaitu 70 persen dari sumber data masukan
- Pilih parameter latihan default

Untuk memilih opsi default

1. Di konsol Amazon ML-pilihAmazon Machine Learning, dan kemudian pilihModel L.
2. PadaModel Lhalaman ringkasan, pilihBuat model ML-baru.
3. PadaData inputhalaman, pastikan bahwaSaya sudah membuat datasource yang menunjuk ke data S3 sayadipilih.
4. Dalam tabel, pilih sumber data Anda, lalu pilihLanjutkan.
5. PadaPengaturan model Lhalaman, untukNama model L, ketik nama untuk model ML-mu.
6. UntukPengaturan pelatihan dan evaluasi, pastikan bahwaDefaultdipilih.
7. UntukBeri nama evaluasi, ketik nama untuk evaluasi, dan kemudian pilihTinjau. Amazon ML-bypass sisa wizard dan membawa Anda keTinjauhalaman.
8. Tinjau data Anda, hapus tag yang disalin dari sumber data yang tidak ingin diterapkan pada model dan evaluasi, lalu pilihSelesai.

Membuat Model ML-nya dengan Opsi Kustom

Menyesuaikan model ML-mu memungkinkan Anda untuk:

- Berikan resep Anda sendiri. Untuk informasi tentang cara menyediakan resep Anda sendiri, lihatReferensi Format.

- Pilih parameter pelatihan. Untuk informasi selengkapnya tentang parameter pelatihan, lihatParameter Pelatihan.
- Pilih rasio pemisahan pelatihan/evaluasi selain rasio 70/30 default atau berikan sumber data lain yang telah Anda siapkan untuk evaluasi. Untuk informasi tentang strategi pemecahan, lihatMemisahkan Data Anda.

Anda juga dapat memilih nilai default untuk pengaturan ini.

Jika Anda telah membuat model menggunakan opsi default dan ingin meningkatkan kinerja prediktif model Anda, gunakanKhususpilihan untuk membuat model baru dengan beberapa pengaturan disesuaikan. Misalnya, Anda dapat menambahkan lebih banyak transformasi fitur ke resep atau meningkatkan jumlah pass dalam parameter pelatihan.

Untuk membuat model dengan opsi khusus

1. Di konsol Amazon ML-pilihAmazon Machine Learning, dan kemudian pilihModel L.
2. PadaModel Lhalaman ringkasan, pilihBuat model ML-baru.
3. Jika Anda telah membuat sumber data, padaData inputhalaman, pilihSaya sudah membuat datasource yang menunjuk ke data S3 saya. Dalam tabel, pilih sumber data Anda, lalu pilihLanjutkan.

   Jika Anda perlu membuat sumber data, pilihData saya ada di S3, dan saya perlu membuat sumber data, pilihLanjutkan. Anda dialihkan keBuat DatasourcePenyihir. Tentukan apakah data Anda adaS3atauRedshift, lalu pilihVerifikasi. Lengkapi prosedur untuk membuat sumber data.

   Setelah Anda membuat datasource, Anda akan diarahkan ke langkah berikutnya dalamBuat Model MLPenyihir.
4. PadaPengaturan model Lhalaman, untukNama model L, ketik nama untuk model ML-mu.
5. MasukPilih pengaturan pelatihan dan evaluasi, pilihKhusus, dan kemudian pilihLanjutkan.
6. PadaResephalaman, Anda bisacustomize a recipe. Jika Anda tidak ingin menyesuaikan resep, Amazon ML-menyarankan satu untuk Anda. Pilih Continue (Lanjutkan).
7. PadaPengaturan lanjutanhalaman, tentukanUkuran model maksimum, yangJumlah maksimum data, yangJenis shuffle untuk data pelatihan, yangJenis regularisasi, danJumlah regularisasi. Jika Anda tidak menentukan ini, Amazon MLakan menggunakan parameter pelatihan default.

   Untuk informasi selengkapnya tentang parameter ini dan defaultnya, lihatParameter Pelatihan.

Pilih Continue (Lanjutkan).

8. PadaEvaluasihalaman, tentukan apakah Anda ingin mengevaluasi model ML-nya segera. Jika Anda tidak ingin mengevaluasi model MLnya sekarang, pilihTinjau.

   Jika Anda ingin mengevaluasi model ML-nya sekarang:

   a. UntukBeri nama evaluasi, ketik nama untuk evaluasi.
   b. UntukPilih data evaluasi, pilih apakah Anda ingin Amazon XML memesan sebagian data input untuk evaluasi dan, jika Anda melakukannya, bagaimana Anda ingin membagi sumber data, atau memilih untuk menyediakan sumber data yang berbeda untuk evaluasi.
   c. Pilih Tinjau.

9. PadaTinjauhalaman, edit pilihan Anda, hapus tag apa pun yang disalin dari sumber data yang tidak ingin Anda terapkan pada model dan evaluasi Anda, lalu pilihSelesai.

Setelah Anda membuat model, lihatLangkah 4: Tinjau Kinerja Prediktif Model L dan Tetapkan Ambang Skor.

Transformasi Data untuk Machine Learning

Model pembelajaran mesin hanya sebagus data yang digunakan untuk melatihnya. Karakteristik utama dari data pelatihan yang baik adalah bahwa hal itu disediakan dengan cara yang dioptimalkan untuk pembelajaran dan generalisasi. Proses menyusun data dalam format optimal ini dikenal di industri sebagaitransformasi fitur.

Topik

Pentingnya Transformasi Fitur

Pertimbangkan model pembelajaran mesin yang tugasnya adalah untuk memutuskan apakah transaksi kartu kredit adalah penipuan atau tidak. Berdasarkan pengetahuan latar belakang aplikasi dan analisis data, Anda mungkin memutuskan bidang data (atau fitur) mana yang penting untuk disertakan dalam data input. Misalnya, jumlah transaksi, nama pedagang, alamat, dan alamat pemilik kartu kredit penting untuk diberikan kepada proses pembelajaran. Di sisi lain, ID transaksi yang dihasilkan secara acak tidak membawa informasi (jika kita tahu bahwa itu benar-benar acak), dan tidak berguna.

Setelah Anda memutuskan bidang mana yang akan disertakan, Anda mengubah fitur ini untuk membantu proses pembelajaran. Transformasi menambah pengalaman latar belakang ke data input, memungkinkan model pembelajaran mesin untuk mendapatkan keuntungan dari pengalaman ini. Sebagai contoh, alamat merchant berikut direpresentasikan sebagai string:

“123 Jalan Utama, Seattle, WA 98101"

Dengan sendirinya, alamat memiliki kekuatan ekspresif terbatas - ini hanya berguna untuk pola belajar yang terkait dengan alamat yang tepat. Memecahnya menjadi bagian-bagian penyusunnya, bagaimanapun, dapat membuat fitur tambahan seperti “Alamat” (123 Main Street), “Kota” (Seattle),

“Negara” (WA) dan “Zip” (98101). Sekarang, algoritma pembelajaran dapat mengelompokkan transaksi yang lebih berbeda bersama-sama, dan menemukan pola yang lebih luas - mungkin beberapa kode pos pedagang mengalami aktivitas yang lebih curang daripada yang lain.

Untuk informasi selengkapnya tentang pendekatan dan proses transformasi fitur, lihatKonsep Machine Learning.

Fitur Transformasi dengan Data Recipes

Ada dua cara untuk mengubah fitur sebelum membuat model ML-nya dengan Amazon IL: Anda dapat mengubah data input Anda secara langsung sebelum menunjukkannya ke Amazon IL, atau Anda dapat menggunakan transformasi data bawaan Amazon ML. Anda dapat menggunakan resep Amazon ML-nya, yang merupakan instruksi yang telah diformat sebelumnya untuk transformasi umum. Dengan resep, Anda dapat melakukan hal berikut ini:

- Pilih dari daftar transformasi pembelajaran mesin umum bawaan, dan terapkan ini ke variabel atau kelompok variabel individual
- Pilih variabel input dan transformasi mana yang tersedia untuk proses pembelajaran mesin

Menggunakan resep Amazon ML-menawarkan beberapa keuntungan. Amazon ML-nya melakukan transformasi data untuk Anda, jadi Anda tidak perlu menerapkannya sendiri. Selain itu, mereka cepat karena Amazon IL menerapkan transformasi saat membaca data input, dan memberikan hasil untuk proses pembelajaran tanpa langkah menengah menyimpan hasil ke disk.

Referensi resep

Resep Amazon XML berisi petunjuk untuk mengubah data Anda sebagai bagian dari proses machine learning. Resep didefinisikan menggunakan sintaks seperti JSON, tetapi mereka memiliki batasan tambahan di luar pembatasan JSON normal. Resep memiliki bagian berikut, yang harus muncul dalam urutan yang ditunjukkan di sini:

- Grupmemungkinkan pengelompokan beberapa variabel, untuk kemudahan menerapkan transformasi. Misalnya, Anda dapat membuat sekelompok semua variabel yang harus dilakukan dengan bagian teks bebas dari halaman web (judul, tubuh), dan kemudian melakukan transformasi pada semua bagian ini sekaligus.
- Tugasmemungkinkan penciptaan variabel bernama menengah yang dapat digunakan kembali dalam pengolahan.

- Keluaranmendefinisikan variabel yang akan digunakan dalam proses pembelajaran, dan apa transformasi (jika ada) berlaku untuk variabel ini.

Grup

Anda dapat menentukan kelompok variabel untuk secara kolektif mengubah semua variabel dalam kelompok, atau menggunakan variabel ini untuk pembelajaran mesin tanpa mengubahnya. Secara default, Amazon IL membuat grup berikut untuk Anda:

ALL_TEXT, ALL_NUMERIC, ALL_CATEGORICAL, ALL_BINARY -Kelompok tipe-spesifik berdasarkan variabel yang didefinisikan dalam skema datasource.

> **Note**
>
> Anda tidak dapat membuat grup dengan`ALL_INPUTS`.

Variabel ini dapat digunakan di bagian output resep Anda tanpa didefinisikan. Anda juga dapat membuat grup kustom dengan menambahkan atau mengurangi variabel dari grup yang ada, atau langsung dari kumpulan variabel. Dalam contoh berikut, kita menunjukkan ketiga pendekatan, dan sintaks untuk tugas pengelompokan:

```
"groups": {

"Custom_Group": "group(var1, var2)",
"All_Categorical_plus_one_other": "group(ALL_CATEGORICAL, var2)"

}
```

Nama grup harus dimulai dengan karakter abjad dan dapat antara 1 dan 64 karakter panjang. Jika nama grup tidak dimulai dengan karakter abjad atau jika mengandung karakter khusus (, "'\ t\ r\ n ()\), maka nama tersebut perlu dikutip untuk dimasukkan dalam resep.

Tugas

Anda dapat menugaskan satu atau lebih variabel menengah, untuk kenyamanan dan keterbacaan. Misalnya, jika Anda memiliki variabel teks bernama email_subject, dan Anda menerapkan

transformasi huruf kecil padanya, Anda dapat memberi nama variabel email_subject_lowercase yang dihasilkan, sehingga mudah untuk melacaknya di tempat lain dalam resep. Tugas juga dapat dirantai, memungkinkan Anda untuk menerapkan beberapa transformasi dalam urutan tertentu. Contoh berikut menunjukkan tugas tunggal dan dirantai dalam sintaks resep:

```
"assignments": {

"email_subject_lowercase": "lowercase(email_subject)",

"email_subject_lowercase_ngram":"ngram(lowercase(email_subject), 2)"

}
```

Nama variabel menengah harus dimulai dengan karakter alfabet dan dapat antara 1 dan 64 karakter panjang. Jika nama tidak dimulai dengan alfabet atau jika mengandung karakter khusus (, "'\ t\ r\ n ()\), maka nama harus dikutip untuk dimasukkan dalam resep.

Output

Bagian output mengontrol variabel input mana yang akan digunakan untuk proses pembelajaran, dan transformasi mana yang berlaku untuk mereka. Bagian output kosong atau tidak ada adalah kesalahan, karena tidak ada data yang akan diteruskan ke proses pembelajaran.

Bagian output yang paling sederhana hanya mencakup yang telah ditetapkanALL_INPUTkelompok, menginstruksikan Amazon XML untuk menggunakan semua variabel yang didefinisikan dalam sumber data untuk pembelajaran:

```
"outputs": [

"ALL_INPUTS"

]
```

Bagian output juga dapat merujuk ke grup yang telah ditetapkan lainnya dengan menginstruksikan Amazon IL untuk menggunakan semua variabel dalam grup ini:

```
"outputs": [
```

```
"ALL_NUMERIC",

"ALL_CATEGORICAL"

]
```

Bagian output juga dapat merujuk ke kelompok kustom. Pada contoh berikut, hanya satu dari kelompok kustom yang didefinisikan dalam bagian penugasan pengelompokan dalam contoh sebelumnya yang akan digunakan untuk pembelajaran mesin. Semua variabel lain akan dijatuhkan:

```
"outputs": [

"All_Categorical_plus_one_other"

]
```

Bagian output juga dapat merujuk ke tugas variabel yang didefinisikan dalam bagian penugasan:

```
"outputs": [

"email_subject_lowercase"

]
```

Dan variabel input atau transformasi dapat didefinisikan langsung di bagian output:

```

"outputs": [

"var1",

"lowercase(var2)"

]
```

Output perlu secara eksplisit menentukan semua variabel dan variabel yang berubah yang diharapkan tersedia untuk proses pembelajaran. Katakanlah, misalnya, bahwa Anda termasuk dalam output produk Cartesian var1 dan var2. Jika Anda ingin memasukkan variabel mentah var1 dan var2 juga, maka Anda perlu menambahkan variabel mentah di bagian output:

```
"outputs": [

"cartesian(var1,var2)",

"var1",

"var2"

]
```

Output dapat mencakup komentar untuk dibaca dengan menambahkan teks komentar bersama dengan variabel:

```
"outputs": [

"quantile_bin(age, 10) //quantile bin age",

"age // explicitly include the original numeric variable along with the
binned version"

]
```

Anda dapat mencampur dan mencocokkan semua pendekatan ini dalam bagian output.

> **Note**
>
> Komentar tidak diizinkan di konsol Amazon XML saat menambahkan resep.

Lengkapi Contoh

Contoh berikut mengacu pada beberapa prosesor data built-in yang diperkenalkan dalam contoh sebelumnya:

```
{

"groups": {
```

```
"LONGTEXT": "group_remove(ALL_TEXT, title, subject)",

"SPECIALTEXT": "group(title, subject)",

"BINCAT": "group(ALL_CATEGORICAL, ALL_BINARY)"

},

"assignments": {

"binned_age" : "quantile_bin(age,30)",

"country_gender_interaction" : "cartesian(country, gender)"

},

"outputs": [

"lowercase(no_punct(LONGTEXT))",

"ngram(lowercase(no_punct(SPECIALTEXT)),3)",

"quantile_bin(hours-per-week, 10)",

"hours-per-week // explicitly include the original numeric variable
along with the binned version",

"cartesian(binned_age, quantile_bin(hours-per-week,10)) // this one is
critical",

"country_gender_interaction",

"BINCAT"

]

}
```

Resep yang Disarankan

Saat Anda membuat datasource baru di Amazon ML. dan statistik dihitung untuk sumber data tersebut, Amazon LL juga akan membuat resep yang disarankan yang dapat digunakan untuk

membuat model ML-baru dari sumber data. Sumber data yang disarankan didasarkan pada data dan atribut target yang ada dalam data, dan menyediakan titik awal yang berguna untuk membuat dan menyempurnakan model ML-mu.

Untuk menggunakan resep yang disarankan di konsol Amazon ML-nya, pilihSumber dataatauModel MLMdariBuat barudaftar drop down. Untuk pengaturan model ML, Anda akan memiliki pilihan pengaturan Default atau Custom Training and Evaluation diPengaturan Model MLMS langkahBuat Model MLMPenyihir. Jika Anda memilih opsi Default, Amazon L akan secara otomatis menggunakan resep yang disarankan. Jika Anda memilih opsi Custom, editor resep di langkah berikutnya akan menampilkan resep yang disarankan, dan Anda akan dapat memverifikasi atau memodifikasinya sesuai kebutuhan.

> **Note**
>
> Amazon XML memungkinkan Anda untuk membuat sumber data dan kemudian segera menggunakannya untuk membuat model ML-nya, sebelum perhitungan statistik selesai. Dalam hal ini, Anda tidak akan dapat melihat resep yang disarankan dalam opsi Custom, tetapi Anda masih dapat melanjutkan langkah itu dan memiliki Amazon ML-nya menggunakan resep default untuk pelatihan model.

Untuk menggunakan resep yang disarankan dengan API Amazon XML, Anda dapat meneruskan string kosong di parameter API Recipe dan recipeURI. Tidak mungkin untuk mengambil resep yang disarankan menggunakan API Amazon ML-nya.

Referensi Transformasi Data

Topik

- Transformasi N-gram
- Transformasi Orthogonal Sparse Bigram (OSB)
- Transformasi huruf kecil
- Hapus Transformasi Tanda baca
- Transformasi Binning
- Transformasi Normalisasi
- Transformasi Produk Cartesian

Transformasi N-gram

Transformasi n-gram mengambil variabel teks sebagai input dan menghasilkan string yang sesuai dengan menggeser jendela (user-configurable) n kata, menghasilkan output dalam proses. Misalnya, perhatikan string teks “Saya sangat menikmati membaca buku ini”.

Menentukan transformasi n-gram dengan ukuran jendela = 1 hanya memberi Anda semua kata individual dalam string itu:

```
{"I", "really", "enjoyed", "reading", "this", "book"}
```

Menentukan transformasi n-gram dengan ukuran jendela =2 memberi Anda semua kombinasi dua kata serta kombinasi satu kata:

```
{"I really", "really enjoyed", "enjoyed reading", "reading this", "this
book", "I", "really", "enjoyed", "reading", "this", "book"}
```

Menentukan transformasi n-gram dengan ukuran jendela = 3 akan menambahkan kombinasi tiga kata ke daftar ini, menghasilkan hal berikut:

```
{"I really enjoyed", "really enjoyed reading", "enjoyed reading this",
"reading this book", "I really", "really enjoyed", "enjoyed reading",
"reading this", "this book", "I", "really", "enjoyed", "reading",
"this", "book"}
```

Anda dapat meminta n-gram dengan ukuran mulai dari 2-10 kata. N-gram dengan ukuran 1 dihasilkan secara implisit untuk semua input yang jenisnya ditandai sebagai teks dalam skema data, sehingga Anda tidak perlu memintanya. Akhirnya, perlu diingat bahwa n-gram dihasilkan dengan melanggar data input pada karakter spasi. Itu berarti bahwa, misalnya, karakter tanda baca akan dianggap sebagai bagian dari token kata: menghasilkan n-gram dengan jendela 2 untuk string “merah, hijau, biru” akan menghasilkan {"merah,”, “hijau,”, “biru,”, “merah, hijau”, “hijau, biru"}. Anda dapat menggunakan prosesor penghilang tanda baca (dijelaskan nanti dalam dokumen ini) untuk menghapus simbol tanda baca jika ini bukan yang Anda inginkan.

Untuk menghitung n-gram ukuran jendela 3 untuk variabel var1:

```
"ngram(var1, 3)"
```

Transformasi Orthogonal Sparse Bigram (OSB)

Transformasi OSB dimaksudkan untuk membantu dalam analisis string teks dan merupakan alternatif untuk transformasi bi-gram (n-gram dengan ukuran jendela 2). OSB dihasilkan dengan menggeser jendela ukuran n di atas teks, dan mengeluarkan setiap pasangan kata yang menyertakan kata pertama di jendela.

Untuk membangun setiap OSB, kata-kata penyusunnya digabungkan dengan karakter “_” (garis bawah), dan setiap token yang dilewati ditunjukkan dengan menambahkan garis bawah lain ke OSB. Dengan demikian, OSB mengkodekan bukan hanya token yang terlihat di dalam jendela, tetapi juga indikasi jumlah token yang dilewati dalam jendela yang sama.

Untuk mengilustrasikan, pertimbangkan string “Rubah coklat cepat melompati dog malas”, dan OSBs ukuran 4. Enam jendela empat kata, dan dua jendela terakhir yang lebih pendek dari akhir string ditunjukkan dalam contoh berikut, serta OSBs yang dihasilkan dari masing-masing:

Jendela, {OSBs dihasilkan}

```
"The quick brown fox", {The_quick, The__brown, The___fox}

"quick brown fox jumps", {quick_brown, quick__fox, quick___jumps}

"brown fox jumps over", {brown_fox, brown__jumps, brown___over}

"fox jumps over the", {fox_jumps, fox__over, fox___the}

"jumps over the lazy", {jumps_over, jumps__the, jumps___lazy}

"over the lazy dog", {over_the, over__lazy, over___dog}

"the lazy dog", {the_lazy, the__dog}

"lazy dog", {lazy_dog}
```

Orthogonal bigrams jarang adalah alternatif untuk n-gram yang mungkin bekerja lebih baik dalam beberapa situasi. Jika data Anda memiliki bidang teks besar (10 kata atau lebih), bereksperimen untuk melihat mana yang bekerja lebih baik. Perhatikan bahwa apa yang merupakan bidang teks besar dapat bervariasi tergantung pada situasi. Namun, dengan bidang teks yang lebih besar, OSBs

telah ditunjukkan secara empiris untuk mewakili teks secara unik karena khususmelewatkansimbol (garis bawah).

Anda dapat meminta ukuran jendela 2 sampai 10 untuk transformasi OSB pada variabel teks input.

Untuk menghitung OSB dengan ukuran jendela 5 untuk variabel var1:

“osb (var1, 5)”

Transformasi huruf kecil

Prosesor transformasi huruf kecil mengubah input teks menjadi huruf kecil. Misalnya, mengingat input “The Quick Brown Fox Jumps Over the Lazy Dog”, prosesor akan menampilkan “rubah coklat cepat melompati dog malas”.

Untuk menerapkan transformasi huruf kecil ke variabel var1:

“huruf kecil (var1)”

Hapus Transformasi Tanda baca

Amazon IL secara implisit membagi input yang ditandai sebagai teks dalam skema data pada spasi. Tanda baca dalam string berakhir baik token kata yang berdampingan, atau sebagai token terpisah seluruhnya, tergantung pada spasi yang mengelilinginya. Jika ini tidak diinginkan, transformasi penghilang tanda baca dapat digunakan untuk menghapus simbol tanda baca dari fitur yang dihasilkan. Misalnya, mengingat string “Selamat datang di AML - silakan kencangkan sabuk pengaman Anda!” , set token berikut ini secara implisit dihasilkan:

```
{"Welcome", "to", "Amazon", "ML", "-", "please", "fasten", "your", "seat-belts!"}
```

Menerapkan prosesor penghilang tanda baca ke string ini menghasilkan set ini:

```
{"Welcome", "to", "Amazon", "ML", "please", "fasten", "your", "seat-belts"}
```

Perhatikan bahwa hanya tanda baca awalan dan akhiran yang dihapus. Tanda baca yang muncul di tengah token, misalnya tanda hubung di “sabuk pengaman”, tidak dilepas.

Untuk menerapkan penghapusan tanda baca ke variabel var1:

“no_punct (var1)”

Transformasi Binning

Prosesor binning mengambil dua input, variabel numerik dan parameter yang disebutnomor bin, dan output variabel kategoris. Tujuannya adalah untuk menemukan non-linearitas dalam distribusi variabel dengan mengelompokkan nilai-nilai yang diamati bersama-sama.

Dalam banyak kasus, hubungan antara variabel numerik dan target tidak linear (nilai variabel numerik tidak meningkat atau menurun secara monoton dengan target). Dalam kasus seperti itu, mungkin berguna untuk bin fitur numerik ke dalam fitur kategoris yang mewakili rentang yang berbeda dari fitur numerik. Setiap nilai fitur kategoris (bin) kemudian dapat dimodelkan sebagai memiliki hubungan linier sendiri dengan target. Misalnya, katakanlah Anda tahu bahwa fitur numerik kontinuaccount_agetidak berkorelasi linear dengan kemungkinan untuk membeli buku. Anda dapat bin usia ke fitur kategoris yang mungkin dapat menangkap hubungan dengan target lebih akurat.

Prosesor binning kuantil dapat digunakan untuk menginstruksikan Amazon FL untuk membuat n bin dengan ukuran yang sama berdasarkan distribusi semua nilai input dari variabel usia, dan kemudian mengganti setiap nomor dengan token teks yang berisi bin. Jumlah optimum sampah untuk variabel numerik tergantung pada karakteristik variabel dan hubungannya dengan target, dan ini paling baik ditentukan melalui eksperimen. Amazon IL menyarankan nomor bin optimal untuk fitur numerik berdasarkan statistik data diResep yang disarankan.

Anda dapat meminta antara 5 dan 1000 sampah kuantil untuk dihitung untuk setiap variabel input numerik.

Untuk contoh berikut menunjukkan bagaimana menghitung dan menggunakan 50 sampah di tempat variabel numerik var1:

“quantile_bin (var1, 50)”

Transformasi Normalisasi

Transformator normalisasi menormalkan variabel numerik untuk memiliki rata-rata nol dan varians satu. Normalisasi variabel numerik dapat membantu proses pembelajaran jika ada perbedaan rentang yang sangat besar antara variabel numerik karena variabel dengan besarnya tertinggi bisa mendominasi model ML-nya, tidak peduli apakah fitur ini informatif sehubungan dengan target atau tidak.

Untuk menerapkan transformasi ini ke variabel numerik var1, tambahkan ini ke resep:

menormalkan (var1)

Transformator ini juga dapat mengambil kelompok didefinisikan pengguna variabel numerik atau kelompok yang telah ditentukan untuk semua variabel numerik (ALL_NUMERIC) sebagai masukan:

menormalkan (ALL_NUMERIC)

Catatan

Hal initidakwajib untuk menggunakan prosesor normalisasi untuk variabel numerik.

Transformasi Produk Cartesian

Transformasi Cartesian menghasilkan permutasi dari dua atau lebih teks atau variabel input kategoris. Transformasi ini digunakan ketika interaksi antara variabel dicurigai. Misalnya, pertimbangkan dataset pemasaran bank yang digunakan dalam Tutorial: Menggunakan Amazon IL untuk Memprediksi Respons terhadap Penawaran Pemasaran. Dengan menggunakan dataset ini, kami ingin memprediksi apakah seseorang akan menanggapi promosi bank secara positif, berdasarkan informasi ekonomi dan demografis. Kita mungkin menduga bahwa tipe pekerjaan seseorang agak penting (mungkin ada korelasi antara dipekerjakan di bidang-bidang tertentu dan memiliki uang yang tersedia), dan tingkat pendidikan tertinggi yang dicapai juga penting. Kita mungkin juga memiliki intuisi yang lebih dalam bahwa ada sinyal kuat dalam interaksi kedua variabel ini—misalnya, bahwa promosi ini sangat cocok untuk pelanggan yang merupakan pengusaha yang mendapatkan gelar sarjana universitas.

Transformasi produk Cartesian mengambil variabel kategoris atau teks sebagai masukan, dan menghasilkan fitur baru yang menangkap interaksi antara variabel input ini. Secara khusus, untuk setiap contoh pelatihan, itu akan menciptakan kombinasi fitur, dan menambahkannya sebagai fitur mandiri. Misalnya, katakanlah baris input yang disederhanakan terlihat seperti ini:

target, pendidikan, pekerjaan

0, university.degree, teknisi

0, high.school, layanan

1, university.degree, admin

Jika kita menentukan bahwa transformasi Cartesian akan diterapkan pada variabel kategoris pendidikan dan bidang pekerjaan, fitur yang dihasilkan education_job_interaction akan terlihat seperti ini:

target, education_job_interaction

0, university.degree_technician

0, high.school_services

1, university.degree_admin

Transformasi Cartesian bahkan lebih kuat ketika datang untuk bekerja pada urutan token, seperti halnya ketika salah satu argumennya adalah variabel teks yang secara implisit atau eksplisit dibagi menjadi token. Misalnya, pertimbangkan tugas mengklasifikasikan buku sebagai buku teks atau tidak. Secara intuitif, kita mungkin berpikir bahwa ada sesuatu tentang judul buku yang dapat memberi tahu kita bahwa itu adalah buku teks (kata-kata tertentu mungkin lebih sering terjadi dalam judul buku teks), dan kita mungkin juga berpikir bahwa ada sesuatu tentang pengikatan buku yang prediktif (buku teks lebih mungkin menjadi hardcover), tapi itu benar-benar kombinasi dari beberapa kata dalam judul dan mengikat yang paling prediktif. Untuk contoh dunia nyata, tabel berikut menunjukkan hasil penerapan prosesor Cartesian ke variabel input yang mengikat dan judul:

Buku Teks	Judul	Mengikat	Produk Cartesian dari no_punct (Judul) dan Binding
1	Ekonomi: Prinsip, Masalah, Kebijakan	Hardcover	{"Economics_Hardcover”, “Prinsip_Hardcover”, “Problems_Hardcover”, “Policies_Hardcover"}
0	Hati Tak Terlihat: Romance Ekonomi	Softcover	{"The_Softcover”, “Invisible_Softcover”, “Heart_Softcover”, “An_Softcover”, “Economics_Softcover”, “Romance_Softcover"}
0	Menyenangkan Dengan Masalah	Softcover	{"Fun_Softcover”, “With_Softcover”, “Problems_Softcover"}

Contoh berikut menunjukkan cara menerapkan transformator Cartesian ke var1 dan var2:

cartesian (var1, var2)

Penataan Data

Fungsi penataan ulang data memungkinkan Anda untuk membuat sumber data yang didasarkan pada hanya sebagian dari data input yang ditunjukkannya. Misalnya, ketika Anda membuat Model L menggunakanBuat Model MLwizard di konsol Amazon ML-nya, dan pilih opsi evaluasi

default, Amazon IL secara otomatis menyimpan 30% data Anda untuk evaluasi model ML-nya, dan menggunakan 70% lainnya untuk pelatihan. Fungsionalitas ini diaktifkan oleh fitur Penataan Ulang Data Amazon ML-nya.

Jika Anda menggunakan API Amazon XML untuk membuat sumber data, Anda dapat menentukan bagian mana dari data input yang akan didasarkan pada sumber data baru. Anda melakukan ini dengan melewati instruksi di`DataRearrangement`parameter ke`CreateDataSourceFromS3`,`CreateDataSourceFromRedshift`atau`CreateDataSourceFromRDSA`
Isi dari string DataArrangement adalah string JSON yang berisi awal dan akhir lokasi data Anda, dinyatakan sebagai persentase, bendera pelengkap, dan strategi membelah. Misalnya, string DataArrangement berikut menentukan bahwa 70% pertama dari data akan digunakan untuk membuat sumber data:

```
{
    "splitting": {
        "percentBegin": 0,
        "percentEnd": 70,
        "complement": false,
        "strategy": "sequential"
    }
}
```

Parameter DatareArrangement

Untuk mengubah cara Amazon ML-membuat sumber data, gunakan parameter berikut.

PersentBegin (Opsional)

Gunakan`percentBegin`untuk menunjukkan di mana data untuk sumber data dimulai. Jika Anda tidak menyertakan`percentBegin`dan`percentEnd`, Amazon IL menyertakan semua data saat membuat sumber data.

Nilai yang valid adalah`0`kepada`100`, inklusif.

PercentEnd (Opsional)

Gunakan`percentEnd`untuk menunjukkan di mana data untuk datasource berakhir. Jika Anda tidak menyertakan`percentBegin`dan`percentEnd`, Amazon IL menyertakan semua data saat membuat sumber data.

Nilai yang valid adalah`0`kepada`100`, inklusif.

Pelengkap (Opsional)

Parametercomplementparameter memberitahu Amazon IL untuk menggunakan data yang tidak termasuk dalam kisaranpercentBeginkepadapercentEnduntuk membuat datasource. Parametercomplementberguna jika Anda perlu membuat sumber data komplementer untuk pelatihan dan evaluasi. Untuk membuat sumber data komplementer, gunakan nilai yang sama untukpercentBegindanpercentEnd, bersama dengancomplementparameter.

Misalnya, dua sumber data berikut tidak berbagi data apapun, dan dapat digunakan untuk melatih dan mengevaluasi model. Sumber data pertama memiliki 25 persen dari data, dan yang kedua memiliki 75 persen dari data.

Sumber data untuk evaluasi:

```
{
    "splitting":{
        "percentBegin":0,
        "percentEnd":25
    }
}
```

Datasource untuk pelatihan:

```
{
    "splitting":{
        "percentBegin":0,
        "percentEnd":25,
        "complement":"true"
    }
}
```

Nilai yang valid adalah `true` dan `false`.

Strategi (Opsional)

Untuk mengubah cara Amazon ML-membagi data untuk sumber data, gunakan`strategy`parameter.

Nilai default untuk`strategy`parameter`sequential`, yang berarti bahwa Amazon ML-mengambil semua data record antarapercentBegindanpercentEndparameter untuk datasource, dalam urutan bahwa catatan muncul dalam input data

Berikut dua`DataRearrangement`garis adalah contoh pelatihan berurutan memerintahkan dan evaluasi datasources:

Sumber data untuk evaluasi:`{"splitting":{"percentBegin":70, "percentEnd":100, "strategy":"sequential"}}`

Datasource untuk pelatihan:`{"splitting":{"percentBegin":70, "percentEnd":100, "strategy":"sequential", "complement":"true"}}`

Untuk membuat datasource dari pilihan acak data, atur`strategy`parameter`random`dan menyediakan string yang digunakan sebagai nilai benih untuk pemisahan data acak (misalnya, Anda dapat menggunakan jalur S3 untuk data Anda sebagai string benih acak). Jika Anda memilih strategi split acak, Amazon ML-memberikan setiap baris data nomor pseudo-acak, dan kemudian memilih baris yang memiliki nomor yang ditetapkan antara`percentBegin`dan`percentEnd`. Nomor pseudo-acak ditugaskan menggunakan byte offset sebagai benih, sehingga mengubah hasil data dalam perpecahan yang berbeda. Setiap pemesanan yang ada dipertahankan. Strategi pemisahan acak memastikan bahwa variabel dalam data pelatihan dan evaluasi didistribusikan sama. Hal ini berguna dalam kasus-kasus di mana data input mungkin memiliki urutan semacam implisit, yang sebaliknya akan menghasilkan pelatihan dan evaluasi sumber data yang berisi catatan data yang tidak serupa.

Berikut dua`DataRearrangement`baris adalah contoh pelatihan non-berurutan memerintahkan dan evaluasi datasources:

Sumber data untuk evaluasi:

```
{
    "splitting":{
        "percentBegin":70,
        "percentEnd":100,
        "strategy":"random",
        "strategyParams": {
            "randomSeed":"RANDOMSEED"
        }
    }
}
```

Datasource untuk pelatihan:

```
{
```

```
    "splitting":{
        "percentBegin":70,
        "percentEnd":100,
        "strategy":"random",
        "strategyParams": {
            "randomSeed":"RANDOMSEED"
        }
        "complement":"true"
    }
}
```

Nilai yang valid adalah `sequential` dan `random`.

Strategi (Opsional): randomSeed

Amazon ML-nyaAcomSeeduntuk membagi data. Benih default untuk API adalah string kosong. Untuk menentukan benih untuk strategi split acak, lulus dalam string. Untuk informasi lebih lanjut tentang biji acak, lihatMemisahkan Data Anda secara acakdiPanduan Pengembang Amazon Machine Learning.

Untuk kode contoh yang menunjukkan cara menggunakan validasi silang dengan Amazon MLnya, bukaSampel Machine Learning.

Mengevaluasi Model ML

Anda harus selalumengevaluasi modeluntuk menentukan apakah itu akan melakukan pekerjaan yang baik memprediksi target pada data baru dan masa depan. Karena instance di masa mendatang memiliki nilai target yang tidak diketahui, Anda perlu memeriksa metrik akurasi model ML-pada data yang sudah Anda ketahui jawaban target, dan menggunakan penilaian ini sebagai proxy untuk akurasi prediktif pada data di masa mendatang.

Untuk mengevaluasi model dengan benar, Anda memegang sampel data yang telah diberi label dengan target (ground truth) dari sumber data pelatihan. Mengevaluasi keakuratan prediktif model L dengan data yang sama yang digunakan untuk pelatihan tidak berguna, karena memberikan penghargaan kepada model yang dapat "mengingat" data pelatihan, yang bertentangan dengan generalisasi darinya. Setelah Anda selesai melatih model MLnya, Anda mengirimkan model pengamatan yang dapat Anda ketahui nilai target. Anda kemudian membandingkan prediksi yang dikembalikan oleh model ML-nya dengan nilai target yang diketahui. Akhirnya, Anda menghitung metrik ringkasan yang memberi tahu Anda seberapa baik nilai yang diprediksi dan benar cocok.

Di Amazon IL, Anda mengevaluasi model ML-nyamenciptakan evaluasi. Untuk membuat evaluasi untuk model ML, Anda memerlukan model ML-yang ingin Anda evaluasi, dan Anda memerlukan data berlabel yang tidak digunakan untuk pelatihan. Pertama, buat sumber data untuk evaluasi dengan membuat sumber data Amazon MLdengan data yang diada-out. Data yang digunakan dalam evaluasi harus memiliki skema yang sama dengan data yang digunakan dalam pelatihan dan mencakup nilai aktual untuk variabel target.

Jika semua data Anda ada dalam satu file atau direktori, Anda dapat menggunakan konsol Amazon XML untuk membagi data. Jalur default dalam wizard model Create MLmembagi sumber data input dan menggunakan 70% pertama untuk sumber data pelatihan dan 30% sisanya untuk sumber data evaluasi. Anda juga dapat menyesuaikan rasio split dengan menggunakanKhususpilihan di wizard model Create ML, di mana Anda dapat memilih untuk memilih sampel 70% acak untuk pelatihan dan menggunakan 30% sisanya untuk evaluasi. Untuk lebih menentukan rasio split kustom, gunakan string penataan ulang data diBuat Sumber DataAPI Setelah Anda memiliki sumber data evaluasi dan model ML, Anda dapat membuat evaluasi dan meninjau hasil evaluasi.

Topik

- Wawasan Model ML
- Insight Model Biner
- Wawasan Model Multiclass

Wawasan Model ML

Saat Anda mengevaluasi model MLnya, Amazon IL menyediakan metrik standar industri dan sejumlah wawasan untuk meninjau keakuratan prediktif model Anda. Di Amazon IL, hasil evaluasi berisi hal-hal berikut:

- Sebuah metrik akurasi prediksi untuk melaporkan keberhasilan keseluruhan model
- Visualisasi untuk membantu mengeksplorasi keakuratan model Anda di luar metrik akurasi prediksi
- Kemampuan untuk meninjau dampak pengaturan ambang skor (hanya untuk klasifikasi biner)
- Peringatan tentang kriteria untuk memeriksa validitas evaluasi

Pilihan metrik dan visualisasi tergantung pada jenis model L yang Anda evaluasi. Penting untuk meninjau visualisasi ini untuk memutuskan apakah model Anda berkinerja cukup baik agar sesuai dengan kebutuhan bisnis Anda.

Insight Model Biner

Menafsirkan Prediksi

Output aktual dari banyak algoritma klasifikasi biner adalah prediksiskor. Skor menunjukkan kepastian sistem bahwa pengamatan yang diberikan milik kelas positif (nilai target sebenarnya adalah 1). Model klasifikasi biner di Amazon MLnya menghasilkan skor yang berkisar dari 0 ke 1. Sebagai konsumen skor ini, untuk membuat keputusan tentang apakah observasi harus diklasifikasikan sebagai 1 atau 0, Anda menafsirkan skor dengan memilih ambang klasifikasi, ataubatas,dan membandingkan skor terhadap itu. Setiap observasi dengan skor lebih tinggi dari batas diperkirakan sebagai target= 1, dan skor lebih rendah dari batas diperkirakan sebagai target= 0.

Di Amazon IL, cut-off skor default adalah 0,5. Anda dapat memilih untuk memperbarui cut-off ini agar sesuai dengan kebutuhan bisnis Anda. Anda dapat menggunakan visualisasi di konsol untuk memahami bagaimana pilihan cut-off akan mempengaruhi aplikasi Anda.

Mengukur Akurasi Model

Amazon IL menyediakan metrik akurasi standar industri untuk model klasifikasi biner yang disebut Area Under the (Receiver Operating Characteristic) Curve (AUC). AUC mengukur kemampuan model untuk memprediksi skor yang lebih tinggi untuk contoh-contoh positif. Karena tidak tergantung pada cut-off skor, Anda bisa mendapatkan rasa keakuratan prediksi model Anda dari metrik AUC tanpa memilih ambang batas.

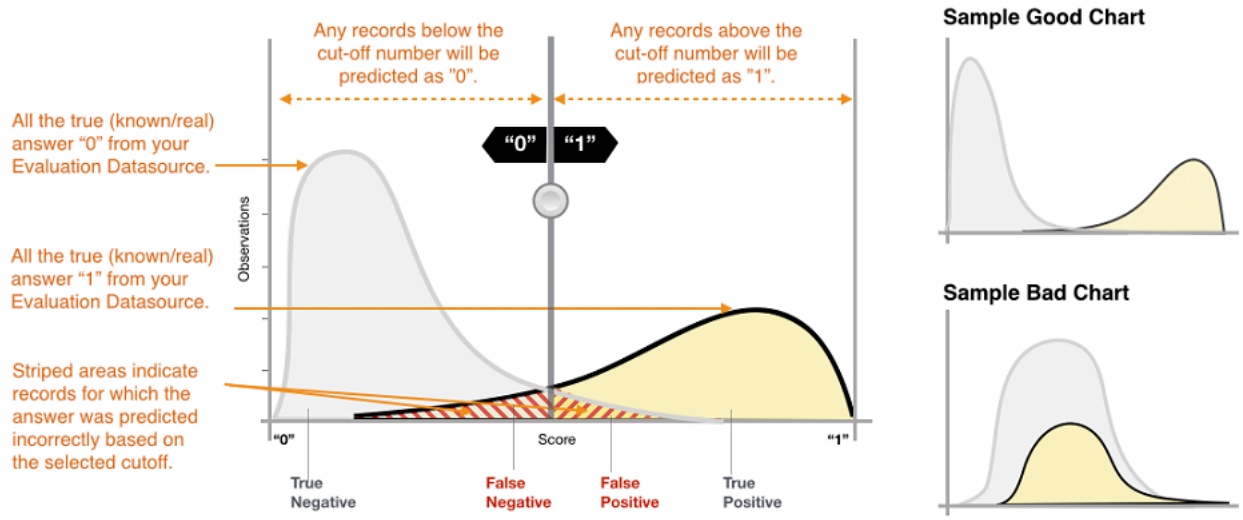
Metrik AUC mengembalikan nilai desimal dari 0 menjadi 1. Nilai AUC dekat 1 menunjukkan model MLnya yang sangat akurat. Nilai di dekat 0,5 menunjukkan model L yang tidak lebih baik daripada menebak secara acak. Nilai dekat 0 tidak biasa untuk dilihat, dan biasanya menunjukkan masalah dengan data. Pada dasarnya, AUC dekat 0 mengatakan bahwa model L telah mempelajari pola yang benar, tetapi menggunakannya untuk membuat prediksi yang dibalik dari kenyataan ('0 diprediksi sebagai '1's dan sebaliknya). Untuk informasi selengkapnya tentang AUC, bukaKarakteristik operasi penerimahalaman di Wikipedia.

Metrik AUC dasar untuk model biner adalah 0,5. Ini adalah nilai untuk model ML hipotetis yang secara acak memprediksi jawaban 1 atau 0. Model ML biner Anda harus berkinerja lebih baik daripada nilai ini untuk mulai menjadi berharga.

Menggunakan Visualisasi Kinerja

Untuk mengeksplorasi keakuratan model ML, Anda dapat meninjau grafik padaevaluasihalaman di konsol Amazon ML. Halaman ini menunjukkan dua histogram: a) histogram skor untuk positif aktual (targetnya adalah 1) dan b) histogram skor untuk negatif aktual (targetnya adalah 0) dalam data evaluasi.

Model L yang memiliki akurasi prediktif yang baik akan memprediksi skor yang lebih tinggi ke 1s aktual dan skor yang lebih rendah ke 0 yang sebenarnya. Sebuah model yang sempurna akan memiliki dua histogram di dua ujung yang berbeda dari sumbu x menunjukkan bahwa sebenarnya positif semua mendapat skor tinggi dan negatif aktual semua mendapat skor rendah. Namun, model L membuat kesalahan, dan grafik khas akan menunjukkan bahwa kedua histogram tumpang tindih pada skor tertentu. Model kinerja yang sangat buruk tidak akan dapat membedakan antara kelas positif dan negatif, dan kedua kelas akan memiliki histogram sebagian besar tumpang tindih.



Dengan menggunakan visualisasi, Anda dapat mengidentifikasi jumlah prediksi yang termasuk dalam dua jenis prediksi yang benar dan dua jenis prediksi yang salah.

Prediksi yang benar

- Benar positif (TP): Amazon IL memprediksi nilainya sebagai 1, dan nilai sebenarnya adalah 1.
- Benar negatif (TN): Amazon IL memprediksi nilainya sebagai 0, dan nilai sebenarnya adalah 0.

Prediksi yang salah

- Positif palsu (FP): Amazon IL memprediksi nilainya sebagai 1, tetapi nilai sebenarnya adalah 0.
- Negatif palsu (FN): Amazon IL memprediksi nilainya sebagai 0, tetapi nilai sebenarnya adalah 1.

> **Note**
>
> Jumlah TP, TN, FP, dan FN tergantung pada ambang skor yang dipilih, dan mengoptimalkan salah satu dari angka-angka ini berarti membuat tradeoff pada yang lain. Sejumlah TPS yang tinggi biasanya menghasilkan sejumlah FP yang tinggi dan sejumlah TNs yang rendah.

Menyesuaikan Skor Cut-off

Model L bekerja dengan menghasilkan skor prediksi numerik, dan kemudian menerapkan cut-off untuk mengubah skor ini menjadi biner 0/1 label. Dengan mengubah skor cut-off, Anda dapat menyesuaikan perilaku model ketika membuat kesalahan. Padaevaluasihalaman di konsol Amazon

XML, Anda dapat meninjau dampak dari berbagai cut-off skor, dan Anda dapat menyimpan potongan skor yang ingin Anda gunakan untuk model Anda.

Ketika Anda menyesuaikan batas cut-off skor, amati trade-off antara dua jenis kesalahan. Memindahkan cut-off ke kiri menangkap positif yang lebih benar, tetapi trade-off adalah peningkatan jumlah kesalahan positif palsu. Memindahkannya ke kanan menangkap kurang dari kesalahan positif palsu, tetapi trade-off adalah bahwa hal itu akan kehilangan beberapa positif yang benar. Untuk aplikasi prediktif Anda, Anda membuat keputusan jenis kesalahan mana yang lebih dapat ditoleransi dengan memilih skor cut-off yang sesuai.

Meninjau Metrik Lanjutan

Amazon IL menyediakan metrik tambahan berikut untuk mengukur keakuratan prediktif model ML-nya: akurasi, presisi, penarikan kembali, dan tingkat positif palsu.

Akurasi

Akurasi(ACC) mengukur fraksi prediksi yang benar. Rentangnya adalah 0 ke 1. Nilai yang lebih besar menunjukkan akurasi prediktif yang lebih baik:

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN}$$

Presisi

Presisimengukur fraksi positif aktual di antara contoh-contoh yang diprediksi positif. Rentangnya adalah 0 ke 1. Nilai yang lebih besar menunjukkan akurasi prediktif yang lebih baik:

$$Precision = \frac{TP}{TP + FP}$$

Recall

Recallmengukur fraksi positif aktual yang diprediksi positif. Rentangnya adalah 0 ke 1. Nilai yang lebih besar menunjukkan akurasi prediktif yang lebih baik:

$$Recall = \frac{TP}{TP + FN}$$

Tingkat Positif Palsu

Parametertingkat positif palsu(FPR) mengukur tingkat alarm palsu atau fraksi negatif aktual yang diprediksi positif. Rentangnya adalah 0 ke 1. Nilai yang lebih kecil menunjukkan akurasi prediktif yang lebih baik:

$$FPR = \frac{\text{FP}}{\text{FP} + \text{TN}}$$

Tergantung pada masalah bisnis Anda, Anda mungkin lebih tertarik pada model yang berkinerja baik untuk subset tertentu dari metrik ini. Misalnya, dua aplikasi bisnis mungkin memiliki persyaratan yang sangat berbeda untuk model ML-nya:

- Satu aplikasi mungkin perlu sangat yakin tentang prediksi positif yang sebenarnya positif (presisi tinggi), dan mampu mengklasifikasikan beberapa contoh positif sebagai negatif (recall moderat).
- Aplikasi lain mungkin perlu memprediksi dengan benar sebanyak mungkin contoh positif (recall tinggi), dan akan menerima beberapa contoh negatif yang salah diklasifikasikan sebagai positif (presisi moderat).

Amazon IL memungkinkan Anda untuk memilih cut-off skor yang sesuai dengan nilai tertentu dari salah satu metrik lanjutan sebelumnya. Hal ini juga menunjukkan tradeoffs yang terjadi dengan mengoptimalkan untuk satu metrik. Misalnya, jika Anda memilih cut-off yang sesuai dengan presisi tinggi, Anda biasanya harus menukarnya dengan penarikan yang lebih rendah.

> **Note**
>
> Anda harus menyimpan cut-off skor agar dapat diterapkan pada mengklasifikasikan prediksi masa depan dengan model ML-mu.

Wawasan Model Multiclass

Menafsirkan Prediksi

Output aktual dari algoritma klasifikasi multiclass adalah seperangkat prediksimencetak. Skor menunjukkan kepastian model bahwa pengamatan yang diberikan milik masing-masing kelas. Tidak seperti untuk masalah klasifikasi biner, Anda tidak perlu memilih skor cut-off untuk membuat prediksi. Jawaban yang diprediksi adalah kelas (misalnya, label) dengan skor prediksi tertinggi.

Mengukur Akurasi Model

Metrik tipikal yang digunakan dalam multiclass sama dengan metrik yang digunakan dalam kasus klasifikasi biner setelah rata-rata di semua kelas. Di Amazon IL, skor F1 makro-rata-rata digunakan untuk mengevaluasi keakuratan prediktif metrik multiclass.

Makro Rata-rata Skor F1

Skor F1 adalah metrik klasifikasi biner yang menganggap kedua metrik biner presisi dan recall. Ini adalah mean harmonik antara presisi dan recall. Rentangnya adalah 0 hingga 1. Nilai yang lebih besar menunjukkan akurasi prediktif yang lebih baik:

$$F1\ score = \frac{2 * \text{precision} * \text{recall}}{\text{precision} + \text{recall}}$$

Skor F1 rata-rata makro adalah rata-rata unweighted dari F1-skor atas semua kelas dalam kasus multiclass. Ini tidak memperhitungkan frekuensi terjadinya kelas dalam dataset evaluasi. Nilai yang lebih besar menunjukkan akurasi prediktif yang lebih baik. Contoh berikut menunjukkan kelas K di datasource evaluasi:

$$Macro\ average\ F1\ score = \frac{1}{K}\sum_{k=1}^{K} F1\ score\ for\ class\ k$$

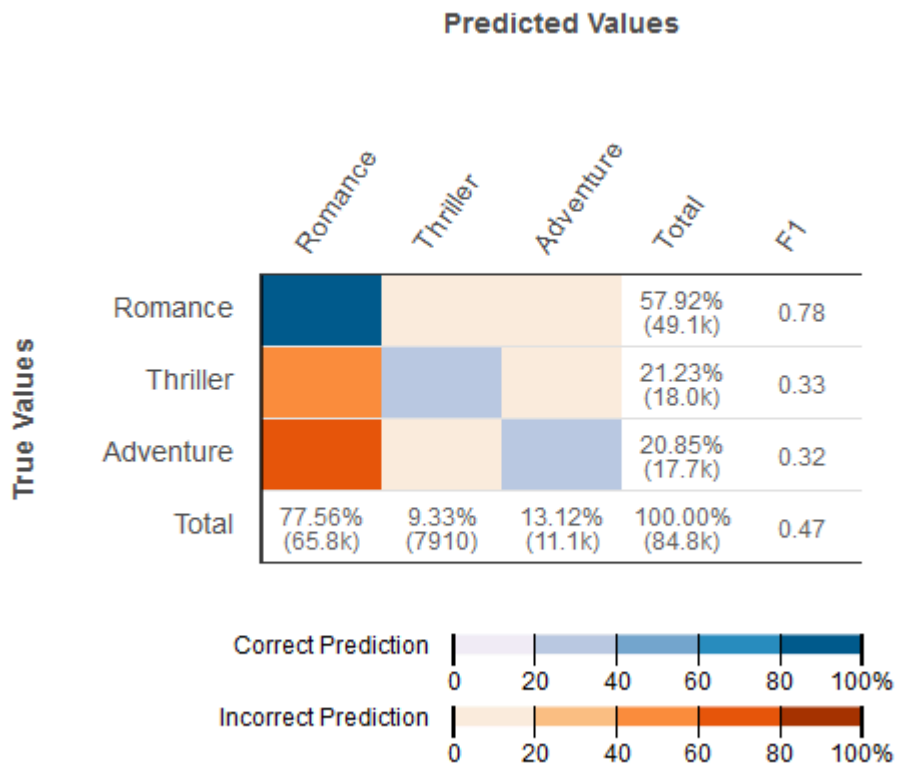
Baseline Makro Rata-rata Skor F1

Amazon IL menyediakan metrik dasar untuk model multiclass. Ini adalah skor F1 rata-rata makro untuk model multiclass hipotetis yang akan selalu memprediksi kelas yang paling sering sebagai jawabannya. Misalnya, jika Anda memprediksi genre film dan genre yang paling umum dalam data pelatihan Anda adalah Romance, maka model dasar akan selalu memprediksi genre sebagai Romance. Anda akan membandingkan model ML-mu dengan baseline ini untuk memvalidasi jika model MLmu lebih baik daripada model ML-mu yang memprediksi jawaban konstan ini.

Menggunakan Visualisasi Kinerja

Amazon ML-nya menyediakanmatriks kebingunganbagai cara untuk memvisualisasikan keakuratan klasifikasi multiclass model prediktif. Matriks kebingungan menggambarkan dalam tabel jumlah atau persentase prediksi yang benar dan salah untuk setiap kelas dengan membandingkan kelas diprediksi pengamatan dan kelas sebenarnya.

Misalnya, jika Anda mencoba mengklasifikasikan film menjadi genre, model prediktif mungkin memprediksi bahwa genre (kelas) adalah Romance. Namun, genre sebenarnya sebenarnya mungkin Thriller. Ketika Anda mengevaluasi keakuratan model ML-klasifikasi multiclass, Amazon IL mengidentifikasi kesalahan klasifikasi ini dan menampilkan hasil dalam matriks kebingungan, seperti yang ditunjukkan dalam ilustrasi berikut.



Informasi berikut ini ditampilkan dalam matriks kebingungan:

- Jumlah prediksi yang benar dan salah untuk setiap kelas: Setiap baris dalam matriks kebingungan sesuai dengan metrik untuk salah satu kelas yang benar. Misalnya, baris pertama menunjukkan bahwa untuk film yang sebenarnya ada dalam genre Romantis, model multiclass ML-nya mendapatkan prediksi yang tepat untuk lebih dari 80% kasus. Ini salah memprediksi genre sebagai Thriller untuk kurang dari 20% dari kasus, dan Adventure kurang dari 20% dari kasus.
- Class-wise F1 Skor: Kolom terakhir menunjukkan F1-skor untuk masing-masing kelas.
- Benar kelas-frekuensi dalam data evaluasi: Kolom kedua untuk terakhir menunjukkan bahwa dalam dataset evaluasi, 57,92% dari pengamatan dalam data evaluasi adalah Romance, 21.23% adalah Thriller, dan 20,85% adalah Petualangan.
- Frekuensi kelas yang diprediksi untuk data evaluasi: Baris terakhir menunjukkan frekuensi masing-masing kelas dalam prediksi. 77.56% pengamatan diprediksi sebagai Romance, 9.33% diprediksi sebagai Thriller, dan 13.12% diprediksi sebagai Petualangan.

Konsol Amazon XML menyediakan tampilan visual yang mengakomodasi hingga 10 kelas dalam matriks kebingungan, yang tercantum dalam urutan kelas yang paling sering hingga paling sering

dalam data evaluasi. Jika data evaluasi Anda memiliki lebih dari 10 kelas, Anda akan melihat 9 kelas paling sering terjadi dalam matriks kebingungan, dan semua kelas lainnya akan runtuh ke dalam kelas yang disebut “orang lain.” Amazon IL juga menyediakan kemampuan untuk mengunduh matriks kebingungan penuh melalui tautan di halaman visualisasi multiclass.

Wawasan Model

Menafsirkan Prediksi

Output dari model ML-regresi adalah nilai numerik untuk prediksi model target. Misalnya, jika Anda memprediksi harga perumahan, prediksi model bisa menjadi nilai seperti 254.013.

> **Note**
>
> Kisaran prediksi dapat berbeda dari kisaran target dalam data pelatihan. Misalnya, katakanlah Anda memprediksi harga perumahan, dan target dalam data pelatihan memiliki nilai dalam kisaran 0 hingga 450.000. Target yang diprediksi tidak perlu berada dalam kisaran yang sama, dan mungkin mengambil nilai positif (lebih besar dari 450.000) atau nilai negatif (kurang dari nol). Penting untuk merencanakan cara mengatasi nilai prediksi yang berada di luar rentang yang dapat diterima untuk aplikasi Anda.

Mengukur Akurasi Model

Untuk tugas regresi, Amazon MLE menggunakan metrik root mean square error (RMSE) standar industri. Ini adalah ukuran jarak antara target numerik yang diprediksi dan jawaban numerik aktual (ground truth). Semakin kecil nilai RMSE, semakin baik akurasi prediktif model. Sebuah model dengan prediksi yang benar sempurna akan memiliki RMSE 0. Contoh berikut menunjukkan data evaluasi yang berisi catatan N:

$$RMSE = \sqrt{1/N \sum_{i=1}^{N} (actual\ target - predicted\ target)^2}$$

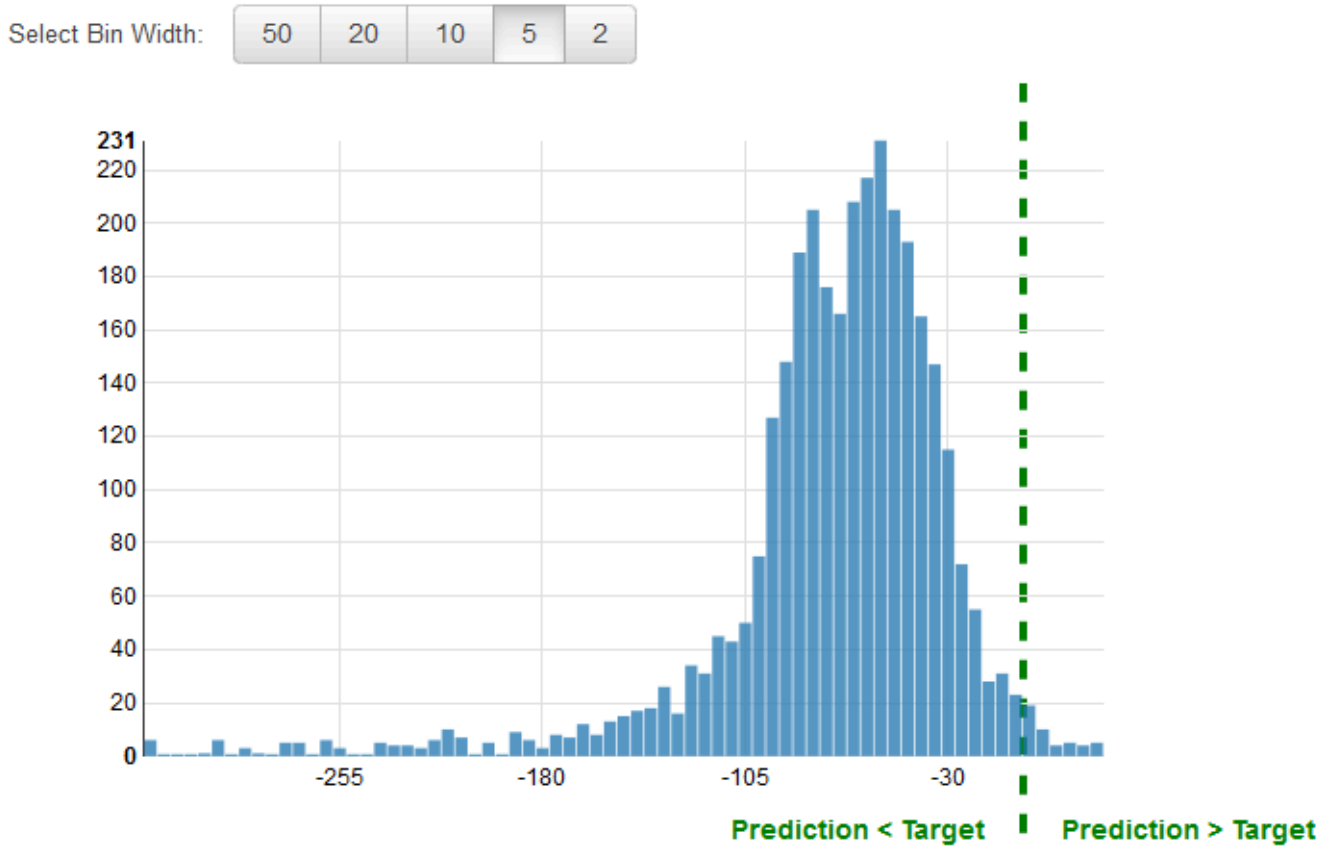
RMSE Dasar

Amazon XML menyediakan metrik dasar untuk model regresi. Ini adalah RMSE untuk model regresi hipotetis yang akan selalu memprediksi rata-rata target sebagai jawabannya. Misalnya,

jika Anda memprediksi usia pembeli rumah dan usia rata-rata untuk pengamatan dalam data pelatihan Anda adalah 35, model dasar akan selalu memprediksi jawabannya sebagai 35. Anda akan membandingkan model ML-mu dengan baseline ini untuk memvalidasi jika model MLmu lebih baik daripada model ML-mu yang memprediksi jawaban konstan ini.

Menggunakan Visualisasi Kinerja

Ini adalah praktik umum untuk meninjau ulangresiduuntuk masalah regresi. Sisa untuk pengamatan dalam data evaluasi adalah perbedaan antara target sebenarnya dan target yang diprediksi. Residu mewakili bagian dari target bahwa model tidak dapat memprediksi. Sisa positif menunjukkan bahwa model meremehkan target (target sebenarnya lebih besar dari target yang diprediksi). Residual negatif menunjukkan overestimation (target sebenarnya lebih kecil dari target yang diprediksi). Histogram residu pada data evaluasi ketika didistribusikan dalam bentuk lonceng dan berpusat pada nol menunjukkan bahwa model membuat kesalahan secara acak dan tidak secara sistematis atas atau di bawah memprediksi kisaran tertentu nilai target. Jika residu tidak membentuk bentuk lonceng berpusat nol, ada beberapa struktur dalam kesalahan prediksi model. Menambahkan lebih banyak variabel ke model mungkin membantu model menangkap pola yang tidak ditangkap oleh model saat ini. Ilustrasi berikut menunjukkan residu yang tidak berpusat di sekitar nol.



Mencegah Overfitting

Saat membuat dan melatih model MLnya, tujuannya adalah memilih model yang membuat prediksi terbaik, yang berarti memilih model dengan pengaturan terbaik (pengaturan model mL atau hyperparameters). Di Amazon Machine Learning, ada empat hyperparameter yang dapat Anda tetapkan: jumlah pass, regularisasi, ukuran model, dan tipe shuffle. Namun, jika Anda memilih pengaturan parameter model yang menghasilkan kinerja prediktif “terbaik” pada data evaluasi, Anda mungkin overfit model Anda. Overfitting terjadi ketika model telah menghafal pola yang terjadi dalam pelatihan dan evaluasi sumber data, tetapi telah gagal untuk menggeneralisasi pola dalam data. Hal ini sering terjadi ketika data pelatihan mencakup semua data yang digunakan dalam evaluasi. Model overfitted tidak baik selama evaluasi, tetapi gagal untuk membuat prediksi akurat pada data tak terlihat.

Untuk menghindari memilih model overfitted sebagai model terbaik, Anda dapat memesan data tambahan untuk memvalidasi kinerja model ML-nya. Misalnya, Anda dapat membagi data Anda menjadi 60 persen untuk pelatihan, 20 persen untuk evaluasi, dan tambahan 20 persen untuk validasi. Setelah memilih parameter model yang bekerja dengan baik untuk data evaluasi, Anda menjalankan evaluasi kedua dengan data validasi untuk melihat seberapa baik model ML-kinerja pada data validasi. Jika model memenuhi harapan Anda pada data validasi, maka model tidak overfitting data.

Menggunakan set ketiga data untuk validasi membantu Anda memilih parameter model ML-yang sesuai untuk mencegah overfitting. Namun, menyimpan data dari proses pelatihan untuk evaluasi dan validasi membuat lebih sedikit data yang tersedia untuk pelatihan. Hal ini terutama masalah dengan set data kecil karena selalu terbaik untuk menggunakan data sebanyak mungkin untuk pelatihan. Untuk mengatasi masalah ini, Anda dapat melakukan cross-validasi. Untuk informasi tentang validasi silang, lihatLintas Validasi.

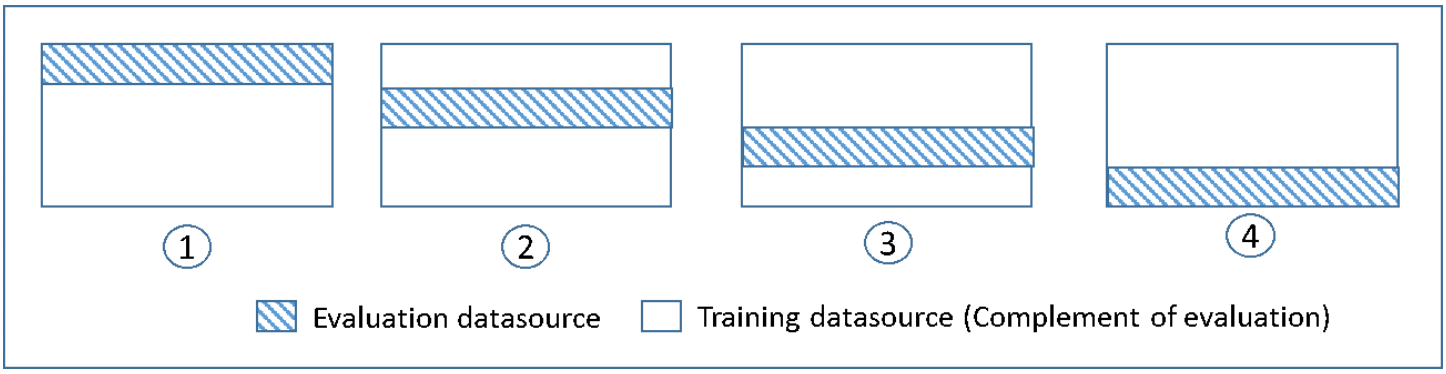
Lintas Validasi

Cross-validasi adalah teknik untuk mengevaluasi model ML-dengan melatih beberapa model ML-subset dari data input yang tersedia dan mengevaluasi mereka pada subset pelengkap data. Gunakan cross-validasi untuk mendeteksi overfitting, yaitu, gagal untuk menggeneralisasi pola.

Di Amazon XML, Anda dapat menggunakan metode validasi silang k-fold untuk melakukan validasi silang. Dalam k-fold cross-validasi, Anda membagi data input menjadi k subset data (juga dikenal sebagai lipatan). Anda melatih model ML-pada semua kecuali satu (k-1) dari subset, dan kemudian

mengevaluasi model pada subset yang tidak digunakan untuk pelatihan. Proses ini diulang k kali, dengan subset yang berbeda disediakan untuk evaluasi (dan dikecualikan dari pelatihan) setiap kali.

Diagram berikut menunjukkan contoh dari subset pelatihan dan subset evaluasi komplementer yang dihasilkan untuk masing-masing dari empat model yang dibuat dan dilatih selama 4 kali lipat cross-validasi. Model satu menggunakan 25 persen data pertama untuk evaluasi, dan 75 persen sisanya untuk pelatihan. Model dua menggunakan subset kedua sebesar 25 persen (25 persen hingga 50 persen) untuk evaluasi, dan tiga subset data yang tersisa untuk pelatihan, dan seterusnya.



Setiap model dilatih dan dievaluasi menggunakan sumber data komplementer - data dalam sumber data evaluasi mencakup dan terbatas pada semua data yang tidak ada dalam sumber data pelatihan. Anda membuat sumber data untuk masing-masing subset ini dengan`DataRearrangement`parameter dalam`createDatasourceFromS3`,`createDatasourceFromRedShift`, dan`createDatasourceFromRDS`API. Di`DataRearrangement`parameter, tentukan subset data yang akan disertakan dalam sumber data dengan menentukan di mana untuk memulai dan mengakhiri setiap segmen. Untuk membuat sumber data komplementer yang diperlukan untuk validasi silang 4 kali lipat, tentukan`DataRearrangement`parameter seperti yang ditunjukkan dalam contoh berikut:

Model satu:

Sumber data untuk evaluasi:

```
{"splitting":{"percentBegin":0, "percentEnd":25}}
```

Sumber data untuk pelatihan:

```
{"splitting":{"percentBegin":0, "percentEnd":25, "complement":"true"}}
```

Model dua:

Sumber data untuk evaluasi:

```
{"splitting":{"percentBegin":25, "percentEnd":50}}
```

Sumber data untuk pelatihan:

```
{"splitting":{"percentBegin":25, "percentEnd":50, "complement":"true"}}
```

Tiga model:

Sumber data untuk evaluasi:

```
{"splitting":{"percentBegin":50, "percentEnd":75}}
```

Sumber data untuk pelatihan:

```
{"splitting":{"percentBegin":50, "percentEnd":75, "complement":"true"}}
```

Model empat:

Sumber data untuk evaluasi:

```
{"splitting":{"percentBegin":75, "percentEnd":100}}
```

Sumber data untuk pelatihan:

```
{"splitting":{"percentBegin":75, "percentEnd":100, "complement":"true"}}
```

Melakukan validasi silang 4 kali lipat menghasilkan empat model, empat sumber data untuk melatih model, empat sumber data untuk mengevaluasi model, dan empat evaluasi, satu untuk setiap model. Amazon ML-menghasilkan metrik kinerja model untuk setiap evaluasi. Misalnya, dalam 4 kali lipat cross-validasi untuk masalah klasifikasi biner, masing-masing evaluasi melaporkan area di bawah kurva (AUC) metrik. Anda bisa mendapatkan ukuran kinerja secara keseluruhan dengan menghitung rata-rata empat metrik AUC. Untuk informasi tentang metrik AUC, lihatMengukur Akurasi Model.

Untuk contoh kode yang menunjukkan cara membuat cross-validasi dan rata-rata skor model, lihatKode sampel Amazon.

Menyesuaikan Model Anda

Setelah Anda memvalidasi model silang, Anda dapat menyesuaikan pengaturan untuk model berikutnya jika model Anda tidak sesuai standar Anda. Untuk informasi lebih lanjut tentang overfitting, lihatCocok model: Overfitting vs. Overfitting. Untuk informasi lebih lanjut tentang regularisasi, lihatRegularisasi. Untuk informasi selengkapnya tentang mengubah pengaturan, lihatMembuat Model ML-nya dengan Opsi Kustom.

Peringatan Evaluasi

Amazon L memberikan wawasan untuk membantu Anda memvalidasi apakah Anda mengevaluasi model dengan benar. Jika salah satu kriteria validasi tidak terpenuhi oleh evaluasi, konsol Amazon ML-memperingatkan Anda dengan menampilkan kriteria validasi yang telah dilanggar, sebagai berikut.

- Evaluasi model ML dilakukan pada data yang diada-out

  Amazon XML memberi tahu Anda jika Anda menggunakan sumber data yang sama untuk pelatihan dan evaluasi. Jika Anda menggunakan Amazon XML untuk membagi data Anda, Anda akan memenuhi kriteria validitas ini. Jika Anda tidak menggunakan Amazon XML untuk membagi data Anda, pastikan untuk mengevaluasi model ML-mu dengan sumber data selain sumber data pelatihan.
- Data yang cukup digunakan untuk evaluasi model prediktif

  Amazon XML memberi tahu Anda jika jumlah pengamatan/catatan dalam data evaluasi Anda kurang dari 10% jumlah pengamatan yang Anda miliki dalam sumber data pelatihan Anda. Untuk mengevaluasi model Anda dengan benar, penting untuk memberikan sampel data yang cukup besar. Kriteria ini memberikan cek untuk memberi tahu Anda jika Anda menggunakan terlalu sedikit data. Jumlah data yang diperlukan untuk mengevaluasi model ML-mu bersifat subjektif. 10% dipilih di sini sebagai stop gap tanpa adanya ukuran yang lebih baik.
- Skema cocok

  Amazon ML-memberi tahu Anda jika skema untuk sumber data pelatihan dan evaluasi tidak sama. Jika Anda memiliki atribut tertentu yang tidak ada dalam sumber data evaluasi atau jika Anda memiliki atribut tambahan, Amazon LL akan menampilkan peringatan ini.
- Semua catatan dari file evaluasi digunakan untuk evaluasi kinerja model prediktif

Penting untuk mengetahui apakah semua catatan yang disediakan untuk evaluasi sebenarnya digunakan untuk mengevaluasi model. Amazon XML memberi tahu Anda jika beberapa catatan dalam sumber data evaluasi tidak valid dan tidak disertakan dalam perhitungan metrik akurasi. Misalnya, jika variabel target hilang untuk beberapa pengamatan dalam sumber data evaluasi, Amazon L tidak dapat memeriksa apakah prediksi model ML's untuk pengamatan ini benar. Dalam hal ini, catatan dengan nilai target yang hilang dianggap tidak valid.

- Distribusi variabel target

  Amazon XML menunjukkan kepada Anda distribusi atribut target dari sumber data pelatihan dan evaluasi sehingga Anda dapat meninjau apakah target didistribusikan sama di kedua sumber data. Jika model dilatih pada data pelatihan dengan distribusi target yang berbeda dari distribusi target pada data evaluasi, maka kualitas evaluasi bisa menderita karena sedang dihitung pada data dengan statistik yang sangat berbeda. Cara terbaik adalah untuk memiliki data didistribusikan sama melalui pelatihan dan evaluasi data, dan memiliki dataset ini meniru sebanyak mungkin data yang model akan hadapi ketika membuat prediksi.

  Jika peringatan ini dipicu, coba gunakan strategi split acak untuk membagi data menjadi sumber data pelatihan dan evaluasi. Dalam kasus yang jarang terjadi, peringatan ini mungkin keliru memperingatkan Anda tentang perbedaan distribusi target meskipun Anda membagi data Anda secara acak. Amazon L menggunakan perkiraan statistik data untuk mengevaluasi distribusi data, kadang-kadang memicu peringatan ini dalam kesalahan.

Menghasilkan dan Menafsirkan Prediksi

Amazon IL menyediakan dua mekanisme untuk menghasilkan prediksi: asinkron (berbasis batch) dan sinkron (satu-at-a-time).

Gunakan prediksi asinkron, atauprediksi batch,ketika Anda memiliki sejumlah pengamatan dan ingin mendapatkan prediksi untuk pengamatan sekaligus. Proses ini menggunakan sumber data sebagai input, dan menghasilkan prediksi ke dalam file.csv yang disimpan dalam bucket S3 pilihan Anda. Anda harus menunggu sampai proses prediksi batch selesai sebelum Anda dapat mengakses hasil prediksi. Ukuran maksimum sumber data yang dapat diproses oleh Amazon L dalam file batch adalah 1 TB (sekitar 100 juta catatan). Jika sumber data Anda lebih besar dari 1 TB, pekerjaan Anda akan gagal dan Amazon IL akan mengembalikan kode kesalahan. Untuk mencegah hal ini, bagi data Anda menjadi beberapa batch. Jika catatan Anda biasanya lebih lama, Anda akan mencapai batas 1 TB sebelum 100 juta catatan diproses. Dalam kasus ini, sebaiknya hubungiDukungan AWSuntuk meningkatkan ukuran pekerjaan untuk prediksi batch Anda.

Gunakan sinkron, atauPrediksi real-time,ketika Anda ingin mendapatkan prediksi pada latensi rendah. API prediksi real-time menerima observasi input tunggal yang diserialisasi sebagai string JSON, dan secara serentak mengembalikan prediksi dan metadata terkait sebagai bagian dari respons API. Anda dapat secara bersamaan memanggil API lebih dari sekali untuk mendapatkan prediksi sinkron secara paralel. Untuk informasi selengkapnya tentang batas throughput API prediksi real-time, lihat batas prediksi real-time diReferensi API Amazon ML-API.

Topik

Membuat Prediksi Batch

Untuk membuat prediksi batch, Anda membuat`BatchPrediction`objek menggunakan konsol Amazon Machine Learning (Amazon ML) atau API. SEBUAH`BatchPrediction`objek menjelaskan seperangkat prediksi yang dihasilkan Amazon MLdengan menggunakan model ML-mu dan satu set observasi masukan. Saat Anda membuat`BatchPrediction`objek, Amazon ML-memulai alur kerja asinkron yang menghitung prediksi.

Anda harus menggunakan skema yang sama untuk sumber data yang Anda gunakan untuk mendapatkan prediksi batch dan sumber data yang Anda gunakan untuk melatih model L yang Anda kueri untuk prediksi. Satu pengecualian adalah bahwa sumber data untuk prediksi batch tidak perlu menyertakan atribut target karena Amazon L memprediksi target. Jika Anda memberikan atribut target, Amazon ML-mengabaikan nilainya.

Membuat Prediksi Batch (Konsol)

Untuk membuat prediksi batch menggunakan konsol Amazon XML, gunakan wizard Create Batch Prediction.

Membuat prediksi batch (console)

1. Masuk keAWS Management Consoledan buka konsol Amazon Machine Learning dihttps://console.aws.amazon.com/machinelearning/.
2. Di dasbor Amazon ML-nya, di bawahObjek, pilihBuat..., dan kemudian pilihPrediksi Batch.
3. Pilih model Amazon ML-yang ingin Anda gunakan untuk membuat prediksi batch.
4. Untuk mengonfirmasi bahwa Anda ingin menggunakan model ini, pilihLanjutkan.
5. Pilih sumber data yang akan dibuatkan prediksi. Sumber data harus memiliki skema yang sama dengan model Anda, meskipun tidak perlu menyertakan atribut target.
6. Pilih Continue (Lanjutkan).
7. UntukTujuan S3, ketik nama bucket S3 Anda.
8. Pilih Tinjau.
9. Tinjau pengaturan Anda dan pilihBuat prediksi batch.

Membuat Prediksi Batch (API)

Membuat`BatchPrediction`objek menggunakan API Amazon MLI, Anda harus memberikan parameter berikut:

ID Sumber data

ID sumber data yang menunjuk ke pengamatan yang Anda inginkan prediksi. Misalnya, jika Anda menginginkan prediksi data dalam file yang disebut`s3://examplebucket/input.csv`, Anda akan membuat objek datasource yang menunjuk ke file data, dan kemudian meneruskan ID dari sumber data tersebut dengan parameter ini.

ID BatchPrediction

ID untuk menetapkan prediksi batch.

ID Model

ID model ML-nya yang harus kueri Amazon ML-nya untuk prediksi tersebut.

URI Output

URI bucket S3 tempat menyimpan output prediksi. Amazon XML harus memiliki izin untuk menulis data ke bucket ini.

Parameter`OutputUri`parameter harus merujuk ke jalur S3 yang berakhir dengan garis miring maju ('/') karakter, seperti yang ditunjukkan pada contoh berikut:

s3: //examplebucket/examplepath/

Untuk informasi tentang mengonfigurasi izin S3, lihat Memberikan Izin Amazon MLuntuk Prediksi Output ke Amazon S3.

(Opsional) Nama BatchPrediction

(Opsional) Nama yang dapat dibaca manusia untuk prediksi batch Anda.

Meninjau Metrik Prediksi Batch

Setelah Amazon Machine Learning (Amazon ML) membuat prediksi batch, Amazon Machine Learning menyediakan dua metrik:`Records seen`dan`Records failed to process`.`Records seen`memberitahu Anda berapa banyak catatan Amazon ML-lihat ketika menjalankan prediksi batch Anda.`Records failed to process`memberi tahu Anda berapa banyak catatan yang tidak dapat diproses oleh Amazon ML-nya.

Untuk memungkinkan Amazon XML memproses data yang gagal, periksa pemformatan catatan dalam data yang digunakan untuk membuat sumber data Anda, dan pastikan semua atribut yang diperlukan ada dan semua data sudah benar. Setelah memperbaiki data Anda, Anda dapat membuat ulang prediksi batch Anda, atau membuat sumber data baru dengan catatan yang gagal, dan kemudian membuat prediksi batch baru menggunakan datasource baru.

Meninjau Metrik Prediksi Batch (Konsol)

Untuk melihat metrik di konsol Amazon ML-nya, bukaRingkasan prediksi Batchhalaman dan lihat diInfo yang Diprosesbagian.

Meninjau Metrik dan Rincian Prediksi Batch (API)

Anda dapat menggunakan API Amazon IL untuk mengambil rincian tentang`BatchPrediction`objek, termasuk metrik rekaman. Amazon IL menyediakan panggilan API prediksi batch berikut:

- `CreateBatchPrediction`
- `UpdateBatchPrediction`
- `DeleteBatchPrediction`
- `GetBatchPrediction`
- `DescribeBatchPredictions`

Untuk informasi lebih lanjut, lihatReferensi API Amazon.

Membaca Batch Prediksi Output File

Lakukan langkah-langkah berikut untuk mengambil file keluaran prediksi:

1. Cari file manifes prediksi batch.
2. Baca file manifes untuk menentukan lokasi file output.
3. Mengambil file output yang berisi prediksi.
4. Menafsirkan isi dari file output. Isi akan bervariasi berdasarkan jenis model ML-nya yang digunakan untuk menghasilkan prediksi.

Bagian berikut menjelaskan langkah-langkah secara lebih rinci.

Menemukan File Manifest Prediksi Batch

File manifes dari prediksi batch berisi informasi yang memetakan file input Anda ke file output prediksi.

Untuk menemukan file manifes, mulailah dengan lokasi output yang Anda tentukan saat Anda membuat objek prediksi batch. Anda dapat query objek prediksi batch selesai untuk mengambil lokasi S3 file ini dengan menggunakan salah satuAPI Amazonatauhttps://console.aws.amazon.com/machinelearning/.

File manifes terletak di lokasi output di jalur yang terdiri dari string statis`batch-prediction/`ditambahkan ke lokasi output dan nama file manifes, yang merupakan ID prediksi batch, dengan ekstensi`.manifest`ditambahkan ke itu.

Misalnya, jika Anda membuat objek prediksi batch dengan ID`bp-example`, dan Anda menentukan lokasi S3`s3://examplebucket/output/`sebagai lokasi output, Anda akan menemukan file manifes Anda di sini:

`s3://examplebucket/output/batch-prediction/bp-example.manifest`

Membaca File Manifes

Isi file.manifest dikodekan sebagai peta JSON, di mana kuncinya adalah string dari nama file data input S3, dan nilainya adalah string dari file hasil prediksi batch terkait. Ada satu baris pemetaan untuk setiap pasangan file input/output. Melanjutkan dengan contoh kita, jika masukan untuk penciptaan`BatchPrediction`objek terdiri dari satu file yang disebut data.csv yang terletak di`s3://examplebucket/input/`, Anda mungkin akan melihat string pemetaan seperti ini:

```
{"s3://examplebucket/input/data.csv":"
s3://examplebucket/output/batch-prediction/result/bp-example-data.csv.gz"}
```

Jika input untuk penciptaan`BatchPrediction`objek terdiri dari tiga file yang disebut data1.csv, data2.csv, dan data3.csv, dan mereka semua disimpan di lokasi S3`s3://examplebucket/input/`, Anda mungkin akan melihat string pemetaan seperti ini:

```
{"s3://examplebucket/input/data1.csv":"s3://examplebucket/output/batch-prediction/
result/bp-example-data1.csv.gz",

"s3://examplebucket/input/data2.csv":"
s3://examplebucket/output/batch-prediction/result/bp-example-data2.csv.gz",

"s3://examplebucket/input/data3.csv":"
s3://examplebucket/output/batch-prediction/result/bp-example-data3.csv.gz"}
```

Mengambil File Output Prediksi Batch

Anda dapat mengunduh setiap file prediksi batch yang diperoleh dari pemetaan manifes dan memprosesnya secara lokal. Format file CSV, dikompresi dengan algoritma gzip. Dalam file itu, ada satu baris per observasi masukan dalam file input yang sesuai.

Untuk bergabung dengan prediksi dengan file input prediksi batch, Anda dapat melakukan penggabungan rekor demi catatan sederhana dari dua file. File output dari prediksi batch selalu berisi jumlah yang sama catatan sebagai file input prediksi, dalam urutan yang sama. Jika pengamatan masukan gagal dalam pemrosesan, dan tidak ada prediksi yang dapat dihasilkan, file output dari prediksi batch akan memiliki garis kosong di lokasi yang sesuai.

Menafsirkan Isi File Prediksi Batch untuk model Binary Classification

Kolom dari file prediksi batch untuk model klasifikasi biner diberi namaJawaban terbaikdanSkor.

ParameterJawaban terbaikkolom berisi label prediksi (“1" atau “0") yang diperoleh dengan mengevaluasi skor prediksi terhadap skor cut-off. Untuk informasi selengkapnya tentang skor cut-off, lihatMenyesuaikan Skor Cut-off. Anda menetapkan skor cut-off untuk model ML dengan menggunakan API Amazon IL atau fungsionalitas evaluasi model di konsol Amazon ML. Jika Anda tidak menetapkan skor cut-off, Amazon IL menggunakan nilai default 0,5.

ParameterSkorkolom berisi skor prediksi baku yang ditetapkan oleh model L untuk prediksi ini. Amazon IL menggunakan model regresi logistik, jadi skor ini mencoba memodelkan probabilitas pengamatan yang sesuai dengan nilai true (“1"). Perhatikan bahwaSkordilaporkan dalam notasi ilmiah, sehingga pada baris pertama dari contoh berikut, nilai 8.7642E-3 sama dengan 0.0087642.

Misalnya, jika skor cut-off untuk model MLnya adalah 0.75, isi file output prediksi batch untuk model klasifikasi biner mungkin terlihat seperti ini:

```
bestAnswer,score

0,8.7642E-3

1,7.899012E-1


0,6.323061E-3

0,2.143189E-2


1,8.944209E-1
```

Pengamatan kedua dan kelima dalam file input telah menerima skor prediksi di atas 0,75, jadi kolom BestAnswer untuk pengamatan ini menunjukkan nilai “1", sementara pengamatan lainnya memiliki nilai “0".

Menafsirkan Isi File Prediksi Batch untuk Model Ml Klasifikasi Multiclass

File prediksi batch untuk model multiclass berisi satu kolom untuk setiap kelas yang ditemukan dalam data pelatihan. Nama kolom muncul di baris header dari file prediksi batch.

Ketika Anda meminta prediksi dari model multiclass, Amazon IL menghitung beberapa skor prediksi untuk setiap pengamatan dalam file input, satu untuk masing-masing kelas yang didefinisikan dalam dataset input. Hal ini setara dengan bertanya “Apa probabilitas (diukur antara 0 dan 1) bahwa pengamatan ini akan jatuh ke dalam kelas ini, sebagai lawan dari salah satu kelas lain?” Setiap skor dapat ditafsirkan sebagai “probabilitas bahwa pengamatan milik kelas ini.” Karena skor prediksi memodelkan probabilitas yang mendasari pengamatan milik satu kelas atau yang lain, jumlah semua skor prediksi di baris adalah 1. Anda perlu memilih satu kelas sebagai kelas yang diprediksi untuk model. Paling umum, Anda akan memilih kelas yang memiliki probabilitas tertinggi sebagai jawaban terbaik.

Misalnya, pertimbangkan untuk mencoba memprediksi peringkat pelanggan dari suatu produk, berdasarkan skala bintang 1-ke-5. Jika kelas diberi nama`1_star`,`2_stars`,`3_stars`,`4_stars`, dan`5_stars`, file output prediksi mungkin akan terlihat seperti ini:

```
1_star, 2_stars, 3_stars, 4_stars, 5_stars

8.7642E-3, 2.7195E-1, 4.77781E-1, 1.75411E-1, 6.6094E-2

5.59931E-1, 3.10E-4, 2.48E-4, 1.99871E-1, 2.39640E-1

7.19022E-1, 7.366E-3, 1.95411E-1, 8.78E-4, 7.7323E-2

1.89813E-1, 2.18956E-1, 2.48910E-1, 2.26103E-1, 1.16218E-1

3.129E-3, 8.944209E-1, 3.902E-3, 7.2191E-2, 2.6357E-2
```

Dalam contoh ini, pengamatan pertama memiliki skor prediksi tertinggi untuk`3_stars`kelas (skor prediksi = 4.77781E-1), sehingga Anda akan menafsirkan hasilnya sebagai menunjukkan kelas itu`3_stars`adalah jawaban terbaik untuk pengamatan ini. Perhatikan bahwa skor prediksi dilaporkan dalam notasi ilmiah, sehingga skor prediksi 4.77781E-1 sama dengan 0.477781.

Mungkin ada keadaan ketika Anda tidak ingin memilih kelas dengan probabilitas tertinggi. Misalnya, Anda mungkin ingin menetapkan ambang minimum di bawah ini yang Anda tidak akan menganggap kelas sebagai jawaban terbaik meskipun memiliki skor prediksi tertinggi. Misalkan Anda mengklasifikasikan film ke dalam genre, dan Anda ingin skor prediksi setidaknya 5E-1 sebelum

Anda menyatakan genre untuk menjadi jawaban terbaik Anda. Anda mendapatkan skor prediksi 3E-1 untuk komedi, 2.5E-1 untuk drama, 2.5E-1 untuk dokumenter, dan 2E-1 untuk film aksi. Dalam kasus ini, model L memprediksi bahwa komedi adalah pilihan Anda yang paling mungkin, namun Anda memutuskan untuk tidak memilihnya sebagai jawaban terbaik. Karena tidak ada skor prediksi melebihi skor prediksi dasar Anda dari 5E-1, Anda memutuskan bahwa prediksi tidak cukup untuk percaya diri memprediksi genre dan Anda memutuskan untuk memilih sesuatu yang lain. Aplikasi Anda kemudian mungkin memperlakukan bidang genre untuk film ini sebagai “tidak diketahui.”

Menafsirkan Isi File Prediksi Batch untuk Model L Regresi

Berkas prediksi batch untuk model regresi berisi satu kolom bernamaSkor. Kolom ini berisi prediksi numerik mentah untuk setiap pengamatan dalam data input. Nilai-nilai yang dilaporkan dalam notasi ilmiah, sehinggaSkornilai -1.526385E1 sama dengan -15.26835 di baris pertama dalam contoh berikut.

Contoh ini menunjukkan file output untuk prediksi batch yang dilakukan pada model regresi:

```
score

-1.526385E1

-6.188034E0

-1.271108E1

-2.200578E1

8.359159E0
```

Memiinta Prediksi Waktu Nyata

Prediksi real-time adalah panggilan sinkron ke Amazon Machine Learning (Amazon ML). Prediksi dibuat ketika Amazon L mendapatkan permintaan, dan respon dikembalikan segera. Prediksi real-time biasanya digunakan untuk mengaktifkan kemampuan prediktif dalam aplikasi web, seluler, atau desktop interaktif. Anda dapat melakukan kueri model ML-yang dibuat dengan Amazon IL untuk prediksi secara real time dengan menggunakan latensi rendah`Predict`API. Parameter`Predict`operasi menerima pengamatan masukan tunggal dalam payload permintaan, dan mengembalikan prediksi serentak dalam respon. Ini membedakannya dari API prediksi batch, yang dipanggil dengan ID objek sumber data Amazon MLyang menunjuk ke lokasi pengamatan masukan,

dan secara asinkron mengembalikan URI ke file yang berisi prediksi untuk semua pengamatan ini. Amazon ML-merespon sebagian besar permintaan prediksi real-time dalam 100 milidetik.

Anda dapat mencoba prediksi real-time tanpa menimbulkan biaya di konsol Amazon ML-nya. Jika Anda kemudian memutuskan untuk menggunakan prediksi real-time, Anda harus terlebih dahulu membuat endpoint untuk generasi prediksi real-time. Anda dapat melakukannya di konsol Amazon MLnya atau menggunakan konsol Amazon`CreateRealtimeEndpoint`API. Setelah Anda memiliki titik akhir, gunakan API prediksi real-time untuk menghasilkan prediksi real-time.

> **Note**
>
> Setelah Anda membuat titik akhir real-time untuk model Anda, Anda akan mulai menimbulkan biaya reservasi kapasitas yang didasarkan pada ukuran model. Untuk informasi selengkapnya, lihat Harga. Jika Anda membuat titik akhir real-time di konsol, konsol akan menampilkan rincian perkiraan biaya yang akan diperoleh endpoint secara berkelanjutan. Untuk berhenti menimbulkan biaya ketika Anda tidak perlu lagi mendapatkan prediksi real-time dari model itu, hapus titik akhir real-time dengan menggunakan konsol atau`DeleteRealtimeEndpoint`operasi.

Sebagai contoh`Predict`permintaan dan tanggapan, lihatPrediksi PrediksidiReferensi API Amazon Machine Learning. Untuk melihat contoh format respons yang tepat yang menggunakan model Anda, lihatPrediksi Waktu Nyata.

Topik

- Prediksi Waktu Nyata
- Membuat Titik Akhir Waktu Nyata
- Menemukan Titik Akhir Prediksi Real-Time (Konsol)
- Menemukan Real-Time Prediction Endpoint (API)
- Membuat Permintaan Prediksi Real-Time
- Menghapus Titik Akhir Waktu Nyata

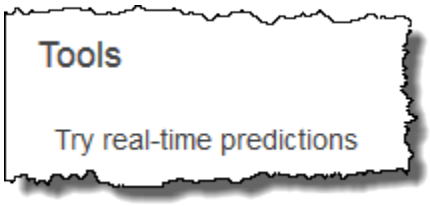
Prediksi Waktu Nyata

Untuk membantu Anda memutuskan apakah akan mengaktifkan prediksi real-time, Amazon XML memungkinkan Anda mencoba membuat prediksi pada data tunggal tanpa menimbulkan biaya

tambahan yang terkait dengan menyiapkan titik akhir prediksi real-time. Untuk mencoba prediksi real-time, Anda harus memiliki model ML-nya. Untuk membuat prediksi real-time pada skala yang lebih besar, gunakanPrediksi PrediksiAPI diReferensi API Amazon Machine Learning.

Untuk mencoba prediksi real-time

1. Masuk keAWS Management Consoledan buka konsol Amazon Machine Learning dihttps://console.aws.amazon.com/machinelearning/.
2. Di bilah navigasi, di bilah navigasiAmazon Machine Learningdrop down, pilihModel L.
3. Pilih model yang ingin Anda gunakan untuk mencoba prediksi real-time, sepertiSubscription propensity modeldari tutorial.
4. Pada halaman laporan model ML, di bawahPrediksi, pilihRingkasan, dan kemudian pilihPrediksi real-time.



   Amazon IL menampilkan daftar variabel yang membentuk catatan data yang digunakan Amazon ML-nya untuk melatih model Anda.
5. Anda dapat melanjutkan dengan memasukkan data di masing-masing bidang dalam formulir atau dengan menempelkan catatan data tunggal, dalam format CSV, ke dalam kotak teks.

   Untuk menggunakan formulir, untuk masing-masingNilailapangan, masukkan data yang ingin Anda gunakan untuk menguji prediksi real-time Anda. Jika catatan data yang Anda masukkan tidak berisi nilai untuk satu atau lebih atribut data, biarkan bidang entri kosong.

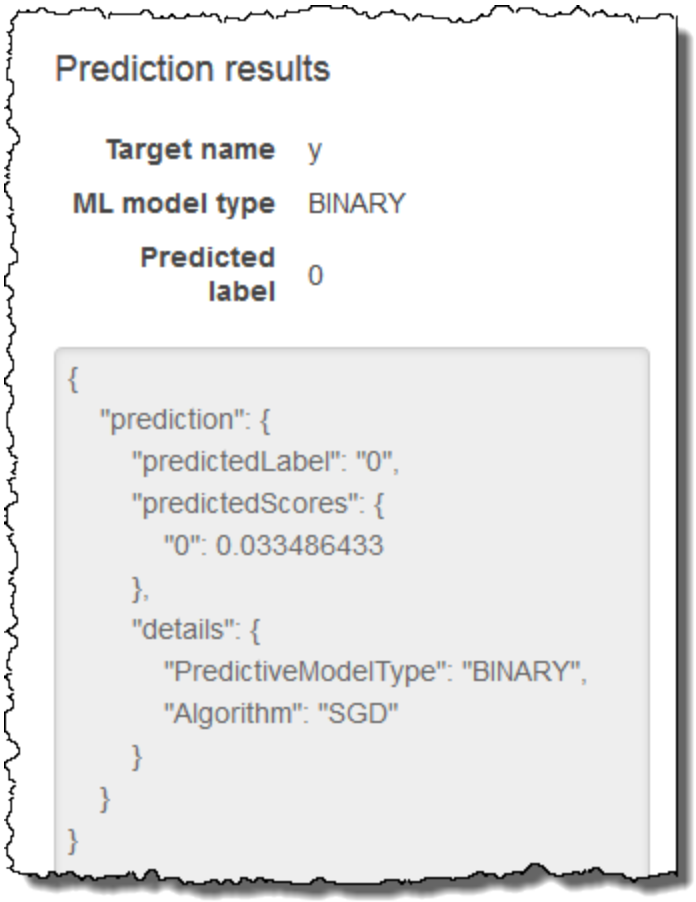
   Untuk menyediakan data record, pilihTempel catatan. Tempel satu baris data yang diformat CSV ke dalam bidang teks, dan pilihKirim. Amazon ML-mengisi secara otomatisNilaibidang untuk Anda.

   > **Note**
   >
   > Data dalam catatan data harus memiliki jumlah kolom yang sama dengan data pelatihan, dan diatur dalam urutan yang sama. Satu-satunya pengecualian adalah bahwa Anda

harus menghilangkan nilai target. Jika Anda menyertakan nilai target, Amazon MLnya mengabaikannya.

6. Di bagian bawah halaman, pilihPrediksi membuat. Amazon ML-segera mengembalikan prediksi.

   DiHasil prediksipanel, Anda melihat objek prediksi bahwa`Predict`Pengembalian panggilan API, bersama dengan tipe model L, nama variabel target, dan kelas atau nilai yang diprediksi. Untuk informasi tentang penafsiran hasil, lihatMenafsirkan Isi File Prediksi Batch untuk model Binary Classification.



Membuat Titik Akhir Waktu Nyata

Untuk menghasilkan prediksi real-time, Anda perlu membuat titik akhir real-time. Untuk membuat titik akhir real-time, Anda harus sudah memiliki model ML-nya yang ingin Anda hasilkan prediksi real-time. Anda dapat membuat titik akhir waktu nyata dengan menggunakan konsol Amazon IL atau dengan memanggil`CreateRealtimeEndpoint`API. Untuk informasi lebih lanjut tentang penggunaan`CreateRealtimeEndpoint`API, lihathttps://docs.aws.amazon.com/machine-learning/

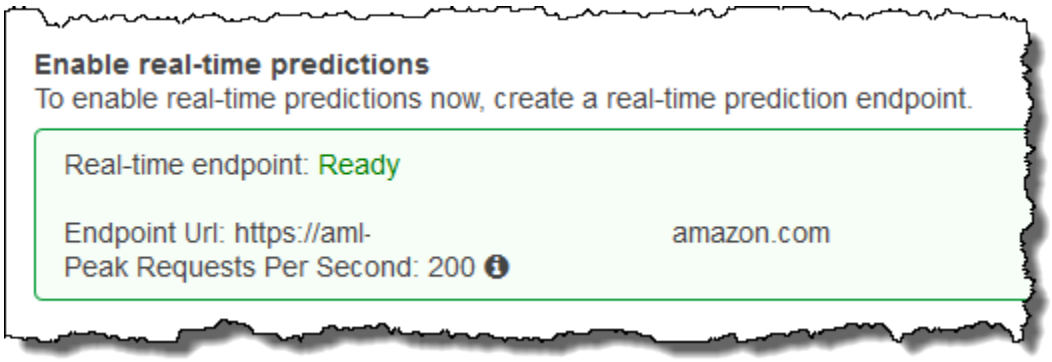
latest/APIReference/API_CreateRealtimeEndpoint.htmldalam Referensi API Amazon Machine Learning.

Untuk membuat titik akhir waktu nyata

1. Masuk keAWS Management Consoledan buka konsol Amazon Machine Learning dihttps://console.aws.amazon.com/machinelearning/.
2. Di bilah navigasi, di bilah navigasiAmazon Machine Learningdrop down, pilihModel L.
3. Pilih model yang ingin Anda hasilkan prediksi waktu nyata.
4. PadaRingkasan model MLMhalaman, di bawahPrediksi, pilihBuat titik akhir waktu nyata.

   Sebuah kotak dialog yang menjelaskan bagaimana prediksi real-time harga muncul.
5. Pilih Create (Buat). Permintaan endpoint real-time dikirim ke Amazon ML-nya dan dimasukkan ke dalam antrian. Status titik akhir waktu nyata adalahMemperbarui.

6. Ketika titik akhir real-time sudah siap, status berubah menjadiSiap, dan Amazon ML-menampilkan URL endpoint. Gunakan URL endpoint untuk membuat permintaan prediksi waktu nyata dengan`Predict`API. Untuk informasi lebih lanjut tentang penggunaan`Predict`API, lihathttps://docs.aws.amazon.com/machine-learning/latest/APIReference/API_Predict.htmldalam Referensi API Amazon Machine Learning.



Menemukan Titik Akhir Prediksi Real-Time (Konsol)

Untuk menggunakan konsol Amazon XML untuk menemukan URL endpoint untuk model ML-navigasikan ke modelRingkasan model MLMhalaman.

Untuk menemukan URL endpoint real-time

1. Masuk keAWS Management Consoledan buka konsol Amazon Machine Learning dihttps://console.aws.amazon.com/machinelearning/.
2. Di bilah navigasi, di bilah navigasiAmazon Machine Learningdrop down, pilihModel L.
3. Pilih model yang ingin Anda hasilkan prediksi waktu nyata.
4. PadaRingkasan model MLMhalaman, gulir ke bawah sampai Anda melihatPrediksibagian.
5. URL endpoint untuk model tercantum dalamPrediksi real-time. Gunakan URL sebagaiUrl Titik AkhirURL untuk panggilan prediksi waktu nyata Anda. Untuk informasi tentang cara menggunakan titik akhir untuk menghasilkan prediksi, lihathttps://docs.aws.amazon.com/machine-learning/latest/APIReference/API_Predict.htmldalam Referensi API Amazon Machine Learning.

Menemukan Real-Time Prediction Endpoint (API)

Saat Anda membuat titik akhir waktu nyata dengan menggunakan`CreateRealtimeEndpoint`operasi, URL dan status endpoint dikembalikan kepada Anda dalam respon. Jika Anda membuat titik akhir real-time dengan menggunakan konsol atau jika Anda ingin mengambil URL dan status titik akhir yang Anda buat sebelumnya, hubungi`GetMLModel`operasi dengan ID model yang ingin Anda kueri untuk prediksi waktu nyata. Informasi endpoint terkandung dalam`EndpointInfo`bagian dari respon. Untuk model yang memiliki titik akhir real-time yang terkait dengannya,`EndpointInfo`mungkin terlihat seperti ini:

```
"EndpointInfo":{
    "CreatedAt": 1427864874.227,
    "EndpointStatus": "READY",
    "EndpointUrl": "https://endpointUrl",
    "PeakRequestsPerSecond": 200
}
```

Sebuah model tanpa titik akhir real-time akan mengembalikan yang berikut ini:

```
EndpointInfo":{
```

```
    "EndpointStatus": "NONE",
    "PeakRequestsPerSecond": 0
}
```

Membuat Permintaan Prediksi Real-Time

SampelPredictmuatan permintaan mungkin terlihat seperti ini:

```
{
    "MLModelId": "model-id",
    "Record":{
        "key1": "value1",
        "key2": "value2"
    },
    "PredictEndpoint": "https://endpointUrl"
}
```

ParameterPredictEndpointbidang harus sesuai denganEndpointUrlbidangEndpointInfostruktur. Amazon IL menggunakan bidang ini untuk merutekan permintaan ke server yang sesuai dalam armada prediksi real-time.

ParameterMLModelIdadalah pengenal model yang terlatih sebelumnya dengan titik akhir real-time.

SEBUAHRecordadalah peta nama variabel untuk nilai-nilai variabel. Setiap pasangan mewakili pengamatan. ParameterRecordpeta berisi input ke model Amazon ML-mu. Hal ini analog dengan satu baris data dalam set data pelatihan Anda, tanpa variabel target. Terlepas dari jenis nilai dalam data pelatihan,Recordberisi pemetaan string-ke-string.

> **Note**
>
> Anda dapat menghilangkan variabel yang Anda tidak memiliki nilai, meskipun ini mungkin mengurangi keakuratan prediksi Anda. Semakin banyak variabel yang dapat Anda sertakan, semakin akurat model Anda.

Format respon yang dikembalikan olehPredictpermintaan tergantung pada jenis model yang sedang ditanyakan untuk prediksi. Dalam semua kasus,detailsbidang berisi informasi tentang permintaan prediksi, terutama termasukPredictiveModelTypebidang dengan tipe model.

Contoh berikut menunjukkan respons untuk model biner:

```
{
    "Prediction":{
        "details":{
            "PredictiveModelType": "BINARY"
        },
        "predictedLabel": "0",
        "predictedScores":{
            "0": 0.47380468249320984
        }
    }
}
```

Perhatikan`predictedLabel`bidang yang berisi label diprediksi, dalam hal ini 0. Amazon IL menghitung label yang diprediksi dengan membandingkan skor prediksi terhadap cut-off klasifikasi:

- Anda dapat memperoleh klasifikasi cut-off yang saat ini terkait dengan model ML dengan memeriksa`ScoreThreshold`bidang dalam respon`GetMLModel`operasi, atau dengan melihat informasi model di konsol Amazon IL. Jika Anda tidak menetapkan ambang skor, Amazon IL menggunakan nilai default yaitu 0,5.
- Anda dapat memperoleh skor prediksi yang tepat untuk model klasifikasi biner dengan memeriksa`predictedScores`peta. Dalam peta ini, label yang diprediksi dipasangkan dengan skor prediksi yang tepat.

Untuk informasi selengkapnya tentang prediksi biner, lihatMenafsirkan Prediksi.

Contoh berikut menunjukkan respons untuk model regresi. Perhatikan bahwa nilai numerik yang diprediksi ditemukan di`predictedValue`Bidang:

```
{
    "Prediction":{
        "details":{
            "PredictiveModelType": "REGRESSION"
        },
        "predictedValue": 15.508452415466309
    }
}
```

Contoh berikut menunjukkan respons untuk model multiclass:

```
{
```

```
    "Prediction":{
        "details":{
            "PredictiveModelType": "MULTICLASS"
        },
        "predictedLabel": "red",
        "predictedScores":{
            "red": 0.12923571467399597,
            "green": 0.08416014909744263,
            "orange": 0.22713537514209747,
            "blue": 0.1438363939523697,
            "pink": 0.184102863073349,
            "violet": 0.12816807627677917,
            "brown": 0.10336143523454666
        }
    }
}
```

Mirip dengan model klasifikasi biner, label/kelas yang diprediksi ditemukan di`predictedLabel`bidang. Anda dapat lebih memahami seberapa kuat prediksi terkait dengan setiap kelas dengan melihat`predictedScores`peta. Semakin tinggi skor kelas dalam peta ini, semakin kuat prediksi terkait dengan kelas, dengan nilai tertinggi akhirnya dipilih sebagai`predictedLabel`.

Untuk informasi selengkapnya tentang prediksi multiclass, lihatWawasan Model Multiclass.

Menghapus Titik Akhir Waktu Nyata

Ketika Anda telah menyelesaikan prediksi real-time, hapus titik akhir waktu nyata untuk menghindari biaya tambahan. Biaya berhenti bertambah segera setelah Anda menghapus titik akhir Anda.

Untuk menghapus titik akhir real-time

1. Masuk keAWS Management Consoledan buka konsol Amazon Machine Learning dihttps://console.aws.amazon.com/machinelearning/.
2. Di bilah navigasi, di bilah navigasiAmazon Machine Learningdrop down, pilihModel L.
3. Pilih model yang tidak lagi membutuhkan prediksi real-time.
4. Pada halaman laporan model ML, di bawahPrediksi, pilihRingkasan.
5. PilihMenghapus titik akhir waktu nyata.
6. DiMenghapus titik akhir waktu nyatakotak dialog, pilihHapus.

Mengelola Objek Amazon XML

Amazon ML-menyediakan empat objek yang dapat Anda kelola melalui konsol Amazon ML-nya atau API Amazon ML-nya:

- Sumber data
- Model L
- evaluasi
- Prediksi Batch

Setiap objek menyajikan tujuan yang berbeda dalam siklus hidup membangun aplikasi pembelajaran mesin, dan setiap objek memiliki atribut dan fungsionalitas khusus yang hanya berlaku untuk objek tersebut. Terlepas dari perbedaan ini, Anda mengelola objek dengan cara yang sama. Misalnya, Anda menggunakan proses yang hampir identik untuk mencantumkan objek, mengambil deskripsi mereka, dan memperbarui atau menghapusnya.

Bagian berikut menggambarkan operasi manajemen yang umum untuk keempat objek dan mencatat perbedaan.

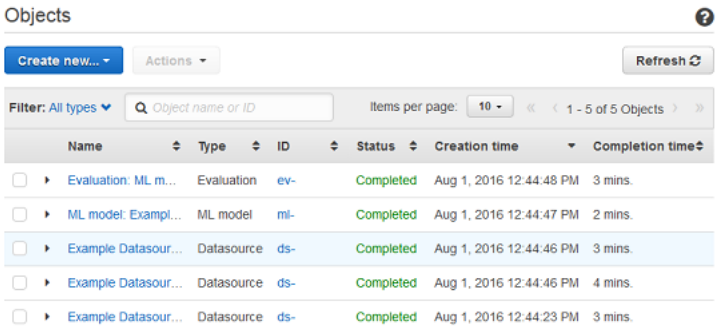
Topik

Daftar Objek

Untuk informasi mendalam tentang sumber data Amazon Machine Learning (Amazon ML-mu), model, evaluasi, dan prediksi batch, cantumkan daftar data tersebut. Untuk setiap objek, Anda akan melihat nama, jenis, ID, kode status, dan waktu pembuatannya. Anda juga dapat melihat rincian yang spesifik untuk jenis objek tertentu. Misalnya, Anda dapat melihat wawasan data untuk sumber data.

Listing Objects (Console)

Untuk melihat daftar 1.000 objek terakhir yang telah Anda buat, di konsol Amazon IL, bukaObjekdasbor. Untuk menampilkanObjekdasbor, masuk ke konsol Amazon ML-nya.



Untuk melihat detail lebih lanjut tentang objek, termasuk detail yang spesifik untuk jenis objek tersebut, pilih nama atau ID objek. Misalnya, untuk melihatWawasan datauntuk datasource, pilih nama datasource.

Kolom padaObjekdashboard menunjukkan informasi berikut tentang setiap objek.

Nama

Nama objek.

Jenis

Jenis objek. Nilai yang valid meliputiSumber data,Model ML,evaluasi, danPrediksi Batch.

> **Note**
>
> Untuk melihat apakah model diatur untuk mendukung prediksi real-time, bukaRingkasan model MLhalaman dengan memilih nama atau model ID.

ID

ID projek.

Status

Status objek. Nilai termasukTertunda,Dalam Progres,Completed (Lengkap), danGagal. Jika statusnyaGagal, periksa data Anda dan coba lagi.

Waktu pembuatan

Tanggal dan waktu ketika Amazon IL selesai membuat objek ini.

Waktu penyelesaian

Lamanya waktu yang dibutuhkan Amazon ML-nya untuk membuat objek ini. Anda dapat menggunakan waktu penyelesaian model untuk memperkirakan waktu pelatihan untuk model baru.

ID sumber data

Untuk objek yang dibuat menggunakan sumber data, seperti model dan evaluasi, ID dari sumber data. Jika Anda menghapus sumber data, Anda tidak dapat lagi menggunakan model L yang dibuat dengan sumber data tersebut untuk membuat prediksi.

Urutkan berdasarkan kolom apa pun dengan memilih ikon segitiga ganda di sebelah header kolom.

Listing Objek (API)

DiAPI Amazon ML, Anda dapat daftar objek, menurut jenis, dengan menggunakan operasi berikut:

- `DescribeDataSources`
- `DescribeMLModels`
- `DescribeEvaluations`
- `DescribeBatchPredictions`

Setiap operasi mencakup parameter untuk penyaringan, penyortiran, dan paginating melalui daftar panjang objek. Tidak ada batasan jumlah objek yang dapat Anda akses melalui API. Untuk membatasi ukuran daftar, gunakan`Limit`parameter, yang dapat mengambil nilai maksimum 100.

Respon API untuk`Describe*`perintah termasuk token pagination (`nextPageToken`), jika sesuai, dan deskripsi singkat dari setiap objek. Deskripsi objek mencakup informasi yang sama untuk masing-masing jenis objek yang ditampilkan di konsol, termasuk rincian yang spesifik untuk jenis objek.

> **Note**
>
> Bahkan jika respon mencakup objek yang lebih sedikit dari batas yang ditentukan, mungkin termasuk`nextPageToken`yang menunjukkan bahwa lebih banyak hasil yang tersedia. Bahkan respon yang berisi 0 item mungkin berisi`nextPageToken`.

Untuk informasi selengkapnya, lihatReferensi API Amazon ML.

Mengambil Deskripsi Objek

Anda dapat melihat deskripsi rinci objek apa pun melalui konsol atau melalui API.

Deskripsi Terperinci di Konsol

Untuk melihat deskripsi di konsol, arahkan ke daftar untuk jenis objek tertentu (sumber data, model L, evaluasi, atau prediksi batch). Selanjutnya, cari baris dalam tabel yang sesuai dengan objek, baik dengan menelusuri daftar atau dengan mencari nama atau ID.

Deskripsi rinci dari API

Setiap jenis objek memiliki operasi yang mengambil rincian lengkap dari objek Amazon ML-nya:

- GetDataSource
- getMLModel
- GetEvaluation
- GetBatchPrediction

Setiap operasi mengambil tepat dua parameter: ID objek dan bendera Boolean disebut Verbose. Panggilan dengan Verbose diatur ke true akan mencakup detail tambahan tentang objek, menghasilkan latensi yang lebih tinggi dan respons yang lebih besar. Untuk mempelajari bidang mana yang disertakan dengan menyetel bendera Verbose, lihatReferensi API Amazon.

Memperbarui Objek

Setiap jenis objek memiliki operasi yang memperbarui rincian objek Amazon ML-nya (LihatReferensi API Amazon):

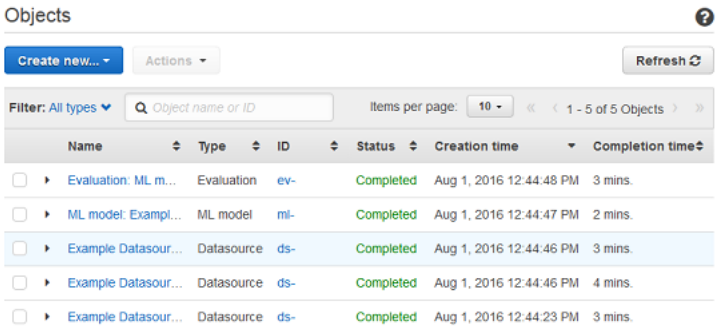
- UpdateDataSource
- UpdateMLModel
- UpdateEvaluation
- UpdateBatchPrediction

Setiap operasi memerlukan ID objek untuk menentukan objek yang sedang diperbarui. Anda dapat memperbarui nama semua objek. Anda tidak dapat memperbarui properti objek lainnya untuk sumber data, evaluasi, dan prediksi batch. Untuk Model L, Anda dapat memperbarui bidang ScoreThreshold, selama model MLnya tidak memiliki titik akhir prediksi real-time yang terkait dengannya.

Menghapus Object

Jika Anda tidak lagi memerlukan sumber data, model, evaluasi, dan prediksi batch, Anda dapat menghapusnya. Meskipun tidak ada biaya tambahan untuk menjaga objek Amazon ML-selain prediksi batch setelah Anda selesai dengan mereka, menghapus objek membuat ruang kerja Anda rapi dan lebih mudah dikelola. Anda dapat menghapus objek tunggal atau beberapa menggunakan konsol Amazon Machine Learning (Amazon ML) atau API.

> **Warning**
>
> Saat Anda menghapus objek Amazon ML-nya, efeknya langsung, permanen, dan tidak dapat diubah.



Menghapus Objects (Konsol)

Anda dapat menggunakan konsol Amazon ML-untuk menghapus objek, termasuk model. Prosedur yang Anda gunakan untuk menghapus model tergantung pada apakah Anda menggunakan model

untuk menghasilkan prediksi real-time atau tidak. Untuk menghapus model yang digunakan untuk menghasilkan prediksi real-time, pertama hapus titik akhir real-time.

Cara menghapus objek Amazon ML-nya (konsol)

1. Masuk keAWS Management Consoledan buka konsol Amazon Machine Learning dihttps://console.aws.amazon.com/machinelearning/.
2. Pilih objek Amazon ML-yang ingin Anda hapus. Untuk memilih lebih dari satu objek, gunakan tombol SHIFT. Untuk membatalkan pilihan semua objek yang dipilih, gunakan 
3. Untuk Tindakan, pilih Hapus.
4. Di kotak dialog, pilihHapusuntuk menghapus model.

Untuk menghapus model Amazon ML-dengan titik akhir real-time (konsol)

1. Masuk keAWS Management Consoledan buka konsol Amazon Machine Learning dihttps://console.aws.amazon.com/machinelearning/.
2. Pilih model yang ingin Anda hapus.
3. UntukTindakan, pilihMenghapus titik akhir waktu nyata.
4. PilihHapusuntuk menghapus endpoint.
5. Pilih model lagi.
6. Untuk Tindakan, pilih Hapus.
7. PilihHapusuntuk menghapus model.

Menghapus Objects (API)

Anda dapat menghapus objek Amazon ML-nya menggunakan panggilan API berikut:

- `DeleteDataSource`- Membawa parameter`DataSourceId`.
- `DeleteMLModel`- Membawa parameter`MLModelId`.
- `DeleteEvaluation`- Membawa parameter`EvaluationId`.
- `DeleteBatchPrediction`- Membawa parameter`BatchPredictionId`.

Untuk informasi selengkapnya, lihatReferensi API Amazon Machine Learning.

Amazon ML-nya dengan Amazon CloudWatch

Amazon ML-otomatis mengirimkan metrik ke Amazon CloudWatch sehingga Anda dapat mengumpulkan dan menganalisis statistik penggunaan untuk model ML-mu. Misalnya, untuk melacak prediksi batch dan real-time, Anda dapat memantau metrik PredictCount sesuai dengan dimensi RequestMode. Metrik secara otomatis dikumpulkan dan dikirim ke Amazon CloudWatch setiap lima menit. Anda dapat memantau metrik ini dengan menggunakan Amazon CloudWatch console, AWS CLI, atau AWS SDKs.

Tidak ada biaya untuk metrik Amazon ML-nya yang dilaporkan melalui CloudWatch. Jika Anda mengatur alarm pada metrik, Anda akan ditagih sesuai standartarif CloudWatch.

Untuk informasi selengkapnya, lihat daftar metrik Amazon ML-nyaReferensi Namespace, Dimensi, dan Metrik Amazon CloudWatchdi Panduan Pengembang Amazon CloudWatch.

Mencatat Panggilan API Amazon denganAWS CloudTrail

Amazon Machine Learning (Amazon ML) terintegrasi denganAWS CloudTrail, layanan yang menyediakan catatan tindakan yang diambil oleh pengguna, peran, atauAWSlayanan di Amazon ML-nya. CloudTrail menangkap semua panggilan API untuk Amazon ML-kejadian. Panggilan yang direkam mencakup panggilan dari konsol Amazon ML-dan panggilan kode ke operasi Amazon ML-API. Jika membuat jejak, Anda dapat mengaktifkan pengiriman peristiwa CloudTrail berkelanjutan ke bucket Amazon S3, termasuk kejadian untuk Amazon ML-nya. Jika Anda tidak mengonfigurasi jejak, Anda masih dapat melihat peristiwa terbaru dalam konsol CloudTrail di Riwayat peristiwa. Menggunakan informasi yang dikumpulkan oleh CloudTrail, Anda dapat menentukan permintaan yang diajukan ke Amazon ML. alamat IP asal permintaan tersebut dibuat, kapan dibuat, dan detail lainnya.

Untuk mempelajari CloudTrail selengkapnya, termasuk cara mengonfigurasi dan mengaktifkannya, lihat AWS CloudTrail Panduan Pengguna.

Informasi Amazon di CloudTrail

CloudTrail diaktifkan pada akun AWS Anda saat Anda membuat akun tersebut. Saat aktivitas peristiwa yang didukung terjadi di Amazon ML-nya, aktivitas tersebut dicatat di peristiwa CloudTrail bersama dengan aktivitas peristiwa lainnyaAWSperistiwa layanan diRiwayat peristiwa. Anda dapat melihat, mencari, dan mengunduh peristiwa terbaru di akun AWS Anda. Untuk informasi selengkapnya, lihat Melihat Peristiwa dengan Riwayat Peristiwa CloudTrail.

Untuk catatan peristiwa yang sedang berlangsung diAWSakun, termasuk peristiwa untuk Amazon ML-nya, buatlah jejak. Sebuah Jejak mengaktifkan CloudTrail untuk mengirim berkas log ke bucket Amazon S3. Secara default, ketika Anda membuat jejak di konsol tersebut, jejak tersebut diterapkan ke semua Wilayah AWS. Log acara jejak dari semua Wilayah di partisi AWS dan mengirimkan berkas log ke bucket Amazon S3 yang Anda tentukan. Selain itu, Anda dapat mengonfigurasi layanan AWS lainnya untuk menganalisis lebih lanjut dan bertindak berdasarkan data peristiwa yang dikumpulkan di log CloudTrail. Untuk informasi selengkapnya, lihat yang berikut:

- Ikhtisar untuk Membuat Jejak
- Layanan yang Didukung dan Integrasi CloudTrail
- Mengonfigurasi Notifikasi Amazon SNS untuk CloudTrail
- Menerima Berkas Log CloudTrail dari Berbagai Wilayah dan Menerima Berkas Log CloudTrail dari Berbagai Akun

Amazon mendukung pencatatan tindakan berikut sebagai kejadian di file log CloudTrail:

- AddTags
- CreateBatchPrediction
- CreateDataSourceFromRDS
- CreateDataSourceFromRedshift
- dibuatasourcefroms3
- CreateEvaluation
- CreateMLModel
- CreateRealtimeEndpoint
- DeleteBatchPrediction
- DeleteDataSource
- DeleteEvaluation
- DeleteMLModel
- DeleteRealtimeEndpoint
- DeleteTags
- DescribeTags
- UpdateBatchPrediction
- UpdateDataSource
- UpdateEvaluation
- UpdateMLModel

Operasi Amazon XML berikut ini menggunakan parameter permintaan yang berisi kredensi. Sebelum permintaan ini dikirim ke CloudTrail, kredensialnya diganti dengan tiga tanda bintang (“***”):

- CreateDataSourceFromRDS
- CreateDataSourceFromRedshift

Saat operasi Amazon ML-nya dilakukan dengan konsol Amazon MLnya, atributnya`ComputeStatistics`tidak termasuk dalam`RequestParameters`komponen log CloudTrail:

- CreateDataSourceFromRedshift

- dibuatasourcefroms3

Setiap entri peristiwa atau log berisi informasi tentang siapa yang membuat permintaan tersebut. Informasi identitas membantu Anda menentukan hal berikut:

- Bahwa permintaan dibuat dengan kredensial pengguna root atau pengguna AWS Identity and Access Management (IAM).
- Bahwa permintaan tersebut dibuat dengan kredensial keamanan sementara untuk peran atau pengguna gabungan.
- Apakah permintaan dibuat oleh layanan AWS lain.

Untuk informasi lebih lanjut, lihat Elemen userIdentity CloudTrail.

Contoh: Entri Berkas Log

Jejak adalah konfigurasi yang memungkinkan pengiriman peristiwa sebagai berkas log ke bucket Amazon S3 yang telah Anda tentukan. File log CloudTrail berisi satu atau beberapa entri log. Peristuwa mewakili satu permintaan dari sumber apa pun dan mencakup informasi tentang tindakan yang diminta, tanggal dan waktu tindakan, parameter permintaan, dan sebagainya. Berkas log CloudTrail bukanlah pelacakan tumpukan terurut dari panggilan API publik, sehingga tidak muncul dalam urutan tertentu.

Contoh berikut menunjukkan entri log CloudTrail yang menunjukkan tindakan .

```
{
    "Records": [
        {
            "eventVersion": "1.03",
            "userIdentity": {
                "type": "IAMUser",
                "principalId": "EX_PRINCIPAL_ID",
                "arn": "arn:aws:iam::012345678910:user/Alice",
                "accountId": "012345678910",
                "accessKeyId": "EXAMPLE_KEY_ID",
                "userName": "Alice"
            },
            "eventTime": "2015-11-12T15:04:02Z",
            "eventSource": "machinelearning.amazonaws.com",
```

```
            "eventName": "CreateDataSourceFromS3",
            "awsRegion": "us-east-1",
            "sourceIPAddress": "127.0.0.1",
            "userAgent": "console.amazonaws.com",
            "requestParameters": {
                "data": {
                    "dataLocationS3": "s3://aml-sample-data/banking-batch.csv",
                    "dataSchema": "{\"version\":\"1.0\",\"rowId\":null,\"rowWeight
\":null,
                        \"targetAttributeName\":null,\"dataFormat\":\"CSV\",
                        \"dataFileContainsHeader\":false,\"attributes\":[
                          {\"attributeName\":\"age\",\"attributeType\":\"NUMERIC\"},
                          {\"attributeName\":\"job\",\"attributeType\":\"CATEGORICAL
\"},
                          {\"attributeName\":\"marital\",\"attributeType\":
\"CATEGORICAL\"},
                          {\"attributeName\":\"education\",\"attributeType\":
\"CATEGORICAL\"},
                          {\"attributeName\":\"default\",\"attributeType\":
\"CATEGORICAL\"},
                          {\"attributeName\":\"housing\",\"attributeType\":
\"CATEGORICAL\"},
                          {\"attributeName\":\"loan\",\"attributeType\":\"CATEGORICAL
\"},
                          {\"attributeName\":\"contact\",\"attributeType\":
\"CATEGORICAL\"},
                          {\"attributeName\":\"month\",\"attributeType\":\"CATEGORICAL
\"},
                          {\"attributeName\":\"day_of_week\",\"attributeType\":
\"CATEGORICAL\"},
                          {\"attributeName\":\"duration\",\"attributeType\":\"NUMERIC
\"},
                          {\"attributeName\":\"campaign\",\"attributeType\":\"NUMERIC
\"},
                          {\"attributeName\":\"pdays\",\"attributeType\":\"NUMERIC\"},
                          {\"attributeName\":\"previous\",\"attributeType\":\"NUMERIC
\"},
                          {\"attributeName\":\"poutcome\",\"attributeType\":
\"CATEGORICAL\"},
                          {\"attributeName\":\"emp_var_rate\",\"attributeType\":
\"NUMERIC\"},
                          {\"attributeName\":\"cons_price_idx\",\"attributeType\":
\"NUMERIC\"},
```

```
                            {\"attributeName\":\"cons_conf_idx\",\"attributeType\":
\"NUMERIC\"},
                            {\"attributeName\":\"euribor3m\",\"attributeType\":\"NUMERIC
\"},
                            {\"attributeName\":\"nr_employed\",\"attributeType\":
\"NUMERIC\"}
                          ],\"excludedAttributeNames\":[]}"
                },
                "dataSourceId": "exampleDataSourceId",
                "dataSourceName": "Banking sample for batch prediction"
            },
            "responseElements": {
                "dataSourceId": "exampleDataSourceId"
            },
            "requestID": "9b14bc94-894e-11e5-a84d-2d2deb28fdec",
            "eventID": "f1d47f93-c708-495b-bff1-cb935a6064b2",
            "eventType": "AwsApiCall",
            "recipientAccountId": "012345678910"
        },
        {
            "eventVersion": "1.03",
            "userIdentity": {
                "type": "IAMUser",
                "principalId": "EX_PRINCIPAL_ID",
                "arn": "arn:aws:iam::012345678910:user/Alice",
                "accountId": "012345678910",
                "accessKeyId": "EXAMPLE_KEY_ID",
                "userName": "Alice"
            },
            "eventTime": "2015-11-11T15:24:05Z",
            "eventSource": "machinelearning.amazonaws.com",
            "eventName": "CreateBatchPrediction",
            "awsRegion": "us-east-1",
            "sourceIPAddress": "127.0.0.1",
            "userAgent": "console.amazonaws.com",
            "requestParameters": {
                "batchPredictionName": "Batch prediction: ML model: Banking sample",
                "batchPredictionId": "exampleBatchPredictionId",
                "batchPredictionDataSourceId": "exampleDataSourceId",
                "outputUri": "s3://EXAMPLE_BUCKET/BatchPredictionOutput/",
                "mLModelId": "exampleModelId"
            },
            "responseElements": {
                "batchPredictionId": "exampleBatchPredictionId"
```

```
            },
            "requestID": "3e18f252-8888-11e5-b6ca-c9da3c0f3955",
            "eventID": "db27a771-7a2e-4e9d-bfa0-59deee9d936d",
            "eventType": "AwsApiCall",
            "recipientAccountId": "012345678910"
        }
    ]
}
```

Menandai Objek Amazon ML-mu

Atur dan kelola objek Amazon Machine Learning (Amazon ML) dengan menetapkan metadata ke objek tersebut dengan tag. SEBUAHmenandaiadalah pasangan nilai-kunci yang Anda tetapkan untuk sebuah objek.

Selain menggunakan tag untuk mengatur dan mengelola objek Amazon ML-mu, Anda dapat menggunakannya untuk mengkategorikan dan melacak biaya AWS Anda. Saat Anda menerapkan tanda ke objek AWS, termasuk model ML-nya, laporan alokasi biaya AWS Anda mencakup penggunaan dan biaya yang diagregasikan berdasarkan tanda. Dengan menerapkan tanda yang mewakili kategori bisnis (seperti pusat biaya, nama aplikasi, atau pemilik), Anda dapat mengatur biaya di berbagai layanan. Untuk informasi selengkapnya, lihat Menggunakan Tanda Alokasi Biaya untuk Laporan Penagihan Khusus dalam Panduan Pengguna AWS Billing.

Isi

Dasar-Dasar Tanda

Gunakan tag untuk mengkategorikan objek Anda agar lebih mudah mengelolanya. Misalnya, Anda dapat mengategorikan objek berdasarkan tujuan, pemilik, atau lingkungan. Kemudian, Anda dapat menentukan satu set tanda yang membantu Anda melacak model berdasarkan pemilik dan aplikasi terkait. Berikut adalah beberapa contoh:

- Proyek: Nama Proyek
- Pemilik: Nama
- Tujuan: Prediksi pemasaran
- Aplikasi: Nama aplikasi
- Lingkungan: Produksi

Anda menggunakan konsol atau API Amazon ML-nya untuk menyelesaikan tugas-tugas berikut:

- Tambahkan tanda ke objek
- Melihat tag untuk objek Anda
- Mengedit tag untuk objek Anda
- Menghapus tanda dari sebuah objek

Secara default, tag yang diterapkan ke objek Amazon ML-disalin ke objek yang dibuat menggunakan objek tersebut. Misalnya, jika sebuah sumber data Amazon Simple Storage Service (Amazon S3) memiliki “Biaya pemasaran: Tag kampanye pemasaran yang ditargetkan, model yang dibuat menggunakan sumber data itu juga akan memiliki “Biaya pemasaran: Target kampanye pemasaran” tag, seperti halnya evaluasi untuk model. Hal ini memungkinkan Anda untuk menggunakan tag untuk melacak objek terkait, seperti semua objek yang digunakan untuk kampanye pemasaran. Jika ada konflik antara sumber tag, seperti model dengan tag “Biaya pemasaran: Target kampanye pemasaran” dan sumber data dengan tag “Biaya pemasaran: Target pelanggan pemasaran”, Amazon ML-menerapkan tag dari model.

Pembatasan Tag

Batasan berikut berlaku untuk tanda.

Batasan dasar:

- Jumlah maksimum tag per objek adalah 50.
- Kunci dan nilai tag peka huruf besar dan kecil.
- Anda tidak dapat mengubah atau mengedit tanda untuk objek yang dihapus.

Batasan kunci tanda:

- Setiap kunci tanda harus unik. Jika Anda menambahkan tanda dengan kunci yang sudah digunakan, tanda baru akan menimpa pasangan nilai-kunci yang sudah ada untuk objek tersebut.
- Anda tidak dapat memulai kunci tag dengan`aws:`karena prefiks ini dicadangkan untuk digunakan oleh AWS. AWS membuat tanda yang dimulai dengan prefiks ini atas nama Anda, tetapi Anda tidak dapat mengedit atau menghapusnya.
- Kunci tanda harus memiliki panjang antara 1 dan 128 karakter Unicode.
- Kunci tag harus terdiri dari karakter berikut: Unicode huruf, digit, ruang putih, dan karakter khusus berikut:`_ . / = + - @`.

Batasan nilai tanda:

- Panjang nilai tanda harus antara 0 dan 255 karakter Unicode.
- Nilai tanda dapat kosong. Jika tidak, mereka harus terdiri dari karakter berikut: Huruf Unicode, digit, white space, dan salah satu karakter khusus berikut:`_ . / = + - @`.

Menandai Objek Amazon ML-nya (Konsol)

Anda dapat melihat, menambahkan, mengedit, dan menghapus tanda menggunakan konsol Amazon ML.

Untuk melihat tanda untuk sebuah objek (konsol)

1. Masuk keAWS Management Consoledan buka konsol Amazon Machine Learning dihttps://console.aws.amazon.com/machinelearning/.
2. Pada bilah navigasi, perluas pemilih wilayah dan pilih wilayah.
3. PadaObjekhalaman, pilih objek.
4. Gulir keTagbagian dari objek yang dipilih. Tanda untuk objek tersebut tercantum di bagian bawah bagian.

Untuk menambahkan tag ke objek (konsol)

1. Masuk keAWS Management Consoledan buka konsol Amazon Machine Learning dihttps://console.aws.amazon.com/machinelearning/.
2. Pada bilah navigasi, perluas pemilih wilayah dan pilih wilayah.
3. PadaObjekhalaman, pilih objek.
4. Gulir keTagbagian dari objek yang dipilih. Tanda untuk objek tersebut tercantum di bagian bawah bagian.
5. Pilih Tambahkan atau edit tag.
6. Di bawahTambahkan Tag, tentukan kunci tag diKuncibidang, opsional menentukan nilai tag diNilaibidang, dan kemudian pilihMenerapkan perubahan.

   JikaMenerapkan perubahantombol tidak aktif, kunci tanda atau nilai tanda yang Anda tentukan tidak memenuhi pembatasan tanda. Untuk informasi selengkapnya, lihat Pembatasan Tag.
7. Untuk melihat tag baru Anda dalam daftar diTagbagian, refresh halaman.

Untuk mengedit tanda (konsol)

1. Masuk keAWS Management Consoledan buka konsol Amazon Machine Learning dihttps://console.aws.amazon.com/machinelearning/.
2. Pada bilah navigasi, perluas pemilih wilayah dan pilih wilayah.
3. PadaObjekhalaman, pilih objek.
4. Gulir keTagbagian dari objek yang dipilih. Tanda untuk objek tersebut tercantum di bagian bawah bagian.
5. Pilih Tambahkan atau edit tag.
6. Di bawahTag Terapan, mengedit nilai tag diNilaibidang, dan kemudian pilihMenerapkan perubahan.

   JikaMenerapkan perubahantombol tidak aktif, nilai tanda yang Anda tentukan tidak memenuhi pembatasan tanda. Untuk informasi selengkapnya, lihat Pembatasan Tag.
7. Untuk melihat tag yang diperbarui dalam daftar diTagbagian, refresh halaman.

Untuk menghapus sebuah tag dari sebuah objek (konsol)

1. Masuk keAWS Management Consoledan buka konsol Amazon Machine Learning dihttps://console.aws.amazon.com/machinelearning/.
2. Pada bilah navigasi, perluas pemilih wilayah dan pilih wilayah.
3. PadaObjekhalaman, pilih objek.
4. Gulir keTagbagian dari objek yang dipilih. Tanda untuk objek tersebut tercantum di bagian bawah bagian.
5. Pilih Tambahkan atau edit tag.
6. Di bawahTag Terapan, pilih tanda yang ingin Anda hapus, lalu pilihMenerapkan perubahan.

Menandai Objek Amazon ML-nya (API)

Anda dapat menambahkan, mencantumkan, dan menghapus tanda menggunakan API Amazon ML-nya. Untuk contoh, lihat dokumentasi berikut:

AddTags

Menambahkan atau mengedit tag untuk objek tertentu.

[DescribeTags]

Mendaftar tanda untuk objek tertentu.

[DeleteTags]

Menghapus tanda dari objek tertentu.

Amazon Machine Learning

Topik

Pemberian Izin Amazon Amazon untuk Membaca Data Anda dari Amazon S3

Untuk membuat objek sumber data dari data input Anda di Amazon S3, Anda harus memberikan izin berikut ke lokasi S3 tempat data input Anda disimpan:

- GetObjectizin pada bucket S3 dan prefiks.
- ListBucketizin pada bucket S3. Tidak seperti tindakan lainnya,ListBucketharus diberikan izin selebar ember (bukan pada awalan). Namun, Anda dapat lingkup izin untuk awalan tertentu dengan menggunakanKondisiklausa.

Jika Anda menggunakan konsol Amazon MLuntuk membuat sumber data, izin ini dapat ditambahkan ke bucket untuk Anda. Anda akan diminta untuk mengonfirmasi apakah Anda ingin menambahkannya saat Anda menyelesaikan langkah-langkah di Wizard.Kebijakan contoh berikut menunjukkan cara memberikan izin kepada Amazon MLuntuk membaca data dari lokasi sampel s3://*examplebucket*/*exampleprefiks*, sementara scopingListBucketizin untuk hanya*exampleprefiks*jalur masukan.

```
{
    "Version": "2008-10-17",
```

```
    "Statement": [
    {
        "Effect": "Allow",
        "Principal": { "Service": "machinelearning.amazonaws.com" },
        "Action": "s3:GetObject",
        "Resource": "arn:aws:s3:::examplebucket/exampleprefix/*"
        "Condition": {
            "StringEquals": { "aws:SourceAccount": "123456789012" }
            "ArnLike": { "aws:SourceArn": "arn:aws:machinelearning:us-
east-1:123456789012:*" }
        }
    },
    {
        "Effect": "Allow",
        "Principal": {"Service": "machinelearning.amazonaws.com"},
        "Action": "s3:ListBucket",
        "Resource": "arn:aws:s3:::examplebucket",
        "Condition": {
            "StringLike": { "s3:prefix": "exampleprefix/*" }
            "StringEquals": { "aws:SourceAccount": "123456789012" }
            "ArnLike": { "aws:SourceArn": "arn:aws:machinelearning:us-
east-1:123456789012:*" }
        }
    }]
}
```

Untuk menerapkan kebijakan ini ke data Anda, Anda harus mengedit pernyataan kebijakan yang terkait dengan bucket S3 tempat data Anda disimpan.

Untuk mengedit kebijakan izin untuk bucket S3 (menggunakan konsol lama)

1. Masuk ke AWS Management Console dan buka konsol Amazon S3 di https://console.aws.amazon.com/s3/.
2. Pilih nama bucket tempat data Anda berada.
3. Pilih Properti.
4. MemiilihMengedit kebijakan bucket
5. Masukkan kebijakan yang ditunjukkan di atas, sesuaikan sesuai dengan kebutuhan Anda, lalu pilihSimpan.
6. Pilih Save (Simpan).

Untuk mengedit kebijakan izin untuk bucket S3 (menggunakan konsol baru)

1. Masuk ke AWS Management Console dan buka konsol Amazon S3 di https://console.aws.amazon.com/s3/.
2. Pilih nama bucket dan kemudian pilihIzin.
3. Pilih Kebijakan Bucket.
4. Masukkan kebijakan yang ditunjukkan di atas, sesuaikan agar sesuai dengan kebutuhan Anda.
5. Pilih Save (Simpan).

Memberikan Izin Amazon MLuntuk Prediksi Output ke Amazon S3

Untuk menampilkan hasil operasi prediksi batch ke Amazon S3, Anda harus memberikan izin berikut ke lokasi keluaran, yang disediakan sebagai masukan ke operasi Buat Prediksi Batch:

- GetObjectizin pada bucket S3 dan prefiks.
- PutObjectizin pada bucket S3 dan prefiks.
- PutObjectAclpada bucket S3 dan prefiks.
  - Amazon MLmemerlukan izin ini untuk memastikannya dapat memberikan kalenganACL bucket-owner-full-control izin ke akun AWS Anda, setelah objek dibuat.
- ListBucketizin pada bucket S3. Tidak seperti tindakan lainnya,ListBucketharus diberikan izin selebar ember (bukan pada awalan). Anda dapat, bagaimanapun, lingkup izin untuk awalan tertentu dengan menggunakanKondisiklausa.

Jika Anda menggunakan konsol Amazon MLuntuk membuat permintaan prediksi batch, izin ini dapat ditambahkan ke bucket untuk Anda. Anda akan diminta untuk mengonfirmasi apakah Anda ingin menambahkannya saat Anda menyelesaikan langkah-langkah di wizard.

Kebijakan contoh berikut menunjukkan cara memberikan izin untuk Amazon MLuntuk menulis data ke lokasi sampel s3://examplebucket/exampleprefix, sementara scopingListBucketizin untuk hanya jalur masukan exampleprefix, dan memberikan izin untuk Amazon MLuntuk mengatur menempatkan objek ACL pada awalan keluaran:

```
{
    "Version": "2008-10-17",
    "Statement": [
    {
        "Effect": "Allow",
```

```
        "Principal": { "Service": "machinelearning.amazonaws.com"},
        "Action": [
            "s3:GetObject",
            "s3:PutObject"
       ],
        "Resource": "arn:aws:s3:::examplebucket/exampleprefix/*"
        "Condition": {
            "StringEquals": { "aws:SourceAccount": "123456789012" }
            "ArnLike": { "aws:SourceArn": "arn:aws:machinelearning:us-
east-1:123456789012:*" }
        }
    },
    {
        "Effect": "Allow",
        "Principal": { "Service": "machinelearning.amazonaws.com"},
        "Action": "s3:PutObjectAcl",
        "Resource": "arn:aws:s3:::examplebucket/exampleprefix/*",
        "Condition": {
            "StringEquals": { "s3:x-amz-acl":"bucket-owner-full-control" }
            "StringEquals": { "aws:SourceAccount": "123456789012" }
           "ArnLike": { "aws:SourceArn": "arn:aws:machinelearning:us-
east-1:123456789012:*" }
       }
    },
    {
       "Effect": "Allow",
        "Principal": {"Service": "machinelearning.amazonaws.com"},
       "Action": "s3:ListBucket",
        "Resource": "arn:aws:s3:::examplebucket",
        "Condition": {
            "StringLike": { "s3:prefix": "exampleprefix/*" }
            "StringEquals": { "aws:SourceAccount": "123456789012" }
            "ArnLike": { "aws:SourceArn": "arn:aws:machinelearning:us-
east-1:123456789012:*" }
        }
    }]
}
```

Untuk menerapkan kebijakan ini ke data Anda, Anda harus mengedit pernyataan kebijakan yang terkait dengan bucket S3 tempat data Anda disimpan.

Untuk mengedit kebijakan izin untuk bucket S3 (menggunakan konsol lama)

1. Masuk ke AWS Management Console dan buka konsol Amazon S3 di https://console.aws.amazon.com/s3/.
2. Pilih nama bucket tempat data Anda berada.
3. Pilih Properti.
4. MemiilihMengedit kebijakan bucket
5. Masukkan kebijakan yang ditunjukkan di atas, sesuaikan sesuai dengan kebutuhan Anda, lalu pilihSimpan.
6. Pilih Save (Simpan).

Untuk mengedit kebijakan izin untuk bucket S3 (menggunakan konsol baru)

1. Masuk ke AWS Management Console dan buka konsol Amazon S3 di https://console.aws.amazon.com/s3/.
2. Pilih nama bucket dan kemudian pilihIzin.
3. Pilih Kebijakan Bucket.
4. Masukkan kebijakan yang ditunjukkan di atas, sesuaikan agar sesuai dengan kebutuhan Anda.
5. Pilih Save (Simpan).

Mengontrol Akses ke Sumber Daya Amazon ML-dengan IAM

AWS Identity and Access Management(IAM) memungkinkan Anda mengontrol akses ke layanan AWS dan sumber daya secara aman bagi pengguna Anda. Dengan menggunakan IAM, Anda dapat membuat dan mengelola pengguna AWS, grup, peran, serta menggunakan izin untuk mengizinkan dan menolak akses mereka ke sumber daya AWS. Dengan menggunakan IAM dengan Amazon Machine Learning (Amazon ML), Anda dapat mengontrol apakah pengguna dalam organisasi Anda dapat menggunakan sumber daya AWS tertentu dan apakah mereka dapat melakukan tugas menggunakan Tindakan API Amazon MLL.

IAM memungkinkan Anda untuk:

- Buat pengguna dan grup di bawah akun AWS Anda.
- Tetapkan kredensial keamanan unik untuk setiap pengguna di bawah akun AWS Anda
- Kontrol setiap izin pengguna untuk melakukan tugas menggunakan sumber daya AWS

- Bagikan sumber daya AWS Anda dengan mudah dengan pengguna di akun AWS Anda
- Buat peran untuk akun AWS Anda dan kelola izinnya untuk menentukan pengguna atau layanan yang dapat mengambil peran tersebut
- Anda dapat membuat peran dalam IAM dan mengelola izin untuk mengontrol operasi mana yang dapat dilakukan oleh entitas, atau layanan AWS, yang mengasumsikan peran tersebut. Anda juga dapat menentukan entitas mana yang diizinkan untuk mengambil peran.

Jika organisasi Anda sudah memiliki identitas IAM, Anda dapat menggunakannya untuk memberikan izin untuk melakukan tugas menggunakan sumber daya AWS.

Untuk informasi lebih lanjut tentang IAM, lihat Panduan Pengguna IAM.

Sintaks Kebijakan IAM

kebijakan IAM adalah dokumen JSON yang terdiri dari satu atau beberapa pernyataan. Setiap pernyataan memiliki struktur sebagai berikut:

```
{
    "Statement":[{
        "Effect":"effect",
        "Action":"action",
        "Resource":"arn",
        "Condition":{
            "condition operator":{
                "key":"value"
            }
        }
    }]
}
```

Sebuah pernyataan kebijakan mencakup elemen-elemen berikut:

- Efek:Mengontrol izin untuk menggunakan sumber daya dan tindakan API yang akan Anda tentukan nanti dalam pernyataan. Nilai yang valid adalah `Allow` dan `Deny`. Secara default, para pengguna IAM tidak memiliki izin untuk menggunakan sumber daya dan tindakan API, jadi semua permintaan akan ditolak. Eksplisit`Allow`mengesampingkan default. Eksplisit`Deny`mengesampingkan`Allows`.
- Action: Tindakan API tertentu atau tindakan yang Anda izinkan atau tolak.
- Resource: Sumber daya yang dipengaruhi oleh tindakan. Untuk menentukan sumber daya dalam pernyataan, gunakan Amazon Resource Name (ARN).

- Kondisi (opsional): Kontrol ketika kebijakan Anda akan berlaku.

Untuk menyederhanakan pembuatan dan pengelolaan kebijakan IAM, Anda dapat menggunakan AWS Policy Generator dan IAM Policy Simulator.

Menentukan Tindakan Kebijakan IAM untuk Amazon MlaMazon

Dalam sebuah pernyataan kebijakan IAM, Anda dapat menentukan tindakan API untuk layanan apa pun yang mendukung IAM. Saat Anda membuat pernyataan kebijakan untuk tindakan Amazon MLAPI, tambahkan`machinelearning:`dengan nama tindakan API, seperti yang ditunjukkan dalam contoh berikut:

- `machinelearning:CreateDataSourceFromS3`
- `machinelearning:DescribeDataSources`
- `machinelearning:DeleteDataSource`
- `machinelearning:GetDataSource`

Untuk menentukan beberapa tindakan dalam satu pernyataan, pisahkan dengan koma:

```
"Action": ["machinelearning:action1", "machinelearning:action2"]
```

Anda juga dapat menentukan beberapa tindakan menggunakan wildcard. Misalnya, Anda dapat menentukan semua tindakan yang namanya dimulai dengan kata “Get”:

```
"Action": "machinelearning:Get*"
```

Untuk menentukan semua tindakan Amazon Amazon, gunakan wildcard:

```
"Action": "machinelearning:*"
```

Untuk daftar lengkap tindakan Amazon MLAPI, lihatReferensi API Amazon Machine Learning.

Menentukan ARN untuk Sumber Daya Amazon MLdalam Kebijakan IAM

Pernyataan kebijakan IAM berlaku untuk satu sumber daya atau lebih. Anda menentukan sumber daya untuk kebijakan Anda berdasarkan ARN mereka.

Untuk menentukan ARN untuk sumber daya Amazon Amazon, gunakan format berikut:

“Sumber daya”:`arn:aws:machinelearning:region:account:resource-type/identifier`

Contoh-contoh berikut menunjukkan cara menentukan ARN umum.

ID Sumber Data:`my-s3-datasource-id`

```
"Resource":
arn:aws:machinelearning:<region>:<your-account-id>:datasource/my-s3-datasource-id
```

ID model ML:`my-ml-model-id`

```
"Resource":
arn:aws:machinelearning:<region>:<your-account-id>:mlmodel/my-ml-model-id
```

ID prediksi Batch:`my-batchprediction-id`

```
"Resource":
arn:aws:machinelearning:<region>:<your-account-id>:batchprediction/my-batchprediction-
id
```

ID evaluasi:`my-evaluation-id`

```
"Resource": arn:aws:machinelearning:<region>:<your-account-id>:evaluation/my-
evaluation-id
```

Contoh Kebijakan untuk Amazon MLs

Contoh 1: Memungkinkan pengguna membaca metadata sumber daya machine learning

Kebijakan berikut ini memungkinkan pengguna atau grup membaca metadata sumber data, model MS, prediksi batch, dan evaluasi dengan melakukanDescribeDataSources,DescribeMLModels,DescribeBatchPredictions,DescribeEvaluations,GetDat danGetEvaluationtindakan pada sumber daya yang ditentukan. Izin operasi Jelaskan * tidak dapat dibatasi pada sumber daya tertentu.

```
{
```

```
    "Version": "2012-10-17",
    "Statement": [{
        "Effect": "Allow",
        "Action": [
            "machinelearning:Get*"
        ],
        "Resource": [
            "arn:aws:machinelearning:<region>:<your-account-id>:datasource/S3-DS-ID1",
            "arn:aws:machinelearning:<region>:<your-account-id>:datasource/REDSHIFT-DS-
ID1",
            "arn:aws:machinelearning:<region>:<your-account-id>:mlmodel/ML-MODEL-ID1",
            "arn:aws:machinelearning:<region>:<your-account-id>:batchprediction/BP-
ID1",
            "arn:aws:machinelearning:<region>:<your-account-id>:evaluation/EV-ID1"
        ]
    },
    {
        "Effect": "Allow",
        "Action": [
            "machinelearning:Describe*"
        ],
        "Resource": [
            "*"
        ]
    }]
}
```

Contoh 2: Memungkinkan pengguna membuat sumber daya pembelajaran mesin

Kebijakan berikut memungkinkan pengguna atau grup untuk membuat sumber data machine learning, model ML, prediksi batch, dan evaluasi dengan melakukan`CreateDataSourceFromS3`,`CreateDataSourceFromRedshift`,`CreateDataSourceFromR` dan`CreateEvaluation`tindakan. Anda tidak dapat membatasi izin untuk tindakan ini ke sumber daya tertentu.

```
{
    "Version": "2012-10-17",
    "Statement": [{
        "Effect": "Allow",
        "Action": [
            "machinelearning:CreateDataSourceFrom*",
            "machinelearning:CreateMLModel",
            "machinelearning:CreateBatchPrediction",
```

```
            "machinelearning:CreateEvaluation"
        ],
        "Resource": [
            "*"
        ]
    }]
}
```

Contoh 3: Memungkinkan pengguna untuk membuat dan menghapus) endpoint real-time dan melakukan prediksi real-time pada model ML

Kebijakan berikut memungkinkan pengguna atau grup untuk membuat dan menghapus titik akhir waktu nyata dan melakukan prediksi waktu nyata untuk model ML tertentu dengan melakukan`CreateRealtimeEndpoint`,`DeleteRealtimeEndpoint`, dan`Predict`tindakan pada model itu.

```
{
    "Version": "2012-10-17",
    "Statement": [{
        "Effect": "Allow",
        "Action": [
            "machinelearning:CreateRealtimeEndpoint",
            "machinelearning:DeleteRealtimeEndpoint",
            "machinelearning:Predict"
        ],
        "Resource": [
            "arn:aws:machinelearning:<region>:<your-account-id>:mlmodel/ML-MODEL"
        ]
    }]
}
```

Contoh 4: Memungkinkan pengguna untuk memperbarui dan menghapus sumber daya tertentu

Kebijakan berikut memungkinkan pengguna atau grup untuk memperbarui dan menghapus sumber daya tertentu di akun AWS Anda dengan memberi mereka izin untuk melakukan`UpdateDataSource`,`UpdateMLModel`,`UpdateBatchPrediction`,`UpdateEvaluation`,`Dele` dan`DeleteEvaluation`tindakan pada sumber daya tersebut di akun Anda.

```
{
    "Version": "2012-10-17",
    "Statement": [{
        "Effect": "Allow",
```

```
        "Action": [
            "machinelearning:Update*",
            "machinelearning:DeleteDataSource",
            "machinelearning:DeleteMLModel",
            "machinelearning:DeleteBatchPrediction",
            "machinelearning:DeleteEvaluation"
        ],
        "Resource": [
            "arn:aws:machinelearning:<region>:<your-account-id>:datasource/S3-DS-ID1",
            "arn:aws:machinelearning:<region>:<your-account-id>:datasource/REDSHIFT-DS-
ID1",
            "arn:aws:machinelearning:<region>:<your-account-id>:mlmodel/ML-MODEL-ID1",
            "arn:aws:machinelearning:<region>:<your-account-id>:batchprediction/BP-
ID1",
            "arn:aws:machinelearning:<region>:<your-account-id>:evaluation/EV-ID1"
        ]
    }]
}
```

Contoh 5: Izinkan Amazon MLAction apa pun

Kebijakan berikut mengizinkan pengguna atau grup menggunakan tindakan Amazon Amazon Amazon. Karena kebijakan ini memberikan akses penuh ke semua sumber daya machine learning Anda, batasi hanya untuk administrator.

```
{
    "Version": "2012-10-17",
    "Statement": [{
        "Effect": "Allow",
        "Action": [
            "machinelearning:*"
        ],
        "Resource": [
            "*"
        ]
    }]
}
```

Pencegahan wakil bingung lintas layanan

Masalah wakil yang bingung adalah masalah keamanan di mana entitas yang tidak memiliki izin untuk melakukan tindakan dapat memaksa entitas yang lebih istimewa untuk melakukan tindakan.

MasukAWS, peniruan lintas layanan dapat mengakibatkan masalah wakil bingung. Peniruan lintas layanan dapat terjadi ketika satu layanan (layanan panggilan) panggilan layanan lain (yangdisebut layanan). Layanan panggilan dapat dimanipulasi untuk menggunakan izinnya untuk bertindak atas sumber daya pelanggan lain dengan cara yang seharusnya tidak memiliki izin untuk mengakses. Untuk mencegah hal ini,AWSmenyediakan alat yang membantu Anda melindungi data Anda untuk semua layanan dengan prinsip-prinsip layanan yang telah diberikan akses ke sumber daya di akun Anda.

Sebaiknya gunakan alat`aws:SourceArn`dan`aws:SourceAccount`kunci konteks kondisi global dalam kebijakan sumber daya untuk membatasi izin yang diberikan Amazon Machine Learning layanan lain ke sumber daya. Jika`aws:SourceArn`value tidak berisi ID akun, seperti ARN bucket Amazon S3, Anda harus menggunakan kedua kunci konteks kondisi global untuk membatasi izin. Jika Anda menggunakan kedua kunci konteks kondisi global dan`aws:SourceArn`nilai berisi ID akun,`aws:SourceAccount`nilai dan akun di`aws:SourceArn`nilai harus menggunakan ID akun yang sama ketika digunakan dalam pernyataan kebijakan yang sama. Gunakan`aws:SourceArn`jika Anda ingin hanya satu sumber daya yang terkait dengan akses lintas layanan. Gunakan`aws:SourceAccount`jika Anda ingin mengizinkan sumber daya apa pun di akun itu dikaitkan dengan penggunaan lintas-layanan.

Cara paling efektif untuk melindungi terhadap masalah wakil yang bingung adalah dengan menggunakan`aws:SourceArn`kunci konteks kondisi global dengan ARN penuh sumber daya. Jika Anda tidak mengetahui ARN lengkap sumber daya atau jika Anda menentukan beberapa sumber daya, gunakan`aws:SourceArn`kunci kondisi konteks global dengan wildcard (*) untuk bagian ARN yang tidak diketahui. Sebagai contoh, `arn:aws:`*`servicename`*`:*:`*`123456789012`*`:*`.

Contoh berikut menunjukkan bagaimana Anda dapat menggunakan`aws:SourceArn`dan`aws:SourceAccount`kunci konteks kondisi global di Amazon MLuntuk mencegah masalah wakil yang membingungkan saat membaca data dari bucket Amazon S3.

```
{
    "Version": "2008-10-17",
    "Statement": [
    {
        "Effect": "Allow",
        "Principal": { "Service": "machinelearning.amazonaws.com" },
        "Action": "s3:GetObject",
        "Resource": "arn:aws:s3:::examplebucket/exampleprefix/*"
        "Condition": {
            "StringEquals": { "aws:SourceAccount": "123456789012" }
```

```
            "ArnLike": { "aws:SourceArn": "arn:aws:machinelearning:us-
east-1:123456789012:*" }
        }
    },
    {
        "Effect": "Allow",
        "Principal": {"Service": "machinelearning.amazonaws.com"},
        "Action": "s3:ListBucket",
        "Resource": "arn:aws:s3:::examplebucket",
        "Condition": {
            "StringLike": { "s3:prefix": "exampleprefix/*" }
            "StringEquals": { "aws:SourceAccount": "123456789012" }
            "ArnLike": { "aws:SourceArn": "arn:aws:machinelearning:us-
east-1:123456789012:*" }
        }
    }]
}
```

Manajemen Ketergantungan Operasi Asinkron

Operasi Batch di Amazon MLbergantung pada operasi lain agar berhasil diselesaikan. Untuk mengelola dependensi ini, Amazon MLmengidentifikasi permintaan yang memiliki dependensi, dan memverifikasi bahwa operasi telah selesai. Jika operasi belum selesai, Amazon MLakan mengesampingkan permintaan awal hingga operasi yang mereka andalkan telah selesai.

Ada beberapa dependensi antara operasi batch. Misalnya, sebelum Anda dapat membuat model ML, Anda harus telah membuat sumber data yang dapat digunakan untuk melatih model MLnya. Amazon MLtidak dapat melatih model ML jika tidak ada sumber data yang tersedia.

Namun, Amazon MLmendukung manajemen dependensi untuk operasi asinkron. Misalnya, Anda tidak perlu menunggu sampai statistik data dihitung sebelum Anda dapat mengirim permintaan untuk melatih model ML pada sumber data. Sebagai gantinya, segera setelah sumber data dibuat, Anda dapat mengirim permintaan untuk melatih model MLmenggunakan sumber data. Amazon MLtidak benar-benar memulai operasi pelatihan sampai statistik sumber data dihitung. Permintaan CreateMLModel dimasukkan ke dalam antrian sampai statistik dihitung; setelah itu selesai, Amazon MLsegera mencoba menjalankan operasi CreateMLModel. Demikian pula, Anda dapat mengirim prediksi batch dan permintaan evaluasi untuk model ML yang belum menyelesaikan pelatihan.

Tabel berikut menunjukkan persyaratan untuk melanjutkan dengan tindakan AmazonMLL yang berbeda

Untuk...	Anda harus memiliki...
Buat model ML(CreateMLModel)	Sumber data dengan statistik data yang dihitung
Buat prediksi batch (createBatchPrediction)	DataSource Model ML
Buat evaluasi batch (createBatchEvaluation)	DataSource Model ML

Memeriksa status permintaan

Ketika Anda mengirimkan permintaan, Anda dapat memeriksa statusnya dengan Amazon Machine Learning (Amazon ML) API. Misalnya, jika Anda mengirimkan`createMLModel`permintaan, Anda dapat memeriksa statusnya dengan menggunakan`describeMLModel`panggilan. Amazon ML. merespons dengan salah satu dari status-status berikut.

Status	Definisi
MENUNGGU	Amazon MLsedang memvalidasi permintaan tersebut. ATAU Amazon MLsedang menunggu sumber daya komputasi tersedia sebelum menjalankan permintaan. Ini mungkin terjadi jika akun Anda telah melebihi jumlah maksimum permintaan operasi batch yang berjalan bersamaan. Jika ini masalahnya, status transisi keInProgressketika permintaan berjalan lainnya telah selesai atau dibatalkan. ATAU Amazon MLsedang menunggu operasi batch yang bergantung pada permintaan Anda untuk diselesaikan.

Status	Definisi
INPROGRESS	Permintaan Anda masih berjalan.
DISELESAIKAN	Permintaan telah selesai, dan objek siap digunakan (model dan sumber data) atau dilihat (prediksi batch dan evaluasi).
GAGAL	Ada yang salah dengan data yang Anda berikan, atau Anda telah membatalkan operasi. Misalnya, jika Anda mencoba menghitun g statistik data pada sumber data yang gagal diselesaikan, Anda mungkin menerimaTidak berlakuatauGagalpesan status. Pesan kesalahan menjelaskan mengapa operasi tidak berhasil diselesai kan.
MENGHAPUS	Objek telah dihapus.

Amazon MLjuga menyediakan informasi tentang suatu objek, seperti saat Amazon MLselesai membuat objek tersebut. Untuk informasi selengkapnya, lihat Daftar Objek.

Pembatasan Sistem

Untuk memberikan layanan yang tangguh dan andal, Amazon MLmemberlakukan batasan tertentu pada permintaan yang Anda buat ke sistem. Sebagian besar masalah ML cocok dengan mudah dalam kendala ini. Namun, jika Anda menemukan bahwa penggunaan Amazon MLnya dibatasi oleh batas-batas ini, Anda dapat menghubungiLayanan pelanggan AWSdan meminta untuk memiliki batas yang dinaikkan. Misalnya, Anda mungkin memiliki batas lima untuk jumlah tugas yang dapat Anda jalankan secara bersamaan. Jika Anda menemukan bahwa Anda sering memiliki pekerjaan antri yang menunggu sumber daya karena batas ini, maka mungkin masuk akal untuk menaikkan batas itu untuk akun Anda.

Tabel berikut menunjukkan batas default per akun di Amazon ML. Tidak semua batasan ini dapat dinaikkan oleh layanan pelanggan AWS.

Jenis Batasan	Batas Sistem
Ukuran setiap pengamatan	100 KB

Jenis Batasan	Batas Sistem
Ukuran data latihan*	100 GB
Ukuran input prediksi batch	1 TB
Ukuran input prediksi batch (jumlah catatan)	100 juta
Jumlah variabel dalam file data (skema)	1.000
Kompleksitas resep (jumlah variabel keluaran yang diproses)	10.000
TPS untuk setiap titik akhir prediksi waktu nyata	200
Total TPS untuk semua titik akhir prediksi waktu nyata	10.000
Total RAM untuk semua titik akhir prediksi waktu nyata	10 GB
Jumlah tugas simultan	25
Waktu aktif terpanjang untuk tugas apa pun	7 hari
Jumlah kelas untuk model ML-kelas	100
Ukuran model ML	Minimal 1 MB, maksimal 2 GB
Jumlah tanda per objek	50

- Ukuran file data Anda terbatas untuk memastikan bahwa pekerjaan selesai tepat waktu. Pekerjaan yang telah berjalan selama lebih dari tujuh hari akan dihentikan secara otomatis, sehingga status GAGAL.

Nama dan ID untuk semua Objek

Setiap objek di Amazon MLharus memiliki pengenal, atau ID. Konsol Amazon MLmenghasilkan nilai ID untuk Anda, tetapi jika Anda menggunakan API, Anda harus membuat nilai ID Anda sendiri. Setiap ID harus unik di antara semua objek Amazon MLdengan jenis yang sama di akun AWS Anda.

Artinya, Anda tidak dapat memiliki dua evaluasi dengan ID yang sama. Dimungkinkan untuk memiliki evaluasi dan sumber data dengan ID yang sama, meskipun tidak disarankan.

Kami menyarankan Anda menggunakan pengidentifikasi yang dibuat secara acak untuk objek Anda, diawali dengan string pendek untuk mengidentifikasi jenisnya. Misalnya, saat konsol Amazon MLmenghasilkan sumber data, konsol tersebut menetapkan sumber data ID acak dan unik seperti "ds-ZScwIuWiOxF". ID ini cukup acak untuk menghindari tabrakan bagi pengguna tunggal, dan juga ringkas dan mudah dibaca. Awalan "ds-" adalah untuk kenyamanan dan kejelasan, tetapi tidak diperlukan. Jika Anda tidak yakin apa yang harus digunakan untuk string ID Anda, sebaiknya gunakan nilai UUID heksadesimal (seperti 28b1e915-57e5-4e6c-a7bd-6fb4e729cb23), yang sudah tersedia di lingkungan pemrograman modern apa pun.

ID string dapat berisi ASCII huruf, angka, tanda hubung dan garis bawah, dan bisa sampai 64 karakter panjang. Hal ini dimungkinkan dan mungkin nyaman untuk mengkodekan metadata ke dalam string ID. Tetapi tidak disarankan karena setelah objek dibuat, ID-nya tidak dapat diubah.

Nama objek memberikan cara mudah bagi Anda untuk mengaitkan metadata yang ramah pengguna dengan setiap objek. Anda dapat memperbarui nama setelah objek telah dibuat. Hal ini memungkinkan nama objek untuk mencerminkan beberapa aspek alur kerja ML-mu. Misalnya, Anda mungkin awalnya nama model "percobaan #3 ", dan kemudian mengubah nama model "model produksi akhir". Nama dapat berupa string yang Anda inginkan, hingga 1.024 karakter.

Umur Objek

Setiap sumber data, model, evaluasi, atau objek prediksi batch yang Anda buat dengan Amazon MLakan tersedia untuk Anda gunakan setidaknya selama dua tahun setelah pembuatan. Amazon MLmungkin secara otomatis menghapus objek yang belum diakses atau digunakan selama lebih dari dua tahun.

Sumber daya

Sumber daya terkait berikut dapat membantu Anda ketika bekerja dengan layanan ini.

- Informasi produk Amazon MLE— Menangkap semua informasi produk terkait tentang Amazon ML-nya di lokasi pusat.
- FAQ Amazon MLE— Meliputi pertanyaan teratas yang telah ditanyakan pengembang tentang produk ini.
- Kode sampel Amazon MLE— Contoh aplikasi yang menggunakan Amazon XML. Anda dapat menggunakan kode sampel sebagai titik awal untuk membuat aplikasi ML-mu sendiri.
- Referensi API Amazon API— Menjelaskan semua operasi API untuk Amazon ML-nya secara rinci. Aplikasi ini juga menyediakan permintaan sampel dan tanggapan untuk protokol layanan web yang didukung.
- Pusat Sumber Daya Pengembang AWS— Menyediakan titik awal pusat untuk menemukan dokumentasi, sampel kode, catatan rilis, dan informasi lainnya untuk membantu Anda membangun aplikasi inovatif dengan AWS.
- Pelatihan dan Kursus AWS— Tautan ke kursus khusus dan berbasis peran serta lab mandiri untuk membantu mempertajam keterampilan AWS Anda dan mendapatkan pengalaman praktis.
- Alat Developer AWS— Tautan ke alat pengembang dan sumber daya yang menyediakan dokumentasi, sampel kode, catatan rilis, dan informasi lainnya untuk membantu Anda membangun aplikasi inovatif dengan AWS.
- Pusat Dukungan AWS— Pusat untuk membuat dan mengelola kasus dukungan AWS Anda. Juga mencakup tautan ke sumber daya yang bermanfaat lainnya, seperti forum, FAQ teknis, status kondisi layanan, dan AWS Trusted Advisor.
- Dukungan AWS— Halaman web utama untuk informasi tentang AWS Support, saluran dukungan jawaban cepat satu per satu untuk membantu Anda membangun dan menjalankan aplikasi di cloud.
- Hubungi Kami— Titik kontak pusat untuk pertanyaan mengenai tagihan AWS, akun, peristiwa, penyalahgunaan, dan masalah lainnya.
- Ketentuan Situs AWS— Informasi lengkap tentang hak cipta dan merek dagang kami; akun, lisensi, dan akses situs Anda; serta topik lainnya.

Riwayat Dokumen

Tabel berikut menjelaskan perubahan penting pada dokumentasi dalam rilis Amazon Machine Learning (Amazon ML).

- Versi API: 2015-04-09
- Pembaruan dokumentasi terakhir: 2016-08-02

Perubahan	Deskripsi	Tanggal yang Diubah
Metrik ditambahkan	Rilis Amazon XML ini menambahkan metrik baru untuk objek Amazon ML-nya. Untuk informasi selengkapnya, lihat Daftar Objek.	2 Agustus 2016
Menghapus beberapa objek	Rilis Amazon XML ini menambahkan kemampuan untuk menghapus beberapa objek Amazon ML-nya. Untuk informasi selengkapnya, lihat Menghapus Object.	20 Juli 2016
Tagging ditambahkan	Rilis Amazon XML ini menambahkan kemampuan untuk menerapkan tag ke objek Amazon ML-nya. Untuk informasi selengkapnya, lihat Menandai Objek Amazon ML-mu.	23 Juni 2016
Sumber data Amazon Redshift	Rilis Amazon XML ini menambahkan kemampuan menyalin pengaturan sumber data Amazon Redshift ke sumber data Amazon Redshift baru. Untuk informasi selengkapnya tentang menyalin pengatura n sumber data Amazon Redshift, lihatMenyalin Datasource (Konsol).	11 April 2016
Shuffling ditambahkan	Rilis Amazon XML ini menambahkan kemampuan untuk mengocokkan data input Anda.	5 April 2016

Perubahan	Deskripsi	Tanggal yang Diubah
	Untuk informasi lebih lanjut tentang penggunaanTipe Shuffleparameter, lihatTipe Shuffle untuk Data Pelatihan.	
Pembuatan sumber data yang ditingkatkan dengan Amazon Redshift	Rilis Amazon XML ini menambahkan kemampuan untuk menguji pengaturan Amazon Redshift Anda saat Anda membuat sumber data Amazon ML-nya di konsol untuk memverifikasi bahwa sambungan berfungsi. Untuk informasi selengkapnya, lihat Membuat Sumber Data dengan Amazon Redshift Data (Konsol).	21 Maret 2016
Konversi skema data Amazon Redshift	Rilis Amazon XML ini meningkatkan konversi skema data Amazon Redshift (Amazon Redshift) ke skema data Amazon ML-nya. Untuk informasi selengkapnya tentang cara menggunakan Amazon Redshift dengan Amazon L, lihatMembuat Sumber Data Amazon ML-dari Data di Amazon Redshift.	9 Februari 2016
Pencatatan CloudTrail ditambahkan	Rilis Amazon XML ini menambahkan kemampuan untuk mencatat permintaan menggunakanAWS CloudTrail(CloudTrail). Untuk informasi selengkapnya tentang cara menggunakan log CloudTrail, lihatMencatat Panggilan API Amazon denganAWS CloudTrail.	10 Desember 2015
Pilihan DataRareArrangement tambahan ditambahkan	Rilis Amazon XML ini menambahkan kemampuan untuk membagi data input Anda secara acak dan membuat sumber data komplementer. Untuk informasi lebih lanjut tentang penggunaan`DataRearrangement` parameter, lihatPenataan Data. Untuk informasi tentang cara menggunakan opsi baru untuk validasi silang, lihatLintas Validasi.	3 Desember 2015

Perubahan	Deskripsi	Tanggal yang Diubah
Mencoba prediksi real-time	Rilis Amazon XML ini menambahkan kemampuan untuk mencoba prediksi real-time di konsol layanan. Untuk informasi selengkapnya tentang mencoba prediksi real-time, lihatMemiinta Prediksi Waktu Nyatadi dalamPanduan Developer Amazon Machine Learning.	19 November 2015
Wilayah baru	Rilis Amazon MLnya menambahkan dukungan untuk wilayah UE (Irlandia). Untuk informasi selengkapnya tentang Amazon ML-wilayah UE (Irlandia), lihatWilayah dan titik akhirdi dalamPanduan Developer Amazon Machine Learning.	20 Agustus 2015
Rilis Awal	Ini adalah rilis pertama dariPanduan Developer Amazon.	9 April 2015