Guía para desarrolladores

Amazon Machine Learning

Version Latest

Copyright © 2025 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon Machine Learning: Guía para desarrolladores

Copyright © 2025 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Las marcas comerciales y la imagen comercial de Amazon no se pueden utilizar en relación con ningún producto o servicio que no sea de Amazon de ninguna manera que pueda causar confusión entre los clientes y que menosprecie o desacredite a Amazon. Todas las demás marcas registradas que no son propiedad de Amazon son propiedad de sus respectivos propietarios, que pueden o no estar afiliados, conectados o patrocinados por Amazon.

Table of Contents

..... ix
¿Qué es Amazon Machine Learning? 1
Amazon Machine Learning: Conceptos clave 1
Fuentes de datos 2
Modelos de ML 4
Evaluaciones 4
Predicciones por lotes 5
Predicciones en tiempo real 6
Acceder a Amazon Machine Learning 6
Regiones y puntos de conexión 7
Precios de Amazon ML 8
Estimación del costo de la predicción por lotes 8
Estimación del costo de la predicción en tiempo real 10
Conceptos del aprendizaje automático 11
Solución de problemas de negocios con Amazon Machine Learning 11
Cuándo utilizar el aprendizaje automático 12
Crear una aplicación de Machine Learning 13
Formulación del problema 13
Recopilación de datos con etiqueta 14
Analizar los datos 15
Procesamiento de características 15
Dividir los datos en datos de formación y evaluación 17
Entrenar a un modelo 18
Evaluación de la precisión del modelo 21
Mejorar la precisión del modelo 26
Uso del modelo para hacer predicciones 27
Retención de modelos en datos nuevos 28
El proceso de Amazon Machine Learning 28
Configuración de Amazon Machine Learning 31
Inscripción en AWS 31
Tutorial: Utilización de Amazon ML para predecir respuestas a una oferta de marketing 32
Requisito previo 32
Pasos 32
Paso 1: prepare los datos 33

Paso 2: cree una fuente de datos de entrenamiento ... 35
Paso 3: Crear una modelo de ML ... 41
Paso 4: Revisar el desempeño predictivo del modelo de ML y establecer un umbral de puntuación ... 42
Paso 5: Uso del modelo de ML para generar predicciones ... 45
Paso 6: eliminación ... 53
Creación y uso de fuentes de datos ... 55
Compresión del formato de datos de Amazon ML ... 55
Atributos ... 56
Requisitos de formato de archivos de entrada ... 56
Uso de varios archivos como datos de entrada para Amazon ML ... 57
End-of-Line Caracteres en formato CSV ... 58
Creación de un esquema de datos para Amazon ML ... 59
Esquema de ejemplo ... 59
Uso del targetAttributeName campo ... 61
Funcionamiento del campo rowID ... 61
Uso del AttributeType campo ... 62
Proporcionar un esquema a Amazon ML ... 64
División de datos ... 65
Pre-división de datos ... 66
División secuencial de datos ... 66
División aleatoria de datos ... 67
Análisis de datos ... 69
Estadísticas descriptivas ... 69
Estadísticas de los datos de acceso de la consola Amazon ML ... 70
Uso de Amazon S3 con Amazon ML ... 80
Carga de datos en Amazon S3 ... 81
Permisos ... 81
Creación de una fuente de datos de Amazon ML a partir de datos de Amazon Redshift ... 82
Parámetros necesarios para el asistente Create Datasource ... 83
Creación de una fuente de datos con datos de Amazon Redshift (consola) ... 87
Temas de solución de problemas de Amazon Redshift ... 91
Uso de datos de una base de datos de Amazon RDS para crear una fuente de datos de Amazon ML ... 97
Identificador de instancias de bases de datos de RDS ... 98
Nombre de la base de datos de MySQL ... 98

Credenciales del usuario de la base de datos ... 98
Información de seguridad de AWS Data Pipeline ... 99
Información de seguridad de Amazon RDS ... 99
Consulta SQL de MySQL ... 100
Ubicación de salida de S3 ... 100
Entrenamiento de modelos de ML ... 101
Tipos de modelos de ML ... 101
Modelo de clasificación binaria ... 102
Modelo de clasificación multiclase ... 102
Modelo de regresión ... 102
Proceso de formación ... 103
Parámetros de entrenamiento ... 103
Tamaño máximo del modelo ... 104
Número máximo de iteraciones en los datos ... 105
Tipo de mezcla para los datos de entrenamiento ... 105
Tipo y cantidad de regularización ... 106
Parámetros de entrenamiento: tipos y valores predeterminados ... 107
Creación de un modelo de ML ... 108
Requisitos previos ... 109
Creación de un modelo de ML con las opciones predeterminadas ... 109
Creación de un modelo de ML con opciones personalizadas ... 110
Transformaciones de datos para aprendizaje automático ... 113
Importancia de la transformación de funciones ... 113
Transformaciones de características con recetas de datos ... 114
Referencia del formato de recetas ... 114
Grupos ... 115
Asignaciones ... 115
Salidas ... 116
Ejemplo completo de receta ... 118
Recetas sugeridas ... 120
Referencia de transformaciones de datos ... 120
Transformación de n-gramas ... 121
Transformación de bigramas dispersos ortogonales (OSB) ... 122
Transformación en minúsculas ... 123
Eliminar la transformación de puntuación ... 123
Transformación de discretización en cuartiles ... 124

Transformación de normalización ... 125
Transformación de producto cartesiana ... 125
Reorganización de datos ... 127
DataRearrangement Parámetros ... 128
Evaluación de modelos de ML ... 132
Información sobre el modelo de ML ... 133
Información sobre modelos binarios ... 133
Interpretación de las predicciones ... 133
Información del modelo multiclase ... 137
Interpretación de las predicciones ... 137
Informaciones sobre el modelo de regresión ... 140
Interpretación de las predicciones ... 140
Prevención del sobreajuste ... 142
Validación cruzada ... 143
Ajuste de los modelos ... 145
Alertas de evaluación ... 146
Creación e interpretación de predicciones ... 148
Creación de una predicción por lotes ... 148
Creación de una predicción por lotes (consola) ... 149
Creación de una predicción por lotes (API) ... 149
Revisión de métricas de predicciones por lotes ... 150
Revisión de métricas de predicciones por lotes (consola) ... 151
Revisión de métricas e información de predicciones por lotes (API) ... 151
Lectura de archivos de salida de predicciones por lotes ... 151
Localización del archivo de manifiesto de predicciones por lotes ... 152
Lectura del archivo de manifiesto ... 152
Recuperación de archivos de salida de predicciones por lotes ... 153
Interpretación del contenido de los archivos de predicciones por lotes para un modelo de ML de clasificación binaria ... 153
Interpretación del contenido de los archivos de predicciones por lotes para un modelo de ML de clasificación multiclase ... 154
Interpretación del contenido de los archivos de predicciones por lotes para un modelo de ML de regresión ... 155
Solicitud de predicciones en tiempo real ... 156
Pruebas con las predicciones en tiempo real ... 157
Creación de un punto de enlace en tiempo real ... 159

Ubicación del punto de enlace de predicciones en tiempo real (consola) 161
Ubicación del punto de enlace de predicciones en tiempo real (API) 161
Creación de una solicitud de predicciones en tiempo real 162
Eliminación de un punto de enlace en tiempo real 164
Administración de objetos de Amazon ML 166
Listas de objetos 166
Listas de objetos (consola) 167
Listas de objetos (API) 168
Recuperación de descripciones de objetos 169
Descripciones detalladas en la consola 169
Descripciones detalladas de la API 169
Actualización de objetos 170
Eliminación de objetos 170
Eliminación de objetos (consola) 171
Eliminación de objetos (API) 171
Supervisión de Amazon ML con Amazon CloudWatch Metrics 173
Registrar llamadas a la API de Amazon ML con AWS CloudTrail 174
Información de Amazon ML en CloudTrail 174
Ejemplo: entradas del archivo de registro de Amazon ML 176
Etiquetado de los objetos 180
Conceptos básicos de etiquetas 180
Restricciones de las etiquetas 181
Etiquetado de objetos de Amazon ML (consola) 182
Etiquetado de objetos de Amazon ML (API) 184
Referencia de Amazon Machine Learning 185
Concesión de permisos de Amazon ML para la lectura de datos desde Amazon S3 185
Concesión de permisos a Amazon ML para enviar predicciones a Amazon S3 187
Control de acceso a los recursos de Amazon ML con IAM 189
Sintaxis de la política de IAM 190
Especificación de las acciones políticas de IAM para Amazon ML MLAmazon 191
Especificación ARNs de los recursos de Amazon ML en las políticas de IAM 192
Ejemplos de políticas para Amazon MLs 193
Prevención de la sustitución confusa entre servicios 196
Administración de la dependencia de operaciones asíncronas 197
Comprobación del estado de la solicitud 198
Límites del sistema 200

Nombres y IDs para todos los objetos 201
Object Lifetimes 202
Recursos 203
Historial de documentos 204

Ya no actualizamos el servicio Amazon Machine Learning ni aceptamos nuevos usuarios para él. Esta documentación está disponible para los usuarios actuales, pero ya no la actualizamos. Para obtener más información, consulte Qué es Amazon Machine Learning.

Las traducciones son generadas a través de traducción automática. En caso de conflicto entre la traducción y la version original de inglés, prevalecerá la version en inglés.

¿Qué es Amazon Machine Learning?

Ya no actualizamos el servicio Amazon Machine Learning (Amazon ML) ni aceptamos nuevos usuarios para él. Esta documentación está disponible para los usuarios actuales, pero ya no la actualizamos.

AWS ahora ofrece un servicio sólido basado en la nube, Amazon SageMaker AI, para que los desarrolladores de todos los niveles de habilidad puedan utilizar la tecnología de aprendizaje automático. SageMaker La IA es un servicio de aprendizaje automático totalmente gestionado que le ayuda a crear modelos potentes de aprendizaje automático. Con la SageMaker IA, los científicos de datos y los desarrolladores pueden crear y entrenar modelos de aprendizaje automático y, después, implementarlos directamente en un entorno hospedado listo para la producción.

Para obtener más información, consulte la documentación sobre SageMaker IA.

Temas

- Amazon Machine Learning: Conceptos clave
- Acceder a Amazon Machine Learning
- Regiones y puntos de conexión
- Precios de Amazon ML

Amazon Machine Learning: Conceptos clave

En esta sección se indican los siguientes conceptos clave y se describen con más detalle cómo se utilizan en Amazon ML:

- Fuentes de datos contienen metadatos asociados con las entradas de datos en Amazon ML
- Modelos de ML generan predicciones utilizando los patrones extraídos de los datos de entrada
- Las Evaluaciones miden la calidad de modelos de ML
- Predicciones por lotes generan predicciones de forma asíncrona para varias observaciones de datos de entrada
- Predicciones en tiempo real generan predicciones de forma sincrónica para observaciones de datos individuales

Fuentes de datos

Una fuente de datos es un objeto que contiene metadatos sobre los datos de entrada. Amazon ML lee los datos de entrada, calcula estadísticas descriptivas sobre sus atributos y almacena las estadísticas, junto con un esquema y otra información, como parte del objeto de fuente de datos. A continuación, Amazon ML utiliza la fuente de datos para entrenar y evaluar un modelo de ML y generar predicciones por lotes.

> **Important**
>
> Una fuente de datos no almacena ninguna copia de los datos de entrada. En su lugar, almacena una referencia a la ubicación de Amazon S3 en la que se encuentran los datos de entrada. Si mueve o cambia el archivo de Amazon S3, Amazon ML no puede obtener acceso a él o utilizarlo para crear un modelo de ML, generar evaluaciones o generar predicciones.

En la siguiente tabla se definen términos relacionados con las fuentes de datos.

Plazo	Definición
Atributo	Una propiedad única con nombre que pertenece a una observación. En los datos formateados por tablas, como las hojas de cálculo o los archivos de valores separados por comas (.csv), los encabezados de columna represent an los atributos y las filas contienen los valores de cada atributo. Sinónimos: variable, nombre de variable, campo, columna
Nombre de la fuente de datos	(Opcional) Permite definir un nombre legible para una fuente de datos. Estos nombres le permiten encontrar y administrar sus fuentes de datos en la consola de Amazon ML.
Datos de entrada	Nombre colectivo para todas las observaciones a las que hace referencia una fuente de datos.
Ubicación	Ubicación de los datos de entrada. En la actualidad, Amazon ML puede utilizar datos que se almacenan en un bucket de Amazon S3, bases de datos de Amazon Redshift o bases de datos MySQL en Amazon Relational Database Service (RDS).

Plazo	Definición
Observación	Una sola unidad de datos de entrada. Por ejemplo, si está creando un modelo de ML para detectar transacciones fraudulentas, los datos de entrada se componen de muchas observaciones, cada una de las cuales representa una transacción individual. Sinónimos: registro, ejemplo, instancia, fila
ID de fila	(Opcional) Una marca que, si se especifica, identifica un atributo en los datos de entrada que se incluye en la salida de predicciones. Este atributo facilita determinar qué predicción se corresponde con cada observación. Sinónimos: identificador de fila
Esquema	La información necesaria para interpretar los datos de entrada, incluidos los nombres de los atributos y sus tipos de datos asignados, así como los nombres de los atributos especiales.
Statistics	Estadísticas de resumen para cada atributo en los datos de entrada. Estas estadísticas tienen dos propósitos: La consola de Amazon ML los muestra en gráficos para ayudarle a entender sus datos at-a-glance e identificar irregularidades o errores. Amazon ML las utiliza durante el proceso de entrenamiento para mejorar la calidad del modelo de ML resultante.
Estado	Indica el estado actual del origen de datos, como, In Progress (En curso), Completed (Completado) o Failed (Error).
Atributo de destino	En el contexto de la formación de un modelo de ML, el atributo de destino identifica el nombre del atributo en los datos de entrada que contiene las respuestas "correctas". Amazon ML lo utiliza para descubrir patrones en los datos de entrada y generar un modelo de ML. En el contexto de la evaluació n y la generación de predicciones, el atributo de destino es el atributo cuyo valor se prevé por un modelo entrenado de ML. Sinónimos: destino

Modelos de ML

Un modelo de ML es un modelo matemático que genera predicciones detectando patrones en los datos. Amazon ML admite tres tipos de modelos de ML: clasificación binaria, clasificación multiclase y regresión.

En la siguiente tabla se definen términos relacionados con los modelos de ML.

Plazo	Definición
Regresión	El objetivo de entrenar un modelo de ML de regresión es predecir un valor numérico.
Multiclase	El objetivo de entrenar un modelo de ML multiclase es predecir valores pertenecientes a un conjunto limitado y predefinido de valores permitidos.
Binario	El objetivo de entrenar un modelo de ML binario es predecir valores que solo pueden tener uno de los dos estados posibles, como verdadero o falso.
Tamaño del modelo	Los modelos de ML capturan y almacenan patrones. Cuantos más patrones almacena un modelo de ML, más grande será. El tamaño del modelo de ML se describe en megabytes.
Número de iteraciones	Cuando entrena un modelo de ML, utiliza datos de una fuente de datos. A veces es beneficioso utilizar más de una vez cada registro de datos en el proceso de aprendizaje. El número de veces que Amazon ML utiliza los mismos registros de datos se denomina número de iteraciones.
Regularización	La regulación es una técnica de machine learning que puede usar para obtener modelos de mayor calidad. Amazon ML ofrece una configuración predeterminada que funciona bien para la mayoría de los casos.

Evaluaciones

Una evaluación mide la calidad del modelo de ML y determina si se está desempeñando bien.

En la siguiente tabla se definen términos relacionados con las evaluaciones.

Plazo	Definición
Informaciones del modelo	Amazon ML le proporciona una métrica y una serie de informaciones que puede utilizar para evaluar el desempeño predictivo de su modelo.
AUC	El parámetro Area Under the ROC Curve (AUC) mide la capacidad de un modelo de ML binario de predecir una mayor puntuación para ejemplos positivos en comparación con ejemplos negativos.
Puntuación F1 macroprom ediada	La puntuación F1 macropromediada se utiliza para evaluar el desempeño predictivo de modelos de ML multiclase.
RMSE	El parámetro Root Mean Square Error (RMSE) es una métrica utilizada para evaluar el desempeño predictivo de modelos de ML de regresión.
Valor de corte	Los modelos de ML trabajan generando puntuaciones de predicción numérica. Al aplicar un valor de corte, el sistema convierte estas puntuacio nes en etiquetas 0 y 1.
Exactitud	La exactitud mide el porcentaje de predicciones correctas.
Precisión	La precision muestra el porcentaje de instancias positivas reales (en lugar de falsos positivos) de las instancias que se han recuperado (las que se predijero n como positivas). En decir, cuántos elementos seleccionados son positivos.
Exhaustividad	La exhaustividad muestra el porcentaje de positivos reales entre el número total de instancias pertinentes (positivos reales). En decir, cuántos elementos positivos se han seleccionado.

Predicciones por lotes

Las predicciones por lotes son para un conjunto de observaciones que se pueden ejecutarse a la vez. Esto es ideal para análisis predictivos que no tienen requisitos en tiempo real.

En la siguiente tabla se definen términos relacionados con las predicciones por lotes.

Plazo	Definición
Ubicación de la salida	Los resultados de una predicción por lotes se almacenan en una ubicación de salida del bucket de S3.
Archivo de manifiesto	Este archivo relaciona cada archivo de datos de entrada con sus resultados de predicción por lotes asociados. Se almacena en la ubicación de salida del bucket de S3.

Predicciones en tiempo real

Las predicciones en tiempo real son para aplicaciones que requieren una latencia baja, como webs interactivas, móviles o aplicaciones de escritorio. Cualquier modelo de ML puede consultarse para obtener predicciones usando la API de predicciones en tiempo real de baja latencia.

En la siguiente tabla se definen términos relacionados con las predicciones en tiempo real.

Plazo	Definición
API de predicción en tiempo real	La API de predicción en tiempo real acepta una única observación de entrada en la solicitud de carga y devuelve la predicción de la respuesta.
Punto de enlace de predicción en tiempo real	Para utilizar un modelo de ML con la API de predicción en tiempo real, debe crear un punto de enlace de predicción en tiempo real. Una vez creado, el punto de enlace contiene la URL que puede utilizar para solicitar las predicciones en tiempo real.

Acceder a Amazon Machine Learning

Puede obtener acceso a Amazon ML utilizando cualquiera de los siguientes:

Consola Amazon ML

Para acceder a la consola de Amazon ML, inicie sesión en la consola de administración de AWS y abra la consola de Amazon ML en https://console.aws.amazon.com/machinelearning/.

CLI de AWS

Para obtener más información sobre cómo instalar y configurar la CLI de AWS, consulte Configuración inicial de la interfaz de línea de comandos de AWS en la Guía del usuario de AWS Command Line Interface.

API de Amazon ML

Para obtener más información acerca de la API de Amazon ML, consulte Referencia de la API de Amazon ML.

AWS SDKs

Para obtener más información sobre AWS SDKs, consulte Herramientas para Amazon Web Services.

Regiones y puntos de conexión

Amazon Machine Learning (Amazon ML) es compatible con los puntos de enlace de predicción en tiempo real en las siguientes dos regiones:

Nombre de la región	Región	Punto de conexión	Protocolo
Este de EE. UU. (Norte de Virginia)	us-east-1	machinelearning.us-east-1.amazonaws.com	HTTPS
Europa (Irlanda)	eu-west-1	machinelearning.eu-west-1.amazonaws.com	HTTPS

Puede alojar conjuntos de datos, entrenar y evaluar modelos y activar predicciones en cualquier región.

Le recomendamos que mantenga todos sus recursos en la misma región. Si los datos de entrada están en una región distinta a la de los recursos Amazon ML, se acumulan las tarifas de transferencia de datos entre regiones. Puede llamar a un punto de enlace de predicción en tiempo real desde cualquier región, pero llamar a un punto de enlace desde una región que no tiene el punto de enlace al que está llamando puede afectar a las latencias de las predicciones en tiempo real.

Precios de Amazon ML

Con AWS los servicios, solo paga por lo que usa. No se requieren pagos mínimos ni compromisos iniciales.

Amazon Machine Learning (Amazon ML) factura una tasa por hora de ejecución para la generación de estadísticas de datos y para entrenar y evaluar modelos y un pago adicional por el número de predicciones generadas para la aplicación. Para las predicciones en tiempo real, también se paga un cargo por hora según la capacidad reservada, basándose en el tamaño del modelo.

Amazon ML solo calcula los costos de las predicciones en la consola de Amazon ML.

Para obtener más información sobre los precios de Amazon ML, consulte Precios de Amazon Machine Learning.

Temas

- Estimación del costo de la predicción por lotes
- Estimación del costo de la predicción en tiempo real

Estimación del costo de la predicción por lotes

En el momento en el que solicita predicciones por lotes de un modelo de Amazon ML utilizando el asistente para crear una predicción por lotes, Amazon ML calcula el costo de estas predicciones. El método para calcular la estimación varía en función del tipo de datos que están disponibles.

Estimación del costo de la predicción por lotes cuando las estadísticas de datos están disponibles

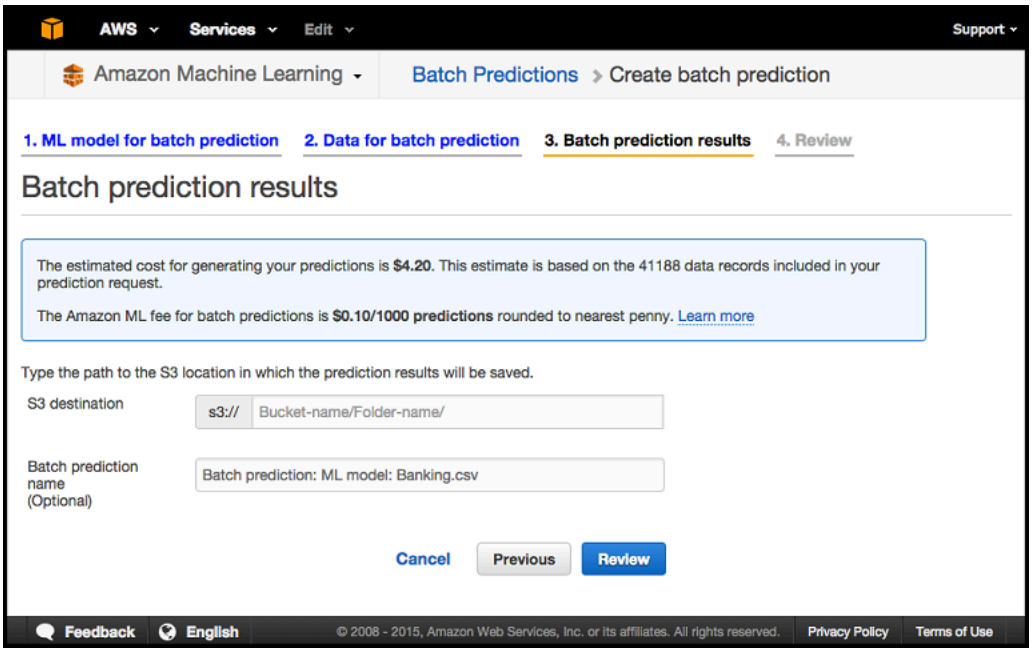
La estimación del costo más precisa se obtiene cuando Amazon ML ya ha calculado las estadísticas de resumen de la fuente de datos que se ha utilizado para solicitar predicciones. Estas estadísticas se suelen calcular para fuentes de datos que se han creado utilizando la consola de Amazon ML. Los usuarios de la API deben `ComputeStatistics` configurar esta opción `True` al crear fuentes de datos mediante programación mediante el CreateDataSourceFromS3 o el RDS CreateDataSourceFromRedshift. CreateDataSourceFrom APIs La fuente de datos debe estar en el estado `READY` para que las estadísticas estén disponibles.

Una de las estadísticas que procesa Amazon ML es el número de registros de datos. Cuando el número de registros de datos está disponible, el asistente para crear predicciones por lotes de

Amazon ML calcula el número de predicciones multiplicando el número de registros de datos por la cuota de las predicciones por lotes.

El costo real puede variar de esta estimación por las siguientes razones:

- Algunos de los registros de datos pueden fallar en el procesamiento. No se le cobrarán las predicciones fallidas de los registros de datos.
- La estimación no tienen en cuenta los créditos preexistentes u otros ajustes que aplica AWS.



Estimación del costo de la predicción por lotes cuando solo está disponible el tamaño de los datos

Cuando solicita una predicción por lotes y las estadísticas de los datos de la fuente de datos de la solicitud no están disponibles, Amazon ML calcula el costo en función de lo siguiente:

- El tamaño total de los datos que se calcula y persiste durante la validación de las fuentes de datos
- El tamaño medio del registro de datos, el cual Amazon ML calcula mediante la lectura y el análisis de los primeros 100 MB del archivo de datos

Para calcular el costo de la predicción por lotes, Amazon ML divide el tamaño total de los datos por el tamaño medio del registro de datos. Este método de predicción de costos es menos preciso que

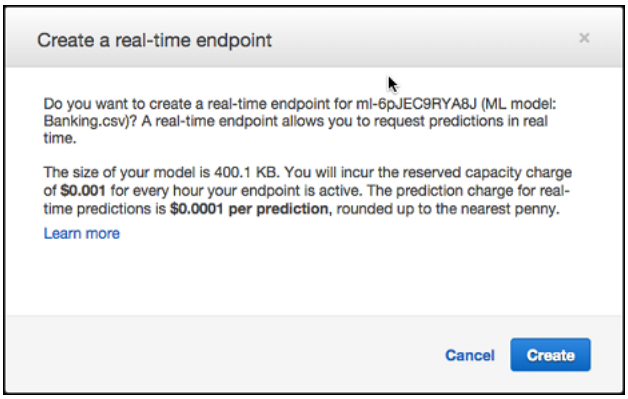
el método que se utiliza cuando el número de registros de datos está disponible, ya que los primeros registros del archivo de datos pueden no representar el tamaño medio del registro de forma precisa.

Estimación del costo de la predicción por lotes cuando no están disponibles ni las estadísticas de datos ni el tamaño de datos

Cuando no están disponibles ni las estadísticas de datos ni el tamaño de datos, Amazon ML no puede calcular el costo de las predicciones por lotes. Este suele ser el caso cuando el origen de los datos que está utilizando para solicitar predicciones por lotes no ha sido validado por Amazon ML aún. Esto puede suceder cuando ha creado una fuente de datos basada en una consulta de Amazon Redshift (Amazon Redshift) o Amazon Relational Database Service (Amazon RDS) y la transferencia de datos aún no se ha completado, o cuando la creación de un origen de datos se pone en cola por detrás de otras operaciones de la cuenta. En este caso, la consola de Amazon ML le informa sobre las cuotas de predicción por lotes. Puede elegir entre continuar con la solicitud de predicción por lotes sin una estimación o cancelar el asistente y volver una vez la fuente de datos que se ha utilizado para las predicciones esté en el estado INPROGRESS o READY.

Estimación del costo de la predicción en tiempo real

Cuando crea el punto de conexión de una predicción en tiempo real con la consola de Amazon ML, se le muestra el cargo estimado de la capacidad de reserva, el cual es un cargo continuo para reservar el punto de conexión para el procesamiento de predicciones. Este cargo varía en función del tamaño del modelo, tal y como se explica en la página de precios del servicio. También se le informará sobre el cargo estándar de las predicciones en tiempo real de Amazon ML.



Conceptos del aprendizaje automático

El aprendizaje automático (ML, por sus siglas en inglés) puede resultar de ayuda a la hora de usar datos históricos para tomar mejores decisiones de negocio. Los algoritmos de ML detectan patrones en los datos y construyen modelos matemáticos con estos descubrimientos. A continuación, podrá usar los modelos para hacer predicciones sobre futuros datos. Por ejemplo, una posible aplicación de un modelo de aprendizaje automático podría ser predecir la probabilidad de que un cliente compre un producto determinado a partir de su comportamiento anterior.

Temas

- Solución de problemas de negocios con Amazon Machine Learning
- Cuándo utilizar el aprendizaje automático
- Crear una aplicación de Machine Learning
- El proceso de Amazon Machine Learning

Solución de problemas de negocios con Amazon Machine Learning

Puede utilizar Amazon Machine Learning para aplicar el aprendizaje automático a los problemas para los que se dispone de ejemplos de respuestas reales. Por ejemplo, si desea utilizar Amazon Machine Learning para predecir si un mensaje de correo electrónico es spam, deberá recopilar ejemplos de correo electrónico que están correctamente etiquetados como spam o no spam. A continuación, puede utilizar el aprendizaje automático para realizar generalizaciones a partir de estos ejemplos de correo electrónico y predecir la posibilidad de que un correo electrónico nuevo sea o no spam. Este enfoque de aprendizaje a partir de los datos que se han etiquetado con la respuesta real se conoce como aprendizaje automático supervisado.

Puede utilizar enfoques de ML supervisados para estas tareas específicas de aprendizaje automático: clasificación binaria (predicción de uno de dos posibles resultados), clasificación multiclase (predicción de uno o de más de dos resultados) y regresión (predicción de un valor numérico).

Ejemplos de problemas de clasificación binaria:

- ¿El cliente comprará o no este producto?
- ¿Este correo electrónico es spam o no?
- ¿Es este producto un libro o una animal de granja?

- ¿Esta revisión la ha escrito un cliente o un robot?

Ejemplos de problemas de clasificación multiclase:

- ¿Este producto es un libro, una película o una prenda de ropa?
- ¿Esta película es una comedia romántica, un documental o un thriller?
- ¿Qué categoría de productos es más interesante para este cliente?

Ejemplos de problemas de clasificación de regresión:

- ¿Cuál será la temperatura en Seattle mañana?
- Para este producto, ¿cuántas unidades se venderán?
- ¿Cuántos días pasarán antes de que este cliente deje de utilizar la aplicación?
- ¿A qué precio se venderá esta casa?

Cuándo utilizar el aprendizaje automático

Es importante recordar que el aprendizaje automático no es una solución para todo tipo de problemas. Existen determinados casos en los que se pueden desarrollar soluciones sólidas sin usar técnicas de aprendizaje automático. Por ejemplo, el aprendizaje automático no es necesario si puede determinar un valor de referencia a través de sencillas reglas, cálculos o pasos predeterminados que pueden programarse sin necesidad de ningún tipo de aprendizaje basado en datos.

Puede utilizar el aprendizaje automático para las siguientes situaciones:

- No puede codificar las reglas: muchas tareas humanas (como reconocer si un mensaje de correo electrónico es spam o no) no pueden resolverse adecuadamente mediante una sencilla solución basada en reglas. La respuesta puede estar influenciada por un gran número de factores. Cuando las reglas dependen de demasiados factores y muchas de estas reglas se solapan o deben ajustarse muy detenidamente, es difícil para una persona codificar las reglas de manera precisa. El aprendizaje automático es útil para resolver este problema de manera eficaz.
- No puede escalar: es posible que pueda reconocer unos cuantos centenares de correos electrónicos y decidir si son spam o no manualmente. Sin embargo, esta tarea pasa a ser tediosa si se trata de millones de correos electrónicos. Las soluciones de aprendizaje automático son eficaces para la gestión de problemas a gran escala.

Crear una aplicación de Machine Learning

La creación de aplicaciones de ML es un proceso iterativo que implica una secuencia de pasos. Para crear una aplicación de ML, siga estos pasos generales:

1. Establezca un marco para los principales problemas de ML en términos de lo que se observa y qué respuesta desea que el modelo prediga.
2. Recopile, limpie y prepare los datos para que resulten adecuados para que los consuman los algoritmos de aprendizaje de modelos de ML. Visualice y analice los datos para ejecutar comprobaciones de datos, validar su calidad y de comprenderlos.
3. A menudo, los datos sin procesar (variables de entrada) y las respuestas (destino) no se representan de forma que se puedan utilizar para el entrenamiento de un modelo predictivo. Por lo tanto, normalmente debería intentar construir representaciones de entrada o características más predictivas a partir de variables sin procesar.
4. Inserte las características resultantes al algoritmo de aprendizaje para crear modelos y evaluar la calidad de estos en los datos que se omitieron de la creación de modelos.
5. Utilice el modelo para generar predicciones de la respuesta de destino para nuevas instancias de datos.

Formulación del problema

El primer paso para el aprendizaje automático consiste en decidir qué desea predecir, que se conoce como la etiqueta o respuesta de destino. Imagine una situación en la que desea fabricar productos, pero la decisión de fabricación de cada producto depende de su número de ventas potenciales. En este caso, es recomendable predecir cuántas veces cada producto se comprará (predecir el número de ventas). Existen varias maneras de definir este problema mediante la utilización del aprendizaje automático. Elegir cómo definir el problema depende de su caso de uso o necesidad empresarial.

¿Desea predecir el número de compras que clientes realizarán para cada producto (en cuyo caso, el destino es numérico y se está resolviendo un problema de regresión)? O bien, ¿quiere predecir qué productos obtendrán más de 10 compras (en cuyo caso, el destino es binario y está resolviendo un problema de clasificación binaria)?

Es importante evitar complicar excesivamente el problema y enmarcar la solución más sencilla que satisfaga sus necesidades. Sin embargo, también es importante evitar la pérdida de información, especialmente información en las respuestas históricas. Aquí, al convertir un número real de ventas en una variable binaria de tipo "más de 10" en lugar de "menos", se perdería información valiosa. Al

dedicar tiempo en decidir qué destino tiene el mayor sentido predecir, evitará la creación de modelos que no responden a su pregunta.

Recopilación de datos con etiqueta

Los problemas de ML comienzan con los datos: preferiblemente una gran cantidad de datos (ejemplos u observaciones) para los que ya tiene la respuesta de destino. Los datos para los que ya conoce la respuesta de destino se denominan datos etiquetados. En la ML supervisada, el algoritmo se enseña a sí mismo a aprender de los ejemplos etiquetados que proporcionamos.

Cada ejemplo u observación en los datos debe contener dos elementos:

- El destino: la respuesta que desea predecir. Usted proporciona datos que están etiquetados con el destino (a la respuesta correcta) al algoritmo de ML para que este aprenda. A continuación, utilizará el modelo de ML entrenado para predecir esta respuesta en los datos para los que no conoce la respuesta de destino.
- Variables/características: estos son atributos del ejemplo que se pueden utilizar para identificar patrones y predecir la respuesta de destino.

Por ejemplo, en el problema de clasificación de correo electrónico, el destino es una etiqueta que indica si un mensaje de correo electrónico es spam o no spam. Algunos ejemplos de variables son el remitente del correo electrónico, el texto del cuerpo de este, el texto en la línea de asunto, la hora de envío del mensaje de correo y la existencia de correspondencia anterior entre el remitente y el receptor.

Con frecuencia, los datos no se encuentran disponibles de forma etiquetada. La recopilación y preparación de las variables y el destino suelen ser los pasos más importantes a la hora de resolver un problema de ML. Los datos de ejemplo deben ser representativos de los datos de los que dispondrá cuando utilice el modelo para realizar una predicción. Por ejemplo, si desea predecir si un mensaje de correo electrónico es spam o no, debe recopilar correos electrónicos positivos (spam) y negativos (correos electrónicos que no son spam) para que el algoritmo de aprendizaje automático pueda buscar patrones que permitirán distinguir entre los dos tipos de correo electrónico.

Una vez que tenga los datos etiquetados, es posible que tenga que convertirlos en un formato que sea aceptable para su algoritmo o software. Por ejemplo, para utilizar Amazon ML, necesita convertir los datos a un formato de valores separados por comas (CSV), con cada ejemplo componiendo una fila del archivo CSV, cada columna conteniendo una variable de entrada y una columna que contiene la respuesta de destino.

Analizar los datos

Antes insertar los datos etiquetados a un algoritmo de ML, es conveniente inspeccionar los datos para identificar problemas y obtener información sobre los datos que utilice. El poder predictivo del modelo será solo tan bueno como los datos que inserte.

Al analizar los datos, que se deben tener en cuenta las siguientes consideraciones:

- Resúmenes de variable y datos de destino: es útil comprender los valores que las variables toman y qué valores son dominantes en los datos. Podría consultar estos resúmenes con un experto en la materia para el problema que desea resolver. Pregúntese a usted mismo o al experto en la materia: ¿los datos coinciden con sus expectativas? ¿Parece que existe un problema de recopilación de datos? ¿Una clase en el destino aparece con más frecuente que las otras clases? ¿Hay más valores que faltan o datos no válidos que los que esperaba?
- Correlaciones de destino de variable: conocer la correlación entre cada variable y la clase de destino es útil, ya que una correlación elevada implica que existe una relación entre la variable y la clase de destino. En general, es recomendable incluir variables con una correlación elevada porque son las que tienen mayor poder predictivo (señal) y omitir variables con una correlación baja porque es probable que sean irrelevantes.

En Amazon ML, puede analizar los datos al crear un origen de datos y revisar el informe de datos resultante.

Procesamiento de características

Después de conocer los datos a través de resúmenes y visualizaciones de datos, es recomendable transformar las variables aún más para que sean más significativas. Esto se conoce como procesamiento de características. Por ejemplo, supongamos que tiene una variable que captura la fecha y hora a las que se ha producido un evento. Esta fecha y hora nunca volverán a producirse y, por lo tanto, no serán útiles para predecir el destino. Sin embargo, si esta variable se transforma en características que representan la hora del día, el día de la semana y el mes, estas variables podrían ser de utilidad para saber si el evento suele suceder en una hora, semana o mes concretos. Este tipo de procesamiento de características para formar puntos de datos más generalizables de los que aprender pueden ofrecer importantes mejoras a los modelos predictivos.

Otros ejemplos de procesamiento de características comunes:

- Sustituir datos que faltan o datos no válidos con valores más significativos (por ejemplo, si sabe que un valor que falta para una variable de tipo de producto en realidad significa que se trata de un libro, puede sustituir todos los valores que faltan en el tipo de producto con el valor de un libro). Una estrategia común que se utiliza para separar valores que faltan consiste en sustituir los valores que faltan con la media o valor de mediana. Es importante comprender los datos antes de elegir una estrategia para la sustitución de valores que faltan.
- Formación cartesiana de productos de una variable con otra. Por ejemplo, si tiene dos variables, por ejemplo, la densidad de la población (urbana, suburbana, rural) y el estado (Washington, Oregón, California), puede haber información útil en las características formadas por un producto cartesiano de estas dos variables, lo que se traduce en características (urban_Washington, suburban_Washington, rural_Washington, urban_Oregon, suburban_Oregon, rural_Oregon, urban_California, suburban_California, rural_California).
- Transformaciones no lineales, como la colocación de variables numéricas en categorías. En muchos casos, la relación entre una característica numérica y el destino no es lineal (el valor de la característica no aumenta ni disminuye de forma monótona con el destino). En estos casos, puede ser útil guardar la característica numérica en características categóricas que representen distintos rangos de la característica numérica. A continuación, cada característica categórica (contenedor) pueden modelarse como si tuviera su propia relación lineal con el destino. Por ejemplo, supongamos que sabe que la característica numérica continua "age" no está linealmente correlacionada con la probabilidad de comprar un libro. Puede guardar la edad en características categóricas que podrían ser capaces de captar la relación con el destino con más precisión. La cantidad óptima de contenedores para una variable numérica depende de las características de la variable y su relación en el destino y se determina mejor con la experimentación. Amazon ML sugiere el número óptimo de contenedores para una característica numérica en función de las estadísticas de datos en la receta sugerida. Consulte la Guía para desarrolladores para obtener más información acerca de la receta sugerida.
- Características específicas de dominio (por ejemplo, dispone de longitud, amplitud y altura como variables independientes; puede crear una nueva característica de volumen para que sea un producto de estas tres variables).
- Características específicas de variables. Algunos tipos de variables como, por ejemplo, características de texto, características que capturan la estructura de una página web o la estructura de una frase, tienen formas genéricas de procesamiento que ayudan a extraer estructura y contexto. Por ejemplo, formar n-grams a partir del texto "the fox jumped over the fence" se pueden representar con unigrams: the, fox, jumped, over, fence o bigrams: the fox, fox jumped, jumped over, over the, the fence.

Incluir características más relevantes ayuda a mejorar el poder de predicción. Es evidente que no siempre es posible conocer de antemano las características con influencia de "señal" o predictiva. Por lo tanto, es conveniente incluir todas las características que podrían estar relacionadas con la etiqueta de destino y dejar que el algoritmo de aprendizaje de modelos seleccione las características con las correlaciones más fuertes. En Amazon ML, el procesamiento de características se puede especificar en la receta al crear un modelo. Consulte la Guía para desarrolladores para obtener una lista de los procesadores de características disponibles.

Dividir los datos en datos de formación y evaluación

El objetivo fundamental de ML consiste en generalizar más allá de las instancias de datos que se utilizan para entrenar a los modelos. Queremos evaluar el modelo para estimar la calidad de su generalización de patrones para los datos en los que el modelo no ha sido entrenado. Sin embargo, dado que las instancias futuras tienen valores de destino desconocidos y no podemos comprobar ahora mismo la precisión de nuestras predicciones para las instancias del futuro, tenemos que utilizar algunos de los datos para los que ya conocemos la respuesta como proxy para los datos futuros. Evaluar el modelo con los mismos datos que se han utilizado para el entrenamiento no es útil, ya que recompensa a los modelos que pueden "recordar" los datos de entrenamiento en lugar de generalizar.

Una estrategia común consiste en tomar todos los datos etiquetados y dividirlos en subconjuntos de entrenamiento y evaluación, normalmente con una proporción del 70 al 80 % para entrenamiento y un 20 al 30 % para evaluación. El sistema de ML utiliza los datos de entrenamiento para entrenar a los modelos a que vean patrones y utiliza los datos de evaluación para evaluar la calidad de predicción del modelo entrenado. El sistema de ML evalúa el rendimiento predictivo al comparar las predicciones en el conjunto de datos de evaluación con valores verdaderos (conocidos dato real) usando una variedad de métricas. Normalmente, puede utilizar el "mejor" modelo en el subconjunto de evaluación para hacer predicciones sobre instancias futuras para las que no conoce la respuesta de destino.

Amazon ML divide los datos enviados para el entrenamiento de un modelo a través de la consola de en 70 % para entrenamiento y 30 % para evaluación. De forma predeterminada, Amazon ML utiliza el primer 70 % de los datos de entrada en el orden en que aparecen en el origen de datos para entrenar la fuente de datos y el 30 % restante de los datos para la evaluación de la fuente de datos. Amazon ML también permite seleccionar un 70 % aleatorio del origen de datos para el entrenamiento en lugar de utilizar el primer 70 % y utilizando el complemento de este subconjunto aleatorio para la evaluación. Puede usar Amazon ML APIs para especificar ratios de división personalizados y para proporcionar datos de formación y evaluación que se dividieron fuera de Amazon ML. Amazon

ML también ofrece estrategias para dividir los datos. Para obtener más información acerca de las estrategias de división, consulte División de datos.

Entrenar a un modelo

Ahora está listo para proporcionar el algoritmo de ML (es decir, el algoritmo de aprendizaje) con los datos de aprendizaje. El algoritmo aprenderá de los patrones de datos de aprendizaje que asignan las variables al destino y tendrá como salida un modelo que captura estas relaciones. A continuación, el modelo de ML se puede utilizar para obtener predicciones sobre datos nuevos para los que no se conoce la respuesta de destino.

Modelos lineales

Hay una gran cantidad de modelos de ML disponibles. Amazon ML aprende un tipo de modelo de ML: modelos lineales. El término "modelo lineal" implica que el modelo se especifica como una combinación línea de características. Según los datos de aprendizaje, el proceso de aprendizaje procesa una ponderación para cada característica para formar un modelo que pueda predecir o estimar el valor de destino. Por ejemplo, si el destino es la cantidad de seguros que comprará un cliente y las variables son edad e ingresos, un modelo lineal sencillo sería el siguiente:

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

Algoritmo de aprendizaje

La tarea del algoritmo de aprendizaje consiste en aprender las ponderaciones para el modelo. Las ponderaciones describen la probabilidad de que los patrones que el modelo está aprendiendo reflejen las relaciones reales en los datos. Un algoritmo de aprendizaje consta de una función de pérdida y una técnica de optimización. La pérdida es la penalización en la que se incurre cuando la estimación del destino que proporciona el modelo de ML no es exactamente igual al destino. Una función de pérdida cuantifica esta penalización como un único valor. Una técnica de optimización pretende minimizar la pérdida. En Amazon Machine Learning, usamos tres funciones de pérdida, una para cada uno de los tres tipos de problemas de predicciones. La técnica de optimización que se utiliza en Amazon ML está online Stochastic Gradient Descent (SGD). SGD hace pases secuenciales sobre los datos de aprendizaje y, durante cada pase, actualiza las ponderaciones de características un ejemplo a la vez con el fin de tratar las ponderaciones óptimas que minimizan la pérdida.

Amazon ML utiliza los siguientes algoritmos de aprendizaje:

- Para la clasificación binaria, Amazon ML utiliza la regresión logística (función de pérdida logística + SGD).

- Para la clasificación multiclase, Amazon ML utiliza la regresión logística multinomial (función de pérdida multinomial + SGD).
- Para la regresión, Amazon ML utiliza la regresión lineal (función de pérdida cuadrada + SGD).

Parámetros de entrenamiento

El algoritmo de aprendizaje de Amazon ML acepta parámetros, llamados hiperparámetros o parámetros de entrenamiento, que permiten controlar la calidad del modelo resultante. Según el hiperparámetro, Amazon ML selecciona automáticamente opciones de configuración o proporciona valores predeterminados estáticos para los hiperparámetros. Aunque la configuración predeterminada de hiperparámetros generalmente produce modelos útiles, es posible que pueda mejorar el rendimiento predictivo de los modelos si cambia los valores de hiperparámetro. En las siguientes secciones se describen los hiperparámetros comunes asociados con los algoritmos de aprendizaje para modelos lineales, como los que crea Amazon ML.

Tasa de aprendizaje

La tasa de aprendizaje es un valor constante del algoritmo Stochastic Gradient Descent (SGD). La tasa de aprendizaje afecta a la velocidad a la que el algoritmo alcanza (se converge en) las ponderaciones óptimas. El algoritmo SGD realiza actualiza las ponderaciones del modelo lineal por cada ejemplo de datos que encuentre. El tamaño de estas actualizaciones se controla mediante la tasa de aprendizaje. Una tasa de aprendizaje demasiado elevada podría impedir que las ponderaciones alcancen la solución óptima. Un valor demasiado pequeño hace que el algoritmo requiera muchos pases para alcanzar las ponderaciones óptimas.

En Amazon ML, la tasa de aprendizaje se selecciona automáticamente en función de los datos.

Tamaño del modelo

Si tiene muchas características de entrada, el número de posibles patrones en los datos puede resultar en una modelo de gran tamaño. Los modelos de gran tamaño tienen implicaciones prácticas, como por ejemplo, requieren más RAM para almacenar el modelo durante el entrenamiento y al generar predicciones. En Amazon ML, puede reducir el tamaño del modelo utilizando la regularización L1 o restringiendo específicamente el tamaño del modelo mediante la especificación del tamaño máximo. Tenga en cuenta que si reduce el tamaño del modelo demasiado, podría reducir su potencia de predicción.

Para obtener información sobre el tamaño predeterminado de modelo, consulte Parámetros de entrenamiento: tipos y valores predeterminados. Para obtener más información acerca de la regularización, consulte Regularización.

Número de iteraciones

El algoritmo SGD hace pases secuenciales sobre los datos de aprendizaje. El parámetro `Number of passes` controla el número de pases que el algoritmo realiza en los datos de aprendizaje. Un número mayor de pases resulta en un modelo que se adapta mejor a los datos (si la tasa de aprendizaje no es demasiado elevada), pero el beneficio disminuye con una creciente cantidad de pases. Para conjuntos de datos más pequeños, puede aumentar significativamente el número de pases, lo que permite que el algoritmo de aprendizaje se adapte de manera más eficaz a los datos. En el caso de conjuntos de datos extraordinariamente grandes, es posible que un pase sea suficiente.

Para obtener información sobre el número predeterminado de pases, consulte Parámetros de entrenamiento: tipos y valores predeterminados.

Distribución de datos

En Amazon ML, debe distribuir los datos porque el algoritmo de SGD se ve influenciado por el orden de las filas de los datos de aprendizaje. La distribución de los datos de aprendizaje resulta en mejores modelos de ML, ya que ayuda que el algoritmo SGD evita soluciones que son óptimas para el primer tipo de datos que encuentra, pero no para todo el rango de datos. La mezcla desordena los datos, de modo que el algoritmo SGD no detecta un tipo de datos por demasiadas observaciones consecutivas. Si solo encuentra un tipo de datos para muchas actualizaciones de ponderación sucesivas, es posible que el algoritmo no pueda corregir las ponderaciones del para el nuevo tipo de datos porque pueda que la actualización sea demasiado grande. Asimismo, cuando los datos no se presentan de forma aleatoria, es difícil para el algoritmo encontrar una solución óptima para todos los tipos de datos de forma rápida; en algunos casos, el algoritmo podría no encontrar nunca la solución óptima. La distribución de los datos de aprendizaje ayuda al algoritmo a converger en la solución óptima con mayor rapidez.

Por ejemplo, supongamos que desea entrenar un modelo de ML para predecir un tipo de producto y los datos de entrenamiento incluyen los tipos de producto película, juegos y videojuegos. Si clasifica los datos por la columna de tipo de productos antes de cargar los datos en Amazon S3, el algoritmo verá los datos alfabéticamente por tipo de producto. El algoritmo observa primero todos los datos de películas y el modelo de ML comienza a aprender patrones para películas. A continuación, cuando el modelo encuentra los datos de juguetes, cada actualización que hace el algoritmo ajustaría el

modelo al tipo de producto de juguete, incluso si estas actualizaciones degradasen los patrones que se ajustan a las películas. Este cambio repentino del tipo de películas a juguetes puede producir un modelo que no aprenderá a predecir los tipos de productos con precisión.

Para obtener información sobre el tipo de distribución, consulte Parámetros de entrenamiento: tipos y valores predeterminados.

Regularización

La regularización ayuda a evitar que los modelos lineales sobreajusten los ejemplos de datos de aprendizaje (es decir, memorizar patrones en lugar de generalizarlos) al penalizar valores de ponderación extremos. La regularización L1 tiene el efecto de reducir el número de características que se utilizan en el modelo al establecer en cero las ponderaciones de características que, de otro modo, tendrían ponderaciones muy reducidas. Como resultado, la regularización L1 produce modelos dispersos y reduce la cantidad de ruido en el modelo. La regularización L2 produce valores de ponderación generales más pequeños y estabiliza las ponderaciones cuando hay gran correlación entre las características de entrada. Puede controlar la cantidad de regularización L1 o L2 que se aplica mediante los parámetros `Regularization type` y `Regularization amount`. Un valor de regularización extremadamente elevado podría resultar en que todas las características tengan ponderaciones cero, lo que impide que el modelo aprenda patrones.

Para obtener información sobre los valores de regularización predeterminados, consulte Parámetros de entrenamiento: tipos y valores predeterminados.

Evaluación de la precisión del modelo

El objetivo del modelo de ML es aprender patrones que generalizan bien en cuanto a datos sin analizar en lugar de memorizar los datos que ha visto durante el entrenamiento. Cuando tenga un modelo, es importante comprobar si este tiene un buen rendimiento en relación con los ejemplos sin analizar que no haya usado para el entrenamiento del modelo. Para ello, utilice el modelo para predecir la respuesta sobre el conjunto de datos de evaluación (datos omitidos) y luego comprar el destino previsto con la respuesta real (dato real).

Un número de métricas se utilizan en ML para medir la precisión predictiva del modelo. La elección de la métrica de precisión depende de la tarea de ML. Es importante revisar estas métricas para decidir si el modelo tiene un buen rendimiento.

Clasificación binaria

El resultado real de muchos algoritmos de clasificación binaria es una puntuación de predicción. La puntuación indica la certeza del modelo de que la observación dada pertenezca a la clase positiva. Para tomar la decisión sobre si la observación debe clasificarse como positiva o negativa, como consumidor de esta puntuación, interpretará la puntuación seleccionando un umbral de clasificación (corte) y comparará la puntuación con dicho umbral. Cualquier observación con puntuaciones superiores al umbral se prevé como la clase positiva y las puntuaciones inferiores al umbral se prevén como la clase negativa.

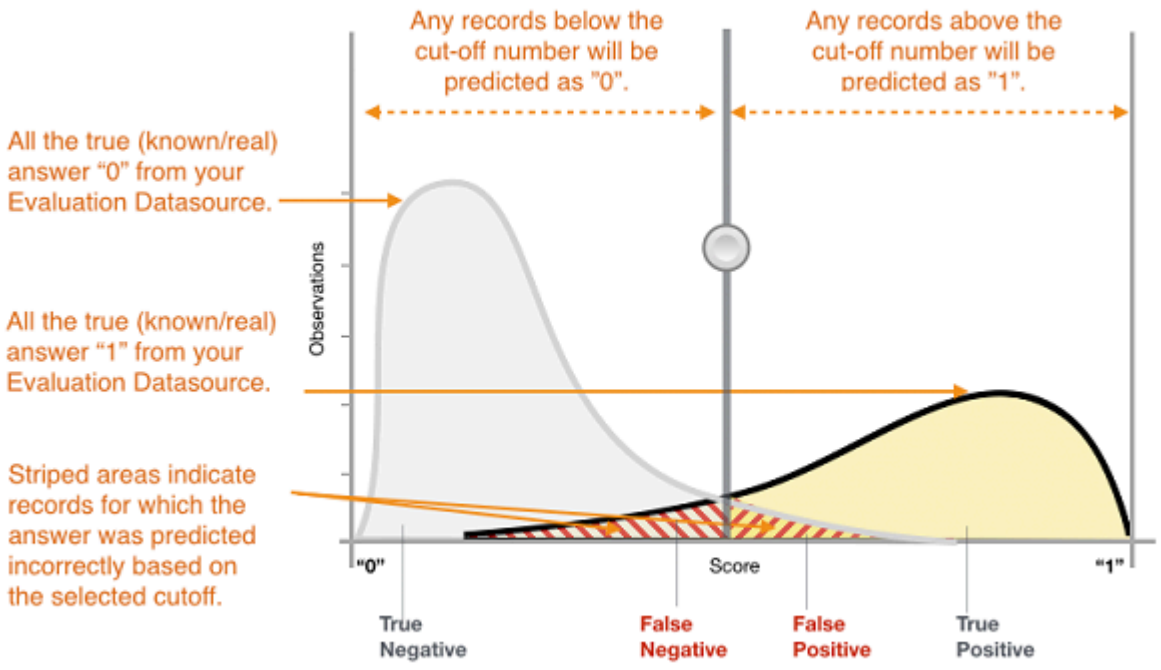


Figura 1: distribución de puntuaciones para el modelo de clasificación binaria

Las predicciones ahora se dividen en cuatro grupos en función de la respuesta conocida real, así como la respuesta predicha: predicciones positivas correctas (positivas reales), predicciones negativas correctas (negativas reales), predicciones positivas incorrectas (positivas falsas) y predicciones negativas incorrectas (negativas falsas).

Las métricas de precisión de clasificación binaria cuantifican los dos tipos de predicciones correctas y dos tipos de errores. Las métricas típicas son exactitud (ACC), precisión, recuperación, tasa de falsos positivos, medición F1. Cada métrica mide otro aspecto del modelo predictivo. La exactitud (ACC) mide la fracción de predicciones correctas. La precisión mide la fracción de positivos reales

entre los ejemplos que se prevén como positivos. La recuperación mide cuantos positivos reales se predijeron como positivos. La medida F1 es la media armónica de la precisión y la recuperación.

AUC es otro tipo de métrica. Mide la capacidad del modelo de predecir una mayor puntuación para ejemplos positivos en comparación con ejemplos negativos. Dado que AUC es independiente del umbral seleccionado, puede obtener una sensación del rendimiento de predicción del modelo a partir de la métrica de AUC sin elegir un umbral.

En función del problema de su negocio, puede que le interese más un modelo que funcione bien para un subconjunto concreto de estas métricas. Por ejemplo, dos aplicaciones empresariales podrían tener requisitos muy diferentes para sus modelos de ML:

- Una aplicación podría necesitar estar muy segura de que las predicciones positivas sean realmente positivas (alta precisión) y podría permitirse la clasificación incorrecta de algunos ejemplos positivos como negativos (recuperación moderada).
- Otra aplicación podría necesitar predecir correctamente el mayor número de ejemplos positivos posible (recuperación elevada) y aceptaría que algunos ejemplos negativos se clasifiquen incorrectamente como positivos (precisión moderada).

En Amazon ML, las observaciones obtienen una puntuación predicha en el rango [0,1]. El umbral de puntuación para tomar la decisión de clasificar ejemplos como 0 o 1 se establece de forma predeterminada en 0,5. Amazon ML permite revisar las implicaciones de elegir umbrales de puntuación diferentes y permite elegir un umbral adecuado que se ajuste a sus necesidades empresariales.

Clasificación multiclase

A diferencia del proceso de los problemas de clasificación binaria, no tiene que elegir un umbral de puntuación para realizar predicciones. La respuesta predicha es la clase (por ejemplo, etiqueta) con la puntuación máxima predicha. En algunos casos, es posible que desee utilizar la respuesta predicha solo si se predice con una puntuación elevada. En este caso, puede elegir un umbral en las puntuaciones predichas en función de si aceptará la respuesta predicha o no.

Las métricas típicas que se utilizan en la multiclase son las mismas que se utilizan en el caso de la clasificación binaria. La métrica se calcula para cada clase al procesarla como un problema de clasificación binaria después de agrupar todas las otras clases como pertenecientes a la segunda clase. A continuación, se calcula el promedio de la métrica entre todas las clases para obtener una métrica de promedio macro (procesar cada clase como igual) o de media ponderada (ponderada

por frecuencia de clase). En Amazon ML, la medición F1 de media se utiliza para evaluar el éxito predictivo de un clasificador multiclase.

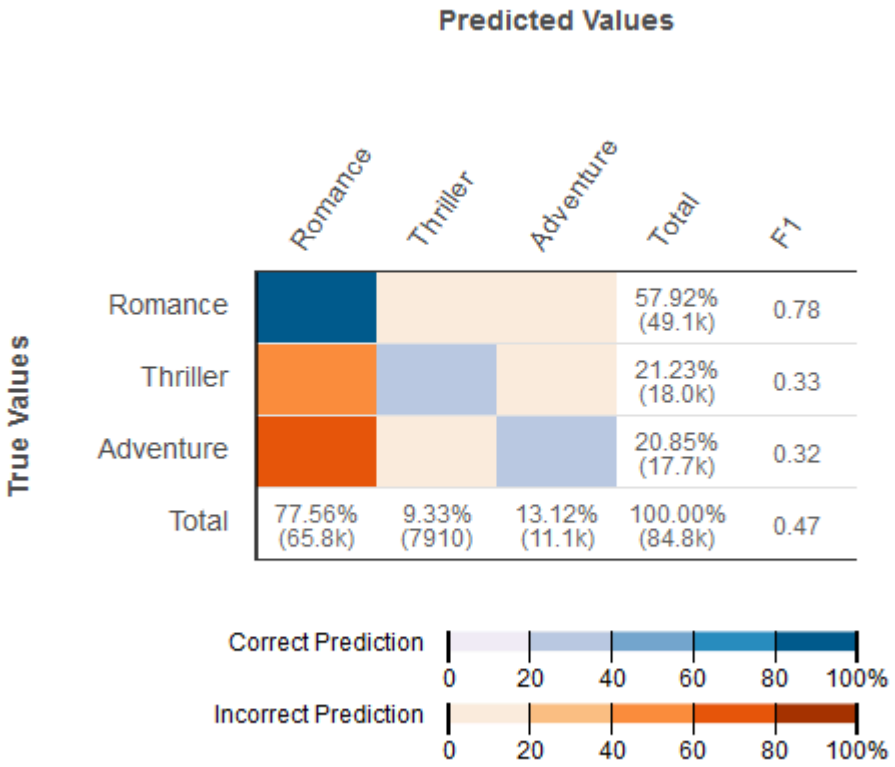


Figura 2: matriz de confusión para un modelo de clasificación multiclase

Se recomienda repasar la matriz de confusión para los problemas de multiclase. La matriz de confusión es una tabla en la que se muestra cada clase de los datos de evaluación y el número o porcentaje de predicciones correctas y predicciones incorrectas.

Regresión

Para las tareas de regresión, las métricas típicas de precisión son el error cuadrado medio raíz (RMSE) y el error porcentual absoluto medio (MAPE). Estas métricas miden la distancia entre el destino numérico predicho y la respuesta numérica real (dato real). En Amazon ML, la métrica RMSE se utiliza para evaluar la exactitud predictiva de un modelo de regresión.

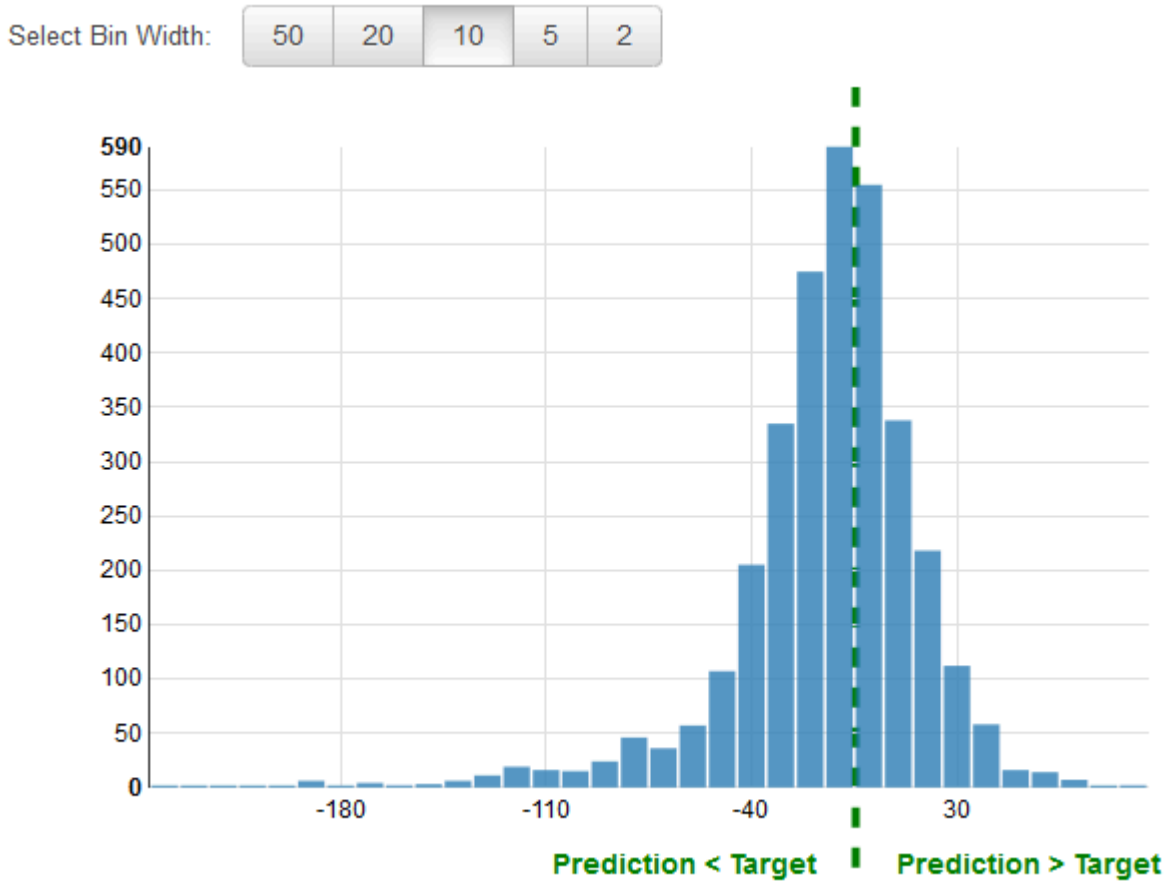


Figura 3: distribución de los residuales de un modelo de regresión

Es una práctica común repasar los residuales para los problemas de regresión. Un residual para una observación en los datos de evaluación es la diferencia entre el destino verdadero y el destino predicho. Los residuales representan la parte del destino que el modelo no puede predecir. Un residual positivo indica que el modelo está subestimando el destino (el destino real es mayor que el destino predicho). Un residual negativo indica una sobreestimación (el destino real es menor que el destino predicho). El histograma de los residuales en los datos de evaluación distribuido en forma de campana y centrado en el cero indica que el modelo comete errores de forma aleatoria y no subestima o sobreestima sistemáticamente un rango determinado de valores de destino. Si los residuales no forman una forma de campana centrada en cero, existe una estructura en el error de predicción del modelo. Añadir más variables al modelo podría ayudar a este a capturar el patrón que no captura el modelo actual.

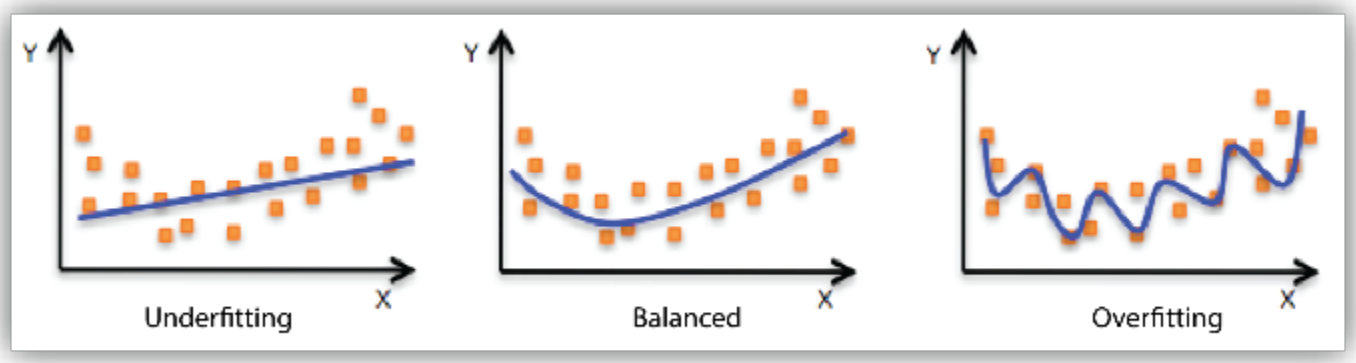
Mejorar la precisión del modelo

Obtener un modelo de ML que coincida con sus necesidades suele implicar la iteración de este proceso de ML y la prueba de algunas variantes. Es posible que no obtenga un modelo muy predictivo en la primera iteración, o quizás desee mejorar el modelo para obtener predicciones aún mejores. Para mejorar el rendimiento, puede iterar por estos pasos:

1. Recopilar datos: aumente el número de ejemplos de entrenamiento.
2. Procesamiento de características: añada más variables y mejor de procesamiento de características.
3. Ajuste de parámetros del modelo: considere valores alternativos para los parámetros de entrenamiento que el algoritmo de aprendizaje utiliza.

Ajuste del modelo: ajustes deficientes vs. ajustes excesivos

Comprender el ajuste del modelo es importante para comprender la causa raíz para una precisión deficiente del modelo. Esta información le ayudará a tomar medidas correctivas. Podemos determinar si un modelo predictivo presenta un ajuste deficiente o un ajuste excesivo de los datos de entrenamiento observando el error de predicción en los datos de entrenamiento y los datos de evaluación.



El modelo presenta ajustes deficientes en los datos de aprendizaje cuando el modelo presenta un rendimiento deficiente en los datos de aprendizaje. Esto se debe a que el modelo no puede capturar la relación entre los ejemplos de entrada (a menudo llamados X) y los valores de destino (a menudo llamados Y). El modelo presenta ajustes excesivos en los datos de entrenamiento cuando el modelo presenta un buen rendimiento con los datos de entrenamiento pero no con los datos de evaluación.

Esto se debe a que el modelo memoriza los datos que ha visto y no puede generalizar para los ejemplos no vistos.

Un rendimiento deficiente en los datos de aprendizaje podría deberse a que el modelo es demasiado sencillo (las características de entrada no son suficientemente expresivas) como para describir el destino adecuadamente. El rendimiento se puede mejorar aumentando la flexibilidad. Para aumentar la flexibilidad del modelo, pruebe lo siguiente:

- Añada nuevas características específicas del dominio y más productos cartesianos de características, y cambie los tipos de procesamiento de características utilizado (por ejemplo, aumentando el tamaño n-grams).
- Reduzca la cantidad de regularización utilizada.

Si el modelo presenta ajustes excesivos en los datos de entrenamiento, es razonable tomar medida que reduzcan la flexibilidad del modelo. Para reducir la flexibilidad del modelo, pruebe lo siguiente:

- Selección de características: considerar el uso de combinaciones de características, la reducción del tamaño n-grams y la reducción del número de contenedores de atributos.
- Aumente la cantidad de regularización utilizada.

La precisión en los datos de entrenamiento y prueba podría ser deficiente porque el algoritmo de aprendizaje no tenía datos suficientes de los que aprender. Podría mejorar el rendimiento haciendo lo siguiente:

- Aumentar el número de ejemplos de datos de entrenamiento.
- Aumentar el número de pases en los datos de entrenamiento existentes.

Uso del modelo para hacer predicciones

Ahora que tiene un modelo de ML con un buen rendimiento, podrá utilizarlo para realizar predicciones. En Amazon Machine Learning, hay dos formas de utilizar un modelo para realizar predicciones:

Predicciones por lotes

Las predicciones por lote son de utilidad cuando desea generar predicciones para un conjunto de observaciones a la vez y luego tomar medidas en un determinado porcentaje o número de

observaciones. Normalmente, no existe un requisito de baja latencia para este tipo de aplicación. Por ejemplo, cuando desea decidir a qué clientes dirigirse como parte de una campaña de publicidad para un producto, obtendrá puntuaciones de predicción para todos los clientes, ordenará las predicciones del modelo para identificar qué clientes presentan mayor probabilidad de compra y luego se dirigirá a quizás el 5 % de clientes principales con mayor probabilidad de compra.

Predicciones online

Los escenarios de predicción en línea son para los casos en los que desea generar predicciones sobre una one-by-one base para cada ejemplo independiente de los demás ejemplos, en un entorno de baja latencia. Por ejemplo, podría utilizar predicciones para tomar decisiones inmediatas sobre la probabilidad de que una transacción determinada sea una transacción fraudulenta.

Retención de modelos en datos nuevos

Para que un modelo haga previsiones con precisión, los datos en los que se basa las previsiones deben tener una distribución similar a los datos en los que se ha entrenado el modelo. Dado que se espera que las distribuciones de datos cambien con el tiempo, la implementación de un modelo no es un ejercicio puntual, sino un proceso continuo. Es una buena práctica monitorizar constantemente los datos entrantes y volver a entrenar el modelo en los datos más reciente si percibe que la distribución de datos se ha desviado significativamente la distribución de datos de entrenamiento original. Si la monitorización de los datos para detectar un cambio en la distribución de datos tiene un costo elevado, una estrategia más sencilla es entrenar al modelo de forma periódica, por ejemplo, de manera diaria, semanal o mensual. Para volver a entrenar los modelos en Amazon ML, debe crear un modelo nuevo basado en los nuevos datos de aprendizaje.

El proceso de Amazon Machine Learning

En la tabla siguiente se describe cómo utilizar la consola de Amazon ML para realizar el proceso de ML descrito en este documento.

Procesamiento de ML	Tarea de Amazon ML
Análisis de sus datos	Para analizar sus datos en Amazon ML, cree una fuente de datos y revise la página de detalles de los datos.

Procesamiento de ML	Tarea de Amazon ML
División de los datos en fuentes de datos de entrenamiento y evaluación	Amazon ML puede dividir la fuente de datos para que utilice el 70% de los datos para entrenar el modelo y el 30% para evaluar el desempeño predictivo del mismo. Cuando utiliza el asistente de creación de un modelo de ML con la configuración predeterminada, Amazon ML divide los datos por usted. Si utiliza el asistente de creación de un modelo de ML con ajustes personalizados y elija evaluar el modelo de ML, verá una opción para permitir que Amazon ML divida los datos y ejecute una evaluación en el 30% de los datos.
Mezcla de los datos de entrenamiento	Cuando utiliza el asistente de creación de un modelo de ML con la configuración predeterminada, Amazon ML mezcla los datos por usted. También puede mezclar sus datos antes de importarlos en Amazon ML.
Procesamiento de funciones	El proceso de recopilar datos de entrenamiento en un formato óptimo para el aprendizaje y la generalización se conoce como transformación de funciones. Cuando utiliza el asistente de creación de un modelo de ML con la configuración predeterminada, Amazon ML sugiere una configuración de procesamiento de características para los datos. Para especificar la configuración de procesamiento de funciones, utilice la opción Custom (Personalizado) del asistente de creación de un modelo de ML (Create ML Model) y proporcione receta de procesami ento de funciones.
Entrenamiento del modelo	Cuando utiliza el asistente de creación de un modelo de ML para crear un modelo en Amazon ML, Amazon ML entrena el modelo.
Selección de los parámetros del modelo	En Amazon ML puede ajustar cuatro parámetros que afectan al desempeño predictivo de su modelo: tamaño del modelo, número de iteraciones, tipo de mezcla y regularización. Puede definir estos parámetros cuando utilice el asistente de creación de un modelo de ML para crear un modelo de ML y elegir la opción Custom (Personalizado).

Procesamiento de ML	Tarea de Amazon ML
Evaluación del desempeño del modelo	Utilice el asistente "Create Evaluation" (crear evaluación) para valorar el desempeño predictivo de su modelo.
Selección de funciones	El algoritmo de aprendizaje de Amazon ML puede descartar funciones que no contribuyen mucho al proceso de aprendizaje. Para indicar que desea descartar estas funciones, elija el parámetro `L1 regularization` cuando cree el modelo de ML.
Establecimiento de un umbral de puntuación para la exactitud de predicciones	Revise el desempeño predictivo del modelo en el informe de evaluación con diferentes umbrales de puntuación y, a continuación, defina el umbral de puntuación en función de su aplicación comercial. El umbral de puntuación determina cómo define el modelo una coincidencia de predicción. Ajuste el número para controlar los falsos positivos y los falsos negativos.
Utilización del modelo	Utilice el modelo para obtener predicciones para un lote de observaciones con el asistente Create Batch Prediction (crear una predicción por lotes). Obtenga predicciones para observaciones individuales bajo demanda habilitando el modelo de ML para procesar predicciones en tiempo real a través de la API `Predict`.

Configuración de Amazon Machine Learning

Necesita una cuenta de AWS para poder empezar a utilizar Amazon Machine Learning. Si no tiene ninguna cuenta, consulte la página de registro de AWS.

Inscripción en AWS

Cuando se inscribe en Amazon Web Services (AWS), la cuenta de AWS se registra automáticamente en todos los servicios de AWS, incluido Amazon ML. Solo se le cobrará por los servicios que utilice. Si ya tiene una cuenta de AWS, omita este paso. Si no dispone de una cuenta de AWS, utilice el siguiente procedimiento para crear una.

Para inscribirse en una cuenta de AWS

1. Vaya a https://aws.amazon.com y elija Registro.
2. Siga las instrucciones en pantalla.

Parte del procedimiento de inscripción consiste en recibir una llamada telefónica e introducir un número PIN con el teclado del teléfono.

Tutorial: Utilización de Amazon ML para predecir respuestas a una oferta de marketing

Con Amazon Machine Learning (Amazon ML), puede crear y entrenar modelos predictivos y alojar sus aplicaciones en una solución escalable en la nube. En este tutorial le mostramos cómo utilizar la consola de Amazon ML para crear una fuente de datos, crear un modelo de machine learning (ML) y utilizar el modelo para generar predicciones que podrá utilizar en sus aplicaciones.

Nuestro ejercicio de muestra ilustra cómo identificar posibles clientes para una campaña de marketing dirigida, pero puede aplicar los mismos principios para crear y utilizar una gran variedad de modelos de ML. Para completar el ejercicio de muestra, utilizará conjuntos de datos de banca y marketing disponibles públicamente que provienen de University of California at Irvine (UCI) Machine Learning Repository. Estos conjuntos de datos contienen información general acerca de los clientes, así como información sobre cómo respondieron a contactos de marketing anteriores. Utilizará estos datos para identificar qué clientes es más probable que se suscriban a su producto nuevo y un depósito de banco de términos, también conocido como certificado de depósito (CD).

> **Warning**
>
> Este tutorial no está incluido en la capa gratuita de AWS. Para obtener más información sobre los precios de Amazon ML, consulte Precios de Amazon Machine Learning.

Requisito previo

Para realizar el tutorial, necesita disponer de una cuenta de AWS. Si no dispone de una cuenta de AWS, consulte Setting Up Amazon Machine Learning.

Pasos

- Paso 1: prepare los datos
- Paso 2: cree una fuente de datos de entrenamiento
- Paso 3: Crear una modelo de ML
- Paso 4: Revisar el desempeño predictivo del modelo de ML y establecer un umbral de puntuación
- Paso 5: Uso del modelo de ML para generar predicciones

- Paso 6: eliminación

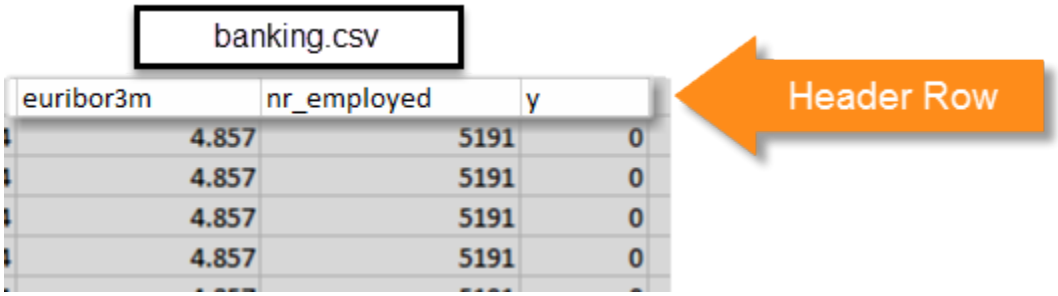
Paso 1: prepare los datos

En el aprendizaje automático, se suelen obtener los datos y asegurarse de que tienen el formato correcto antes de comenzar el proceso de entrenamiento. A efectos de este tutorial, hemos obtenido un conjunto de datos de muestra de UCI Machine Learning Repository, les hemos dado formato para cumplir con las directrices de Amazon ML y los hemos puesto a disposición para que los descargue. Descargue el conjunto de datos desde nuestra ubicación de almacenamiento de Amazon Simple Storage Service (Amazon S3) y cárguelo a su propio bucket de S3 siguiendo los procedimientos de este tema.

Para los requisitos de formato de Amazon ML, consulte Compresión del formato de datos de Amazon ML.

Descarga de los conjuntos de datos

1. Descargue el archivo que contiene los datos históricos de los clientes que han adquirido productos similares a su depósito de banco de términos haciendo clic en banking.zip. Descomprima la carpeta y guarde el archivo banking.csv en su equipo.
2. Descargue el archivo que utilizará para predecir si los clientes potenciales responderán a su oferta haciendo clic en banking-batch.zip. Descomprima la carpeta y guarde el archivo banking-batch.csv en su equipo.
3. Abra `banking.csv`. Verá filas y columnas de datos. La fila de encabezado contiene los nombres de atributo para cada columna. Un atributo es una propiedad con un nombre único que describe una característica particular de cada cliente; por ejemplo, "nr_employed" indica la situación profesional del cliente. Cada fila representa la colección de observaciones acerca de un único cliente.

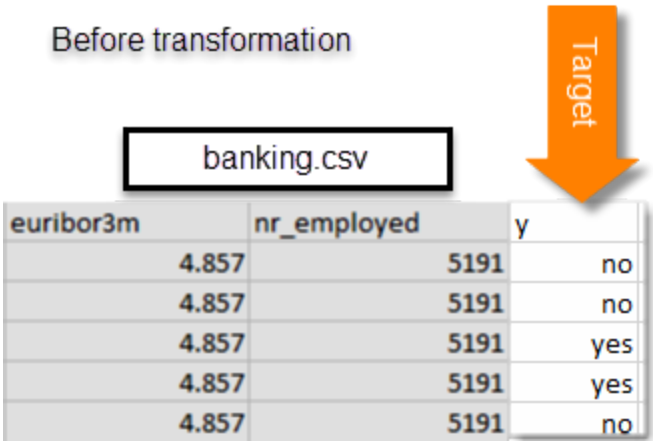


Desea que su modelo de ML responda a la pregunta "¿Este cliente va a suscribirse a mi nuevo producto?". En el conjunto de datos `banking.csv`, la respuesta a esta pregunta es el atributo

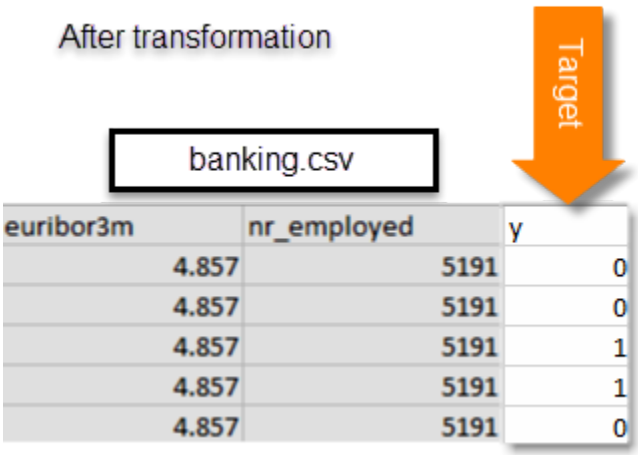
y, que contiene los valores 1 (para "sí") o 0 (para "no"). El atributo que desea que Amazon ML aprenda a predecir se conoce como el atributo de destino.

> **Note**
>
> El atributo y es un atributo binario. Puede contener solo uno de los dos valores; en este caso, 0 o 1. En el conjunto de datos de UCI original el atributo y es Sí o No. Hemos editado el conjunto de datos original. Todos los valores del atributo y que significan "sí" son 1 y todos los valores que significan "no" son 0. Si utiliza sus datos propios, puede utilizar otros valores para un atributo binario. Para obtener más información acerca de los valores válidos, consulte Uso del AttributeType campo.

Los siguientes ejemplos muestran los datos antes y después de que se cambiaran los valores del atributo y a los atributos binarios 0 y 1.



euribor3m	nr_employed	y
4.857	5191	no
4.857	5191	no
4.857	5191	yes
4.857	5191	yes
4.857	5191	no



euribor3m	nr_employed	y
4.857	5191	0
4.857	5191	0
4.857	5191	1
4.857	5191	1
4.857	5191	0

El archivo `banking-batch.csv` no contiene el atributo y. Una vez que haya creado un modelo de ML, podrá utilizar el modelo para predecir y para cada registro en dicho archivo.

A continuación, cargue los archivos `banking.csv` y `banking-batch.csv` a Amazon S3.

Carga de los archivos a una ubicación de Amazon S3

1. Inicie sesión en la consola de Amazon S3 AWS Management Console y ábrala en https://console.aws.amazon.com/s3/.
2. En la lista All Buckets (Todos los buckets), cree un bucket o elija la ubicación donde desee cargar los archivos.
3. En la barra de navegación, elija Upload (Cargar).
4. Seleccione Add Files (Añadir archivos).
5. En el cuadro de diálogo, diríjase al escritorio, elija `banking.csv` y `banking-batch.csv` y, a continuación, seleccione Open (Abrir).

Ahora está preparado para crear su origen de datos de entrenamiento.

Paso 2: cree una fuente de datos de entrenamiento

Después de cargar el conjunto de datos de `banking.csv` a su ubicación de Amazon Simple Storage Service (Amazon S3), la utilizará para crear un origen de datos de entrenamiento. Una fuente de datos es un objeto de Amazon Machine Learning (Amazon ML) que contiene la ubicación de los datos de entrada y metadatos importantes sobre los datos de entrada. Amazon ML utiliza la fuente de datos para operaciones como el entrenamiento y la evaluación del modelo de ML.

Para crear una fuente de datos, proporcione los siguientes datos:

- Ubicación de Amazon S3 de sus datos de y permisos para obtener acceso a ellos
- El esquema, que incluye los nombres de los atributos en los datos y el tipo de cada atributo (Numeric, Text, Categorical o Binary)
- El nombre del atributo que contiene la respuesta que desea que aprenda a predecir Amazon ML el atributo de destino

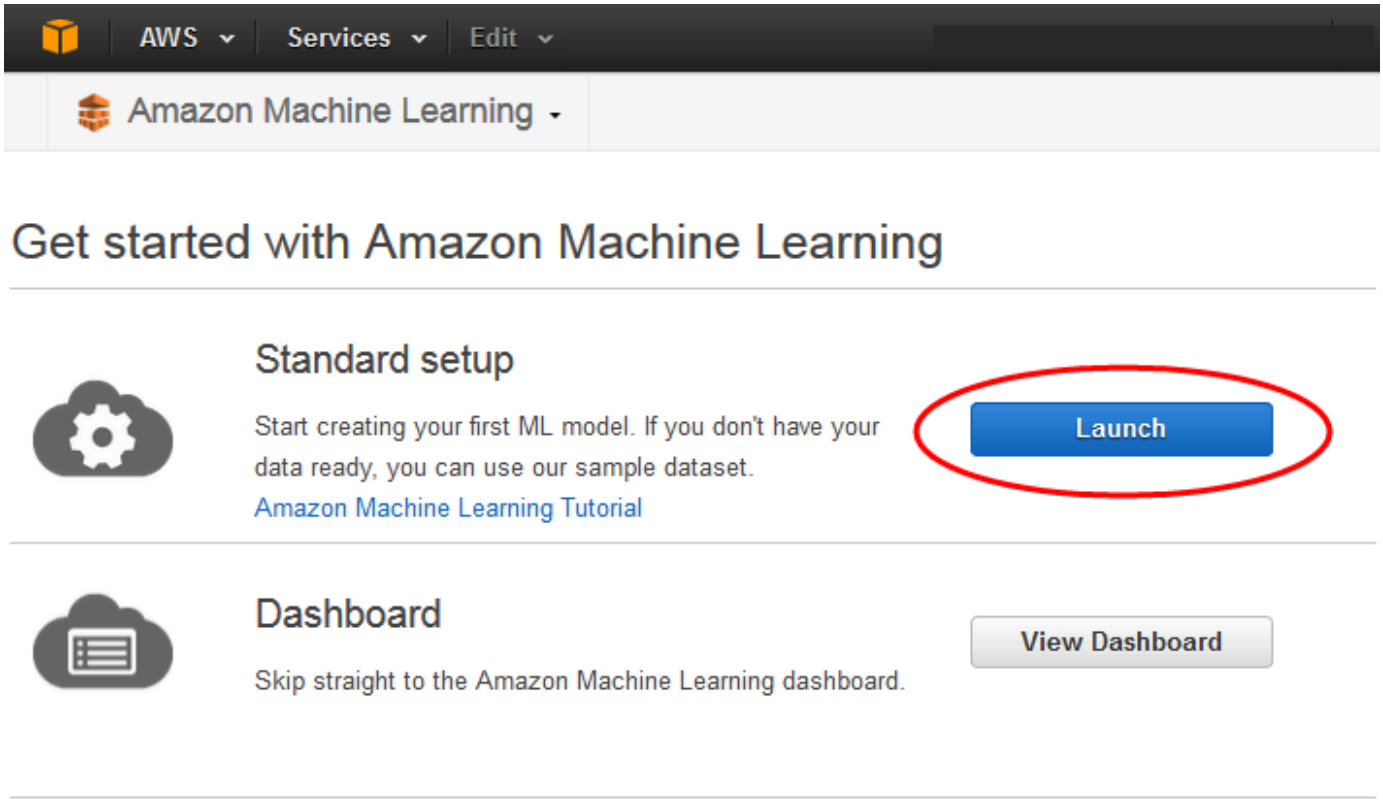
> **Note**
>
> La fuente de datos realmente no almacena sus datos, sino que solo les hace referencia. Evite mover o cambiar los archivos almacenados en Amazon S3. Si los mueve o los cambia, Amazon ML no puede obtener acceso a ellos para crear un modelo de ML, generar evaluaciones o generar predicciones.

Creación de la fuente de datos de entrenamiento

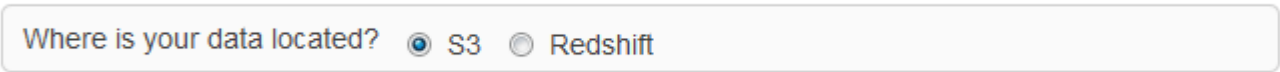
1. Abra la consola Amazon Machine Learning en https://console.aws.amazon.com/machinelearning/.
2. Elija Comenzar.

   > **Note**
   >
   > Este tutorial supone que es la primera vez que utiliza Amazon ML. Si ha usado Amazon ML antes, puede utilizar la lista desplegable Crear nuevo… en el panel de Amazon ML para crear un origen de datos nuevo.

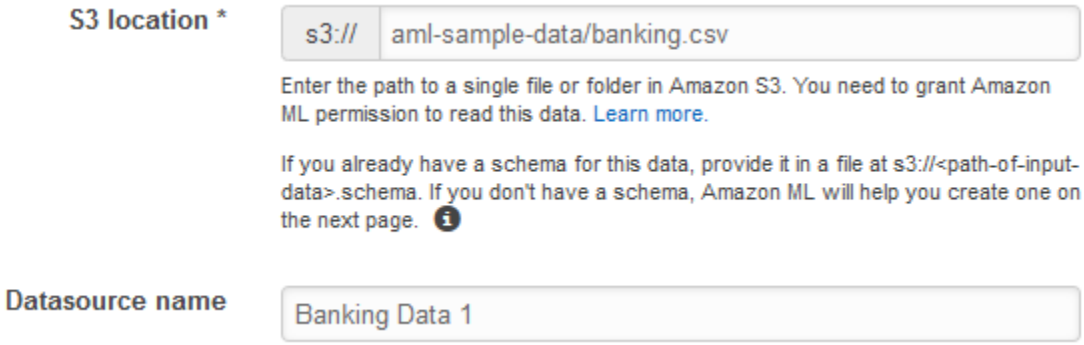
3. En la página Introducción a Amazon Machine Learning, seleccione Lanzar.



4. En la página Input Data (Datos de entrada), para Where is your data located? (¿Dónde están sus datos?), asegúrese de que está marcado S3.

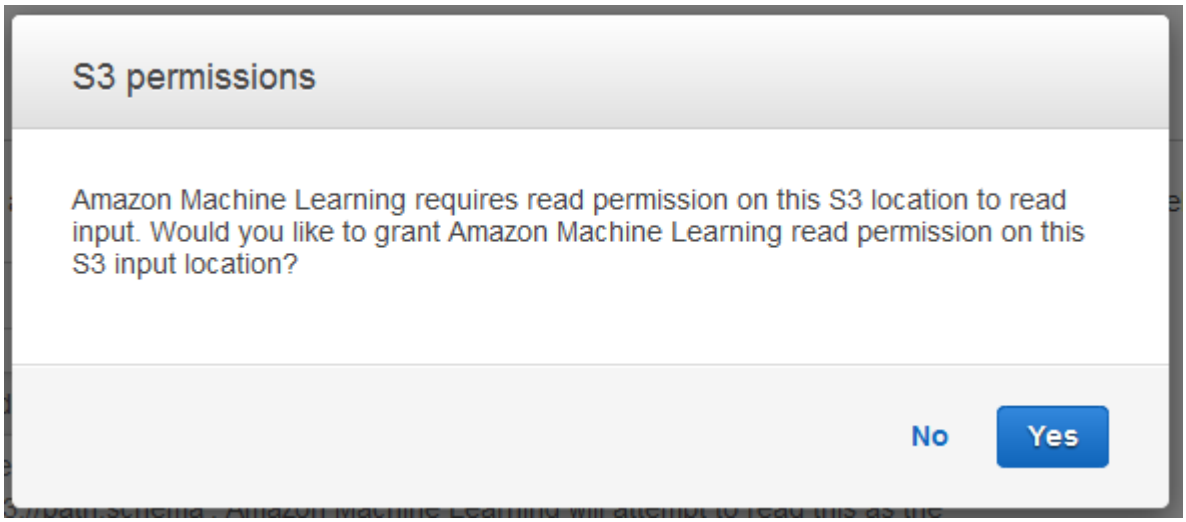


5. Para S3 Location (Ubicación de S3), escriba la ubicación completa del archivo `banking.csv` del paso 1: prepare los datos. Por ejemplo: *`your-bucket`*`/banking.csv`. Amazon ML añade s3:// al nombre de su bucket por usted.
6. En Datasource name (Nombre de origen de datos), escriba **`Banking Data 1`**.

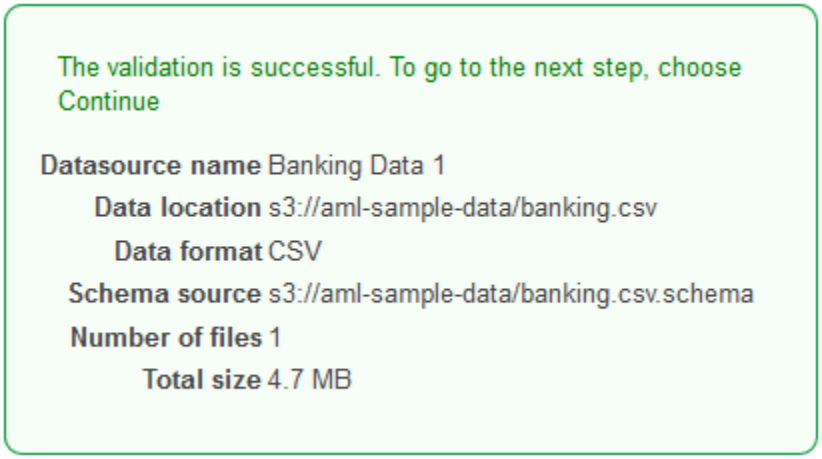


7. Seleccione Verificar.

8. En el cuadro de diálogo S3 permissions (Permisos de S3), elija Yes (Sí).



9. Si Amazon ML puede obtener acceso al archivo de datos y leerlo en la ubicación de S3, verá una página similar a la siguiente. Revise las propiedades y, a continuación, elija Continue (Continuar).



A continuación, establezca un esquema. Un esquema es la información que necesita Amazon ML para interpretar los datos de entrada de un modelo de ML, incluidos los nombres de los atributos y sus tipos de datos asignados, así como los nombres de los atributos especiales. Hay dos formas de proporcionar un esquema a Amazon ML:

- Proporcione un archivo de esquema independiente al cargar los datos de Amazon S3.
- Permitir que Amazon ML infiera los tipos de atributo y cree un esquema por usted.

En este tutorial, pediremos a Amazon ML que infiera el esquema.

Para obtener información sobre la creación de archivo de esquema independiente, consulte Creación de un esquema de datos para Amazon ML.

Permisos para que Amazon ML infiera el esquema

1. En la página Esquema, Amazon ML muestra el esquema que infirió. Revise los tipos de datos que ha inferido Amazon ML para los atributos. Es importante que los atributos estén señalados con el tipo de datos correcto para ayudar a que Amazon ML reciba los datos correctamente y habilitar el procesamiento de características correcto en los atributos.

   - Los atributos que solo tienen dos estados posibles, como sí o no, deberían estar marcados como Binary (Binario).
   - Los atributos que son números o cadenas que se utilizan para denotar una categoría deberían estar marcados como Categorical (Categórico).
   - Los atributos que son cantidades numéricas cuyo orden es relevante deberían estar marcados como Numeric (Numérico).
   - Los atributos que son cadenas que desea tratar como palabras delimitadas por espacios deberían estar marcados como Text (Texto).

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. En este tutorial, Amazon ML ha identificado correctamente los tipos de datos para todos los atributos. Por lo tanto, seleccione Continuar.

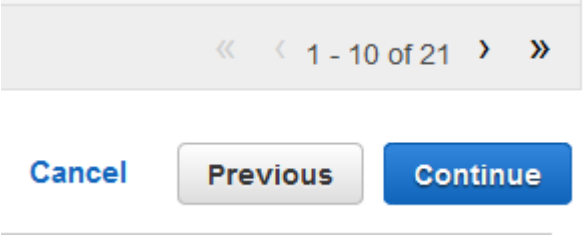
A continuación, seleccione un atributo de destino.

Recuerde que el destino es el atributo que el modelo de ML debe aprender a predecir. El atributo y indica si un individuo se ha suscrito a una campaña en el pasado: 1 (sí) o 0 (no).

> **Note**
>
> Elija un atributo de destino solo si utilizará la fuente de datos para entrenar y evaluar modelos de ML.

Selección de "y" como el atributo de destino

1. En la parte inferior derecha de la tabla, elija la flecha simple para avanzar a la última página de la tabla, donde aparece el atributo con el nombre y.



2. En la columna Target (Destino), seleccione y.

   Amazon ML confirma que y está seleccionado en el destino.

3. Elija Continuar.

4. En la página Row ID (ID de fila), en Does your data contain an identifier? (Los datos contienen un identificador?), asegúrese de que está seleccionado No, el valor predeterminado.
5. Seleccione Review (Revisar) y, a continuación, Continue (Continuar).

Ahora que tiene un origen de datos de entrenamiento, está listo para crear su modelo.

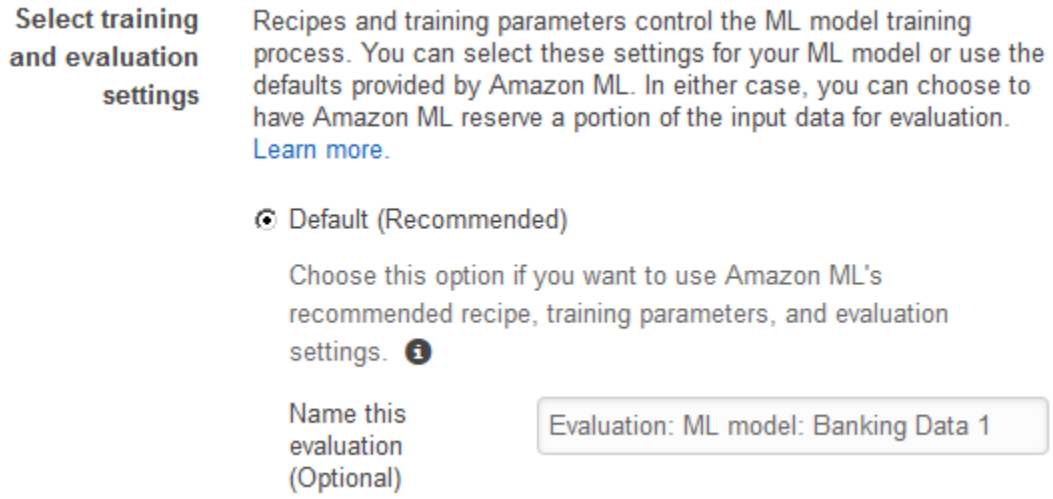
Paso 3: Crear una modelo de ML

Una vez que haya creado la fuente de datos de formación, se utiliza para crear un modelo de ML, formar el modelo y, a continuación, evaluar los resultados. El modelo de ML es un conjunto de patrones que Amazon ML busca en sus datos durante la formación. El modelo se utiliza para crear predicciones.

Para crear un modelo de ML

1. Puesto que el asistente Get started crea un origen de datos de formación y un modelo, Amazon Machine Learning (Amazon ML) utiliza automáticamente el origen de datos de formación que acaba de crear y le lleva directamente a la página Configuración de modelo de ML. En la página ML model settings (Configuración de modelo de ML), para ML model name (Nombre de modelo de ML), asegúrese de que se muestra **`ML model: Banking Data 1`**, el valor predeterminado.

   El uso de un nombre sencillo, como el predeterminado, le ayuda a identificar y administrar fácilmente el modelo de ML.
2. Para Training and evaluation settings (Configuración de entrenamiento y evaluación), asegúrese de que se selecciona Default (Predeterminado).



3. En Name this evaluation (Asignar nombre a esta evaluación), acepte el valor predeterminado, **`Evaluation: ML model: Banking Data 1`**.
4. Elija Review (Revisar), revise los ajustes y, a continuación, seleccione Finish (Finalizar).

   Después de seleccionar Finalizar, Amazon ML añade el modelo a la cola de procesamiento. Cuando Amazon ML crea el modelo, se aplica la configuración predeterminada y realiza las siguientes acciones:

   - Divide el origen de datos de entrenamiento en dos secciones, una que contiene el 70% de los datos y otra que contiene el 30% restante
   - Forma el modelo de ML en la sección que contiene el 70% de los datos de entrada
   - Evalúa el modelo mediante el 30% restante de los datos de entrada

   Mientras el modelo está en la cola, Amazon ML informa del estado como Pendiente. Mientras Amazon ML crea el modelo, informa del estado como En curso. Cuando ha finalizado todas las acciones, informa del estado como Completed (Completado). Espere a que finalice la evaluación antes de continuar.

Ahora ya está listo para revisar el rendimiento del modelo y establecer un corte de puntuación.

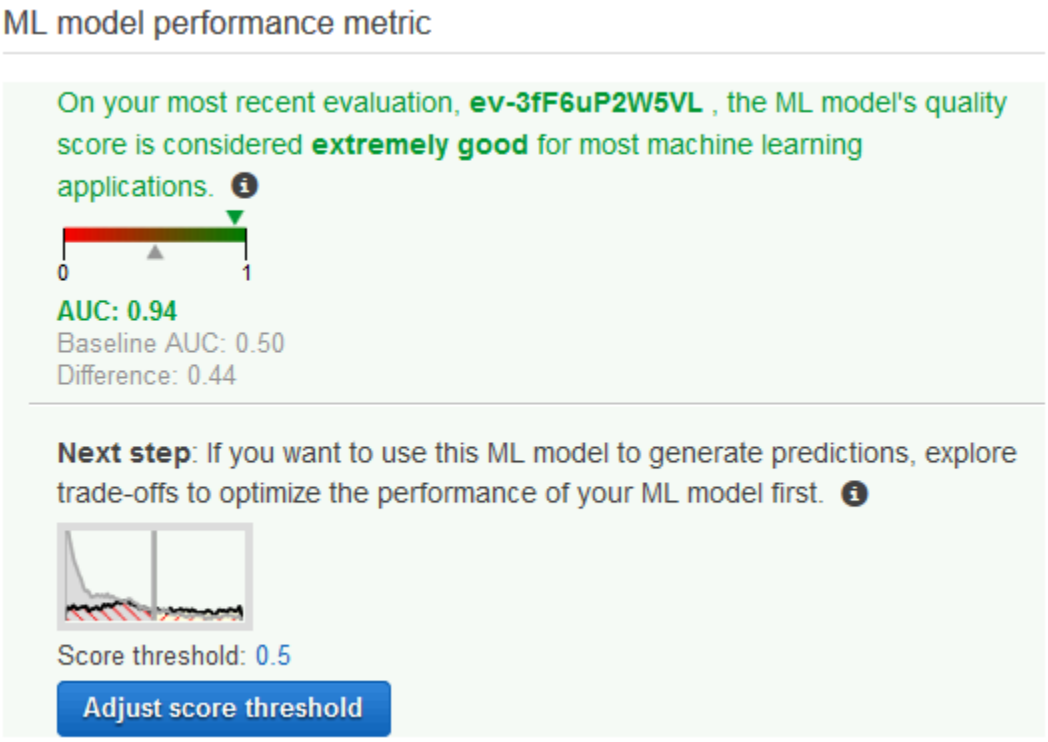
Para obtener más información acerca de los modelos de formación y evaluación, consulte Entrenamiento de modelos de ML y evaluate an ML model.

Paso 4: Revisar el desempeño predictivo del modelo de ML y establecer un umbral de puntuación

Ahora que ha creado el modelo de ML y Amazon Machine Learning (Amazon ML) lo ha evaluado, vamos a ver si es suficientemente bueno para ponerlo en práctica. Durante la evaluación, Amazon ML ha calculado una métrica de calidad estándar del sector, denominada métrica Area Under a Curve (AUC), que expresa la calidad del desempeño del modelo de ML. Amazon ML también interpreta la métrica AUC para indicarle si la calidad del modelo de ML es suficiente para la mayoría de las aplicaciones de machine learning. (Para obtener más información sobre AUC, consulte Medición de la precisión del modelo de ML.) Vamos a revisar la métrica AUC y, a continuación, a ajustar el umbral o límite de puntuación para optimizar el desempeño predictivo del modelo.

Revisar la métrica AUC del modelo de ML

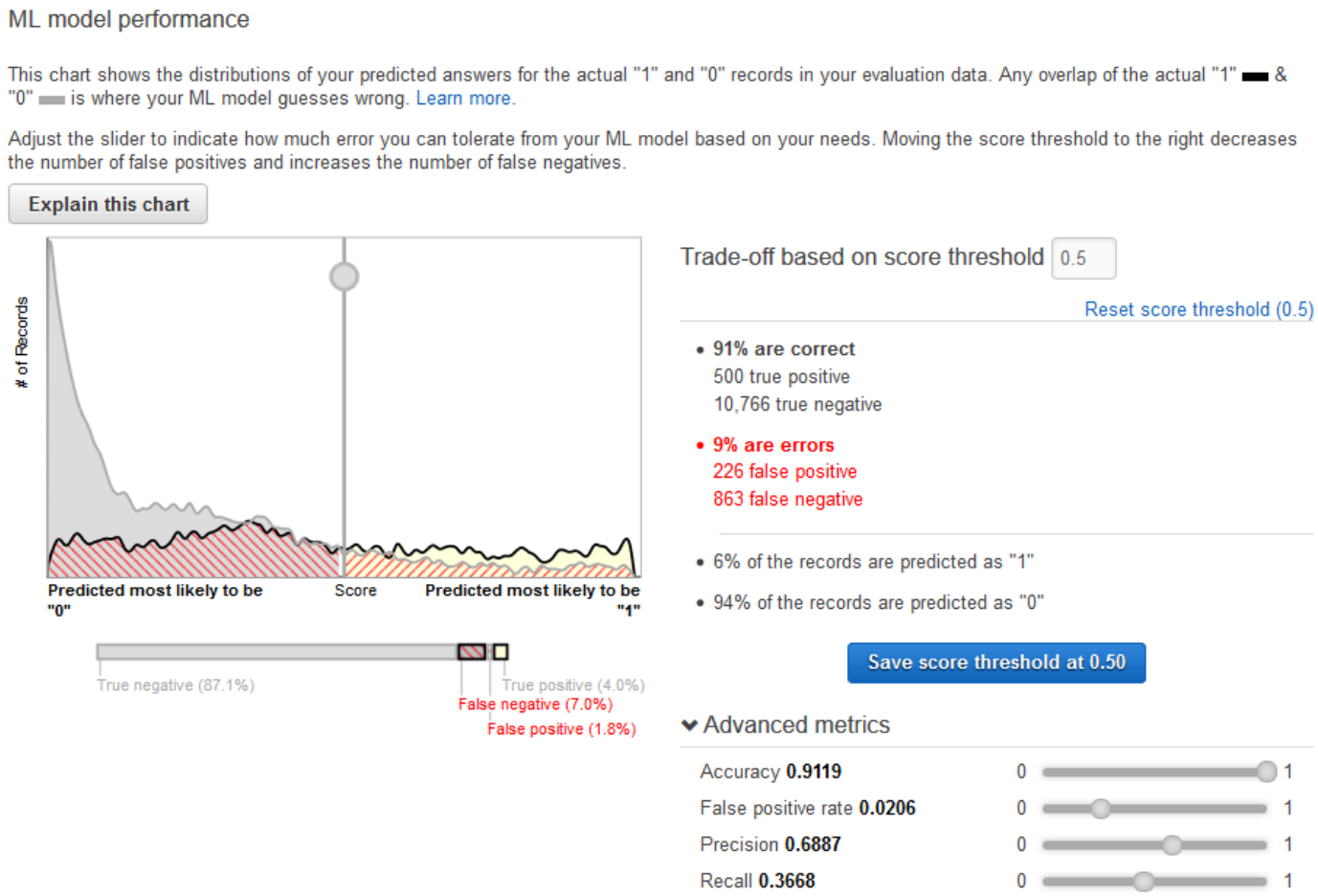
1. En la página ML model summary (Resumen de modelos de ML), en el panel de navegación ML model report (Informe de modelos de ML), seleccione Evaluations (Evaluaciones), Evaluation: ML model: Banking model 1 (Evaluación: modelo de ML: modelo bancario 1) y, a continuación, Summary (Resumen).
2. En la página Evaluation summary (Resumen de evaluación), revise el resumen de la evaluación, incluida la métrica de desempeño de AUC del modelo.



El modelo de ML genera puntuaciones numéricas de predicciones para cada registro de una fuente de datos de predicciones y, a continuación, aplica un umbral para convertir estas puntuaciones en etiquetas binarias de 0 (para no) o 1 (para sí). Al cambiar el umbral de puntuación, puede ajustar la manera en la que el modelo de ML asigna estas etiquetas. A continuación, establezca el umbral de puntuación.

Establecer un umbral de puntuación para el modelo de ML

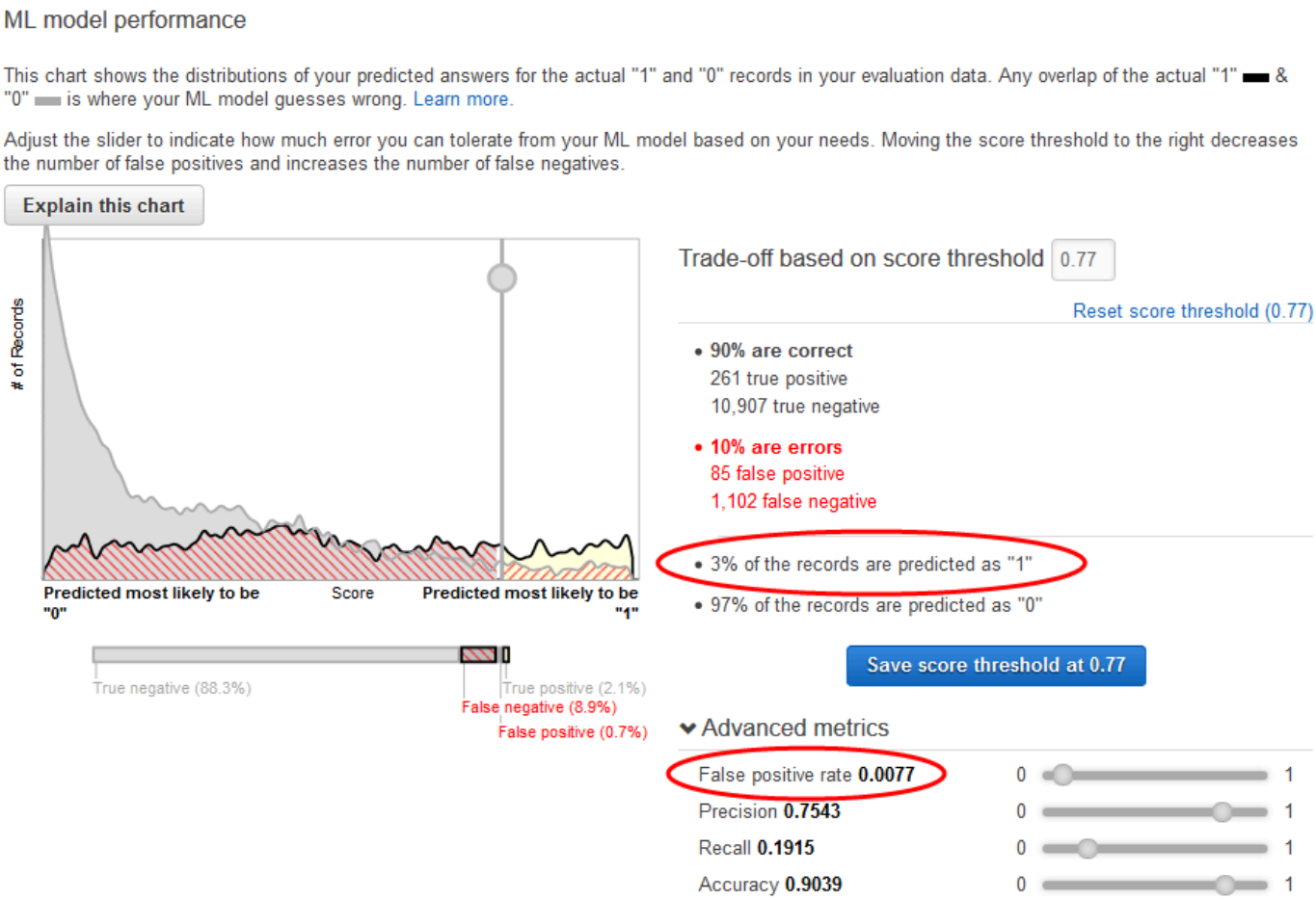
1. En la página Evaluation Summary (Resumen de evaluación), seleccione Adjust Score Threshold (Ajustas umbral de puntuación).



Puede ajustar las métricas de desempeño del modelo de ML ajustando el umbral de puntuación. Ajustar este valor cambia el nivel de confianza que debe tener el modelo en una predicción antes de considerar que la predicción es positiva. También cambia la cantidad de falsos negativos y falsos positivos que está dispuesto a tolerar en sus predicciones.

Puede controlar el límite de lo que el modelo considera una predicción positiva incrementando el umbral de puntuación hasta que solo considere que las predicciones con más probabilidad de ser positivos reales son positivas. También puede reducir el umbral de puntuación hasta que deje de tener falsos negativos. Seleccione un límite para reflejar las necesidades de su empresa. En este tutorial, cada falso positivo cuesta dinero de campaña, por lo que queremos una proporción alta de positivos reales respecto a falsos positivos.

2. Digamos que desea dirigirse al primer 3% de clientes que se suscribirán al producto. Deslice el selector vertical para establecer el umbral de puntuación en un valor que corresponda a 3% of the records are predicted as "1" (3% de los registros están previstos como "1").



Observe el impacto de este umbral de puntuación en el rendimiento del modelo de ML: la tasa de falsos positivos es de 0,007. Supongamos que esa proporción de falsos positivos es aceptable.

3. Seleccione Save score threshold at 0.77 (Guardar umbral de puntuación en 0,77).

Cada vez que usa este modelo de ML para realizar predicciones, predecirá los registros con puntuaciones superiores al 0,77 como "1", y el resto de los registros como "0".

Para obtener más información sobre el umbral de puntuación, consulte Clasificación binaria.

Ahora está preparado para generar predicciones utilizando un modelo.

Paso 5: Uso del modelo de ML para generar predicciones

Amazon Machine Learning (Amazon ML) puede generar dos tipos de predicciones: por lotes y en tiempo real.

Una predicción en tiempo real es una predicción para una única observación que Amazon ML genera bajo demanda. Las predicciones en tiempo real son ideales para aplicaciones móviles, sitios web y otras aplicaciones que necesitan utilizar los resultados de forma interactiva.

Una predicción por lotes es un conjunto de predicciones para un grupo de observaciones. Amazon ML procesa los registros en una predicción por lotes conjuntamente, por lo que el procesamiento puede tardar un tiempo. Utilice las predicciones por lote para aplicaciones que requieren predicciones para un conjunto de observaciones o predicciones que no utilicen los resultados de forma interactiva.

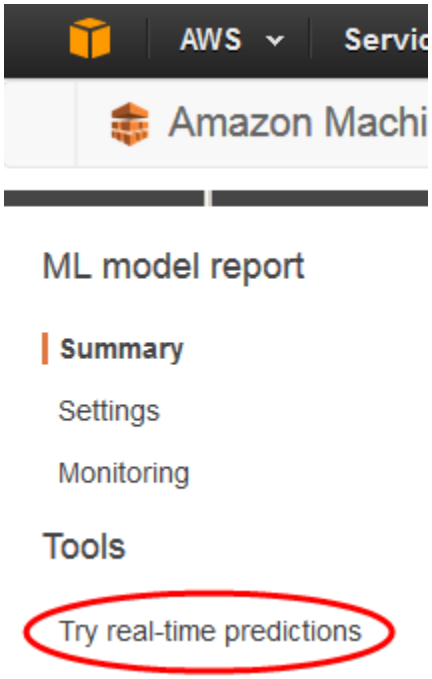
En este tutorial, generará una predicción en tiempo real que prediga si un cliente potencial se suscribirá al nuevo producto. Asimismo, generará predicciones para un gran lote de clientes potenciales. Para la predicción por lotes, se utilizará el archivo `banking-batch.csv` que ha cargado en el Paso 1: prepare los datos.

Empecemos con una predicción en tiempo real.

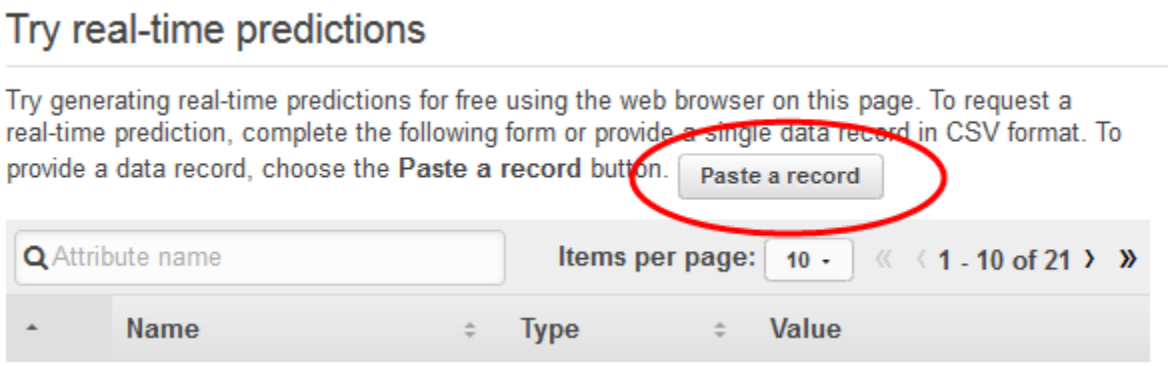
> **Note**
>
> Para las aplicaciones que requieren predicciones en tiempo real, debe crear un punto de enlace en tiempo real para el modelo de ML. Se acumulan cargos mientras se encuentra disponible un punto de enlace en tiempo real. Antes de comprometerse a utilizar las predicciones en tiempo real y empezar a incurrir en el costo derivado de las mismas, puede probar a utilizar la característica de predicción en tiempo real en el navegador web, sin necesidad de crear un punto de enlace en tiempo real. Eso es lo que vamos a hacer para este tutorial.

Probar una predicción en tiempo real

1. En el panel de navegación ML model report, seleccione Try real-time predictions.



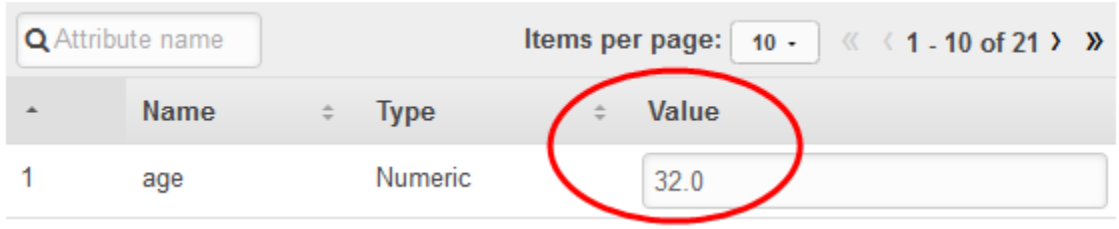
2. Elija Paste a record.



3. En el cuadro de diálogo Paste a record, pegue la siguiente observación:

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

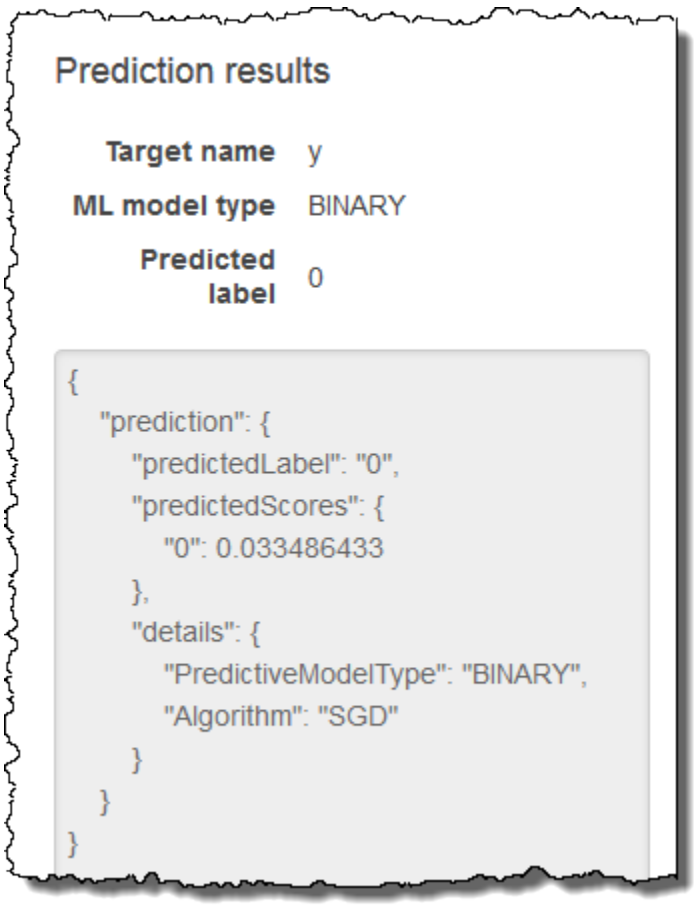
4. En el cuadro de diálogo Pegar un registro, elija Enviar para confirmar que desea generar una predicción para esta observación. Amazon ML rellena los valores en el formulario de predicciones en tiempo real.



> **Note**
>
> También puede rellenar los campos Value (Valor) escribiendo los valores individuales. Independientemente del método que elija, debe proporcionar una observación que no se haya utilizado para formar el modelo.

5. En la parte inferior de la página, elija Create prediction.

   La predicción aparece en el panel Prediction results a la derecha. Esta predicción tiene una Predicted label de `0`, lo que significa que no es probable que este cliente potencial responda a la campaña. Una Predicted label (Etiqueta predicha) de 1 significaría que el cliente posiblemente responda a la campaña.

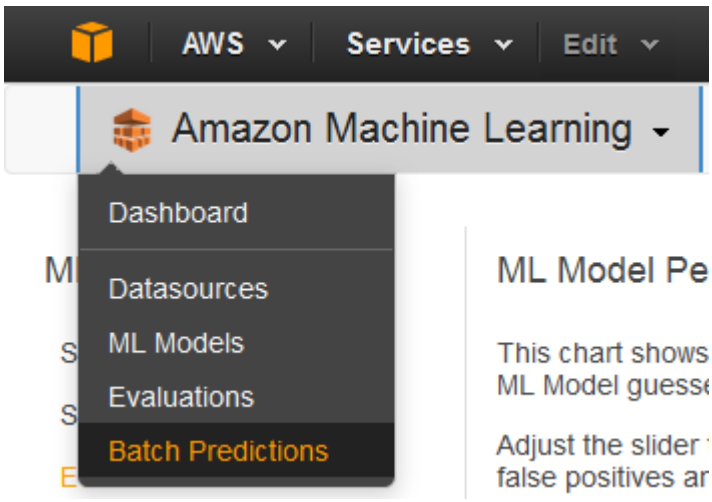


Ahora, cree una predicción por lotes. Proporcionará a Amazon ML el nombre del modelo de ML que está usando, la ubicación de Amazon Simple Storage Service (Amazon S3) de los datos de entrada

para los que desea generar predicciones (Amazon ML creará un origen de datos de predicciones por lotes a partir de estos datos) y la ubicación de Amazon S3 para almacenar los resultados.

Para crear una predicción por lotes

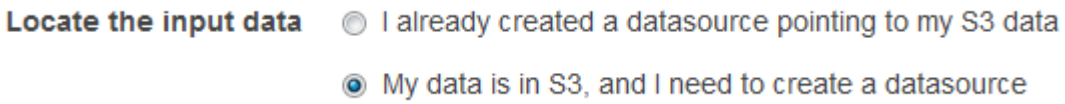
1. Elija Amazon Machine Learning y, a continuación, elija Batch Predictions (Predicciones por lotes).



2. Elija Create new batch prediction (Crear nueva predicción por lotes).
3. En la página ML model for batch predictions (Modelo de ML para predicciones por lotes), elija ML model: Banking Data 1 (Modelo de ML: datos bancarios 1).

   Amazon ML muestra el nombre del modelo de ML, el ID, la hora de creación y el ID de la fuente de datos asociada.
4. Elija Continuar.
5. Para generar predicciones, debe proporcionar a Amazon ML los datos para los que necesita predicciones. Estos se denominan datos de entrada. En primer lugar, coloque los datos de entrada en una fuente de datos de forma que Amazon ML pueda acceder a ellos.

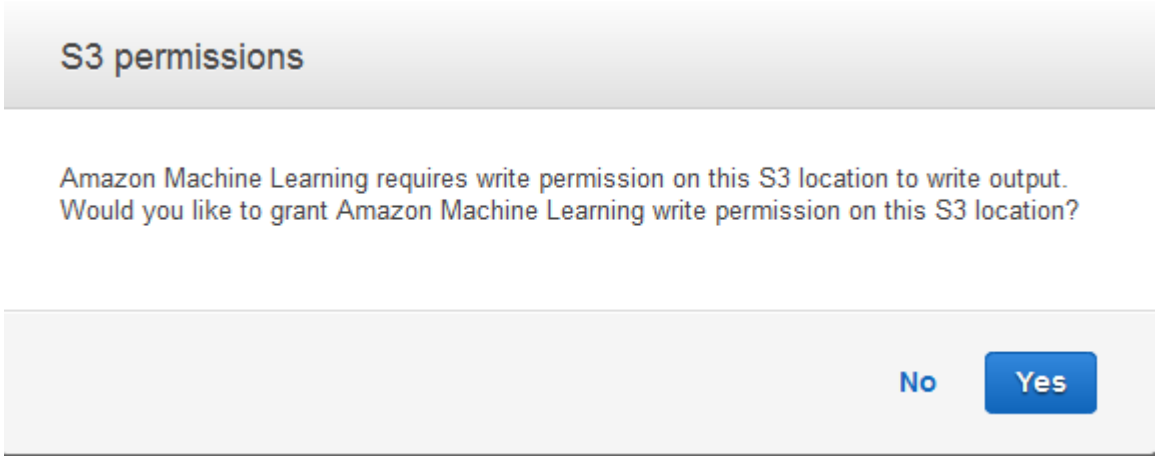
   Para Locate the input data (Localizar los datos de entrada), elija My data is in S3, and I need to create a datasource (Mis datos están en S3 y debo crear un origen de datos).



6. En Datasource name (Nombre de origen de datos), escriba **`Banking Data 2`**.
7. En S3 Location, escriba la ubicación completa del `banking-batch.csv` archivo: *`your-bucket`*`/`**`banking-batch.csv`**.

8. En Does the first line in your CSV contain the column names? (¿La primera línea del CSV contiene los nombres de columna?), elija Yes (Sí).
9. Seleccione Verificar.

   Amazon ML valida la ubicación de los datos.
10. Elija Continuar.
11. Para Destino S3, escriba el nombre de la ubicación de Amazon S3 donde ha cargado los archivos en el Paso 1: Preparación de sus datos. Amazon ML carga los resultados de la predicción ahí.
12. Para el nombre de predicción por lotes, acepte el valor predeterminado, **`Batch prediction: ML model: Banking Data 1`**. Amazon ML elige el nombre predeterminado en función del modelo que utilizará para crear predicciones. En este tutorial, el modelo y las previsiones se nombran según la fuente de datos de formación, `Banking Data 1`.
13. Elija Revisar.
14. En el cuadro de diálogo S3 permissions (Permisos de S3), elija Yes (Sí).

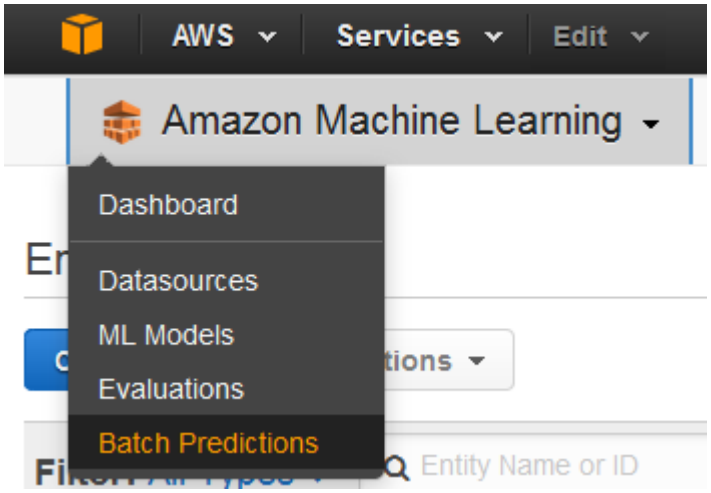


15. En la página Revisar, elija Finalizar.

    La solicitud de predicciones por lote se envía a Amazon ML y se introduce en una cola. El tiempo que tarda Amazon ML en procesar una predicción por lotes depende del tamaño de la fuente de datos y de la complejidad de su modelo de ML. Aunque Amazon ML procesa la solicitud, notifica el estado En curso. Después de que haya finalizado la predicción por lotes, el estado de la solicitud cambia a Completed (Completado). Ahora puede ver los resultados.

Para ver las predicciones

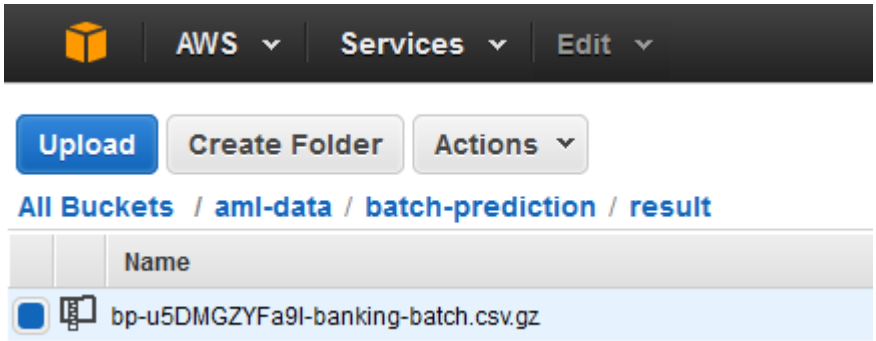
1. Elija Amazon Machine Learning y, a continuación, elija Batch Predictions (Predicciones por lotes).



2. En la lista de predicciones, elija Batch prediction: ML model: Banking Data 1 (Predicciones por lotes: modelo de ML: datos bancarios 1). Aparece la página Batch prediction info (Información de predicción por lotes).

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. Para ver los resultados de la predicción de lotes, vaya a la consola de Amazon S3 en https://console.aws.amazon.com/s3/y navegue hasta la ubicación de Amazon S3 a la que se hace referencia en el campo URL de salida de S3. A partir de ahí, desplácese hasta la carpeta de resultados, que tendrá un nombre similar a `s3://aml-data/batch-prediction/result`.



La predicción se almacena en un archivo comprimido .gzip con la extensión .gz.

4. Descargue el archivo de predicción en el escritorio, descomprímalo y ábralo.

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

El archivo tiene dos columnas, bestAnswer y score, y una fila para cada una de las observaciones del origen de datos. Los resultados de la columna bestAnswer se basan en el umbral de puntuación de 0,77 que estableció en el Paso 4: Revisar el desempeño predictivo del modelo de ML y establecer un umbral de puntuación. Un valor de score superior a 0,77 genera una bestAnswer de 1, que es una respuesta o una predicción positiva, y un valor de score inferior a 0,77 genera una bestAnswer de 0, que es una respuesta o una predicción negativa.

Los siguientes ejemplos muestran predicciones positivas y negativas en función del umbral de puntuación de 0,77.

Predicción positiva:

bestAnswer	score
1	0.8228876

En este ejemplo, el valor de bestAnswer es 1 y el valor de score es 0,8228876. El valor de bestAnswer es 1 porque el valor de score es mayor que el umbral de puntuación de 0,77. Un valor de bestAnswer de 1 indica que el cliente posiblemente adquiera el producto y, por lo tanto, se considera una predicción positiva.

Predicción negativa:

bestAnswer	score
0	0.7695356

En este ejemplo, el valor de bestAnswer es 0 porque el valor de score es 0,7695356, que es menor que el umbral de puntuación 0,77. El valor de bestAnswer de 0 indica que no es probable que el cliente adquiera el producto y, por lo tanto, se considera una predicción negativa.

Cada fila del resultado por lotes se corresponde con una fila de su entrada por lotes (un comentario de su fuente de datos).

Después de analizar las predicciones, puede ejecutar su campaña de marketing dirigida; por ejemplo, puede enviar folletos a todas las personas con una puntuación de predicción de 1.

Ahora que ha creado, revisado y utilizado el modelo, limpie los datos y los recursos de AWS que ha creado para evitar incurrir en gastos innecesarios y a mantener su espacio de trabajo despejado.

Paso 6: eliminación

Para evitar acumular cargos adicionales de Amazon Simple Storage Service (Amazon S3), elimine los datos almacenados en Amazon S3. No se le aplicará ningún cargo por otros recursos de Amazon ML no utilizados, pero le recomendamos que los elimine para mantener limpio su espacio de trabajo.

Eliminación de los datos de entrada almacenados en Amazon S3

1. Abra la consola de Amazon S3 en https://console.aws.amazon.com/s3/.
2. Acceda a la ubicación de Amazon S3 donde almacenó los archivos `banking.csv` y `banking-batch.csv`.
3. Seleccione los archivos `banking.csv`, `banking-batch.csv` y `.writePermissionCheck.tmp`.
4. Elija Acciones y, a continuación, elija Eliminar.
5. Cuando se pida confirmación, elija Aceptar.

Aunque no se le aplica ningún cargo por mantener el registro de la predicción por lotes que Amazon ML ejecutó en las fuentes de datos, el modelo y la evaluación que creó durante el tutorial, le recomendamos que los elimine para no desordenar su espacio de trabajo.

Eliminación de las predicciones por lotes

1. Vaya a la ubicación de Amazon S3 donde almacenó el resultado de las predicciones por lotes.
2. Elija la carpeta `batch-prediction`.
3. Elija Acciones y, a continuación, elija Eliminar.
4. Cuando se pida confirmación, elija Aceptar.

Para eliminar los recursos de Amazon ML

1. En el panel de Amazon ML, seleccione los siguientes recursos.
   - La fuente de datos `Banking Data 1`
   - La fuente de datos `Banking Data 1_[percentBegin=0, percentEnd=70, strategy=sequential]`
   - La fuente de datos `Banking Data 1_[percentBegin=70, percentEnd=100, strategy=sequential]`
   - La fuente de datos `Banking Data 2`
   - El modelo de ML `ML model: Banking Data 1`
   - La evaluación `Evaluation: ML model: Banking Data 1`
2. Elija Acciones y, a continuación, elija Eliminar.
3. En el cuadro de diálogo, seleccione Delete (Eliminar) para eliminar todos los recursos seleccionados.

Acaba de completar correctamente el tutorial. Para seguir utilizando la consola para crear fuentes de datos, modelos y predicciones, consulte Guía para desarrolladores de machine learning de Amazon. Para obtener información sobre cómo utilizar la API, consulte la referencia de la API de Amazon Machine Learning.

Creación y uso de fuentes de datos

Puede utilizar fuentes de datos de Amazon ML para entrenar un modelo de ML, evaluarlo y generar predicciones en lote con él. Los objetos de las fuentes de datos contienen metadatos sobre los datos de entrada. Cuando crea una fuente de datos, Amazon ML lee los datos de entrada, calcula las estadísticas descriptivas sobre sus atributos y almacena las estadísticas, un esquema y otra información, como parte del objeto de origen de datos. Después de crear un origen de datos, puede utilizar las estadísticas de datos de Amazon ML para explorar las propiedades de los datos de entrada, y utilizar ese origen de datos para entrenar un modelo de ML.

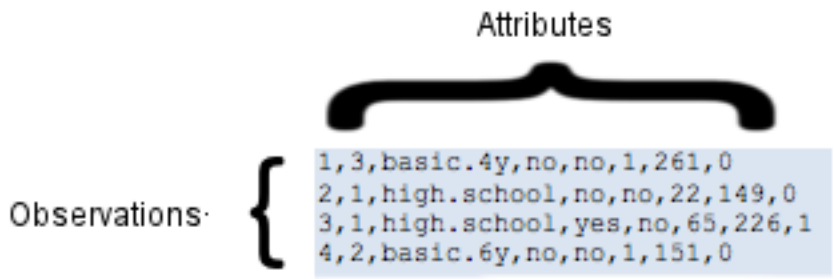
> **Note**
>
> En esta sección se presupone que se conocen los conceptos de Amazon Machine Learning.

Temas

- Compresión del formato de datos de Amazon ML
- Creación de un esquema de datos para Amazon ML
- División de datos
- Análisis de datos
- Uso de Amazon S3 con Amazon ML
- Creación de una fuente de datos de Amazon ML a partir de datos de Amazon Redshift
- Uso de datos de una base de datos de Amazon RDS para crear una fuente de datos de Amazon ML

Compresión del formato de datos de Amazon ML

Los datos de entrada son aquellos que se utilizan para crear una fuente de datos. Debe guardar los datos de entrada en el formato de valores separados por comas (.csv). Cada fila en el archivo .csv es un solo registro de datos u observación. Cada columna en el archivo .csv contiene un atributo de la observación. Por ejemplo, el siguiente gráfico muestra el contenido de un archivo .csv que tiene cuatro observaciones, cada una en su propia fila. Cada observación contiene ocho atributos, separados por comas. Los atributos representan la siguiente información sobre cada persona representada por una observación: CustomeriD, JoBid, educación, vivienda, préstamo, campaña, duración, campaña. willRespondTo



Atributos

Amazon ML requiere nombres para cada atributo. Puede especificar los nombres de los atributos del siguiente modo:

- Incluyendo los nombres de los atributos en la primera línea (también conocida como línea de encabezado) del archivo .csv que utilice como datos de entrada
- Incluyendo los nombres de los atributos en un archivo de esquema por separado que se encuentra en el mismo bucket de S3 que los datos de entrada

Para obtener más información acerca de la utilización de archivos de esquema, consulte Creación de un esquema de datos.

A continuación, se muestra un ejemplo de archivo .csv que incluye los nombres de los atributos en la línea de encabezado.

```
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
```

Requisitos de formato de archivos de entrada

El archivo .csv que contiene los datos de entrada debe cumplir los siguientes requisitos:

- Debe estar en texto sin formato y con un conjunto de caracteres como ASCII, Unicode o EBCDIC.
- Consta de observaciones, una observación por línea.
- Para cada observación, los valores de atributos deben estar separados por comas.
- Si un valor de atributo contiene una coma (el delimitador), todo el valor del atributo debe estar entre comillas dobles.
- Cada observación debe terminar con un end-of-line carácter, que es un carácter especial o una secuencia de caracteres que indica el final de una línea.
- Los valores de los atributos no pueden incluir end-of-line caracteres, incluso si el valor del atributo está entre comillas dobles.
- Cada observación debe tener el mismo número de atributos y la misma secuencia de los atributos.
- Cada observación no puede ser superior a 100 KB. Amazon ML rechaza cualquier observación con un tamaño superior a 100 KB durante el procesamiento. Si Amazon ML rechaza más de 10 000 observaciones, se rechaza todo el archivo .csv.

Uso de varios archivos como datos de entrada para Amazon ML

Puede proporcionar sus entradas a Amazon ML como un único archivo, o bien como una colección de archivos. Las colecciones deberán cumplir estas condiciones:

- Todos los archivos deben tener el mismo esquema de datos.
- Todos los archivos deben encontrarse en el mismo prefijo de Amazon Simple Storage Service (Amazon S3) y la ruta que proporcione para la colección debe acabar con el carácter de barra inclinada ('/').

Por ejemplo, si los archivos de datos se nombran input1.csv, input2.csv e input3.csv y el nombre del bucket de S3 es s3:// examplebucket, las rutas de sus archivos deberían ser del siguiente modo:

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

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

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

Proporcionaría la siguiente ubicación de S3 como entrada para Amazon ML:

's3:///' examplebucket/path/to/data

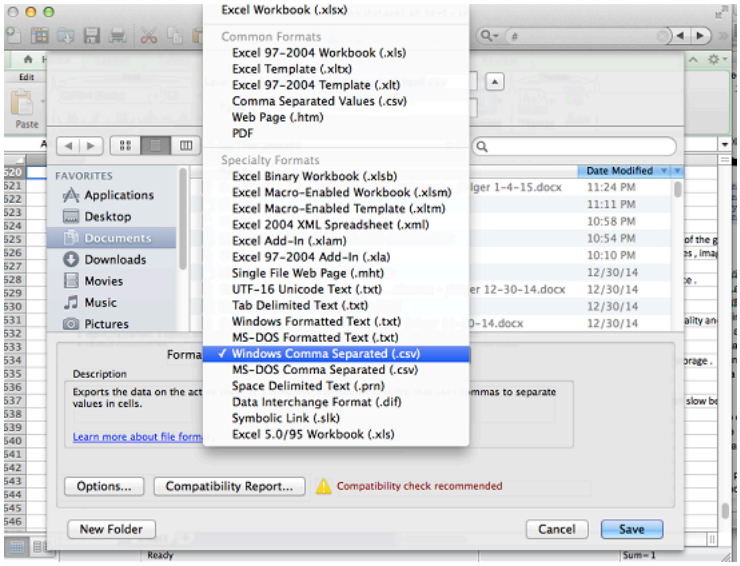
End-of-Line Caracteres en formato CSV

Al crear el archivo.csv, cada observación terminará con un end-of-line carácter especial. Este carácter no es visible, pero se incluirá automáticamente al final de cada comentario al pulsar la tecla Intro o Retorno. El carácter especial que lo representa end-of-line varía en función del sistema operativo. Los sistemas Unix, como Linux u OS X, utilizan un carácter de salto de línea que se indica mediante "\n" (código ASCII 10 en decimal o 0x0a en hexadecimal). Microsoft Windows utiliza dos caracteres llamados retorno de carro y avance de línea que se indican con "\n" (códigos ASCII 13 y 10 en decimal o 0x0d y 0x0a en hexadecimal).

Si desea utilizar OS X y Microsoft Excel para crear su archivo.csv, realice el siguiente procedimiento. Asegúrese de elegir el formato correcto.

Para guardar un archivo .csv si utiliza OS X y Excel

1. Al guardar el archivo .csv, elija Formato y, a continuación, elija Valores separados por comas de Windows (.csv).
2. Seleccione Guardar.



> **Important**
>
> No guarde el archivo .csv usando los formatos Valores separados por comas (.csv) o Valores separados por comas de MS-DOS (.csv) porque Amazon ML no puede leeros.

Creación de un esquema de datos para Amazon ML

Un esquema se compone de todos los atributos de los datos de entrada y sus tipos de datos correspondientes. Permite que Amazon ML entienda los datos de la fuente de datos. Amazon ML utiliza la información del esquema para leer e interpretar los datos de entrada, calcular estadísticas, aplicar las transformaciones de atributo correctas y ajustar los algoritmos de aprendizaje. Si no proporciona ningún esquema, Amazon ML infiere uno a partir de los datos.

Esquema de ejemplo

Para que Amazon ML lea los datos de entrada correctamente y genere predicciones precisas, cada atributo debe estar asignado al tipo de datos correcto. Veamos un ejemplo para ver cómo se asignan los tipos de datos a atributos y cómo los atributos y los tipos de datos se incluyen en un esquema. Denominaremos a nuestro ejemplo "Customer Campaign" porque queremos predecir qué clientes responderán a nuestra campaña de correo electrónico. Nuestro archivo de entrada es un archivo.csv con nueve columnas:

```
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
```

Este es el esquema de estos datos:

```
{
    "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"
        }

    ]
}
```

En el archivo de esquema de este ejemplo, el valor del `rowId` es `customerId`:

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

El atributo `willRespondToCampaign` se define como el atributo de destino:

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

El atributo `customerId` y el tipo de datos `CATEGORICAL` están asociados a la primera columna, el atributo `jobId` y el tipo de datos `CATEGORICAL` están asociados a la segunda columna, el atributo `jobDescription` y el tipo de datos `TEXT` están asociados a la tercera columna, el atributo `education` y el tipo de datos `CATEGORICAL` están asociados a la cuarta columna y así sucesivamente. La novena columna está asociada al atributo `willRespondToCampaign` con un tipo de datos `BINARY`, y este atributo también está definido como atributo de destino.

Uso del targetAttributeName campo

El valor `targetAttributeName` es el nombre del atributo que desea predecir. Debe asignar un valor `targetAttributeName` al crear o evaluar un modelo.

Durante la formación o evaluación de un modelo de ML, el `targetAttributeName` identifica el nombre del atributo de los datos de entrada que contiene las respuestas "correctas" para el atributo de destino. Amazon ML utiliza el destino, el cual incluye las respuestas correctas, para detectar patrones y generar un modelo de ML.

Cuando evalúa el modelo, Amazon ML utiliza el destino para comprobar la exactitud de las predicciones. Una vez que haya creado y evaluado el modelo de ML, puede utilizar los datos con un `targetAttributeName` que no esté asignado para generar predicciones con el modelo de ML.

Puede definir el atributo de destino en la consola Amazon ML al crear una fuente de datos o en un archivo de esquema. Si crea su propio archivo de esquemas, utilice la siguiente sintaxis para definir el atributo de destino:

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

En este ejemplo, `exampleAttributeTarget` es el nombre del atributo del archivo de origen que es el atributo de destino.

Funcionamiento del campo rowID

El `row ID` es un marcador opcional asociado a un atributo de los datos de entrada. Si se especifica, el atributo marcado como `row ID` se incluye en la predicción de salida. Este atributo facilita determinar qué predicción se corresponde con cada observación. Un ejemplo de un buen `row ID` es un ID de cliente o un atributo exclusivo similar.

> **Note**
>
> El ID de fila solo es para fines de referencia. Amazon ML no la utiliza al entrenar un modelo de ML. Seleccionar un atributo como ID de fila lo excluye de que se utilice para el entrenamiento de un modelo de ML.

Puede definir el `row ID` en la consola Amazon ML al crear una fuente de datos o en un archivo de esquema. Si crea su propio archivo de esquema, utilice la siguiente sintaxis para definir el `row ID`:

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

En el anterior ejemplo, `exampleRow` es el nombre del atributo del archivo de origen que se define como el ID de fila.

Al generar predicciones por lotes, es posible que aparezca el siguiente resultado:

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

En este ejemplo, `RowID` representa al atributo `customerId`. Por ejemplo, un customerId de `55` significa que responde a nuestra campaña de correo electrónico con una confianza baja (0,46317), mientras un `customerId` de `102` significa que responde a nuestras campañas de correo electrónico con una confianza alta (0,89625).

Uso del AttributeType campo

En Amazon ML existen cuatro tipos de datos de atributos:

Binario

Seleccione `BINARY` para un atributo que solo tiene dos estados posibles, como por ejemplo `yes` o no.

Por ejemplo, el atributo `isNew`, para controlar si una persona es un nuevo cliente, tendría que tener un valor `true` para indicar que la persona es un nuevo cliente, y un valor `false` para indicar que no es un nuevo cliente.

Los valores negativos válidos son `0`, `n`, `no`, `f` y `false`.

Los valores positivos válidos son `1`, `y`, `yes`, `t` y `true`.

Amazon ML ignora el caso de entradas binarias y elimina el espacio blanco de alrededor. Por ejemplo, `" FaLSe "` es un valor binario válido. Puede combinar los valores binarios que utiliza en la misma fuente de datos, como `true`, noy `1`. Amazon ML solo genera `0` y `1` para atributos binarios.

Categórico

Seleccione `CATEGORICAL` para un atributo que admite un número limitado de valores de cadena únicos. Por ejemplo, un ID de usuario, el mes y un código postal son valores categóricos. Los atributos categóricos se tratan como una cadena única y no se tokenizan más.

Numérico

Seleccione `NUMERIC` para un atributo que admite una cantidad como un valor.

Por ejemplo, la temperatura, el peso y la el número de clics son valores numéricos.

No todos los atributos que contienen números son numéricos. Los atributos categóricos, como los días del mes y IDs, suelen representarse como números. Para que se consideren numéricos, un número debe ser comparable a otro número. Por ejemplo, el ID de cliente `664727` no le indica nada sobre el ID de cliente `124552`, pero un peso de `10` le indica que ese atributo es más pesado que un atributo con un peso de `5`. Los días del mes no son numéricos, porque el primero de un mes podría ocurrir antes o después del segundo de otro mes.

> **Note**
>
> Al utilizar Amazon ML para crear su esquema, se asignará el tipo de datos `Numeric` para todos los atributos que utilizan los números. Si Amazon ML crea su esquema, compruebe la existencia de asignaciones incorrectas y definir los atributos `CATEGORICAL`.

Text

Elija `TEXT` para un atributo que es una cadena de palabras. Al leer en los atributos de texto, Amazon ML convierte en tokens, delimitado por los espacios en blanco.

Por ejemplo, `email subject` vuelve a estar en buen estado `email` y `subject`y `email-subject here` se convierte en `email-subject` y `here`.

Si el tipo de datos de una variable en el esquema de formación no coincide con el tipo de datos de esa variable en el esquema de evaluación, Amazon ML cambia el tipo de datos de evaluación para que coincida con el tipo de datos de formación. Por ejemplo, si el esquema de datos de formación asigna un tipo de datos de `TEXT` a la variable `age`, pero el esquema de evaluación asigna un tipo de datos de `NUMERIC` a `age`, Amazon ML considera que la envejecen en la evaluación de los datos como variables `TEXT` en vez de `NUMERIC`.

Para obtener información sobre las estadísticas asociadas a cada tipo de datos, consulte Estadísticas descriptivas.

Proporcionar un esquema a Amazon ML

Cada fuente de datos necesita un esquema. Puede elegir entre dos formas para proporcionar un esquema a Amazon ML:

- Permitir que Amazon ML infiera los tipos de datos de cada atributo en el archivo de datos de entrada y que cree un esquema automáticamente.
- Proporcione un archivo de esquema cuando cargue sus datos de Amazon Simple Storage Service (Amazon S3).

Permitir que Amazon ML cree un esquema

Al utilizar la consola de Amazon ML para crear un origen de datos, Amazon ML utiliza reglas sencillas basadas en los valores de las variables para crear un esquema. Le recomendamos que revise el esquema creado por Amazon ML y que corrija los tipos de datos que no sean precisos.

Proporcionar un esquema

Después de crear su archivo de esquema, debe volver a Amazon ML. Dispone de dos opciones para hacerlo:

1. Proporcione el esquema utilizando la consola de Amazon ML.

   Utilice la consola para crear la fuente de datos e incluya el archivo de esquema añadiendo la extensión .schema al nombre del archivo de los datos de entrada. Por ejemplo, si el URI de Amazon Simple Storage Service (Amazon S3) de los datos de entrada es s3:my-bucket-name///

data/input.csv, the URI to your schema will be s3://my-bucket-name/data/input.csv.schema. Amazon ML localiza automáticamente el archivo de esquema que proporcione en lugar de intentar inferir el esquema de los datos.

Para utilizar un directorio de archivos como datos de entrada a Amazon ML, añada la extensión .schema, a la ruta del directorio. Por ejemplo, si sus archivos de datos residen en la ubicación s3:///.schema. examplebucket/path/to/data/, the URI to your schema will be s3:// examplebucket/path/to/data

2. Proporcione el esquema utilizando la API de Amazon ML.

   Si pretende llamar a la API de Amazon ML para crear un origen de datos, puede cargar el archivo de esquema a Amazon S3 y, a continuación, proporcionar la URI a dicho archivo del atributo `DataSchemaLocationS3` de la API `CreateDataSourceFromS3`. Para obtener más información, consulte S3. CreateDataSourceFrom

   Puede proporcionar el esquema directamente en la carga de `CreateDataSource*` `APIs` en lugar de guardarlo primero en Amazon S3. Para ello, coloque la cadena de esquema completa en el `DataSchema` atributo `CreateDataSourceFromS3CreateDataSourceFromRDS`, o `CreateDataSourceFromRedshift` APIs. Para obtener más información, consulte Referencia de la API de Amazon Machine Learning.

División de datos

El objetivo fundamental de un modelo de ML es realizar predicciones precisas sobre futuras instancias de datos más allá de aquellos que se utilizan para entrenar modelos. Antes de utilizar un modelo de ML para realizar predicciones, debemos evaluar el rendimiento predictivo del modelo. Para valorar la calidad de las predicciones de un modelo de ML con datos que no ha visto, podemos reservar, o dividir, una parte de los datos de los que ya sabemos la respuesta como un proxy para futuros datos y evaluar la capacidad del modelo de ML para predecir las respuestas correctas para dichos datos. El origen de datos se divide en una parte para el entrenamiento del origen de datos y otra parte para la evaluación del origen de datos.

Amazon ML ofrece tres opciones para dividir los datos:

- Predivisión de datos: los datos se dividen en dos ubicaciones de entrada de datos antes de cargarlos a Amazon Simple Storage Service (Amazon S3) y crear dos orígenes de datos con ellos.
- División secuencial de Amazon ML: podemos pedirle a Amazon ML que divida los datos de forma secuencial durante la creación de fuentes de datos para la formación y la evaluación.

- División aleatoria de Amazon ML: podemos pedirle a Amazon ML que divida los datos con un método aleatorio durante la creación de orígenes de datos para el entrenamiento y la evaluación.

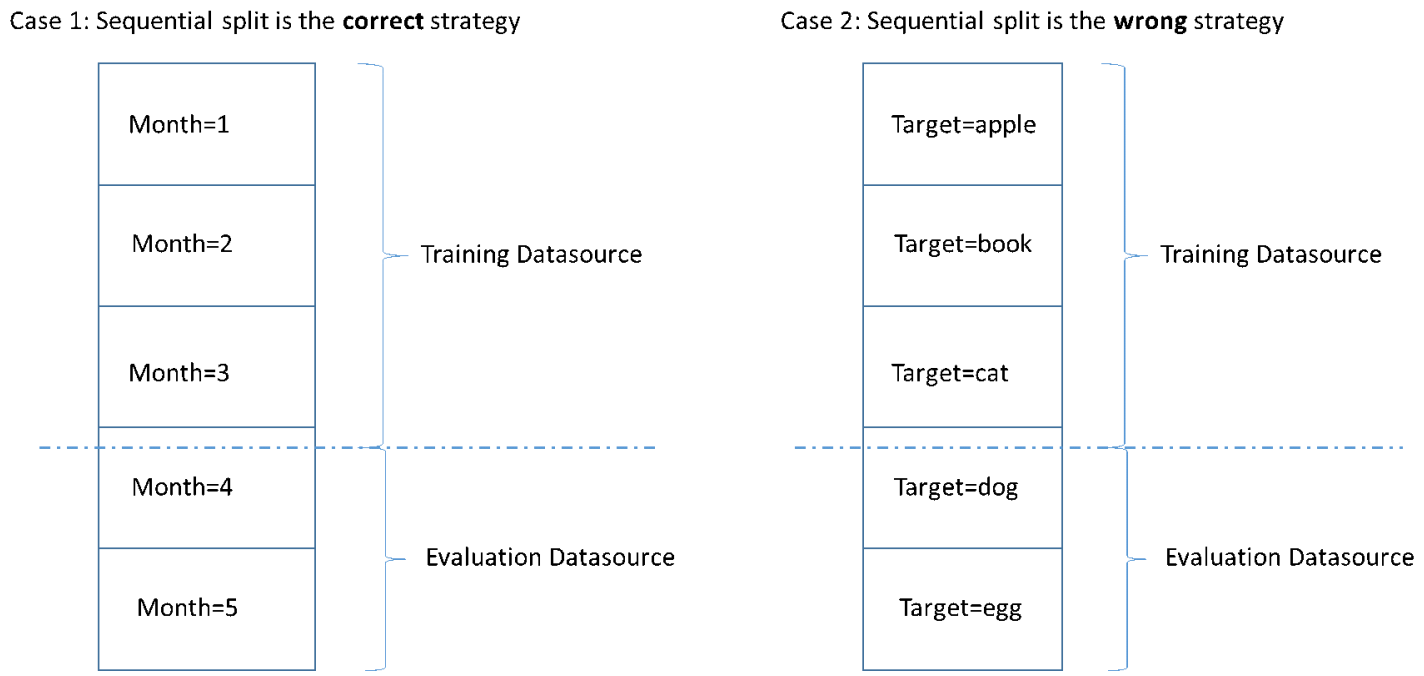
Pre-división de datos

Si desea tener un control explícito de los datos durante el entrenamiento y la evaluación de las fuentes de datos, divida los datos en diferentes ubicaciones de datos y cree otra fuente de datos para las ubicaciones de entrada y de evaluación.

División secuencial de datos

Una manera sencilla de dividir los datos de entrada para el entrenamiento y la evaluación consiste en seleccionar subconjuntos de datos que no se solapen y, al mismo tiempo, preservar el orden de los registros de datos. Este enfoque resulta útil si desea evaluar sus modelos de ML en relación con los datos de una determinada fecha o dentro de un determinado intervalo de tiempo. Por ejemplo, digamos que tiene los datos de compromiso de los clientes de los últimos cinco meses y que desea utilizar estos datos históricos para predecir el compromiso de los clientes del siguiente mes. Con el principio del rango para la formación y los datos del final del rango para la evaluación podría producirse una estimación más precisa de la calidad del modelo de datos que no utilizando los datos de los registros de todo el rango de datos.

La siguiente figura muestra ejemplos de cuándo debe utilizar una estrategia de división secuencial y cuándo debe utilizar una estrategia aleatoria.

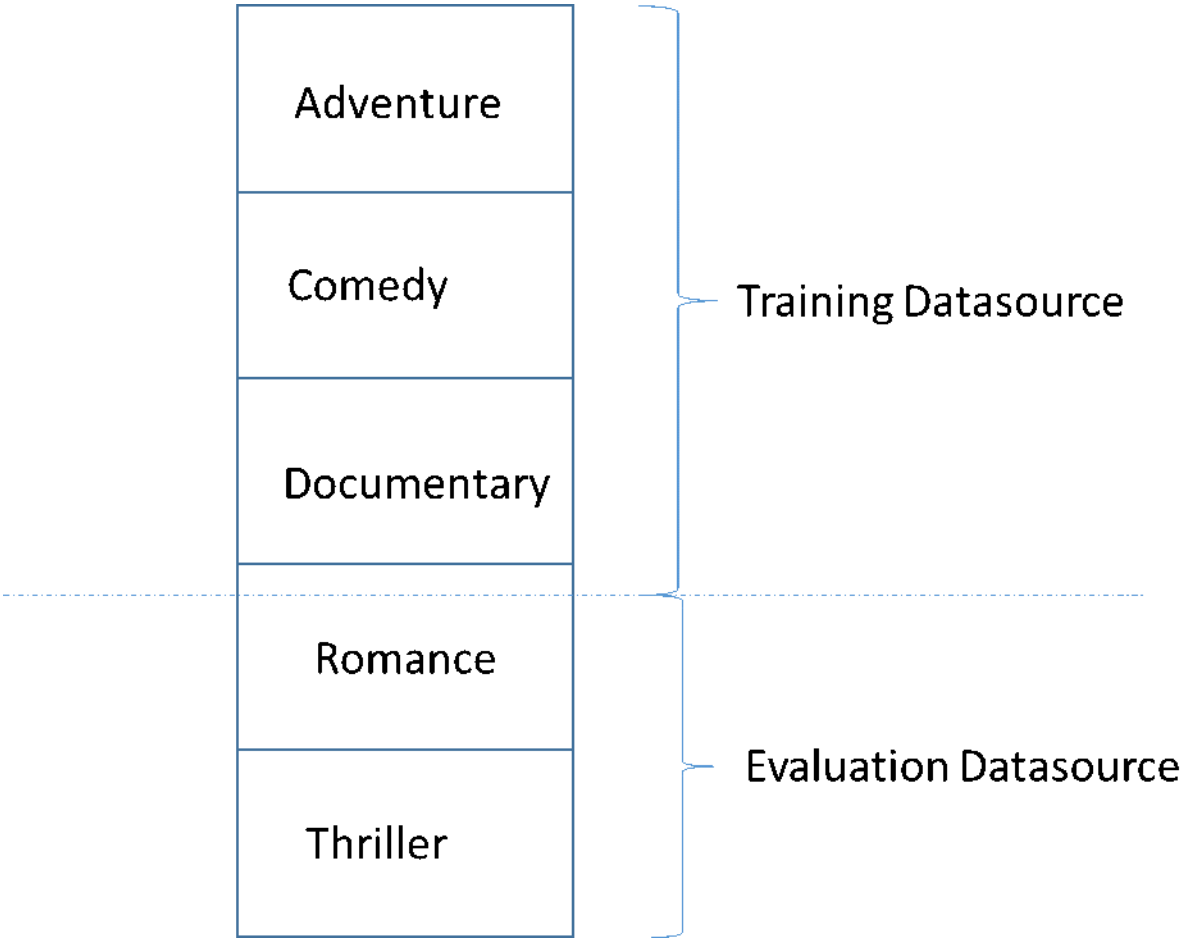


Cuando crea una fuente de datos, puede optar por dividir su fuente de datos de forma secuencial, y Amazon ML utilizará el primer 70 % de los datos para la formación y el 30 restante de los datos para la evaluación. Este es el enfoque predeterminado cuando utilizar la consola de Amazon ML para dividir los datos.

División aleatoria de datos

Dividir los datos de entrada de forma aleatoria en fuentes de datos para la formación y la evaluación garantiza que la distribución de los datos sea similar en la formación y en la evaluación de fuentes de datos. Elija esta opción cuando no necesite conservar el orden de los datos de entrada.

Amazon ML usa un método pseudo-aleatorio para la generación de números para dividir los datos. El origen se basa en parte, en el valor de una cadena de entrada y, en parte, en el contenido de los propios datos. Por defecto, la consola Amazon ML usa la ubicación S3 de los datos de entrada de la cadena. Los usuarios de la API pueden proporcionar una cadena personalizada. Esto quiere decir que proporcionando el mismo bucket de S3 y datos, Amazon ML divide los datos de la misma forma cada vez. Para cambiar la manera en la que Amazon ML divide los datos, puede utilizar la API `CreateDatasourceFromS3`, `CreateDatasourceFromRedshift` o `CreateDatasourceFromRDS` y proporcionar un valor para la cadena de origen. Al utilizarlos APIs para crear fuentes de datos independientes para la formación y la evaluación, es importante utilizar el mismo valor de cadena inicial para ambas fuentes de datos y el indicador de complemento para una fuente de datos, a fin de garantizar que no se superpongan los datos de formación y de evaluación.



Un problema común en el desarrollo de un modelo de ML de alta calidad es la evaluación del modelo de ML en relación con los datos que no se parecen a los datos utilizados para la formación. Por ejemplo, digamos que está utilizando ML para predecir el género de películas y que sus datos de formación contienen películas de aventuras, comedias y documentales. Sin embargo, los datos de evaluación solo contienen datos de películas románticas y de suspense. En este caso, el modelo de ML no había almacenado ninguna información sobre las películas románticas y de suspense, y en la evaluación no se ha evaluado la eficacia del modelo con el aprendizaje de patrones para películas de aventuras, comedias y documentales. Como resultado, la información sobre el género no es útil y la calidad de las predicciones del modelo de ML para todos los géneros no es fiable. El modelo y la evaluación son demasiado diferentes (tienen estadísticas descriptivas extremadamente diferentes) para ser útiles. Esto puede ocurrir cuando los datos de entrada se ordenan por una de las columnas del conjunto de datos y, a continuación, se dividen de forma secuencial.

Si las fuentes de datos para el entrenamiento y la evaluación tienen diferentes distribuciones de datos, verá una alerta de evaluación en la evaluación de su modelo. Para obtener más información sobre las alertas de evaluación, consulte Alertas de evaluación.

No es necesario utilizar la división aleatoria de Amazon ML si ya se han aleatorizado los datos de entrada, por ejemplo, mediante la transferencia aleatoria de los datos de entrada a Amazon S3, o mediante la utilización de la `random()` función de una consulta SQL de Amazon Redshift o la función `rand()` de una consulta MySQL de SQL al crear orígenes de datos. En estos casos, puede confiar en la opción de la división secuencial para crear fuentes de datos para la formación y la evaluación con distribuciones similares.

Análisis de datos

Amazon ML procesa estadísticas descriptivas de los datos de entrada que puede utilizar para comprender los datos.

Estadísticas descriptivas

Amazon ML procesa las siguientes estadísticas descriptivas de diferentes tipos de atributo:

Numérico:

- Histogramas de la distribución
- Número de valores no válidos
- Valores mínimos, medios y máximos

Binario y categórico:

- Cantidad (de valores distintos por categoría)
- Histogramas de la distribución de valores
- Valores más frecuentes
- Cantidad de valores exclusivos
- Porcentaje del valor "true" (solo binario)
- Palabras más destacadas
- Palabras más frecuentes

Texto:

- Nombre del atributo
- Correlación con el destino (si el destino está establecido)

- Total de palabras
- Palabras exclusivas
- Alcance del número de palabras en una línea
- Alcance de la longitud de palabra
- Palabras más destacadas

Estadísticas de los datos de acceso de la consola Amazon ML

En la consola Amazon ML, puede elegir el nombre o ID de cualquier origen de datos para ver la página Estadísticas de datos. Esta página proporciona métricas y visualizaciones que permiten obtener más información acerca de los datos de entrada asociados a la fuente de datos, incluyendo la siguiente información:

- Resumen de datos
- Distribuciones de destino
- Valores faltantes
- Valores no válidos
- Estadísticas de resumen de las variables por tipo de datos
- Distribuciones de variables por tipo de datos

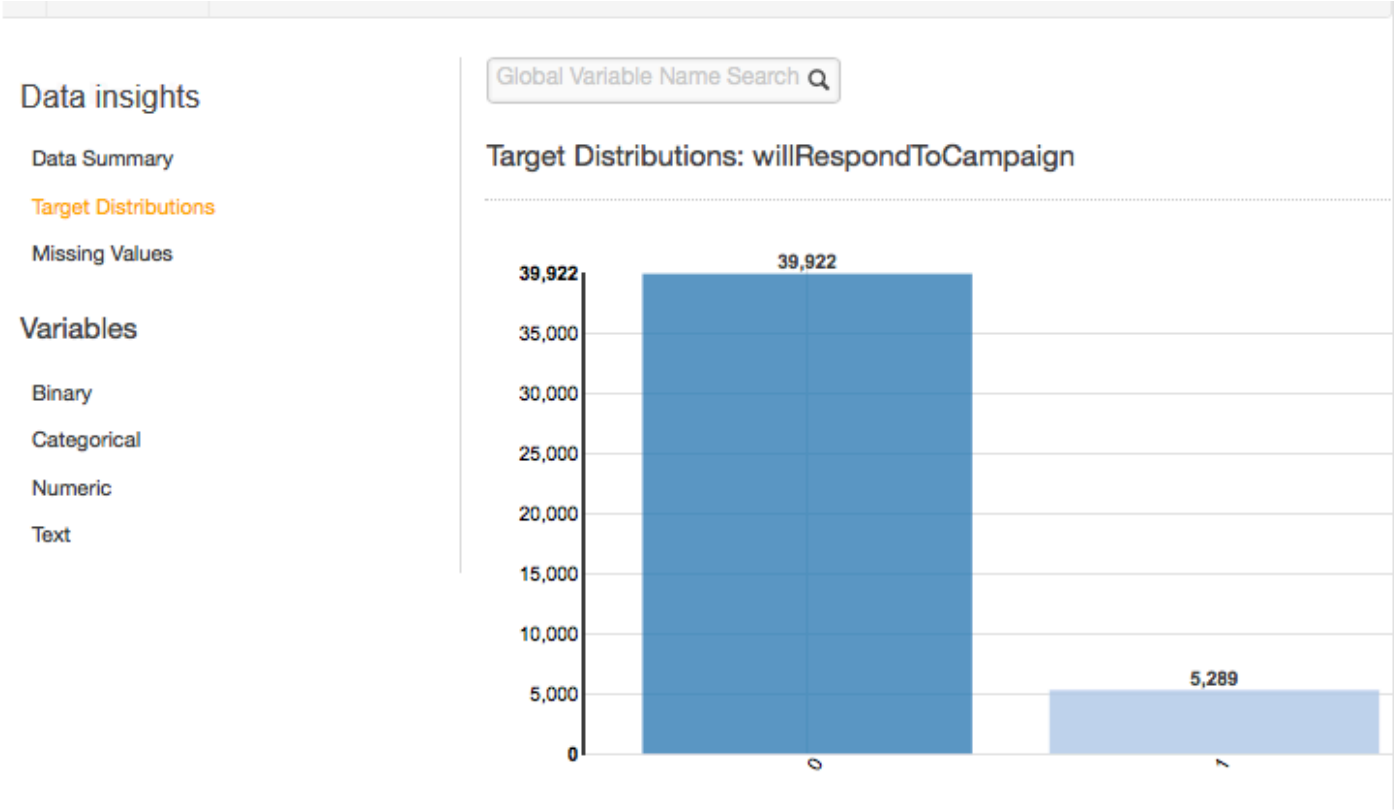
Las siguientes secciones describen las métricas y visualizaciones de forma más detallada.

Resumen de datos

El informe del resumen de los datos de una fuente de datos muestra información resumida, incluido el ID y el nombre de la fuente de datos, dónde se ha completado, el estado actual, el atributo de destino, la información de los datos de entrada (la ubicación del depósito de S3, el formato de datos, el número de registros procesados y el número de registros incorrectos encontrados durante el procesamiento), así como el número de variables por tipo de datos.

Distribuciones de destino

El informe de las distribuciones de destino muestra la distribución del atributo de destino de la fuente de datos. En el siguiente ejemplo, hay 39.922 observaciones en las que el atributo de objetivo de la willRespondTo campaña es igual a 0. Este es el número de clientes que no han respondido a la campaña de correo electrónico. Hay 5.289 observaciones en las que willRespondTo Campaign es igual a 1. Este es el número de clientes que han respondido a la campaña de correo electrónico.



Valores que faltan

El informe de los valores que faltan muestra una lista de los atributos de los datos de entrada para los que faltan valores. Solo pueden faltar valores a los atributos con tipos numéricos de datos. Teniendo en cuenta que los valores que faltan pueden influir a la calidad del entrenamiento de un modelo de ML, recomendamos que se proporcionen a ser posible.

Durante el entrenamiento del modelo de ML, si falta el atributo de destino, Amazon ML rechaza el registro correspondiente. Si el atributo de destino está en el registro, pero falta un valor de otro atributo numérico, Amazon ML ignora el valor que falta. En este caso, Amazon ML crea un atributo sustituto y lo establece a 1 para indicar que este atributo falta. Esto permite que Amazon ML aprenda patrones a partir de la aparición de valores que faltan.

Valores no válidos

Los valores no válidos solo pueden aparecer con los tipos de datos numéricos y binarios. Puede encontrar valores no válidos si consulta las estadísticas de resumen de las variables de los informes de tipos de datos. En los siguientes ejemplos, aparecen un valor no válido para el atributo numérico de duración y dos valores no válidos para el tipo de datos binario (uno en el atributo del alojamiento y otro en el atributo de préstamo).

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%)	

Correlación variable-destino

Después de crear una fuente de datos, Amazon ML puede evaluar la fuente de datos e identificar la correlación, o el impacto, entre las variables y el destino. Por ejemplo, el precio de un producto podría tener un impacto significativo sobre el hecho de si es un éxito de ventas o no, mientras que las dimensiones del producto tendrían poco poder predictivo.

Se suele recomendar incluir tantas variables como sea posible en los datos de aprendizaje. Sin embargo, el ruido que se genera al incluir muchas variables con poco poder predictivo podría afectar negativamente a la calidad y precisión del modelo de ML.

Puede que pueda mejorar el rendimiento predictivo del modelo eliminando variables que tengan poco impacto cuando formes el modelo. Puede definir las variables que estarán disponibles para el proceso de machine learning en una receta, que consiste en un mecanismo de transformación de Amazon ML. Para obtener más información acerca de las recetas, consulte la Transformación de datos para el aprendizaje automático.

Estadísticas de resumen de los atributos por tipo de datos

En el informe de las estadísticas de datos, puede ver las estadísticas de resumen de los atributos por los siguientes tipos de datos:

- Binario
- Categórico
- Numérico
- Texto

Las estadísticas de resumen del tipo de datos binario muestran todos los atributos binarios. La columna Correlations to target (Correlaciones con el destino) muestra la información que comparten la columna del destino y la columna del atributo. La columna Percent true (Porcentaje de "true") muestra el porcentaje de las observaciones que tienen el valor 1. La columna Invalid values (Valores no válidos) muestra el número de valores no válidos, así como el porcentaje de valores no válidos para cada atributo. La columna Preview (Vista previa) proporciona un enlace a una distribución gráfica para cada atributo.

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%)	

Las estadísticas de resumen del tipo de datos categórico muestran todos los atributos categóricos con la cantidad de valores únicos, el valor más frecuente y el valor menos frecuente. La columna Preview (Vista previa) proporciona un enlace a una distribución gráfica para cada atributo.

Categorical Variables

Variables	Correlations to Target	Unique Values	Most Frequent	Least Frequent	Preview
campaign	0.00433	49	1	39	
customerId	NA	45211	45211	1	
education	0.00355	5	secondary		
housing	0.01846	4	1		
jobId	0.00671	13	blue-collar		
willRespondToCampaign	NA	3	0		

Las estadísticas de resumen del tipo de datos numérico muestran todos los atributos numéricos con el número de valores que faltan, de valores no válidos, del alcance de los valores y la media. La columna Preview (Vista previa) proporciona un enlace a una distribución gráfica para cada atributo.

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	

Las estadísticas de resumen del tipo de datos textual muestran todos los atributos textuales, el número total de palabras de dicho atributo, el número de palabras de dicho atributo exclusivo, el alcance de las palabras de un atributo, el alcance de la longitud de palabra y las palabras más destacadas. La columna Preview (Vista previa) proporciona un enlace a una distribución gráfica para cada atributo.

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.*

El siguiente ejemplo muestra las estadísticas del tipo de datos textual para una variable textual denominada "review", con cuatro registros.

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

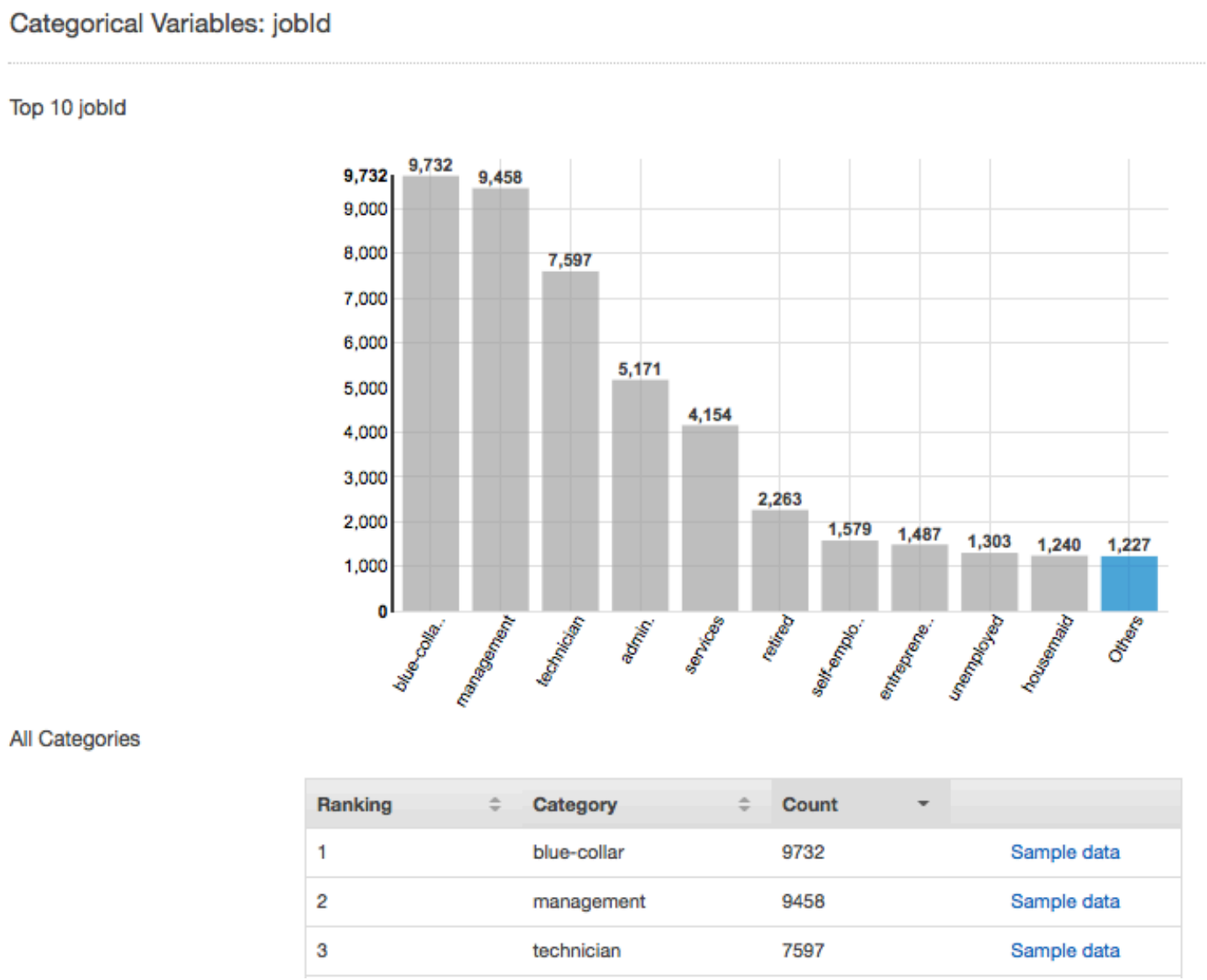
Las columnas de este ejemplo muestran la siguiente información.

- La columna Attributes (Atributos) muestra el nombre de la variable. En este ejemplo, en esta columna aparecería "review".
- La columna Correlations to target (Correlaciones con el destino) solo existe si se especifica un destino. La correlación calcula la cantidad de información que proporciona este atributo sobre el destino. Cuanto mayor sea la correlación, más información proporciona este atributo sobre el destino. La correlación se calcula en términos de información mutua entre una representación del atributo del texto y el destino.
- La columna Total words (Total de palabras) muestra el número de palabras generadas a partir de la tokenización de cada registro, delimitando las palabras con espacios en blanco. En este ejemplo, en esta columna aparecería "12".
- La columna Unique word (Palabra única) muestra el número de palabras únicas de un atributo. En este ejemplo, en esta columna aparecería "10".
- La columna Words in attribute (range) (Palabras en el atributo [rango]) muestra el número de palabras en una sola fila del atributo. En este ejemplo, en esta columna aparecería "0-6".
- La columna Word length (range) (Longitud de palabra [rango]) muestra cuántos caracteres hay en las palabras. En este ejemplo, en esta columna aparecería "2-11".
- La columna Most prominent words (Palabras más destacadas) muestra una lista ordenada por importancia de las palabras que aparecen en el atributo. Si existe un atributo de destino, las palabras se clasifican según su correlación con el destino, lo que significa que las palabras con mayor correlación se enumerarán en primer lugar. Si no hay destino en los datos, las palabras se clasifican según su entropía.

Distribución de atributos binarios y categóricos

Al hacer clic en el enlace Preview (Vista previa) asociado a un atributo categórico o binario, puede ver la distribución de dicho atributo, así como los datos de muestra del archivo inicial de cada valor categórico del atributo.

Por ejemplo, la siguiente instantánea muestra la distribución del atributo categórico jobId. La distribución muestra los 10 valores categóricos más importantes, con todos los demás valores agrupados como "other". Clasifica cada uno de los 10 valores categóricos más importantes con el número de observaciones del archivo de entrada que contiene dicho valor, así como un enlace para ver ejemplos de observaciones de los archivos de los datos de entrada.

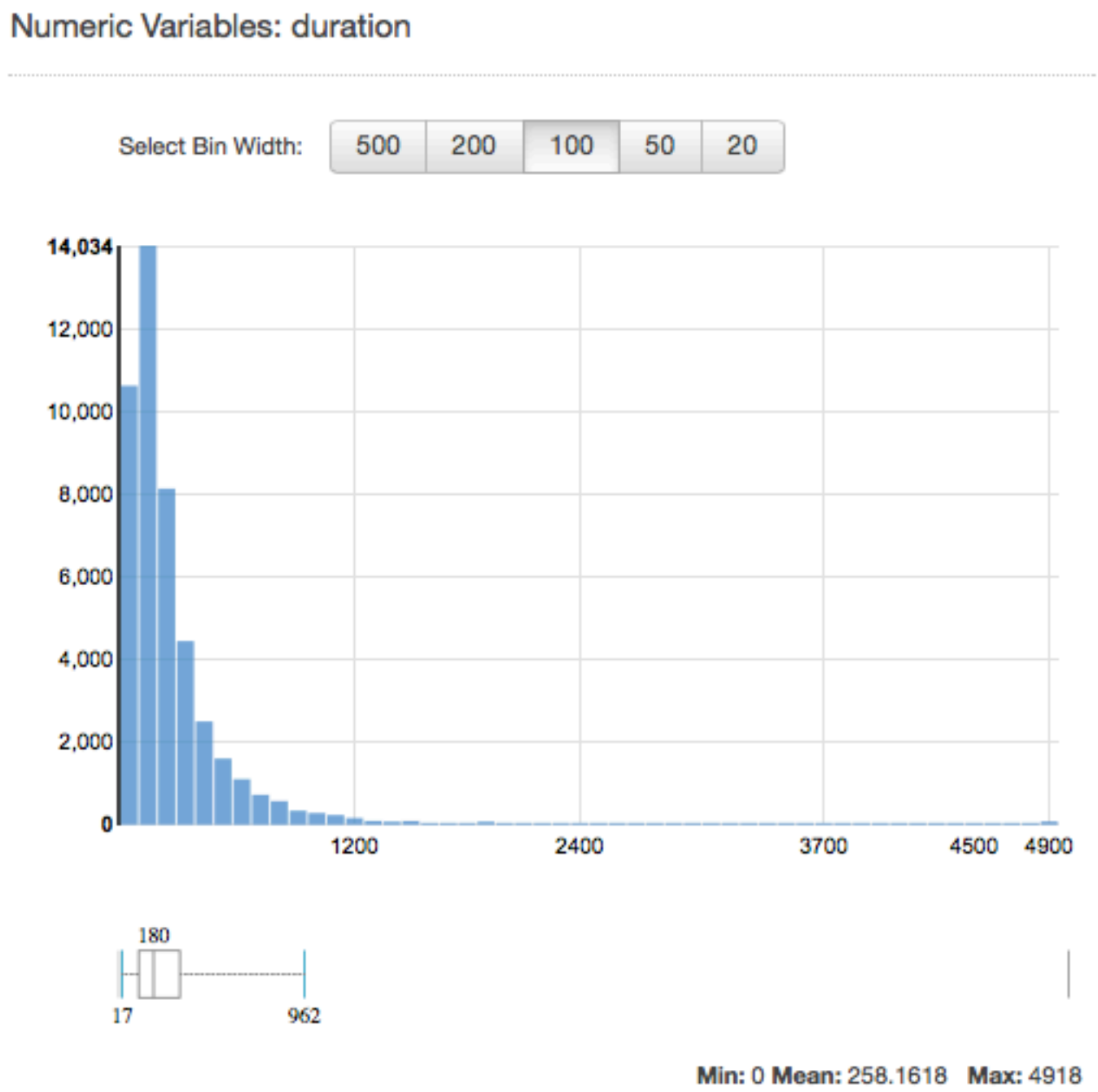


Ranking	Category	Count	
1	blue-collar	9732	Sample data
2	management	9458	Sample data
3	technician	7597	Sample data

Distribución de atributos numéricos

Para consultar la distribución de un atributo numérico, haga clic en el enlace Preview (Vista previa) del atributo. Cuando vea la distribución de un atributo numérico, puede elegir el tamaño de la papelera entre 500, 200, 100, 50 o 20. Cuanto mayor sea el tamaño de la papelera, menor será el número de gráficos de barras que se mostrará. Además, la resolución de la distribución será amplia para los tamaños de la papelera más grandes. En cambio, ajustando el tamaño del depósito a 20 aumenta la resolución de la distribución que se muestra.

También se muestran los valores mínimos, medios y máximos, tal como se muestra en la siguiente captura de pantalla.



Distribución de atributos textuales

Para consultar la distribución de un atributo textual, haga clic en el enlace Preview (Vista previa) del atributo. Cuando vea la distribución de un atributo de texto, verá la siguiente información.

Text attributes: Phrase

Ranking	Token	Word prominence	Count	
1	enters	0.01105	7	0.0%
2	trust	0.00884	28	0.0%
3	bad	0.00735	833	0.2%
4	film	0.00669	4747	1.3%
5	movie	0.00611	4242	1.2%
6	unwieldy	0.00605	11	0.0%
7	good	0.00574	1620	0.5%
8	ashamed	0.00551	7	0.0%
9	funny	0.00550	1078	0.3%
10	wankery	0.00498	9	0.0%

« ‹ 1 - 10 of 11091 › »

Clasificación

Los tokens textuales se clasifican según la cantidad de información que transmiten, del más informativo al menos informativo.

Token

Los tokens muestran la palabra del texto de entrada sobre la fila de las estadísticas.

Importancia de las palabras

Si existe un atributo de destino, las palabras se clasificarán según su correlación con el destino, lo que significa que las palabras que tengan mayor correlación se enumerarán en primer lugar. Si no hay destino en los datos, las palabras se clasificarán según su entropía, es decir, según la cantidad de información que pueden comunicar.

Cantidad

La cantidad muestra el número de registros de entrada en los que apareció el token.

Porcentaje

El porcentaje muestra el porcentaje de las líneas de los datos de entrada en los que apareció el token.

Uso de Amazon S3 con Amazon ML

Amazon Simple Storage Service (Amazon S3) es un servicio de almacenamiento para Internet. Puede utilizar Amazon S3 para almacenar y recuperar cualquier cantidad de datos en cualquier momento y desde cualquier parte de la web. Amazon ML utiliza Amazon S3 como repositorio principal de datos para las siguientes tareas:

- Obtener acceso a sus archivos de entrada y crear objetos de fuente de datos para la formación y evaluación de sus modelos de ML.
- Obtener acceso a sus archivos de entrada para generar predicciones por lotes.
- Cuando genere predicciones por lotes mediante la utilización de los modelos de ML, emitir el archivo de predicciones a un bucket de S3 que especifique.
- Copiar los datos que ha almacenado en Amazon Redshift o Amazon Relational Database Service (Amazon RDS) en un archivo .csv y cargarlos en Amazon S3.

Para permitir que Amazon ML realice estas tareas, debe concederle permisos a Amazon ML para que obtenga acceso a sus datos de Amazon S3.

> **Note**
>
> No puede extraer archivos de predicción por lotes en un bucket de S3 que solo acepta archivos cifrados en el servidor. Asegúrese de que la directiva de bucket permite cargar archivos sin cifrar confirmando que la política no incluye un efecto `Deny` para la acción `s3:PutObject` cuando no existe ningún encabezado `s3:x-amz-server-side-encryption` en la solicitud. Para obtener más información sobre las políticas de buckets de cifrado del lado del servidor de S3, consulte Protección de datos mediante cifrado del lado del servidor en la Guía del usuario de Amazon Simple Storage Service.

Carga de datos en Amazon S3

Debe cargar los datos de entrada en Amazon Simple Storage Service (Amazon S3), ya que Amazon ML lee datos de las ubicaciones de Amazon S3. Puede cargar sus datos directamente a Amazon S3 (por ejemplo, desde su equipo) o Amazon ML puede copiar los datos almacenados en Amazon Redshift o Amazon Relational Database Service (RDS) a un archivo .csv y cargarlo en Amazon S3.

Para obtener más información sobre cómo copiar los datos de Amazon Redshift o Amazon RDS, consulte Using Amazon Redshift with Amazon ML o Using Amazon RDS with Amazon ML, respectivamente.

El resto de esta sección describe cómo cargar los datos de entrada directamente desde su equipo a Amazon S3. Antes de comenzar los procedimientos de esta sección, debe disponer de los datos en un archivo .csv. Para obtener información sobre cómo dar el formato correcto al archivo .csv para que Amazon ML pueda utilizarlo, consulte Descripción del formato de datos para Amazon ML.

Cargar los datos desde su equipo a Amazon S3

1. Inicie sesión en la consola de administración de AWS y abra la consola de Amazon S3 en https://console.aws.amazon.com/s3.
2. Cree un bucket o elija uno existente.
   a. Para crear un bucket, elija Create Bucket (Crear bucket). Asigne un nombre a su bucket, elija una región (puede elegir cualquier región disponible) y, a continuación, seleccione Create (Crear). Para obtener más información, consulte Crear un bucket en la Guía de introducción de Amazon Simple Storage.
   b. Para utilizar un bucket existente, busque el bucket eligiéndolo en la lista All Buckets (Todos los buckets). Cuando aparezca el nombre del bucket, selecciónelo y, a continuación, elija Upload (Cargar).
3. En el cuadro de diálogo Upload (Cargar), seleccione Add Files (Añadir archivos).
4. Vaya a la carpeta que contiene el archivo .csv de los datos de entrada y, a continuación, seleccione Open (Abrir).

Permisos

Para conceder permisos para que Amazon ML obtenga acceso a uno de los buckets de S3, debe editar la política del bucket.

Para obtener información sobre la concesión de permiso a Amazon ML para leer datos de su bucket en Amazon S3, consulte Concesión de permisos de Amazon ML para la lectura de datos desde Amazon S3.

Para obtener información sobre la concesión de permisos a Amazon ML para extraer los resultados de la predicción por lotes a su bucket en Amazon S3, consulte Concesión de permisos a Amazon ML para enviar predicciones a Amazon S3.

Para obtener información sobre la administración de permisos de acceso a los recursos de Amazon S3, consulte la Guía para desarrolladores de Amazon S3.

Creación de una fuente de datos de Amazon ML a partir de datos de Amazon Redshift

Si dispone de datos almacenados en Amazon Redshift, puede utilizar el asistente Crear origen de datos de la consola de Amazon Machine Learning (Amazon ML) para crear un objeto del origen de datos. Cuando se crea una fuente de datos a partir de datos de Amazon Redshift, debe especificar el clúster que contiene los datos y la consulta SQL para recuperar los datos. Amazon ML ejecuta la consulta invocando el comando `Unload` del clúster de Amazon Redshift. Amazon ML almacena los resultados en la ubicación de Amazon Simple Storage Service (Amazon S3) que elija y, a continuación, utiliza los datos almacenados en Amazon S3 para crear la fuente de datos. La fuente de datos, el clúster de Amazon Redshift y el bucket de S3 deben estar en la misma región.

> **Note**
>
> Amazon ML no admite la creación de fuentes de datos a partir de clústeres de Amazon Redshift de forma privada. VPCs El clúster debe tener una dirección IP pública.

Temas

- Parámetros necesarios para el asistente Create Datasource
- Creación de una fuente de datos con datos de Amazon Redshift (consola)
- Temas de solución de problemas de Amazon Redshift

Parámetros necesarios para el asistente Create Datasource

Para permitir que Amazon ML se conecte a la base de datos de Amazon Redshift y lea datos en su nombre, debe proporcionar lo siguiente:

- El Amazon Redshift `ClusterIdentifier`
- El nombre de la base de datos de Amazon Redshift
- Las credenciales de la base de datos de Amazon Redshift (nombre de usuario y contraseña)
- La función Amazon ML Amazon Redshift AWS Identity and Access Management (IAM)
- La consulta SQL en Amazon Redshift
- (Opcional) La ubicación del esquema de Amazon ML
- La ubicación de almacenamiento provisional de Amazon S3 (donde Amazon ML almacena los datos antes de crear el origen de datos)

Además, debe asegurarse de que los usuarios o los roles de IAM que crean orígenes de datos de Amazon Redshift (ya sea a través de la consola o a través de la acción `CreateDatasourceFromRedshift`) tengan el permiso `iam:PassRole`.

Amazon Redshift **`ClusterIdentifier`**

Use este parámetro que distingue entre mayúsculas y minúsculas para habilitar que Amazon ML encuentre y se conecte al clúster. Puede obtener el identificador del clúster (nombre) desde la consola de Amazon Redshift. Para obtener más información sobre clústeres, consulte Clústeres de Amazon Redshift.

Nombre de la base de datos de Amazon Redshift

Use este parámetro para indicarle a Amazon ML qué base de datos del clúster de Amazon Redshift contiene los datos que desea utilizar como origen de datos.

Credenciales de la base de datos Amazon Redshift

Utilice estos parámetros para especificar el nombre de usuario y la contraseña del usuario de la base de datos de Amazon Redshift en el contexto del cual se ejecutará la consulta de seguridad.

> **Note**
>
> Amazon ML requiere un nombre de usuario y una contraseña de Amazon Redshift para conectarse a la base de datos de Amazon Redshift. Después de descargar los datos a Amazon S3, Amazon ML no suele reutilizar ni almacenar la contraseña.

Función de Amazon ML en Amazon Redshift

Use este parámetro para especificar el nombre del rol de IAM que debería utilizar Amazon ML para configurar los grupos de seguridad para el clúster de Amazon Redshift y la política de buckets para la ubicación de almacenamiento provisional de Amazon S3.

Si no dispone de ningún rol de IAM que pueda acceder a Amazon Redshift, Amazon ML puede crear un rol automáticamente. Cuando Amazon ML crea un rol, crea y asocia una política administrada por el cliente a un rol de IAM. La política que crea Amazon ML solo concede a Amazon ML permiso para acceder al clúster que especifique.

Si ya posee un rol de IAM para acceder a Amazon Redshift, puede escribir el ARN del rol o elegir el rol de la lista desplegable. Los roles de IAM con acceso a Amazon Redshift aparecen en la parte superior de la lista desplegable.

El rol de IAM debe tener el siguiente contenido:

```
{
    "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/*" }
        }
    }]
}
```

Para obtener más información sobre las políticas administradas por los clientes, consulte Políticas administradas por el cliente en la Guía del usuario de IAM.

Consulta SQL en Amazon Redshift

Use este parámetro para especificar la consulta SQL SELECT que ejecuta Amazon ML en la base de datos de Amazon Redshift para seleccionar los datos. Amazon ML utiliza la acción DESCARGAR de Amazon Redshift para copiar los resultados de la consulta a una ubicación de Amazon S3 de forma segura.

> **Note**
>
> Amazon ML funciona mejor cuando los registros de entrada se encuentran en orden aleatorio (mezclados). Es fácil mezclar los resultados de la consulta SQL de Amazon Redshift mediante el rol random() de Amazon Redshift. Por ejemplo, imaginemos que esta es la consulta original:
>
> ```
> "SELECT col1, col2, … FROM training_table"
> ```
>
> Puede incrustar la mezcla aleatoria mediante la actualización de la consulta de esta manera:
>
> ```
> "SELECT col1, col2, … FROM training_table ORDER BY random()"
> ```

Ubicación de los esquemas (opcional)

Use este parámetro para especificar la ruta de Amazon S3 del esquema para los datos de Amazon Redshift que exportará Amazon ML.

Si no proporciona ningún esquema para la fuente de datos, la consola de Amazon ML creará un esquema de Amazon ML basado en el esquema de datos de la consulta SQL de Amazon Redshift automáticamente. Los esquemas de Amazon ML tienen menos tipos de datos que los esquemas de Amazon Redshift, por lo que no se trata de una conversión. one-to-one La consola de Amazon ML convierte tipos de datos de Amazon Redshift a tipos de datos de Amazon ML mediante el siguiente esquema de conversión.

Tipos de datos de Amazon Redshift	Alias de Amazon Redshift	Tipo de datos de Amazon ML
SMALLINT	INT2	NUMERIC
INTEGER	INT, INT4	NUMERIC
BIGINT	INT8	NUMERIC
DECIMAL	NUMERIC	NUMERIC
REAL	FLOAT4	NUMERIC
DOUBLE PRECISION	FLOAT8, FLOTAR	NUMERIC
BOOLEAN	BOOL	BINARIO
CHAR	CHARACTER, NCHAR, BPCHAR	CATEGÓRICO
VARCHAR	CHARACTER VARYING, NVARCHAR, TEXT	TEXT
DATE		TEXT
TIMESTAMP	TIMESTAMP WITHOUT TIME ZONE	TEXT

Para convertirlos a tipos de datos `Binary` de Amazon ML, los valores binarios de Amazon Redshift deben admitir los valores de los booleanos. Si el tipo de datos booleanos no admite algunos valores, Amazon ML los convierte al tipo de datos más específico posible. Por ejemplo, si un booleano de Amazon Redshift tiene los valores `0`, `1` y `2`, Amazon ML convierte el booleano a un tipo de datos `Numeric`. Para obtener más información sobre los valores binarios admitidos, consulte Uso del AttributeType campo.

Si Amazon ML no puede averiguar un tipo de datos, se establece el tipo predeterminado `Text`.

Una vez que Amazon ML convierte el esquema, puede revisar y corregir los tipos de datos de Amazon ML asignados al asistente de creación de orígenes de datos y revisar el esquema antes de que Amazon ML cree el origen de datos.

Ubicación de almacenamiento provisional de Amazon S3

Use este parámetro para especificar el nombre de la ubicación de almacenamiento provisional de Amazon S3 donde Amazon ML almacena los resultados de la consulta SQL de Amazon Redshift. Después de crear el origen de datos, Amazon ML utiliza los datos de la ubicación de almacenamiento provisional en lugar de volver a Amazon Redshift.

> **Note**
>
> Como Amazon ML asume el rol de IAM definido por la función Amazon Redshift de Amazon ML, Amazon ML tiene permisos para acceder a cualquier objeto en la ubicación provisional de Amazon S3 especificada. Por ello, le recomendamos que solo almacene archivos que no contengan información confidencial en la ubicación de almacenamiento provisional de Amazon S3. Por ejemplo, si el bucket raíz es `s3://mybucket/`, le sugerimos que cree una ubicación para almacenar solo los archivos a los que desea que tenga acceso Amazon ML, como por ejemplo `s3://mybucket/AmazonMLInput/`.

Creación de una fuente de datos con datos de Amazon Redshift (consola)

La consola de Amazon ML ofrece dos formas para crear un origen de datos utilizando datos de Amazon Redshift. Puede crear una fuente de datos completando el asistente para la creación de una fuente de datos o, si ya dispone de una fuente de datos creada a partir de datos de Amazon Redshift, puede copiar la fuente de datos original y modificar su configuración. Copiar una fuente de datos le permite crear múltiples fuentes de datos similares fácilmente.

Para obtener información sobre cómo crear una fuente de datos mediante la API, consulte. CreateDataSourceFromRedshift

Para obtener más información sobre los parámetros de los siguientes procedimientos, consulte Parámetros necesarios para el asistente Create Datasource.

Temas

- Creación de una fuente de datos (consola)
- Copiar una fuente de datos (consola)

Creación de una fuente de datos (consola)

Para descargar datos de Amazon Redshift en un origen de datos de Amazon ML, utilice el asistente para la creación de un origen de datos.

Para crear una fuente de datos a partir de datos de Amazon Redshift

1. Abra la consola Amazon Machine Learning en https://console.aws.amazon.com/machinelearning/.
2. En el panel de Amazon ML, en Entidades, seleccione Crear nuevo... y, a continuación, seleccione Origen de datos.
3. En la página Datos de entrada, elija Amazon Redshift.
4. En el asistente de creación de orígenes de datos, en Cluster identifier (Identificador de clúster), escriba el nombre del clúster.
5. En Nombre de la base de datos, escriba el nombre de la base de datos de Amazon Redshift.
6. En Database user name (Nombre de usuario de base de datos), escriba el nombre de usuario de la base de datos.
7. En Database password, escriba la contraseña de la base de datos.
8. En IAM role, seleccione el rol de IAM. Si aún no tiene uno, seleccione Crear nuevo rol. Amazon ML crea un rol de IAM de Amazon Redshift para usted.
9. Para probar la configuración de Amazon Redshift, elija Probar acceso (junto a Rol de IAM). Si Amazon ML no puede conectarse a Amazon Redshift con la configuración proporcionada, no puede continuar la creación de un origen de datos. Para obtener ayuda sobre la resolución de problemas, consulte Solución de errores.
10. En SQL query, escriba la consulta SQL.
11. En Ubicación del esquema, seleccione si desea que Amazon ML cree un esquema automáticamente. Si ya ha creado un esquema, escriba la ruta de Amazon S3 del archivo del esquema.
12. En Ubicación de almacenamiento provisional de Amazon S3, escriba la ruta de Amazon S3 del bucket donde desea que Amazon ML almacene los datos que descarga de Amazon Redshift.
13. (Opcional) En Datasource name, escriba un nombre para el origen de datos.
14. Seleccione Verificar. Amazon ML verifica que pueda conectarse a la base de datos de Amazon Redshift.
15. En la página Schema (Esquema), revise los tipos de datos de todos los atributos y corríjalos, según sea necesario.

16. Elija Continuar.
17. Si desea usar este origen de datos para crear o evaluar un modelo de ML, en Do you plan to use this dataset to create or evaluate an ML model? (¿Va a usar este conjunto de datos para crear o evaluar un modelo de ML?), elija Yes (Sí). Si elige Yes, seleccione la línea de destino. Para obtener más información sobre destinos, consulte Uso del targetAttributeName campo.

    Si desea utilizar este origen de datos junto con un modelo que ya ha creado para generar predicciones, seleccione No.
18. Elija Continuar.
19. En Does your data contain an identifier? (¿Sus datos contienen un identificador?), si sus datos no contienen un identificador de fila, elija No.

    Si los datos contienen un identificador de línea, seleccione Yes. Para obtener información sobre identificadores de línea, consulte Funcionamiento del campo rowID.
20. Elija Revisar.
21. En la página Review, revise los ajustes y, a continuación, seleccione Finish.

Una vez que haya creado una fuente de datos, puede usarla para create an ML model. Si ya ha creado un modelo, puede usar la fuente de datos para evaluate an ML model o generate predictions.

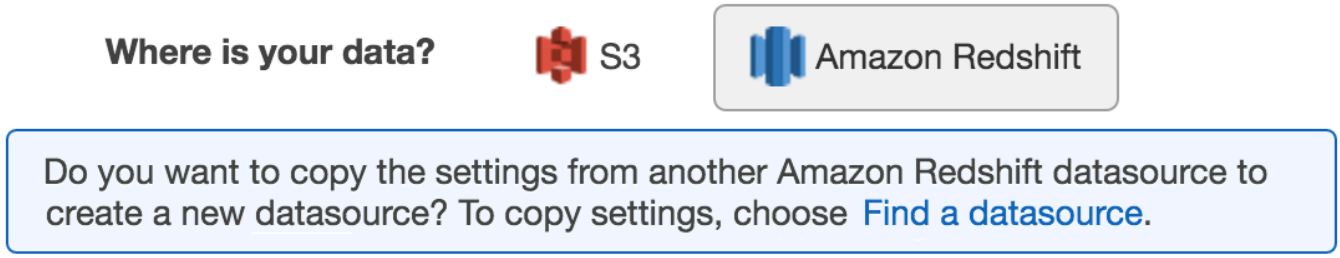
Copiar una fuente de datos (consola)

Cuando desee crear una fuente de datos que sea similar a una fuente de datos existente, puede utilizar la consola de Amazon ML para copiar la fuente de datos original y modificar su configuración. Por ejemplo, puede optar por empezar con una fuente de datos existente y, a continuación, modificar el esquema de datos para que se ajuste mejor a sus datos; cambiar la consulta SQL utilizada para descargar datos de Amazon Redshift; o especificar un usuario diferente AWS Identity and Access Management (IAM) para acceder al clúster de Amazon Redshift.

Para copiar y modificar una fuente de datos de Amazon Redshift

1. Abra la consola Amazon Machine Learning en https://console.aws.amazon.com/machinelearning/.
2. En el panel de Amazon ML, en Entidades, seleccione Crear nuevo... y, a continuación, seleccione Origen de datos.

3. En la página Datos de entrada, en ¿Dónde están los datos?, elija Amazon Redshift. Si ya tiene una fuente de datos creada a partir de datos de Amazon Redshift, tiene la opción de copiar la configuración de otra fuente de datos.



   Si aún no tiene ninguna fuente de datos creada a partir de datos de Amazon Redshift, esta opción no aparecerá.

4. Seleccione Find a datasource (Buscar un origen de datos).
5. Seleccione la fuente de datos que desee copiar y seleccione Copiar configuración. Amazon ML rellena la mayor parte de la configuración de la fuente de datos automáticamente con la configuración de la fuente de datos original. No copia la contraseña de la base de datos, la ubicación del esquema o el nombre de la fuente de datos de la fuente de datos original.
6. Modifique la configuración rellenada automáticamente que desee cambiar. Por ejemplo, si desea cambiar los datos que descarga Amazon ML de Amazon Redshift, cambie la consulta SQL.
7. En Database password, escriba la contraseña de la base de datos. Amazon ML no almacena ni reutiliza la contraseña, por lo que debe proporcionarla siempre.
8. (Opcional) En Ubicación del esquema, Amazon ML selecciona Deseo que Amazon ML genere un esquema recomendado de forma previa y automática. Si ya ha creado un esquema, seleccione Deseo usar el esquema que he creado y almacenado en Amazon S3 y escriba la ruta del archivo del esquema de Amazon S3.
9. (Opcional) En Datasource name, escriba un nombre para el origen de datos. De lo contrario, Amazon ML genera un nuevo nombre de la fuente de datos.
10. Seleccione Verificar. Amazon ML verifica que pueda conectarse a la base de datos de Amazon Redshift.
11. (Opcional) Si Amazon ML infiere el esquema por usted, en la página Esquema, revise los tipos de datos de todos los atributos y corríjalos, según sea necesario.
12. Elija Continuar.
13. Si desea usar este origen de datos para crear o evaluar un modelo de ML, en Do you plan to use this dataset to create or evaluate an ML model? (¿Va a usar este conjunto de datos para crear

o evaluar un modelo de ML?), elija Yes (Sí). Si elige Yes, seleccione la línea de destino. Para obtener más información sobre destinos, consulte Uso del targetAttributeName campo.

Si desea utilizar este origen de datos junto con un modelo que ya ha creado para generar predicciones, seleccione No.

14. Elija Continuar.
15. En Does your data contain an identifier? (¿Sus datos contienen un identificador?), si sus datos no contienen un identificador de fila, elija No.

    Si los datos contienen un identificador de fila, seleccione Yes y seleccione la fila que desee utilizar como identificador. Para obtener información sobre identificadores de línea, consulte Funcionamiento del campo rowID.
16. Elija Revisar.
17. Revise la configuración y, a continuación seleccione Finish.

Una vez que haya creado una fuente de datos, puede usarla para create an ML model. Si ya ha creado un modelo, puede usar la fuente de datos para evaluate an ML model o generate predictions.

Temas de solución de problemas de Amazon Redshift

A medida que crea su fuente de datos, modelos de ML y evaluación de Amazon Redshift, Amazon Machine Learning (Amazon ML) informa el estado de sus objetos de Amazon ML en la consola de Amazon ML. Si Amazon ML devuelve mensajes de error, utilice la siguiente información y los siguientes recursos para solucionar los problemas.

Para obtener respuestas a preguntas generales sobre Amazon ML, consulte Amazon Machine Learning FAQs. También puede buscar respuestas y publicar preguntas en el foro de Amazon Machine Learning.

Temas

- Solución de errores
- Cómo ponerse en contacto con AWS Support

Solución de errores

El formato del rol no es válido. Proporcione un rol de IAM válido. Por ejemplo, arn:aws:iam:: ID:Role/. YourAccount YourRedshiftRole

Causa

El formato del nombre de recurso de Amazon (ARN) de la función de IAM es incorrecto.

Solución

En el asistente Create Datasource, corrija el ARN de su rol. Para obtener información sobre la función ARNs de formato, consulte IAM en la Guía del usuario de IAM. ARNs La región es opcional para el rol de IAM. ARNs

El rol no es válido. Amazon ML no puede asumir el rol de IAM <rol de ARN>. Proporcione un rol de IAM válido y haga que sea accesible para Amazon ML.

Causa

El rol no está configurado para permitir que Amazon ML lo asuma.

Solución

En la consola de IAM, edite su rol de forma que tenga una política de confianza que permita a Amazon ML asumir el rol adjunto a ella.

Este usuario <ARN de usuario> no está autorizado para transferir el rol de IAM <ARN de rol>.

Causa

El usuario de IAM no tiene una política de permisos que le permita transferir un rol a Amazon ML.

Solución

Adjunte una política de permisos al usuario de IAM que le permita transferir roles a Amazon ML. Puede adjuntar una política de permisos al usuario de IAM en la consola de IAM.

No se permite transferir un rol de IAM entre cuentas. El rol de IAM debe pertenecer a esta cuenta.

Causa

No se puede transferir un rol que pertenece a otra cuenta de IAM.

Solución

Inicie sesión en la cuenta de AWS que utilizó para crear el rol. Puede ver sus roles de IAM en la consola de IAM.

El rol especificado no tiene permisos para realizar la operación. Proporcione un rol que tenga una política que proporcione a Amazon ML los permisos necesarios.

Causa

El rol de IAM no tiene permisos para realizar la operación solicitada.

Solución

Edite la política de permisos adjunta al rol en la consola de IAM para proporcionar los permisos necesarios.

Amazon ML no puede configurar un grupo de seguridad en ese clúster de Amazon Redshift con el rol de IAM especificado.

Causa

Su rol de IAM no tiene los permisos necesarios para configurar un clúster de seguridad Amazon Redshift.

Solución

Edite la política de permisos adjunta al rol en la consola de IAM para proporcionar los permisos necesarios.

Se ha producido un error cuando Amazon ML ha intentado configurar un grupo de seguridad en el clúster. Inténtelo de nuevo más tarde.

Causa

Cuando Amazon ML ha intentado conectarse a su clúster de Amazon Redshift, se ha producido un problema.

Solución

Verifique que el rol de IAM que ha facilitado en el asistente Create Datasource tiene todos los permisos necesarios.

El formato del ID del clúster no es válido. El clúster IDs debe empezar por una letra y debe contener únicamente caracteres alfanuméricos y guiones. No pueden contener dos guiones consecutivos ni acabar con guion.

Causa

El formato de ID de clúster Amazon Redshift es incorrecto.

Solución

En el asistente Create Datasource, corrija el ID de clúster de forma que contenga únicamente caracteres alfanuméricos y guiones y no contenga dos guiones consecutivos ni finalice con un guion.

No existe ningún clúster <nombre de clúster de Amazon Redshift>, o bien el clúster no está en la misma región que el servicio de Amazon ML. Especifique un clúster en la misma región que este Amazon ML.

Causa

Amazon ML no puede encontrar el clúster de Amazon Redshift porque no está ubicado en la región en la que está creando un origen de datos de Amazon ML.

Solución

Verifique que el clúster existe en la página de Clústeres de la consola Amazon Redshift, que está creando una fuente de datos en la misma región en la que se encuentra el clúster Amazon Redshift y que el ID de clúster especificado en el asistente Create Datasource es correcto.

Amazon ML no puede leer los datos del clúster de Amazon Redshift. Proporcione el ID de clúster de Amazon Redshift correcto.

Causa

Amazon ML no puede leer los datos en el clúster de Amazon Redshift que ha especificado.

Solución

En el asistente Create Datasource, especifique el ID de clúster Amazon Redshift correcto, verifique que está creando una fuente de datos en la misma región que el clúster Amazon Redshift y que el clúster se encuentra en la lista de la página Clústeres de Amazon Redshift.

El clúster <nombre de clúster de Amazon Redshift> no se encuentra accesible al público.

Causa

Amazon ML no puede obtener acceso al clúster porque este no es de acceso público y no tiene una dirección IP pública.

Solución

Haga que el clúster sea accesible al público y asígnele una dirección IP pública. Para obtener información acerca de cómo hacer que los clústeres sean accesibles al público, consulte Modificación de un clúster en la Guía de administración de Amazon Redshift.

El estado de clúster de <Redshift> no está disponible para Amazon ML. Utilice la consola de Amazon Redshift para ver y resolver este estado de clúster. El estado del clúster debe ser "disponible".

Causa

Amazon ML no puede ver el estado del clúster.

Solución

Asegúrese de que el clúster está disponible. Para obtener información acerca de cómo comprobar el estado del clúster, consulte Obtención de información general acerca del estado del clúster en la Guía de administración de Amazon Redshift. Para obtener información acerca de cómo reiniciar el clúster para que se encuentre disponible, consulte Reinicio de un clúster en la Guía de administración de Amazon Redshift.

No existe ninguna base de datos <nombre de base de datos> en este clúster. Verifique que el nombre de la base de datos es correcta o especifique otro clúster y otra base de datos.

Causa

Amazon ML no puede encontrar la base de datos especificada en el clúster especificado.

Solución

Verifique que el nombre de la base de datos introducido en el asistente Create Datasource es correcto o especifique los nombres correctos de clúster y base de datos.

Amazon ML no ha podido obtener acceso a la base de datos. Proporcione una contraseña válida para el usuario de base de datos <nombre de usuario>.

Causa

La contraseña que ha facilitado en el asistente de creación de orígenes de datos para permitir que Amazon ML obtenga acceso a la base de datos de Amazon Redshift es incorrecta.

Solución

Proporcione la contraseña correcta para el usuario de la base de datos de Amazon Redshift.

Se ha producido un error cuando Amazon ML ha intentado validar la consulta.

Causa

Ha surgido un problema con la consulta SQL.

Solución

Verifique que la consulta SQL es válida.

Se ha producido un error al ejecutar la consulta SQL. Verifique el nombre de la base de datos y la consulta proporcionada. Causa raíz: {serverMessage}.

Causa

Amazon Redshift no ha podido ejecutar la consulta.

Solución

Verifique que ha especificado el nombre correcto de la base de datos en el asistente Create Datasource y que su consulta SQL es válida.

Se ha producido un error al ejecutar la consulta SQL. Causa raíz: {serverMessage}.

Causa

Amazon Redshift no ha podido encontrar la tabla especificada.

Solución

Verifique que la tabla que ha especificado en el asistente Create Datasource se encuentra presente en la base de datos de clúster de Amazon Redshift y que ha introducido el ID de clúster, el nombre de la base de datos y la consulta SQL correctos.

Cómo ponerse en contacto con AWS Support

Si dispone de AWS Premium Support, puede crear un caso de soporte técnico en el AWS Support Center.

Uso de datos de una base de datos de Amazon RDS para crear una fuente de datos de Amazon ML

Amazon ML le permite crear un objeto de origen de datos a partir de los datos almacenados en una base de datos Amazon Relational Database Service (Amazon RDS). Al realizar esta acción, Amazon ML crea un objeto de AWS Data Pipeline que ejecuta la consulta SQL que especifique y coloca el resultado en un bucket de S3 de su elección. Amazon ML utiliza esos datos para crear la fuente de datos.

> **Note**
>
> Amazon ML solo admite bases de datos MySQL en VPCs.

Para que Amazon ML pueda leer los datos de entrada, debe exportarlos al Amazon Simple Storage Service (Amazon S3). Puede configurar Amazon ML para que realice la exportación con la API. (RDS se limita a la API y no está disponible desde la consola).

Para que Amazon ML se conecte a la base de datos de MySQL en Amazon RDS y lea datos en su nombre, debe proporcionar lo siguiente:

- El identificador de instancias de bases de datos de RDS
- El nombre de la base de datos de MySQL
- La función AWS Identity and Access Management (IAM) que se utiliza para crear, activar y ejecutar la canalización de datos
- Las credenciales del usuario de la base de datos:
  - Nombre de usuario
  - Contraseña
- La información de seguridad de AWS Data Pipeline:
  - El rol de recursos de IAM
  - El rol de servicio de IAM
- La información de seguridad de Amazon RDS:
  - El ID de subred
  - El grupo de seguridad IDs
- La consulta SQL que especifica los datos que desea utilizar para crear la fuente de datos
- La ubicación de salida de S3 (bucket) utilizada para almacenar los resultados de la consulta

- (Opcional) La ubicación del archivo de esquema de datos

Además, debe asegurarse de que los usuarios o roles de IAM que crean las fuentes de datos de Amazon RDS mediante la operación CreateDataSourceFromRDS tengan el permiso. `iam:PassRole` Para obtener más información, consulte Control de acceso a los recursos de Amazon ML con IAM.

Temas

- Identificador de instancias de bases de datos de RDS
- Nombre de la base de datos de MySQL
- Credenciales del usuario de la base de datos
- Información de seguridad de AWS Data Pipeline
- Información de seguridad de Amazon RDS
- Consulta SQL de MySQL
- Ubicación de salida de S3

Identificador de instancias de bases de datos de RDS

El identificador de instancias de bases de datos de RDS es un nombre único que suministra y que identifica la instancia de base de datos que Amazon ML debe utilizar a la hora de interactuar con Amazon RDS. Puede encontrar el identificador de instancias de bases de datos de RDS en la consola de Amazon RDS.

Nombre de la base de datos de MySQL

El nombre de la base de datos de MySQL especifica el nombre de la base de datos de MySQL en la instancia de bases de datos de RDS.

Credenciales del usuario de la base de datos

Para conectarse a la instancia de base de datos de RDS, debe proporcionar el nombre de usuario y la contraseña de usuario de la base de datos que tenga permisos suficientes para ejecutar la consulta SQL que proporcione.

Información de seguridad de AWS Data Pipeline

Para habilitar el acceso seguro de AWS Data Pipeline, debe proporcionar los nombres del rol de recursos de IAM y del rol de servicios de IAM.

Una EC2 instancia asume la función de recurso para copiar datos de Amazon RDS a Amazon S3. La forma más sencilla de crear este rol de recursos es usar la plantilla `DataPipelineDefaultResourceRole` e incorporar **`machinelearning.aws.com`** como servicio de confianza. Para obtener información acerca de la plantilla, consulte Configuración de roles de IAM en la Guía para desarrolladores de AWS Data Pipeline.

Si crea su propia función, debe tener el siguiente contenido:

```
{
    "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 asume el rol de servicio para monitorizar el progreso de copia de datos de Amazon RDS a Amazon S3. La forma más sencilla de crear este rol de recursos es usar la plantilla `DataPipelineDefaultRole` e incorporar `machinelearning.aws.com` como servicio de confianza. Para obtener información acerca de la plantilla, consulte Configuración de roles de IAM en la Guía para desarrolladores de AWS Data Pipeline.

Información de seguridad de Amazon RDS

Para habilitar el acceso seguro de Amazon RDS, debe proporcionar el `VPC Subnet ID` y el `RDS Security Group IDs`. También tendrá que configurar las reglas de entrada adecuadas para la

subred VPC a la que apunta el parámetro `Subnet ID` y proporcionar el ID del grupo de seguridad que tiene este permiso.

Consulta SQL de MySQL

El parámetro `MySQL SQL Query` especifica la consulta SQL SELECT que desea ejecutar en la base de datos MySQL. Los resultados de la consulta se copian en la ubicación de salida de S3 (bucket) que especifique.

> **Note**
>
> La tecnología de aprendizaje automático funciona mejor cuando los registros de entrada se presentan en orden aleatorio (desordenados). Puede mezclar fácilmente los resultados de la consulta SQL de MySQL mediante la función `rand()`. Por ejemplo, imaginemos que esta es la consulta original:
> "SELECT col1, col2, … FROM training_table"
> Puede añadir una mezcla aleatoria actualizando la consulta de esta manera:
> "SELECT col1, col2, … FROM training_table ORDER BY rand()"

Ubicación de salida de S3

El parámetro `S3 Output Location` especifica el nombre de la ubicación de Amazon S3 "provisional" en la que se ubican los resultados de la consulta SQL de MySQL.

> **Note**
>
> Debe asegurarse de que Amazon ML tiene permisos para leer datos desde esta ubicación una vez que los datos se exportan desde Amazon RDS. Para obtener información acerca de cómo ajustar estos permisos, consulte Concesión de permisos a Amazon ML para leer datos desde Amazon S3.

Entrenamiento de modelos de ML

El proceso de entrenamiento de un modelo de ML consiste en proporcionar datos de entrenamiento de los cuales aprender a un algoritmo de ML (es decir, el algoritmo de aprendizaje). El término modelo de ML se refiere al artefacto de modelo que se crea en el proceso de entrenamiento.

Los datos de entrenamiento deben contener la respuesta correcta, que se conoce como destino o atributo de destino. El algoritmo de aprendizaje encuentra patrones en los datos de entrenamiento que asignan los atributos de los datos de entrada al destino (la respuesta que desea predecir) y genera un modelo de ML que captura dichos patrones.

Puede utilizar el modelo de ML para obtener predicciones sobre datos nuevos para los que no se conoce la respuesta de destino. Por ejemplo, si desea entrenar un modelo de ML para que prediga si un mensaje de correo electrónico es spam o no. Le proporcionaría datos de entrenamiento a Amazon ML que contienen correos electrónicos para los que conoce el destino (es decir, una etiqueta que indica si un mensaje es spam o no). Amazon ML entrenaría un modelo de ML mediante la utilización de estos datos, lo que se traduce en un modelo que intenta predecir si los correos electrónicos nuevos son spam o no.

Para obtener información general sobre los modelos de ML y algoritmos de ML, consulte Conceptos del aprendizaje automático.

Temas

- Tipos de modelos de ML
- Proceso de formación
- Parámetros de entrenamiento
- Creación de un modelo de ML

Tipos de modelos de ML

Amazon ML admite tres tipos de modelos de ML: clasificación binaria, clasificación multiclase y regresión. El tipo de modelo que debe elegir depende del tipo de destino que desee predecir.

Modelo de clasificación binaria

Los modelos de ML de problemas de clasificación binaria predicen un resultado binario (una de las dos clases posibles). Para formar modelos de clasificación binaria, Amazon ML utiliza el algoritmo de aprendizaje estándar del sector conocido como regresión logística.

Ejemplos de problemas de clasificación binaria

- "¿Este correo electrónico es spam o no?"
- "¿El cliente comprará este producto?"
- "¿Es este producto un libro o una animal de granja?"
- "¿Esta revisión la ha escrito un cliente o un robot?"

Modelo de clasificación multiclase

Los modelos de ML de problemas de clasificación multiclase le permiten generar predicciones para varias clases (predecir uno de más de dos resultados). Para formar modelos multiclase, Amazon ML utiliza el algoritmo de aprendizaje estándar del sector conocido como regresión logística multinomial.

Ejemplos de problemas multiclase

- "¿Este producto es un libro, una película o una prenda de ropa?"
- "¿Esta película es una comedia romántica, un documental o un thriller?"
- "¿Qué categoría de productos es más interesante para este cliente?"

Modelo de regresión

Los modelos de ML para problemas de regresión predicen un valor numérico. Para formar modelos de regresión, Amazon ML utiliza el algoritmo de aprendizaje estándar del sector conocido como regresión lineal.

Ejemplos de problemas de regresión

- "¿Cuál será la temperatura en Seattle mañana?"
- "Para este producto, ¿cuántas unidades se venderán?"
- "¿A qué precio se venderá esta casa?"

Proceso de formación

Para formar un modelo de ML, debe especificar lo siguiente:

- Fuente de datos de formación de entrada
- Nombre del atributo de datos que contiene el destino que se va a predecir
- Instrucciones de transformación de datos necesaria
- Parámetros de formación para controlar el algoritmo de aprendizaje

Durante el proceso de formación, Amazon ML selecciona automáticamente el algoritmo de aprendizaje correcto para usted, en función del tipo de destino que haya especificado en la fuente de datos de formación.

Parámetros de entrenamiento

Normalmente, los algoritmos de aprendizaje automático aceptan parámetros que pueden utilizarse para controlar determinadas propiedades del proceso de aprendizaje y del modelo de ML resultante. En Amazon Machine Learning, se denominan parámetros de entrenamiento. Puede definir estos parámetros utilizando la consola, la API o la interfaz de la línea de comandos (CLI) de Amazon ML. Si no configura ningún parámetro, Amazon ML utilizará los valores predeterminados que funcionan bien para una gran variedad de tareas de machine learning.

Puede especificar valores para los siguientes parámetros de entrenamiento:

- Tamaño máximo del modelo
- Número máximo de iteraciones en los datos de aprendizaje
- Tipo de mezcla
- Tipo de regularización
- Cantidad de regularización

En la consola de Amazon ML, los parámetros de aprendizaje están establecidos de forma predeterminada. La configuración predeterminada es suficiente para la mayoría de problemas de ML, pero puede elegir otros valores para afinar el desempeño. Algunos otros parámetros de entrenamiento, como por ejemplo la tasa de aprendizaje, están configuradas automáticamente en función de los datos.

En las secciones siguientes se proporciona más información acerca de los parámetros de entrenamiento.

Tamaño máximo del modelo

El tamaño máximo del modelo es el tamaño total, en unidades de bytes, de patrones que Amazon ML crea durante el entrenamiento de un modelo de ML.

De forma predeterminada, Amazon ML crea un modelo de 100 MB. Puede indicarle a Amazon ML que cree un modelo más grande o más pequeño especificando un tamaño diferente. Para el rango de tamaños disponibles, consulte Tipos de modelos de ML

Si Amazon ML no encuentra patrones suficientes para rellenar el tamaño del modelo, crea un modelo más pequeño. Por ejemplo, si especifica un tamaño máximo de modelo de 100 MB pero Amazon ML encuentra patrones que suman un total de 50 MB, el modelo resultante será de 50 MB. Si Amazon ML encuentra más patrones de los que cabrán en el tamaño especificado, impone un valor de corte máximo recortando los patrones que menos afectan a la calidad del modelo aprendido.

La elección del tamaño del modelo le permite controlar el equilibrio entre la calidad de predicción del modelo y el costo de su uso. Los modelos más pequeños pueden provocar que Amazon ML elimine muchos patrones para que quepan en el límite de tamaño máximo, hecho que afectará a la calidad de las predicciones. Los modelos más grandes, por otro lado, son más costosos de consultar para obtener predicciones en tiempo real.

> **Note**
>
> Si utiliza un modelo de ML para generar predicciones en tiempo real, producirá una pequeña carga de reserva de capacidad que se determina en función del tamaño del modelo. Para obtener más información, consulte Precios de Amazon ML.

Los conjuntos de datos de entrada grandes no generarán necesariamente modelos más grandes. ya que los modelos almacenan patrones y no datos de entrada; si los patrones son pocos y sencillos, el modelo resultante será pequeño. Los datos de entrada que tienen un gran número de atributos sin formato (columnas de entrada) o funciones derivados (salidas de las transformaciones de datos de Amazon ML) generarán probablemente más patrones y estos se almacenarán durante el proceso de entrenamiento. Es mejor realizar algunas pruebas antes de seleccionar el tamaño del modelo correcto para sus datos y problemas. El registro de entrenamiento de modelos de Amazon ML (que puede descargar desde la consola o mediante la API) contiene mensajes sobre la cantidad de recortes de modelos (si los hubo) que se produjeron durante el proceso de entrenamiento, lo que le permite estimar la hit-to-prediction calidad potencial.

Número máximo de iteraciones en los datos

Para obtener el mejor resultado, quizá sea necesario que Amazon ML realice varias iteraciones en los datos para descubrir patrones. Amazon ML realiza 10 iteraciones de forma predeterminada, pero puede cambiar el valor predeterminado estableciendo un número de hasta 100. Amazon ML lleva un seguimiento de la calidad de los patrones (convergencia del modelo) mientras va avanzando y detiene automáticamente el entrenamiento cuando no hay más puntos de datos o patrones por descubrir. Por ejemplo, si establece el número de iteraciones a 20 pero Amazon ML descubre que no se pueden encontrar patrones nuevos después de 15 iteraciones, detendrá la formación a 15.

En general, los conjuntos de datos con solo unas cuantas observaciones suelen requerir más iteraciones en los datos para obtener una mayor calidad del modelo. Los conjuntos de datos grandes suelen contener muchos puntos de datos similares, hecho que elimina la necesidad de un gran número de iteraciones. El impacto de la elección de más iteraciones en los datos es doble: el aprendizaje del modelo necesita más tiempo y tiene un costo mayor.

Tipo de mezcla para los datos de entrenamiento

En Amazon ML, debe mezclar sus datos de entrenamiento. La mezcla desordena los datos, de modo que el algoritmo SGD no detecta un tipo de datos por demasiadas observaciones consecutivas. Por ejemplo, si está entrenando un modelo de ML para predecir un tipo de producto y los datos de entrenamiento incluyen tipos de producto como películas, juguetes y videojuegos, si ordena los datos por la columna del tipo de producto antes de cargarlos, el algoritmo ve los datos alfabéticamente por tipo de producto. El algoritmo observa primero todos los datos de películas y el modelo de ML comienza a aprender patrones para películas. A continuación, cuando el modelo encuentra los datos de juguetes, cada actualización que hace el algoritmo ajustaría el modelo al tipo de producto de juguete, incluso si estas actualizaciones degradasen los patrones que se ajustan a las películas. Este cambio repentino del tipo de películas a juguetes puede producir un modelo que no aprenderá a predecir los tipos de productos con precisión.

Debe mezclar sus datos de entrenamiento incluso si selecciona la opción de división aleatoria al dividir la fuente de datos de entrada en partes de formación y evaluación. La estrategia de división aleatoria elige un subconjunto aleatorio de los datos para cada fuente de datos, pero no cambia el orden de las filas en la fuente de datos. Para obtener más información acerca de la división de los datos, consulte División de datos.

Al crear un modelo de ML mediante la consola, Amazon ML mezcla los datos con una técnica de mezcla seudoaleatoria de manera predeterminada. Independientemente de la cantidad de iteraciones establecidas, Amazon ML mezcla los datos solo una vez antes de entrenar el modelo de

ML. Si mezcló los datos antes de proporcionarlos a Amazon ML y no desea que Amazon ML vuelva a mezclarlos de nuevo, puede establecer el valor de tipo de mezcla en `none`. Por ejemplo, si mezcló aleatoriamente los registros en su archivo .csv antes de cargarlo en Amazon S3, usó la función `rand()` en su consulta SQL de MySQL al crear su fuente de datos desde Amazon RDS o usó la función `random()` en su consulta SQL de Amazon Redshift al crear su fuente de datos de Amazon Redshift, configurar el tipo de mezcla en `none` no afectará la precisión predictiva de su modelo de aprendizaje automático. Mezclar los datos solo una vez reduce el tiempo de ejecución y el costo para la creación de un modelo de ML.

> **Important**
>
> Al crear un modelo de ML mediante la API de Amazon ML, Amazon ML no mezcla sus datos de forma predeterminada. Si utiliza la API en lugar de la consola para crear su modelo de ML, recomendamos encarecidamente que mezcle sus datos ajustando el parámetro `sgd.shuffleType` en `auto`.

Tipo y cantidad de regularización

El desempeño predictivo de modelos de ML complejos (los que tienen muchos atributos de entrada) se ve perjudicado cuando los datos contienen demasiados patrones. Como aumenta el número de patrones, aumenta también la probabilidad de que el modelo aprenda artefactos de datos involuntarios en lugar de patrones de datos reales. En este caso, el modelo rinde muy bien con los datos de entrenamiento pero no puede generalizar bien los datos nuevos. Este fenómeno se conoce como sobreajuste de los datos de aprendizaje.

La regularización ayuda a prevenir que los modelos lineales realicen un sobreajuste de los ejemplos de datos de entrenamiento penalizando los valores de peso extremos. La regularización L1 reduce el número de funciones utilizadas en el modelo y aumenta el peso de características que, de otro modo, tendrían a cero los pesos muy reducidos. La regularización L1 produce modelos dispersos y reduce la cantidad de ruido en el modelo. La regularización L2 produce en general valores de peso más pequeños, lo que estabiliza las ponderaciones cuando hay gran correlación entre las funciones. Puede controlar la cantidad de regularización L1 o L2 mediante el parámetro `Regularization amount`. Especificar un valor extremadamente grande de `Regularization amount` puede provocar que todas las funciones tengan un peso igual a cero.

La selección y el ajuste del valor de regularización óptimo es un tema candente en la investigación del aprendizaje automático. Probablemente se beneficiará de seleccionar una cantidad moderada

de regularización L2, que es la opción predeterminada en la consola de Amazon ML. Los usuarios avanzados pueden elegir entre tres tipos de regularización (ninguna, L1, o L2) y de cantidad. Para obtener más información sobre la regularización, consulte Regularization (mathematics) (en inglés).

Parámetros de entrenamiento: tipos y valores predeterminados

En la siguiente tabla se muestran los parámetros de entrenamiento de Amazon ML, junto con los valores predeterminados y el rango admisible para cada uno de ellos.

Parámetro de entrenamiento	Tipo	Default Value (Valor predeterminado)	Descripción
máximo MLModel SizeInBytes	Entero	100 000 000 bytes (100 MiB)	Rango admisible: desde 100 000 (100 KiB) hasta 2 147 483 648 (2 GiB) En función de los datos de entrada, el tamaño del modelo podría afectar al desempeño.
sgd.maxPasses	Entero	10	Rango admisible: desde 1 hasta 100
sgd.shuffleType	Cadena	auto	Valores admisibles: `auto` o `none`
sgd.l1 RegularizationAmount	Doble	0 (de forma predeterminada no se utiliza L1)	Rango admisible: desde 0 hasta MAX_DOUBLE Se ha observado que los valores de L1 entre 1E-4 y 1E-8 producen buenos resultados. Los valores más grandes pueden producir modelos que no son muy útiles. No puede establecer tanto L1 como L2. Debe elegir uno o el otro.
sgd.l2 RegularizationAmount	Doble	1E-6 (de forma predeterminada,	Rango admisible: desde 0 hasta MAX_DOUBLE

Parámetro de entrenamiento	Tipo	Default Value (Valor predeterminado)	Descripción
		se utiliza L2 con este nivel de regularización)	Se ha observado que los valores de L2 entre 1E-2 y 1E-6 producen buenos resultados. Los valores más grandes pueden producir modelos que no son muy útiles. No puede establecer tanto L1 como L2. Debe elegir uno o el otro.

Creación de un modelo de ML

Una vez que haya creado una fuente de datos, está listo para crear un modelo de ML. Si utiliza la consola de Amazon Machine Learning para crear un modelo, puede optar por utilizar la configuración predeterminada o personalizar el modelo mediante la aplicación de opciones personalizadas.

Entre las opciones personalizadas se incluyen:

- Configuración de la evaluación: puede hacer que Amazon ML reserve una parte de los datos de entrada para evaluar la calidad de predicción del modelo de ML. Para obtener información sobre evaluaciones, consulte la sección Evaluación de modelos de ML.
- Receta: una receta indica a Amazon ML qué atributos y transformaciones de atributo están disponibles para el entrenamiento de modelos. Para obtener más información sobre las recetas de Amazon ML, consulte Transformaciones de características con recetas de datos.
- Parámetros de entrenamiento: los parámetros controlan determinadas propiedades del proceso de entrenamiento y del modelo de ML resultante. Para obtener más información sobre los parámetros de entrenamiento, consulte Parámetros de entrenamiento.

Para seleccionar o especificar valores para estos ajustes, elija la opción Custom (Personalizado) cuando utilice el asistente de creación de modelos de ML. Si desea que Amazon ML aplique la configuración predeterminada, elija Predeterminado.

Al crear un modelo de ML, Amazon ML selecciona el tipo de algoritmo de aprendizaje que se utilizará en función del tipo de atributo del atributo de destino. (El atributo de destino es el atributo que

contiene las respuestas "correctas".) Si el atributo de destino es "Binario", Amazon ML crea un modelo de clasificación binaria que utiliza el algoritmo de regresión logístico. Si el atributo de destino es "Categórico", Amazon ML crea un modelo multiclase que utiliza un algoritmo de regresión logístico multinomial. Si el atributo de destino es "Numérico", Amazon ML crea un modelo de regresión que utiliza un algoritmo de regresión lineal.

Temas

- Requisitos previos
- Creación de un modelo de ML con las opciones predeterminadas
- Creación de un modelo de ML con opciones personalizadas

Requisitos previos

Antes de utilizar la consola de Amazon ML para crear un modelo de ML, debe crear dos fuentes de datos, una para entrenar el modelo y otra para evaluarlo. Si no ha creado dos fuentes de datos, consulte Paso 2: cree una fuente de datos de entrenamiento en el tutorial.

Creación de un modelo de ML con las opciones predeterminadas

Elija las opciones Predeterminado si desea que Amazon ML:

- Divida los datos de entrada para utilizar el primer 70% para el entrenamiento y el 30% restante la evaluación
- Sugiera una receta en función de las estadísticas recopiladas en la fuente de datos de entrenamiento, que es el 70% de la fuente de datos de entrada
- Elección de los parámetros de entrenamiento predeterminados

Elección de las opciones predeterminadas

1. En la consola de Amazon ML, elija Amazon Machine Learning y, a continuación, elija los modelos de machine learning.
2. En la página de resumen ML models (Modelos ML), elija Create a new ML model (Crear un nuevo modelo de ML).
3. En la página Input data (Datos de entrada), asegúrese de que está seleccionado I already created a datasource pointing to my S3 data (Siempre he creado un origen de datos que apunta a mis datos de S3).

4. En la tabla, elija el origen de datos y, a continuación, elija Continue (Continuar).
5. En la página ML model settings (Configuración de modelo de ML), en ML model name (Nombre de modelo de ML), escriba un nombre para su modelo de ML.
6. Para Training and evaluation settings (Configuración de entrenamiento y evaluación), asegúrese de que se selecciona Default (Predeterminado).
7. En Asignar nombre a esta evaluación, escriba un nombre para la evaluación y después elija Revisar. Amazon ML ignora el resto del asistente y le lleva a la página Revisar.
8. Revise los datos, elimine las etiquetas copiadas del origen de datos que no desea aplicar a su modelo y evaluaciones y, a continuación, elija Finish (Finalizar).

Creación de un modelo de ML con opciones personalizadas

Personalizar su modelo de ML le permite:

- Proporcionar su propia receta. Para obtener más información sobre cómo proporcionar su propia receta, consulte Recipe Format Reference (Referencia de formato de receta).
- Elija los parámetros de entrenamiento. Para obtener más información sobre los parámetros de entrenamiento, consulte Parámetros de entrenamiento.
- Elija una relación de la división de entrenamiento y evaluación distinta de la proporción 70/30 predeterminada o proporcione otra fuente de datos que ya haya preparado para su evaluación. Para obtener información sobre las estrategias de división, consulte División de datos.

También puede elegir los valores predeterminados para cualquiera de estos ajustes.

Si ya se ha creado un modelo utilizando las opciones predeterminadas y desea mejorar el rendimiento predictivo de su modelo, utilice la opción Custom (Personalizado) para crear un modelo nuevo con algunos ajustes personalizados. Por ejemplo, puede agregar más transformaciones de funciones a la receta o aumentar el número de iteraciones en el parámetro de entrenamiento.

Creación de un modelo con opciones personalizadas

1. En la consola de Amazon ML, elija Amazon Machine Learning y, a continuación, elija los modelos de machine learning.
2. En la página de resumen ML models (Modelos ML), elija Create a new ML model (Crear un nuevo modelo de ML).

3. Si ya ha creado un origen de datos, en la página Input data (Datos de entrada), elija I already created a datasource pointing to my S3 data (Ya he creado un origen de datos que apunta a mis datos de S3). En la tabla, elija el origen de datos y, a continuación, elija Continue (Continuar).

   Si necesita crear un origen de datos, elija My data is in S3, and I need to create a datasource (Mis datos están en S3 y necesito crear un origen de datos) y seleccione Continue (Continuar). Se le redirigirá al asistente Create a Datasource (Crear un origen de datos). Especifique si sus datos están en S3 o en Redshift y, a continuación, seleccione Verify (Verificar). Complete el procedimiento para la creación de una fuente de datos.

   Después de crear un origen de datos, se le redirigirá el siguiente paso en el asistente Create ML Model (Crear modelo de ML).
4. En la página ML model settings (Configuración de modelo de ML), en ML model name (Nombre de modelo de ML), escriba un nombre para su modelo de ML.
5. En Select training and evaluation settings (Seleccionar configuración de entrenamiento y evaluación), seleccione Custom (Personalizado) y, a continuación, elija Continue (Continuar).
6. En la página Recipe (Receta) puede customize a recipe. Si no desea personalizar una receta, Amazon ML le sugiere una. Elija Continuar.
7. En la página Advanced settings (Configuración avanzada), especifique los valores de Maximum ML model Size (Tamaño de modelo de ML máximo), Maximum number of data passes (Número máximo de pasadas de datos), Shuffle type for training data (Tipo de mezcla para datos de entrenamiento), Regularization type (Tipo de regularización) y Regularization amount (Cantidad de regularización). Si no los especifica, Amazon ML utiliza los parámetros de entrenamiento predeterminados.

   Para obtener más información sobre estos parámetros y sus opciones predeterminadas, consulte Parámetros de entrenamiento.

   Elija Continuar.
8. En la página Evaluation (Evaluación), especifique si desea evaluar el modelo de ML inmediatamente. Si no desea evaluar el modelo de ML ahora, seleccione Review (Revisar).

   Si desea evaluar el modelo de ML ahora:

   a. En Name this evaluation (Asignar nombre a esta evaluación), escriba un nombre para la evaluación.

b. En Seleccionar datos de evaluación, elija si desea que Amazon ML reserve una parte de los datos de entrada para la evaluación y, si lo hace, cómo desea dividir el origen de datos o si decide proporcionar un origen de datos diferente para la evaluación.

c. Elija Revisar.

9. En la página Review (Revisar), edite las selecciones, elimine las etiquetas copiadas del origen de datos que no desea aplicar a su modelo y evaluaciones y, a continuación, elija Finish (Finalizar).

Una vez que haya creado el modelo, consulte Paso 4: Revisar el desempeño predictivo del modelo de ML y establecer un umbral de puntuación.

Transformaciones de datos para aprendizaje automático

La calidad de los modelos de aprendizaje dependerá de la calidad de los datos que se utilizan para formarlos. Una de las principales características de los buenos datos de formación es que se proporcionan de un modo optimizado para el aprendizaje y la generalización. El proceso de elaborar los datos en este formato óptimo se conoce en el sector como transformación de características.

Temas

- Importancia de la transformación de funciones
- Transformaciones de características con recetas de datos
- Referencia del formato de recetas
- Recetas sugeridas
- Referencia de transformaciones de datos
- Reorganización de datos

Importancia de la transformación de funciones

Imagine un modelo de aprendizaje automático que decide si una transacción con tarjeta de crédito es fraudulenta o no. Basándose en la información de fondo de la aplicación y en el análisis de datos, puede decidir qué campos de datos (o funciones) son importantes para incluir en los datos de entrada. Por ejemplo, es importante proporcionar el importe de la transacción, el nombre y la dirección del comerciante y la dirección del propietario de la tarjeta de crédito al proceso de aprendizaje. Por otra parte, un ID de transacciones generadas de forma aleatoria no aporta información (si sabemos que es aleatorio) y no es útil.

Una vez que haya decidido los campos en los que las incluirá, transforme estas funciones para ayudar al proceso de aprendizaje. Las transformaciones añaden experiencia de fondo a los datos de entrada, de manera que el modelo de aprendizaje automático se puede beneficiar de esta experiencia. Por ejemplo, la siguiente dirección de comerciante está representada como cadena:

"123 Main Street, Seattle, WA 98101"

Por sí sola, la dirección tiene un poder de expresión limitado: solo es útil para patrones de aprendizaje asociados a esa dirección exacta. Sin embargo, si se divide en partes constituyentes, se pueden crear funciones adicionales como "Address" (123 Main Street), "City" (Seattle), "State" (WA) y "Zip" (98101). De este modo, el algoritmo de aprendizaje puede agrupar transacciones más

dispares y descubrir patrones más amplios; quizás algunos códigos postales de comerciantes experimenten actividades más fraudulentas que otros.

Para obtener más información sobre el enfoque y el proceso de la transformación de funciones, consulte Conceptos de aprendizaje automático.

Transformaciones de características con recetas de datos

Hay dos formas de transformar características antes de la creación de modelos de ML con Amazon ML: puede transformar los datos de entrada directamente antes de mostrarlos en Amazon ML, o bien puede utilizar las transformaciones de datos integradas de Amazon ML. Puede utilizar recetas de Amazon ML, que son instrucciones formateadas previamente para transformaciones comunes. Con las recetas, puede hacer lo siguiente:

- Elegir entre una lista de transformaciones de aprendizaje automático comunes integradas y aplicarlas a variables individuales o grupos de variables
- Seleccionar cuáles de las variables de entrada y de las transformaciones están disponibles para el proceso de aprendizaje automático

El uso de recetas de Amazon ML ofrece varias ventajas. Amazon ML se encarga de realizar las transformaciones de datos, por lo que ya no necesita implementarlas usted mismo. Además, son rápidas, ya que Amazon ML aplica las transformaciones al leer datos de entrada y ofrece resultados para el proceso de aprendizaje sin el paso intermedio de guardar resultados en disco.

Referencia del formato de recetas

Las recetas de Amazon ML contienen instrucciones para transformar los datos como parte del proceso de machine learning. Las recetas se definen utilizando una sintaxis similar a JSON, pero que tienen restricciones adicionales más allá de las restricciones JSON normales. Las recetas tienen las secciones siguientes, que debe aparecer en el orden en el que se muestran aquí:

- Los grupos permiten agrupar diversas variables para facilitar la aplicación de transformaciones. Por ejemplo, puede crear un grupo de todas las variables relacionadas con las partes de texto sin formato de una página web (título, cuerpo) y, a continuación, realizar una transformación en todas estas partes a la vez.
- Las asignaciones permiten la creación de variables con nombres intermedios que se pueden reutilizar en el procesamiento.

- Las salidas definen qué variables se utilizarán en el proceso de entrenamiento y qué transformaciones (si procede) se aplicarán a estas variables.

Grupos

Puede definir grupos de variables a fin de transformar colectivamente todas las variables dentro de los grupos o utilizar estas variables para el aprendizaje automático sin transformarlas. De forma predeterminada, Amazon ML crea los siguientes grupos:

ALL_TEXT, ALL_NUMERIC, ALL_CATEGORICAL, ALL_BINARY: grupos para tipos específicos en función de las variables definidas en el esquema del origen de datos.

> **Note**
>
> No puede crear un grupo con el grupo `ALL_INPUTS`.

Estas variables se pueden utilizar en la sección de la salida de la receta sin estar definidos. También puede crear grupos personalizados añadiendo o restando variables de grupos existentes, o directamente desde una colección de variables. En el siguiente ejemplo se muestran los tres enfoques y la sintaxis de la asignación de agrupamiento:

```
"groups": {

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

}
```

Los nombres de grupo tienen que comenzar con un carácter alfabético y puede tener entre 1 y 64 caracteres. Si el nombre de grupo no comienza con un carácter alfabético o si contiene caracteres especiales (, ' " \t \r \n ( ) \), debe estar entre comillas para que se incluya en la receta.

Asignaciones

Puede asignar una o varias transformaciones a una variable intermedia por comodidad y legibilidad. Por ejemplo, si tiene una variable de texto denominada "email_subject" y le aplica la transformación a minúscula, puede llamar a la variable resultante "email_subject_lowercase", lo que facilita la tarea

de realizar un seguimiento de ella en otras partes de la receta. Las asignaciones también pueden estar encadenadas, lo que le permite aplicar varias transformaciones en un orden especificado. El siguiente ejemplo muestra asignaciones únicas y asignaciones encadenadas en la sintaxis de la receta:

```
"assignments": {

"email_subject_lowercase": "lowercase(email_subject)",

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

}
```

Los nombres de variables intermedias tienen que comenzar con un carácter alfabético y pueden tener entre 1 y 64 caracteres. Si el nombre no comienza con un carácter alfabético o si contiene caracteres especiales (, ' " \t \r \n ( ) \), debe estar entre comillas para que se incluya en la receta.

Salidas

La sección de salidas controla qué variables de entrada se utilizarán para el proceso de aprendizaje y qué transformaciones se les aplicarán. Una sección de salida vacía o inexistente es un error, ya que no se transferirá ningún dato al proceso de aprendizaje.

La sección de salidas más simple incluye el grupo predefinido ALL_INPUTS, que indica a Amazon ML que utilice todas las variables definidas en el origen de datos para el aprendizaje:

```
"outputs": [

"ALL_INPUTS"

]
```

La sección de salida también puede consultar los demás grupos predefinidos indicando a Amazon ML que utilice todas las variables en estos grupos:

```
"outputs": [
```

```
"ALL_NUMERIC",

"ALL_CATEGORICAL"

]
```

La sección de salida también puede consultar grupos personalizados. En el siguiente ejemplo, solo uno de los grupos personalizados definidos en la sección de asignaciones de agrupamiento del ejemplo anterior se utilizará para el aprendizaje automático. Todas las demás variables quedarán descartadas:

```
"outputs": [

"All_Categorical_plus_one_other"

]
```

La sección de salidas también puede consultar las asignaciones de variables definidas en la sección de asignación:

```
"outputs": [

"email_subject_lowercase"

]
```

Las variables de entrada o las transformaciones se pueden definir directamente en la sección de salidas:

```

"outputs": [

"var1",

"lowercase(var2)"

]
```

La salida tiene que especificar de forma explícita todas las variables y las variables transformadas que se espera que estén disponibles para el proceso de aprendizaje. Por ejemplo, cuando incluye

en la salida un producto cartesiano de "var1" y "var2". Si desea incluir también las dos variables sin procesar "var1" y "var2", debe añadir las variables sin procesar en la sección de salida:

```
"outputs": [

"cartesian(var1,var2)",

"var1",

"var2"

]
```

Las salidas pueden incluir comentarios para facilitar la legibilidad añadiendo el texto del comentario junto con la variable:

```
"outputs": [

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

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

]
```

Puede combinar y asociar todos estos enfoques en la sección de salidas.

> **Note**
>
> Los comentarios no están permitidos en la consola de Amazon ML cuando se añade una receta.

Ejemplo completo de receta

El siguiente ejemplo se refiere a varios procesadores de datos integrados que se introdujeron en ejemplos anteriores:

```
{

"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"

]

}
```

Recetas sugeridas

Al crear una nueva fuente de datos en Amazon ML y al calcular las estadísticas de dicha fuente de datos, Amazon ML también creará una receta sugerida que puede utilizarse para crear un nuevo modelo de ML a partir de la fuente de datos. La fuente de datos sugerida se basa en los datos y el atributo de destino presente en los datos y proporciona un útil punto de partida para la creación y el ajuste de modelos de ML.

Para utilizar la receta sugerida en la consola de Amazon ML, elija Datasource (Origen de datos) o Datasource and ML model (Origen de datos y modelo de ML) en la lista desplegable Create new (Crear nuevo). Para los ajustes del modelo de ML, podrá elegir entre los ajustes Default (Predeterminado) o Custom Training y Evaluation (Entrenamiento y evaluación personalizados) en el paso ML Model Settings (Configuración de modelo de ML) del asistente Create ML Model (Crear modelo de ML). Si elige la opción Default, Amazon ML utilizará automáticamente la receta sugerida. Si elige la opción Custom, el editor de recetas en el siguiente paso mostrará la receta sugerida y podrá verificar o modificarla según sea necesario.

> **Note**
>
> Amazon ML le permite crear una fuente de datos y, a continuación, usarla inmediatamente para crear un modelo de ML, antes de que se complete el cálculo de estadísticas. En este caso, no podrá ver la receta sugerida en la opción Custom, pero podrá avanzar después de ese paso y hacer que Amazon ML utilice la receta predeterminada para la formación de modelos.

Para usar la receta sugerida con la API de Amazon ML, puedes pasar una cadena vacía en los parámetros Recipe y RecipeUri API. No es posible recuperar la receta sugerida con la API de Amazon ML.

Referencia de transformaciones de datos

Temas

- Transformación de n-gramas
- Transformación de bigramas dispersos ortogonales (OSB)
- Transformación en minúsculas
- Eliminar la transformación de puntuación

- Transformación de discretización en cuartiles
- Transformación de normalización
- Transformación de producto cartesiana

Transformación de n-gramas

La transformación de n-gramas toma una variable de texto como entrada y produce cadenas correspondientes al deslizamiento de un recuadro de n palabras (configurables por el usuario), con lo que se generan resultados en el proceso. Por ejemplo, tomemos como ejemplo la cadena de texto "I really enjoyed reading this book".

Al especificar la transformación de n-gramas con tamaño de recuadro = 1 simplemente le ofrece todas las palabras individuales en dicha cadena:

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

Al especificar la transformación de n-gramas con un tamaño de recuadro =2, obtiene todas las combinaciones de dos palabras y todas las combinaciones de una palabra:

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

Al especificar la transformación de n-gramas con un tamaño de recuadro = 3 añadiremos las combinaciones de tres palabras a esta lista, con lo que se obtiene lo siguiente:

```
{"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"}
```

Puede solicitar n-gramas con un tamaño de 2 a 10 palabras. Los n-gramas con tamaño 1 se generan de forma implícita para todas las entradas cuyo tipo se marca como texto en el esquema de datos, por lo que no tiene que solicitarlos. Por último, tenga en cuenta que los n-gramas se generan dividiendo los datos de entrada en espacios en blanco. Esto significa que, por ejemplo, los caracteres de puntuación se considerarán una parte de los tokens de palabras: al generar n-gramas con un

recuadro de 2 para la cadena "rojo, verde, azul" se obtendrá {"rojo", "verde", "azul", "rojo, verde", "verde, azul"}. Puede utilizar el procesador de eliminación de puntuación (que se describe más adelante en este documento) para eliminar los símbolos de puntuación si no es lo que desea.

Para calcular los n-gramas de tamaño de recuadro 3 para la variable var1:

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

Transformación de bigramas dispersos ortogonales (OSB)

La transformación OSB está destinada a ayudar en el análisis de cadenas de texto y es una alternativa a la transformación de bigramas (n-gramas con tamaño de ventana 2). OSBs se generan deslizando la ventana de tamaño n sobre el texto y generando cada par de palabras que incluya la primera palabra de la ventana.

Para crear cada OSB, sus palabras constitutivas se unen mediante el carácter "_" (guion bajo) y cada token de caracteres que se ha saltado se indica añadiendo otro guion bajo en el OSB. De este modo, el OSB codifica no solo los tokens vistos dentro de un recuadro, sino también una indicación del número de tokens omitidos dentro de ese mismo recuadro.

Para ilustrarlo, consideremos la cadena «El zorro marrón rápido salta sobre el perro perezoso», y OSBs de tamaño 4. En el siguiente ejemplo se muestran las seis ventanas de cuatro palabras y las dos últimas ventanas más cortas situadas al final de la cadena, OSBs generadas a partir de cada una de ellas:

Ventana, {OSBs generada}

```
"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}
```

Los bigramas dispersos ortogonales son una alternativa a los n-gramas que pueden funcionar mejor en algunas situaciones. Si el texto contiene campos de texto largo (10 palabras o más), pruebe a ver qué opción funciona mejor. Tenga en cuenta que el concepto de campo de texto largo puede variar en función de la situación. Sin embargo, con campos de texto más grandes, se ha OSBs demostrado empíricamente que representan el texto de forma única debido al símbolo especial de omisión (el subrayado).

Puede solicitar una tamaño de recuadro de 2 a 10 para transformaciones de OSB en variables de texto de entrada.

Para calcular OSBs con un tamaño de ventana 5 para la variable var1:

"osb(var1, 5)"

Transformación en minúsculas

El procesador de transformación de minúsculas convierte entradas de texto en minúsculas. Por ejemplo, con la entrada "The Quick Brown Fox Jumps Over the Lazy Dog", el procesador devolverá "the quick brown fox jumps over the lazy dog".

Para aplicar la transformación de minúscula a la variable var1:

"lowercase(var1)"

Eliminar la transformación de puntuación

Amazon ML divide implícitamente las entradas marcadas como texto en el esquema de datos en espacios en blanco. La puntuación en la cadena termina conectando tokens de palabras, o como tokens totalmente independientes, en función de los espacios en blanco alrededor de ella. Si no es lo que se desea, se puede utilizar la transformación de eliminación de puntuación para eliminar los símbolos de puntuación de las características generadas. Por ejemplo, en la cadena "Welcome to AML - please fasten your seat-belts!", se genera implícitamente el siguiente conjunto de tokens:

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

Al aplicar el procesador de eliminación de la puntuación a esta cadena se obtiene este conjunto:

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

Tenga en cuenta que solo se eliminan los símbolos de puntuación de prefijo y sufijo. Los símbolos de puntuación que aparecen en la mitad de un token, como por ejemplo, el guion en "seat-belts", no se eliminan.

Para aplicar la eliminación de la puntuación a la variable var1:

"no_punct(var1)"

Transformación de discretización en cuartiles

El proceso de discretización en cuartiles toma dos entradas, una variable numérica y un parámetro denominado número de contenedor y da como resultado una variable categórica. El objetivo consiste en descubrir la ausencia de linealidad en la distribución de la variable agrupando valores observados de manera conjunta.

En muchos casos, la relación entre una variable numérica y el destino no es lineal (el valor de la variable numérica no aumenta ni disminuye de forma monótona con el destino). En estos casos, puede ser útil guardar la característica numérica en una característica categórica que represente distintos rangos de la característica numérica. A continuación, cada valor de característica categórica (contenedor) pueden modelarse como si tuviera su propia relación lineal con el destino. Por ejemplo, supongamos que sabe que la característica numérica continua account_age no está linealmente correlacionada con la probabilidad de comprar un libro. Puede guardar la edad en características categóricas que podrían ser capaces de captar la relación con el destino con más precisión.

El procesador de discretización en cuartiles puede utilizarse para indicar a Amazon ML que establezca n contenedores de igual tamaño en función de la distribución de todos los valores de entrada de la variable de la edad y, a continuación, que sustituya cada número con un token de texto que contenga el contenedor. La cantidad óptima de contenedores para una variable numérica depende de las características de la variable y su relación en el destino y se determina mejor con la experimentación. Amazon ML sugiere el número óptimo de contenedores para una característica numérica en función de las estadísticas de datos en la receta sugerida.

Puede solicitar entre 5 y 1000 contenedores de cuartiles para que se calculen para cualquier variable de entrada numérica.

El siguiente ejemplo muestra cómo calcular y utilizar 50 contenedores en el lugar de la variable numérica var1:

"quantile_bin(var1, 50)"

Transformación de normalización

El transformador de normalización normaliza las variables numéricas para que tengan una media de cero y una varianza de uno. La normalización de variables numéricas puede ayudar en el proceso de aprendizaje si hay diferencias de rango muy grandes entre variables numéricas, porque las variables con la mayor magnitud podrían dominar el modelo de ML, independientemente de si la característica es o no informativa con respecto al destino.

Para aplicar esta transformación a variables numéricas var1, añada esto a la receta:

normalize(var1)

Este transformador también puede tomar un grupo de variables numéricas definidas por el usuario o el grupo predefinido de todas las variables numéricas (ALL_NUMERIC) como entrada:

normalize(ALL_NUMERIC)

Nota

No es obligatorio utilizar el procesador de normalización para las variables numéricas.

Transformación de producto cartesiana

La transformación cartesiana genera permutaciones de dos o más variables de texto o de entrada categórica. Esta transformación se utiliza cuando se sospecha una interacción entre variables. Por ejemplo, pensemos en el conjunto de datos de marketing bancario que se utiliza en Tutorial: Uso de Amazon ML para predecir respuestas a una oferta de marketing. Al usar este conjunto de datos, nos gustaría predecir si una persona responde positivamente a la promoción de un banco, en función de la información económica y demográfica. Podemos sospechar que el tipo de trabajo de la persona es algo importante (quizás existe una correlación entre trabajar en determinados campos y disponer de más dinero) y también lo es el máximo nivel de educación alcanzado. Puede que también tengamos una mayor intuición de que existe una señal sólida en la interacción de estas dos variables: por ejemplo, que la promoción es especialmente idónea para clientes que sean empresarios que han obtenido un título universitario.

La transformación de producto cartesiana toma variables categóricas o texto como entrada y produce nuevas características que captan la interacción entre estas variables de entrada. Concretamente, para cada ejemplo de formación, creará una combinación de características y las añadirá como una

característica independiente. Por ejemplo, supongamos que nuestras filas de entradas simplificadas tienen este aspecto:

objetivo, formación, trabajo

0, título.universitario, técnico

0, educación.secundaria, servicios

1, título.universitario, administración

Si se especifica que la transformación cartesiana se debe aplicar a los campos de trabajo y educación de variables categóricas, la característica resultante educación_trabajo_interacción tendrá este aspecto:

objetivo, educación_trabajo_interacción

0, título.universitario_técnico

0, educación.secundaria_servicios

1, título.universitario_administración

La transformación cartesiana es incluso más potente cuando se trata de trabajar en secuencias de tokens, como ocurre cuando uno de sus argumentos es una variable de texto que se divide de forma implícita o explícita en tokens. Por ejemplo, tomemos el caso de la tarea de clasificar un libro como un libro de texto o no. De forma intuitiva, podríamos pensar que hay algo sobre el título del libro que nos indica que es un libro de texto (determinadas palabras pueden aparecer con mayor frecuencia en los títulos de los libros de texto) y también podríamos pensar que existe algo acerca de la tapa del libro que es predictivo (los libros de texto tienen más posibilidades de tener tapa dura), pero realmente es la combinación de algunas palabras en el título y la tapa lo que resulta más predictivo. Para ofrecer un ejemplo real, la siguiente tabla muestra los resultados de aplicar el procesador cartesiano a las variables de entrada de tapa y título:

Libro de texto	Título	Tapa	Producto cartesiano de no_punct(Title) y tapa
1	Economics : Principles, Problems, Policies	Tapa dura	{"Economics_Hardcover", "Principles_Hardcover", "Problems_Hardcover", "Policies_Hardcover"}

Libro de texto	Título	Tapa	Producto cartesiano de no_punct(Title) y tapa
0	The Invisible Heart: An Economics Romance	Tapa blanda	{"The_Softcover", "Invisible_Softcover", "Heart_Softcover", "An_Softcover", "Economics_Softcover", "Romance_Softcover"}
0	Fun With Problems	Tapa blanda	{"Fun_Softcover", "With_Softcover", "Problems_Softcover"}

El siguiente ejemplo muestra cómo aplicar el transformador cartesiano a var1 y var2:

cartesian(var1, var2)

Reorganización de datos

La funcionalidad de reorganización de datos le permite crear una fuente de datos que se basa solo en una parte de los datos de entrada a los que señala. Por ejemplo, si crea un modelo de ML utilizando el asistente Crear modelo de ML en la consola de Amazon ML y elige la opción de evaluación predeterminada, Amazon ML reserva automáticamente el 30% de los datos para la evaluación de modelos de ML y utiliza el 70% restante para el entrenamiento. Esta funcionalidad se habilita a través de la característica de Reorganización de datos de Amazon ML.

Si utiliza la API de Amazon ML para crear fuentes de datos, puede especificar en qué parte de los datos de entrada se basará una nueva fuente de datos. Para ello, pase las instrucciones del `DataRearrangement` parámetro a`CreateDataSourceFromS3`, `CreateDataSourceFromRedshift` o. `CreateDataSourceFromRDS` APIs El contenido de la DataRearrangement cadena es una cadena JSON que contiene las ubicaciones inicial y final de los datos, expresadas en porcentajes, un indicador de complemento y una estrategia de división. Por ejemplo, la siguiente DataRearrangement cadena especifica que el primer 70% de los datos se utilizará para crear la fuente de datos:

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

```
        "strategy": "sequential"
    }
}
```

DataRearrangement Parámetros

Para cambiar el modo en que Amazon ML crea una fuente de datos, utilice los siguientes parámetros.

PercentBegin (Opcional)

Utilice `percentBegin` para indicar dónde comienzan los datos para la fuente de datos. Si no incluye `percentBegin` y `percentEnd`, Amazon ML incluye todos los datos al crear la fuente de datos.

Los valores válidos son `0` a `100`, ambos incluidos.

PercentEnd (Opcional)

Utilice `percentEnd` para indicar dónde acaban los datos para la fuente de datos. Si no incluye `percentBegin` y `percentEnd`, Amazon ML incluye todos los datos al crear la fuente de datos.

Los valores válidos son `0` a `100`, ambos incluidos.

Complement (opcional)

El parámetro `complement` indica a Amazon ML que utilice los datos que no se incluyen en el rango de `percentBegin` a `percentEnd` para crear un origen de datos. El parámetro `complement` es útil si necesita crear fuentes de datos complementarias para formación y evaluación. Para crear una fuente de datos complementaria, utilice los mismos valores para `percentBegin` y `percentEnd`, junto con el parámetro `complement`.

Por ejemplo, las siguientes dos fuentes de datos no comparten ningún dato y se pueden utilizar para formar y evaluar un modelo. La primera fuente de datos tiene un 25 por ciento de los datos y la segunda el 75 por ciento de los datos.

Origen de datos para evaluación:

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

```
    }
}
```

Origen de datos para entrenamiento:

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

Los valores válidos son `true` y `false`.

Strategy (opcional)

Para cambiar cómo divide Amazon ML los datos de una fuente de datos, utilice el parámetro `strategy`.

El valor predeterminado para el parámetro `strategy` es `sequential`, lo que significa que Amazon ML toma todos los registros de datos entre los parámetros `percentBegin` y `percentEnd` del origen de datos, en el orden en el que aparecen los registros en los datos de entrada

Las siguientes dos líneas de `DataRearrangement` son ejemplos de fuentes de datos de formación y evaluación ordenadas de forma secuencial:

Fuente de datos para evaluación: `{"splitting":{"percentBegin":70, "percentEnd":100, "strategy":"sequential"}}`

Fuente de datos para formación: `{"splitting":{"percentBegin":70, "percentEnd":100, "strategy":"sequential", "complement":"true"}}`

Para crear una fuente de datos a partir de una selección aleatoria de los datos, defina el parámetro `strategy` en `random` y proporcione una cadena que se utilice como valor de inicio para la división aleatoria de los datos (por ejemplo, puede utilizar la ruta de S3 a los datos como la cadena de origen aleatoria). Si elige la estrategia de división aleatoria, Amazon ML asigna a cada fila de datos un número pseudoaleatorio y, a continuación, selecciona las filas que tienen un número asignado entre `percentBegin` y `percentEnd`. Los números pseudoaleatorios se asignan utilizando el desplazamiento en byte como inicio, por lo que se cambian los resultados

de los datos en una división diferente. Se conserva cualquier orden existente. La estrategia de la división aleatoria garantiza que las variables en los datos de formación y evaluación se distribuyen de forma similar. Es útil en los casos en los que los datos de entrada pueden tener un orden implícito, que de otro modo haría que las fuentes de datos de formación y evaluación tuvieran registros de datos no similares.

Las siguientes dos líneas de `DataRearrangement` son ejemplos de fuentes de datos de entrenamiento y evaluación ordenadas de forma no secuencial:

Origen de datos para evaluación:

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

Origen de datos para entrenamiento:

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

Los valores válidos son `sequential` y `random`.

Estrategia (opcional): RandomSeed

Amazon ML utiliza randomSeed para dividir los datos. El inicio predeterminado para la API es una cadena vacía. Para especificar un inicio para la estrategia de división aleatoria, transfiérala

en una cadena. Para obtener más información sobre las semillas aleatorias, consulte División aleatoria de datos en la Guía para desarrolladores de Amazon Machine Learning.

Para obtener el código de muestra que indica cómo utilizar la validación con Amazon ML visite Github Machine Learning.

Evaluación de modelos de ML

Siempre debe evaluar un modelo para determinar si realizará un buen trabajo de predicción para nuevos y futuros datos de destino. Dado que las futuras instancias tienen valores de destino desconocidos, debe comprobar la métrica de precisión del modelo de ML en relación con los datos de los que ya sabe la respuesta de destino y utilizar esta comprobación como proxy de precisión predictiva para futuros datos.

Para poder evaluar un modelo correctamente, separe una muestra de datos que han sido etiquetados con el destino (dato real) de la fuente de datos de entrenamiento. Evaluar la precisión predictiva de un modelo de ML con los mismos datos que se han utilizado para el entrenamiento no es útil, ya que compensa a los modelos que pueden "recordar" los datos de entrenamiento en lugar de generalizar. Una vez que haya entrenado el modelo de ML, envíe al modelo las observaciones separadas para las que conoce los valores de destino. A continuación, compare las predicciones devueltas por el modelo de ML con el valor de destino conocido. Por último, genere una métrica de resumen que indique la efectividad de la predicción y la coincidencia de valores reales.

En Amazon ML, evalúe un modelo de ML mediante la creación de una evaluación. Para realizar una evaluación de un modelo de ML, necesita el modelo de ML que desea evaluar, y también datos etiquetados que no se hayan utilizado para el entrenamiento. En primer lugar, cree una fuente de datos para la evaluación mediante la creación de una fuente de datos de Amazon ML con los datos separados. Los datos utilizados en la evaluación deben tener el mismo esquema que los utilizados en el entrenamiento e incluir valores reales para la variable de destino.

En el caso de que todos los datos se encuentren en un único archivo o directorio, puede utilizar la consola de Amazon ML para dividirlos. La ruta predeterminada del asistente para la creación de un modelo de ML divide la fuente de datos de entrada y utiliza el primer 70 % para una fuente de datos de entrenamiento y el 30 % restante para una fuente de datos de evaluación. También puede personalizar las proporciones de la división con la opción Custom (Personalizado) del asistente para la creación de un modelo de ML. Esta opción le permitirá seleccionar una muestra aleatoria del 70 % para el entrenamiento y utilizar el 30 % restante para la evaluación. Para especificar proporciones de división personalizadas, utilice la cadena de reorganización de datos de la API Create Datasource. Una vez que tenga una fuente de datos de evaluación y un modelo de ML, podrá realizar una evaluación y revisarlos resultados obtenidos.

Temas

- Información sobre el modelo de ML

- Información sobre modelos binarios
- Información del modelo multiclase
- Informaciones sobre el modelo de regresión
- Prevención del sobreajuste
- Validación cruzada
- Alertas de evaluación

Información sobre el modelo de ML

Al evaluar un modelo de ML, Amazon ML proporciona una métrica estándar del sector y una serie de información para revisar la precisión predictiva del modelo. En Amazon ML, el resultado de una evaluación contiene lo siguiente:

- Una métrica de la precisión de la predicción que informa sobre el éxito general del modelo
- Visualizaciones para ayudarle a explorar la precisión de su modelo más allá de la métrica de precisión de la predicción
- La posibilidad de revisar el impacto de establecer un umbral de puntuación (solo para clasificación binaria)
- Alertas en los criterios para comprobar la validez de la evaluación

La elección de la métrica y la visualización depende del tipo de modelo de ML que está evaluando. Es importante revisar estas visualizaciones para decidir si su modelo está funcionando lo suficientemente bien como para responder a los requisitos de su negocio.

Información sobre modelos binarios

Interpretación de las predicciones

El resultado real de muchos algoritmos de clasificación binaria es una puntuación de predicción. La puntuación indica la certeza del sistema de que la observación determinada pertenece a la clase positiva (el valor objetivo real es de 1). Los modelos de clasificación binaria en Amazon ML producen una puntuación que oscila entre 0 y 1. Como consumidor de esta puntuación, para tomar la decisión sobre si la observación debe clasificarse como 1 o 0, interpreta la puntuación seleccionando un umbral de clasificación o de corte y compara la puntuación con dicho umbral. Cualquier observación

con puntuaciones superiores al valor de corte se predice como objetivo = 1 y las puntuaciones inferiores al corte, se predicen como objetivo = 0.

En Amazon ML, el corte de puntuación predeterminado es 0,5. Puede decidir actualizar este límite para que se ajuste a sus necesidades de negocio. Puede utilizar las visualizaciones en la consola para comprender cómo afectará la selección de corte a su aplicación.

Medición de la precisión del modelo de ML

Amazon ML proporciona una métrica de precisión estándar del sector para modelos de clasificación binaria denominada Area Under the (Receiver Operating Characteristic) Curve (AUC). AUC mide la capacidad del modelo de predecir una mayor puntuación para ejemplos positivos en comparación con ejemplos negativos. Dado que es independiente del corte de puntuación, puede hacerse una idea de la precisión de la predicción de su modelo a partir de la métrica de AUC sin elegir un umbral.

La métrica AUC devuelve un valor decimal comprendido entre 0 y 1. Los valores de AUC próximos a 1 indican un modelo de aprendizaje automático muy preciso. Los valores cercanos a 0,5 indican un modelo de ML que no es mejor que hacer una suposición al azar. No es habitual ver valores cercanos a 0 y suelen indicar un problema con los datos. Básicamente, una AUC cercana a 0 indica que el modelo de ML ha aprendido los patrones correctos, pero que los utiliza para realizar predicciones contrarios a la realidad (Los '0' se predicen como '1' y viceversa). Para obtener más información acerca de la AUC, vaya a la página de Curva ROC en Wikipedia.

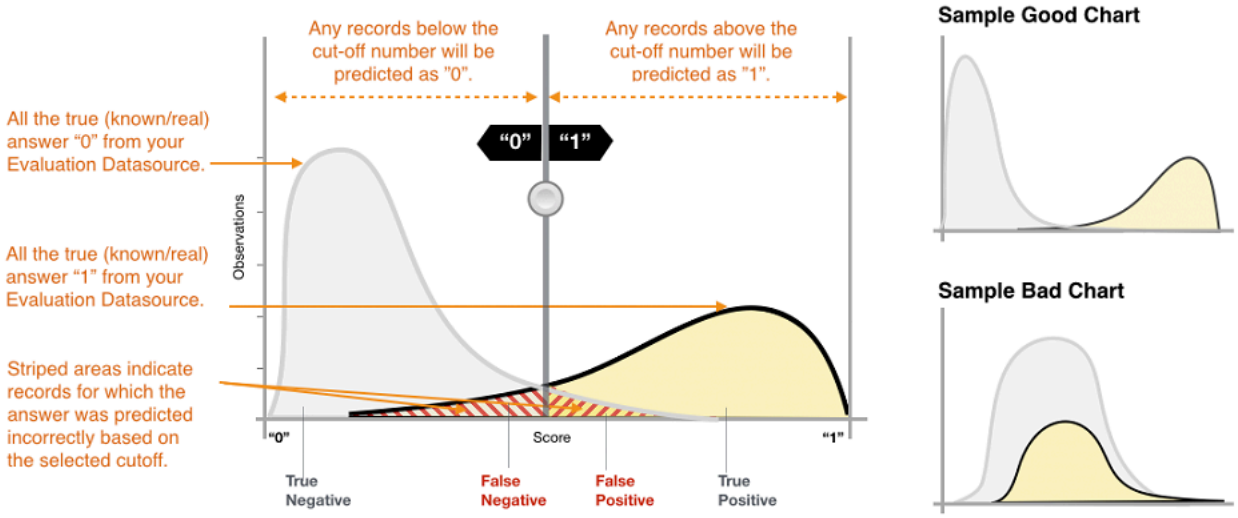
La métrica de AUC de referencia para un modelo binario es 0,5. Es el valor de un modelo de ML hipotético que predice aleatoriamente una respuesta de 1 o 0. El modelo de ML binario debe funcionar mejor que este valor para comenzar a ser de utilidad.

Uso de la visualización del desempeño

Para explorar la precisión del modelo de ML, puede revisar los gráficos en la página Evaluación en la consola de Amazon ML. Esta página muestra dos histogramas: a) un histograma de las puntuaciones de los positivos reales (el objetivo es 1) y b) un histograma de las puntuaciones de negativos reales (el objetivo es 0) en los datos de evaluación.

Un modelo de ML que tiene buena precisión predictiva predecirá puntuaciones más altas que los 1 reales y puntuaciones más bajas que los 0 reales. Un modelo perfecto tendrá los dos histogramas en dos extremos diferentes del eje x que demuestren que todos los positivos reales han obtenido puntuaciones altas y que todos los negativos reales han obtenido puntuaciones bajas. No obstante, los modelos de ML cometen errores y un gráfico típico demostrará que los dos histogramas se

solapan en determinadas puntuaciones. Un modelo de rendimiento extremadamente deficiente no podrá distinguir entre las clases positivas y negativas, y ambas clases tendrán principalmente histogramas que se solapan.



Mediante las visualizaciones, puede identificar el número de predicciones que pertenecen a los dos tipos de predicciones correctas y a los dos tipos de predicciones incorrectas.

Predicciones correctas

- Positivo real (TP): Amazon ML ha predicho el valor como 1 y el valor real es 1.
- Negativo real (TN): Amazon ML ha predicho el valor como 0 y el valor real es 0.

Predicciones erróneas

- Positivo falso (FP): Amazon ML ha predicho el valor como 1 pero el valor real es 0.
- Negativo falso (FN): Amazon ML ha predicho el valor como 0 pero el valor real es 1.

> **Note**
>
> El número de TP, TN, FP y FN depende del umbral de puntuación seleccionado y la optimización de cualquiera de uno de estos números supondría realizar una compensación en los demás. Un número alto de TPs normalmente da como resultado un número alto FPs y un número bajo de TNs.

Ajuste del corte de la puntuación

Los modelos de ML funcionan generando puntuaciones de predicciones numéricas y, a continuación, aplicando un corte para convertir estas puntuaciones en etiquetas 0/1 binarias. Al cambiar el corte de puntuación, puede ajustar el comportamiento del modelo cuando comete un error. En la página Evaluación de la consola de Amazon ML, puede comprobar el impacto de distintos cortes de puntuación y puede guardar el corte de puntuación que desee utilizar para su modelo.

Al ajustar el umbral de corte de puntuación, observe la compensación entre los dos tipos de errores. Al desplazar el corte a la izquierda, capta más positivos reales, pero la compensación produce un aumento del número de errores positivos falsos. Al desplazarlo a la derecha, capta menos errores positivos falsos, pero a cambio no captará algunos positivos verdaderos. Para su aplicación predictiva, usted decide qué tipo de error es más tolerable seleccionando una puntuación de corte adecuada.

Revisión de la métrica avanzada

Amazon ML ofrece las siguientes métricas adicionales para medir la precisión predictiva del modelo de ML: exactitud, precisión, exhaustividad y tasa de falsos positivos.

Exactitud

La exactitud (ACC) mide la fracción de predicciones correctas. El rango va de 0 a 1. Un mayor valor indica una mayor exactitud predictiva:

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

Precisión

La precisión mide la fracción de positivos reales entre los ejemplos que se prevén como positivos. El rango va de 0 a 1. Un mayor valor indica una mayor exactitud predictiva:

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

Exhaustividad

La exhaustividad mide la fracción de positivos reales que se prevén como positivos. El rango va de 0 a 1. Un mayor valor indica una mayor exactitud predictiva:

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

Tasa de positivos falsos

La tasa de positivos falsos (FPR) mide la tasa de alarma falsa o la fracción de los negativos reales que se prevén como positivos. El rango va de 0 a 1. Un valor bajo indica una mayor exactitud predictiva:

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

En función del problema de su negocio, puede que le interese más un modelo que funcione bien para un subconjunto concreto de estas métricas. Por ejemplo, dos aplicaciones de negocio podrían tener requisitos muy diferentes para su modelos de ML:

- Una aplicación podría necesitar estar muy segura de que las predicciones positivas sean realmente positivas (alta precisión) y podría permitirse la clasificación incorrecta de algunos ejemplos positivos como negativos (exhaustividad moderada).
- Otra aplicación podría necesitar predecir correctamente el mayor número de ejemplos positivos posible (exhaustividad elevada) y aceptaría que algunos ejemplos negativos se clasifiquen incorrectamente como positivos (precisión moderada).

Amazon ML le permite elegir un corte de puntuación que se corresponda con un valor determinado de cualquiera de las métricas avanzadas anteriores. También muestra las compensaciones que se generan con la optimización de cualquier métrica. Por ejemplo, si selecciona un corte que se corresponde con una alta precisión, normalmente tendrá que compensarlo con una menor exhaustividad.

> **Note**
>
> Debe guardar el corte de puntuación para que surta efecto al clasificar las futuras predicciones realizadas por modelo de ML.

Información del modelo multiclase

Interpretación de las predicciones

La salida real de un algoritmo de clasificación multiclase es un conjunto de puntuaciones de predicción. Las puntuaciones indican la certeza del modelo de que la observación dada pertenezca

a cada una de las clases. A diferencia de los problemas de clasificación binaria, no tiene que elegir una puntuación de corte para realizar predicciones. La respuesta predicha es la clase (por ejemplo, etiqueta) con la puntuación máxima predicha.

Medición de la precisión del modelo de ML

Las métricas típicas utilizadas en la multiclase son las mismas que las utilizadas en el caso de la clasificación binaria después de calcular su promedio en todas las clases. En Amazon ML, la puntuación F1 de macropromedio se utiliza para evaluar la exactitud predictiva de una métrica multiclase.

Puntuación F1 de macropromedio

La puntuación F1 es una métrica de clasificación binaria que considera las métricas binarias de precisión y exhaustividad. Es la media armónica entre la precisión y la exhaustividad. El rango va de 0 a 1. Un mayor valor indica una mayor exactitud predictiva:

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

La puntuación F1 de macropromedio es el promedio no ponderado de la puntuación F1 en todas las clases del caso multiclase. No tiene en cuenta la frecuencia de aparición de las clases en conjunto de datos de evaluación. Un valor mayor indica mejor exactitud predictiva. El siguiente ejemplo muestra K clases en la fuente de datos de evaluación:

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

Puntuación F1 de macropromedio de referencia

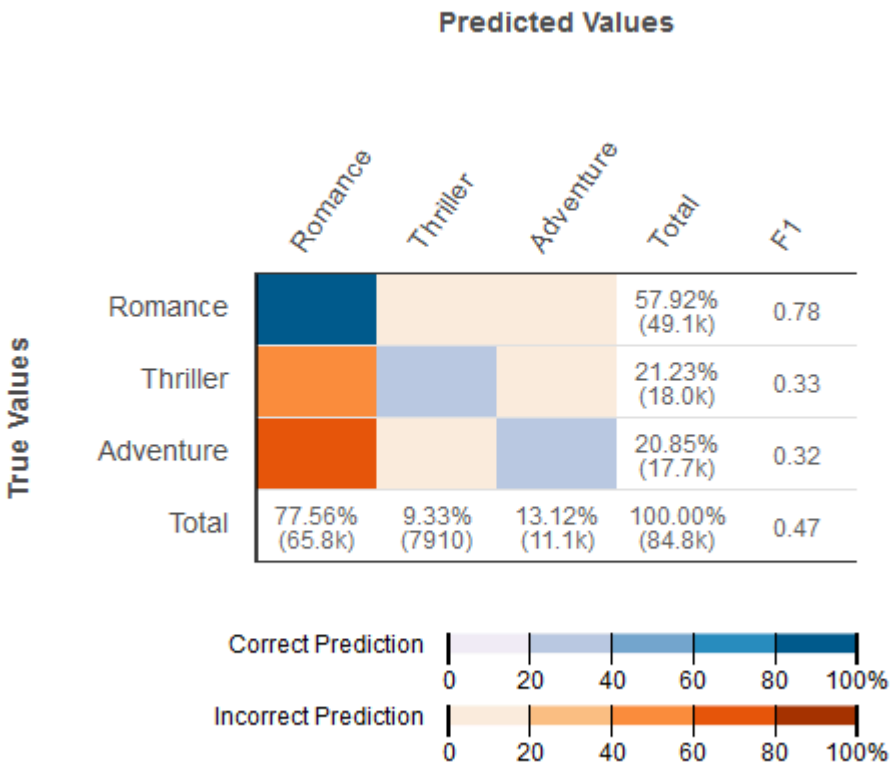
Amazon ML proporciona una métrica de referencia para modelos multiclase. Es la puntuación F1 de macropromedio para un modelo multiclase hipotético cuya respuesta sería siempre la clase más frecuente. Por ejemplo, si quisiera predecir el género de una película y el género más común en sus datos de entrenamiento fuera "Romance", el modelo de referencia siempre predeciría el género como "Romance". Compararía su modelo de ML con esta referencia para validar si el modelo de ML es mejor que un modelo de ML que predice esta respuesta constante.

Uso de la visualización del desempeño

Amazon ML proporciona una matriz de confusión como un mecanismo para visualizar la exactitud de los modelos predictivos de clasificación multiclase. La matriz de confusión ilustra en una tabla el

número o el porcentaje de predicciones correctas e incorrectas para cada clase comparando una clase predicha en la observación y su clase verdadera.

Por ejemplo, si está intentando clasificar una película en un género, el modelo predictivo podría predecir que su género (clase) es "Romance". Sin embargo, su género verdadero podría ser "Thriller". Al evaluar la exactitud de un modelo de ML de clasificación multiclase, Amazon ML identifica estos errores de clasificación y muestra los resultados en la matriz de confusión, tal y como se muestra en la siguiente ilustración.



La información siguiente se muestra en una matriz de confusión:

- Número de predicciones correctas e incorrectas para cada clase: cada fila de la matriz de confusión corresponde a las métricas para una de las clases verdaderas. Por ejemplo, en la primera fila se muestra que, para películas que realmente son del género "Romance", el modelo de ML multiclase obtiene las predicciones correctas para más del 80% de los casos. Predice incorrectamente el género como "Thriller" para menos del 20% de los casos y "Adventure" para menos del 20% de los casos.
- Puntuación F1 en clases: la última columna muestra la puntuación F1 para cada una de las clases.

- Frecuencias de clase verdaderas en los datos de evaluación: a partir de la segunda columna y hasta la última se muestra que, en el conjunto de datos de evaluación, el 57,92% de las observaciones en los datos de evaluación son "Romance", el 21,23% son "Thriller" y el 20,85% son "Adventure".
- Frecuencias de clase previstas para los datos de evaluación: la última fila muestra la frecuencia de cada clase en las predicciones. El 77,56% de las observaciones se predice como Romance, el 9,33% se predice como Thriller y el 13,12% se predice como Aventura.

La consola de Amazon ML ofrece una representación visual que permite alojar hasta 10 clases en la matriz de confusión, ordenadas de clase más frecuente a menos frecuente en los datos de evaluación. Si los datos de evaluación tienen más de 10 clases, verá las 9 clases que ocurren más frecuentemente en la matriz de confusión y todas las demás clases se juntarán en una clase denominada "otras". Amazon ML también ofrece la posibilidad de descargar la matriz de confusión a través de un enlace en la página de visualizaciones multiclase.

Informaciones sobre el modelo de regresión

Interpretación de las predicciones

La salida de un modelo de regresión de ML es un valor numérico para la predicción del destino que hace el modelo. Por ejemplo, si prevé precios inmobiliarios, la predicción del modelo podría ser un valor como 254 013.

> **Note**
>
> El rango de las predicciones puede ser diferente del rango del destino en los datos de entrenamiento. Por ejemplo, supongamos que está prediciendo precios inmobiliarios y que el destino en los datos de entrenamiento tenía valores con un rango de 0 a 450 000. El destino predicho no tiene por qué encontrarse en el mismo rango y puede tener cualquier valor positivo (mayor que 450 000) o valor negativo (inferior a cero). Es importante planificar cómo afrontar los valores de predicciones que superen un rango aceptable para su aplicación.

Medición de la precisión del modelo de ML

Para las tareas de regresión, Amazon ML utiliza la métrica estándar en el sector, la desviación cuadrática media o, en inglés, "root mean square error" (RMSE). Es una medida de distancia entre

el destino numérico predicho y la respuesta numérica real (dato real). Cuanto más pequeño sea el valor RMSE, mejor será la exactitud predictiva del modelo. Un modelo con unas predicciones perfectamente correctas tendría un valor RMSE de 0. El siguiente ejemplo muestra los datos de evaluación que contienen N registros:

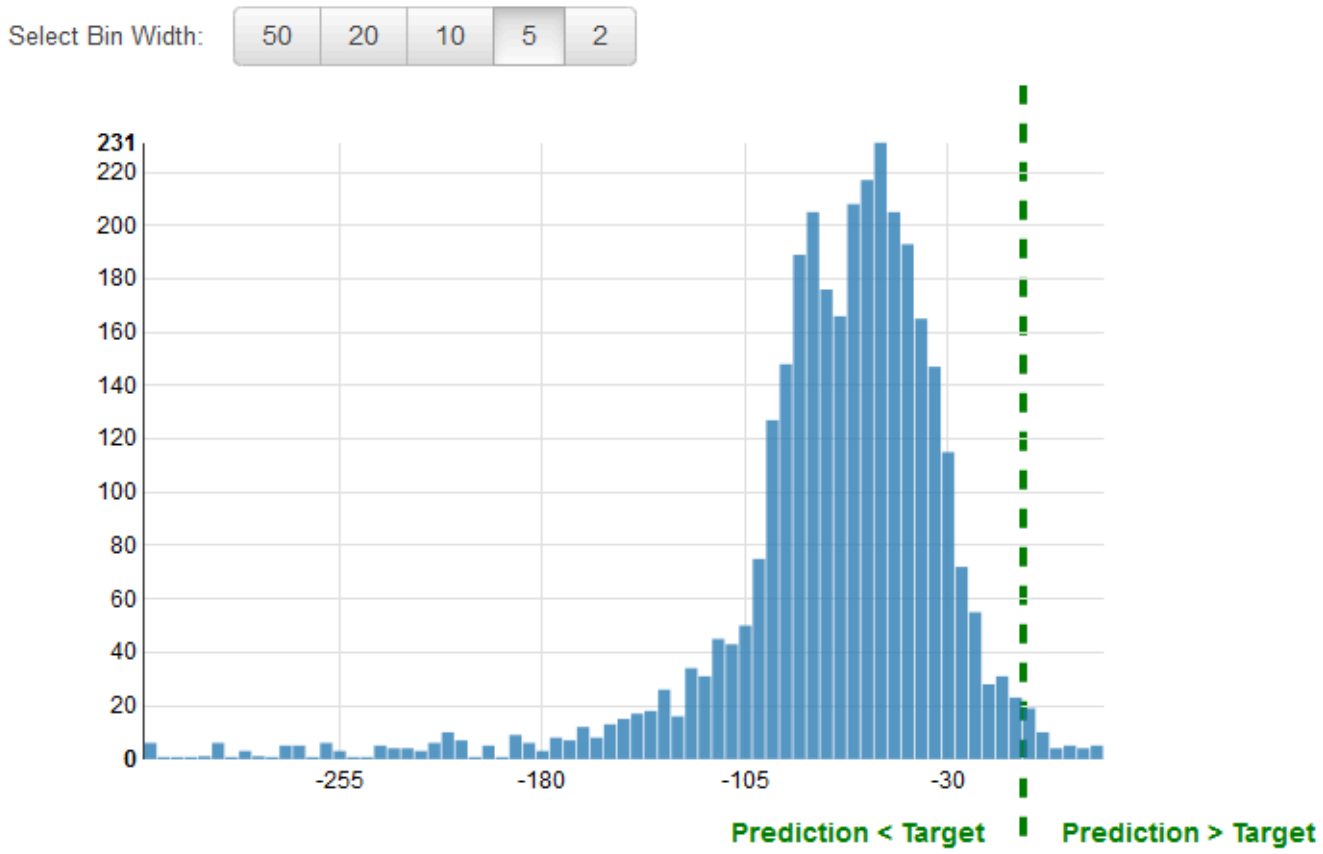
$$RMSE = \sqrt{1/N \sum_{i=1}^{N} (actual\ target - predicted\ target)^2}$$

RMSE de referencia

Amazon ML proporciona una métrica de referencia para modelos de regresión. Es el valor RMSE para un modelo de regresión hipotético que predeciría siempre la media del destino como respuesta. Por ejemplo, si estuviera prediciendo la edad de un comprador de una casa y la edad media para las observaciones en sus datos de entrenamiento fuera 35, el modelo de referencia siempre predeciría la respuesta como 35. Compararía entonces su modelo de ML con esta referencia para validar si el modelo de ML es mejor que un modelo de ML que predice esta respuesta constante.

Uso de la visualización del desempeño

Es una práctica común repasar los residuales para los problemas de regresión. Un residual para una observación en los datos de evaluación es la diferencia entre el destino verdadero y el destino predicho. Los residuales representan la parte del destino que el modelo no puede predecir. Un residual positivo indica que el modelo está subestimando el destino (el destino real es mayor que el destino predicho). Un residual negativo indica una sobreestimación (el destino real es menor que el destino predicho). El histograma de los residuales en los datos de evaluación distribuido en forma de campana y centrado en el cero indica que el modelo comete errores de forma aleatoria y no subestima o sobreestima sistemáticamente un rango determinado de valores de destino. Si los residuales no dibujan una forma de campana centrada en el cero, existe una estructura en el error de predicción del modelo. Añadir más variables al modelo podría ayudar a este a capturar el patrón que no captura el modelo actual. En la siguiente ilustración aparecen los residuales que no están centrados en el cero.



Prevención del sobreajuste

Al crear y entrenar un modelo de ML, el objetivo es seleccionar el modelo que realice las mejores predicciones, lo cual significa seleccionar el modelo con la mejor configuración (configuración o hiperparámetros del modelo de ML). En Amazon Machine Learning, pueden establecerse cuatro hiperparámetros: número de iteraciones, regularización, tamaño de modelo y tipo de reorganización. Sin embargo, si selecciona la configuración de parámetros del modelo que presenta el "mejor" rendimiento predictivo en los datos de evaluación, es posible que dicho modelo se sobreajuste. El sobreajuste ocurre cuando un modelo tiene patrones memorizados que aparecen en las fuentes de datos de entrenamiento y evaluación, pero falla al generalizar los patrones de los datos. Esto ocurre a menudo cuando los datos de entrenamiento incluyen todos los datos utilizados en la evaluación. Un modelo sobreajustado es efectivo durante las evaluaciones, pero falla en la elaboración de predicciones precisas sobre datos no vistos anteriormente.

Para evitar seleccionar un modelo sobreajustado como mejor modelo, puede reservar datos adicionales para validar el rendimiento del modelo de ML. Por ejemplo, puede dividir los datos en un 60 % para el entrenamiento, un 20 % para la evaluación y un 20 % adicional para la validación.

Después de seleccionar los parámetros del modelo que funcionan bien para los datos de evaluación, ejecute una segunda evaluación con los datos de validación para ver el rendimiento de ML en la validación de los datos. Si el modelo cumple sus expectativas en relación con los datos de validación, entonces el modelo no estará sobreajustando a los datos.

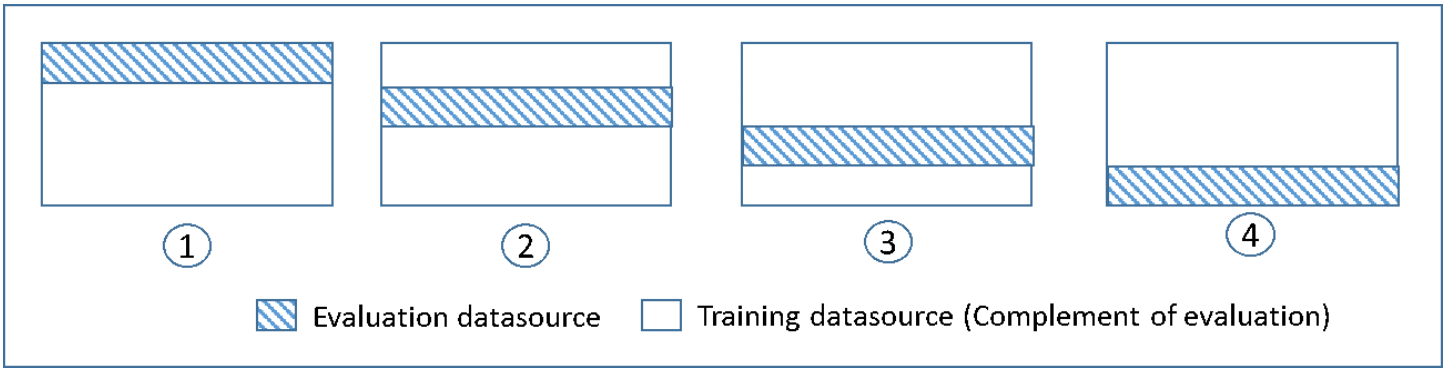
El uso de un tercer conjunto de datos para la validación permite seleccionar los parámetros del modelo de ML adecuados para evitar el sobreajuste. Sin embargo, la separación de los datos del proceso de entrenamiento tanto para la evaluación como para la validación hace que haya menos datos disponibles para el entrenamiento. Esto resulta un problema, especialmente para conjuntos de datos pequeños, porque siempre es mejor utilizar el mayor número de datos posible para el entrenamiento. Para solucionarlo, puede utilizar la validación cruzada. Para obtener más información sobre la validación cruzada, consulte Validación cruzada.

Validación cruzada

La validación cruzada es una técnica para evaluar modelos de ML mediante el entrenamiento de varios modelos de ML en subconjuntos de los datos de entrada disponibles y evaluarlos con el subconjunto complementario de los datos. Utilice la validación cruzada para detectar el sobreajuste, es decir, en aquellos casos en los que no se logre generalizar un patrón.

En Amazon ML, puede utilizar el método de la validación cruzada de K iteraciones para realizar la validación cruzada. En la validación cruzada de K iteraciones se dividen los datos de entrada en K subconjuntos de datos (también conocido como iteraciones). Puede entrenar un modelo de ML en todos menos uno (k-1) de los subconjuntos y, a continuación, evaluar el modelo en el subconjunto que no se ha utilizado para el entrenamiento. Este proceso se repite K veces, con un subconjunto diferente reservado para la evaluación (y excluido del entrenamiento) cada vez.

En el siguiente diagrama se muestra un ejemplo de los subconjuntos de entrenamiento y de los subconjuntos de evaluación complementarios generados para cada uno de los cuatro modelos que se crean y se entrenan durante una validación cruzada de 4 iteraciones. El modelo uno utiliza el primer 25% de los datos para la evaluación y el 75% restante para el entrenamiento. El modelo dos utiliza el segundo subconjunto del 25% (del 25% al 50%) para la evaluación y los tres subconjuntos restantes de los datos para el entrenamiento y así sucesivamente.



Cada modelo se entrena y se evalúa utilizando fuentes de datos complementarias: los datos de la fuente de datos incluyen y se limitan a todos los datos que no están en la fuente de datos para el entrenamiento. Las fuentes de datos para cada uno de estos subconjuntos se crean con el `DataRearrangement` parámetro en`createDatasourceFromS3`, y`createDatasourceFromRedShift`. `createDatasourceFromRDS` APIs En el parámetro `DataRearrangement`, especifique qué subconjunto de datos se incluye en una fuente de datos especificando dónde comienza y finaliza cada segmento. Para crear las fuentes de datos complementarias necesarias para la validación cruzada de 4 iteraciones, especifique el parámetro `DataRearrangement` tal y como se muestra en el ejemplo siguiente:

Modelo uno:

Origen de datos para evaluación:

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

Origen de datos para entrenamiento:

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

Modelo dos:

Origen de datos para evaluación:

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

Origen de datos para entrenamiento:

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

Modelo tres:

Origen de datos para evaluación:

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

Origen de datos para entrenamiento:

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

Modelo cuatro:

Origen de datos para evaluación:

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

Origen de datos para entrenamiento:

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

Llevar a cabo una validación cruzada de 4 iteraciones genera cuatro modelos, cuatro fuentes de datos para entrenar los modelos, cuatro fuentes de datos para evaluar los modelos y cuatro evaluaciones, una para cada modelo. Amazon ML genera una métrica de desempeño del modelo para cada evaluación. Por ejemplo, en una validación cruzada de 4 iteraciones para un problema de clasificación binaria, cada una de las evaluaciones genera una métrica de Area Under the ROC Curve (AUC). Puede obtener la medición del desempeño general calculando la media de las cuatro métricas AUC. Para obtener información sobre la métrica AUC, consulte Medición de la precisión del modelo de ML.

Para ver código de muestra que ilustra cómo crear una validación cruzada y calcular la media de las puntuaciones del modelo, consulte el código de muestra de Amazon ML.

Ajuste de los modelos

Una vez que le haya realizado una validación cruzada entre los modelos, puede ajustar la configuración del siguiente modelo si el modelo no alcanza sus estándares de rendimiento. Para obtener más información acerca del sobreajuste, consulte Ajuste del modelo: ajustes deficientes vs. ajustes excesivos. Para obtener más información acerca de la regularización, consulte

Regularización. Para obtener más información acerca de cómo realizar cambios en la configuración de la regularización, consulte Creación de un modelo de ML con opciones personalizadas.

Alertas de evaluación

Amazon ML proporciona estadísticas para ayudarle a validar si ha evaluado el modelo correctamente. Si la evaluación no cumple alguno de los criterios de validación, la consola de Amazon ML le informa mostrándole el criterio de validación que se haya infringido, tal y como se indica a continuación.

- La evaluación del modelo de ML se realiza a partir de datos separados

  Amazon ML le informa si utiliza la misma fuente de datos para el entrenamiento y para la evaluación. Si utiliza Amazon ML para dividir los datos, cumplirá este criterio de validez. Si no utiliza Amazon ML para dividir los datos, asegúrese de evaluar el modelo de ML con una fuente de datos que no sea la fuente de datos de entrenamiento.
- Se utilizaron suficientes datos para la evaluación del modelo de predicción

  Amazon ML le informa si el número de observaciones o registros de la evaluación de los datos es inferior al 10% del número de observaciones que tiene en la fuente de datos de entrenamiento. Para poder evaluar el modelo, es importante proporcionar una muestra de datos suficientemente grande. Este criterio proporciona una comprobación para informarle de si no está utilizando suficientes datos. La cantidad de datos necesaria para evaluar su modelo de machine learning es subjetiva. Aquí se selecciona el 10% como medida provisional a falta de una medida mejor.
- Esquema asignado

  Amazon ML le informa si el esquema de la fuente de datos para el entrenamiento y para la evaluación no son iguales. Si tiene determinados atributos que no existen en la evaluación de datos o si tiene atributos adicionales, Amazon ML muestra esta alerta.
- Todos los registros de archivos de evaluación utilizados para la evaluación del rendimiento de modelo predictivo

  Es importante saber si todos los registros previstos de evaluación se utilizan para evaluar el modelo. Amazon ML le informa si algunos registros en la evaluación de datos no son válidos y no se incluyen en el cálculo de las métricas de precisión. Por ejemplo, si falta la variable de destino para algunas de las observaciones en la evaluación de datos, Amazon ML no podrá comprobar si las predicciones del modelo de ML son correctas. En este caso, los registros a los que les falten valores de destino se considerarán no válidos.

- Distribución de variables de destino

  Amazon ML le muestra la distribución del atributo de destino de las fuentes de datos del entrenamiento y la evaluación para que pueda revisar si el destino se distribuye igualmente en ambas fuentes de datos. En caso de que el modelo se entrenara con datos de entrenamiento con una distribución de destino diferente a la distribución de destino de los datos de la evaluación, la calidad de la evaluación podría verse afectada puesto que se calcularía basándose en datos con estadísticas muy dispares. Es mejor distribuir los datos de forma similar entre el entrenamiento y la evaluación, para que estos conjuntos de datos imiten en la máxima medida posible los datos que el modelo encontrará a la hora de realizar predicciones.

  Si esta alerta se activa, pruebe a aplicar la estrategia de división aleatoria para dividir los datos entre fuentes de datos de formación y evaluación. En casos excepcionales, esta alerta podría avisarle erróneamente de las diferencias de la distribución de destino, a pesar de dividir los datos de forma aleatoria. Amazon ML utiliza estadísticas de datos aproximadas para evaluar la distribución de los datos, generando a veces esta alerta por error.

Creación e interpretación de predicciones

Amazon ML proporciona dos mecanismos para generar predicciones: asíncrono (basado en lotes) y sincrónico (). one-at-a-time

Utilice predicciones asíncronas o predicciones en lote, cuando tenga una serie de observaciones y desea obtener predicciones para todas las observaciones a la vez. El proceso utiliza una fuente de datos como entrada y genera predicciones en un archivo.csv que se almacena en el depósito de S3 que desee. Debe esperar hasta que se complete el proceso de predicciones en lote antes de poder acceder a los resultados de las predicciones. El tamaño máximo de una fuente de datos que puede procesar Amazon ML en lote en un archivo es de 1 TB (aproximadamente 100 millones de registros). Si la fuente de datos supera 1 TB, su proceso fallará y Amazon ML devolverá un código de error. Para evitarlo, divida los datos en varios lotes. Si sus registros acostumbran a ser más largos, alcanzará el límite de 1 TB antes de que se procesen 100 millones de registros. En este caso, le recomendamos que se ponga en contacto con AWS Support para aumentar el tamaño del proceso de predicciones en lote.

Utilice predicciones sincrónicas o predicciones en tiempo real cuando desee obtener predicciones de baja latencia. La API de predicciones en tiempo real acepta una sola observación de entrada serializada como cadena JSON, y devuelve, de forma sincrónica, la predicción y los metadatos asociados como parte de la respuesta de la API. Al mismo tiempo, puede invocar la API más de una vez para obtener predicciones sincrónicas en paralelo. Para obtener más información sobre los límites de rendimiento de la API de predicciones en tiempo real, consulte los límites de predicciones en tiempo real en las referencias de la API de Amazon ML.

Temas

- Creación de una predicción por lotes
- Revisión de métricas de predicciones por lotes
- Lectura de archivos de salida de predicciones por lotes
- Solicitud de predicciones en tiempo real

Creación de una predicción por lotes

Para crear una predicción por lotes, se crea un objeto `BatchPrediction` a través de la consola Amazon Machine Learning (Amazon ML) o la API. Un objeto `BatchPrediction` se describe como

un conjunto de predicciones que genera Amazon ML mediante su modelo de ML y un conjunto de observaciones de entrada. Al crear un objeto `BatchPrediction`, Amazon ML inicia un flujo de trabajo asíncrono que calcula las predicciones.

Debe utilizar el mismo esquema para la fuente de datos que utilice para obtener predicciones por lotes y la fuente de datos que utilizó para formar el modelo de ML al que consulta las predicciones. La única excepción es que la fuente de datos para una predicción por lotes no tiene por qué incluir el atributo de destino porque Amazon ML predice el destino. Si proporciona el atributo de destino, Amazon ML pasa por alto su valor.

Creación de una predicción por lotes (consola)

Para crear una predicción por lotes con la consola Amazon ML utilice el asistente Create Batch Prediction.

Para crear una predicción por lotes (consola)

1. Inicie sesión en la consola Amazon Machine Learning AWS Management Console y ábrala en https://console.aws.amazon.com/machinelearning/.
2. En el panel de Amazon ML, en Objetos, elija Crear nuevo... y, a continuación, elija Predicción por lotes.
3. Elija el modelo de Amazon ML que desee utilizar para crear la predicción por lotes.
4. Para confirmar que desea utilizar este modelo, seleccione Continue (Continuar).
5. Elija la fuente de datos para la que desee crear predicciones. La fuente de datos debe tener el mismo esquema que su modelo, aunque no tiene que incluir el atributo de destino.
6. Elija Continuar.
7. Para S3 destination (Destino de S3), escriba el nombre del bucket de S3.
8. Elija Revisar.
9. Revise la configuración y seleccione Create batch prediction (Crear predicciones por lotes).

Creación de una predicción por lotes (API)

Para crear un objeto `BatchPrediction` utilizando la API de Amazon ML, debe proporcionar los siguientes parámetros:

Datasource ID

El ID de la fuente de datos que apunta a las observaciones para las que desea las predicciones. Por ejemplo, si desea predicciones para los datos en un archivo llamado `s3://examplebucket/input.csv`, crearía un objeto de fuente de datos que apunte al archivo de datos y, a continuación, trasladaría el ID de esa fuente de datos con este parámetro.

BatchPrediction ID

El ID que se asigna a la predicción por lotes.

ML Model ID

El ID del modelo de ML que Amazon ML debe consultar para las predicciones.

Output Uri

El URI del bucket de S3 en el que almacenar el resultado de la predicción. Amazon ML debe tener permisos para escribir datos en este bucket.

El parámetro `OutputUri` debe hacer referencia a una ruta de S3 que termine con una barra inclinada ('/'), tal y como se muestra en el ejemplo siguiente:

s3://examplebucket/examplepath/

Para obtener más información acerca de la configuración de permisos de S3, consulte Concesión de permisos a Amazon ML para enviar predicciones a Amazon S3.

(Opcional) BatchPrediction Nombre

(Opcional) Un nombre legible para la predicción por lotes.

Revisión de métricas de predicciones por lotes

Después de que Amazon Machine Learning (Amazon ML) cree una predicción por lotes, proporciona dos métricas: `Records seen` y `Records failed to process`. `Records seen` le indica la cantidad de registros de Amazon ML que se han analizado cuando se ha ejecutado la predicción por lotes. `Records failed to process` le indica la cantidad de registros de Amazon ML que no se han podido procesar.

Para permitir que Amazon ML procese registros fallidos, compruebe el formato de los registros de los datos utilizados para crear la fuente de datos y asegúrese de que todos los atributos obligatorios

estén indicados y que todos los datos sean correctos. Después de reparar los datos, puede volver a crear su predicción por lotes o crear una nueva fuente de datos con los registros fallidos y, a continuación, crear una nueva predicción por lotes utilizando la nueva fuente de datos.

Revisión de métricas de predicciones por lotes (consola)

Para consultar las métricas en la consola de Amazon ML, abra la página Resumen de predicción por lotes y busque en la sección Información procesada.

Revisión de métricas e información de predicciones por lotes (API)

Puede utilizar Amazon ML APIs para recuperar detalles sobre los `BatchPrediction` objetos, incluidas las métricas de registro. Amazon ML ofrece las siguientes llamadas a la API de predicciones por lotes:

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

Para obtener más información, consulte la Referencia de las API de Amazon ML.

Lectura de archivos de salida de predicciones por lotes

Realice los siguientes pasos para recuperar los archivos de salida de predicciones por lotes:

1. Localice el archivo de manifiesto de predicciones por lotes.
2. Lea el archivo de manifiesto para determinar las ubicaciones de los archivos de salida.
3. Recupere los archivos de salida que contengan las predicciones.
4. Interprete el contenido de los archivos de salida. El contenido variará en función del tipo de modelo de ML que se utilizó para generar las predicciones.

En las siguientes secciones se describen los pasos de manera más detallada.

Localización del archivo de manifiesto de predicciones por lotes

Los archivos de manifiesto de predicciones por lotes contienen la información que une los archivos de entrada y los archivos de salida de las predicciones.

Para localizar el archivo de manifiesto, comience con la ubicación de salida especificada al crear el objeto de predicciones por lotes. Puede consultar un objeto de predicción de lotes completado para recuperar la ubicación S3 de este archivo mediante la API Amazon ML o la https://console.aws.amazon.com/machinelearning/.

El archivo de manifiesto se encuentra en la ubicación de salida, en una ruta que consta de la cadena estática `/batch-prediction/` anexada a la ubicación de salida y el nombre del archivo de manifiesto, el cual se corresponde con el ID de la predicción por lotes, con la extensión `.manifest` anexada.

Por ejemplo, si crea un objeto de predicciones por lotes con el ID `bp-example` y especifica la ubicación de S3 `s3://examplebucket/output/` como la ubicación de salida, encontrará su archivo de manifiesto aquí:

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

Lectura del archivo de manifiesto

El contenido del archivo de manifiesto está codificado como un mapa JSON, donde la clave es una cadena del nombre de un archivo de datos de entrada de S3 y el valor es una cadena del archivo asociado de resultados de predicciones por lotes. Hay una línea de asignación para cada par de archivos de entrada o salida. Siguiendo con nuestro ejemplo, si la entrada de la creación del objeto `BatchPrediction` consta de un único archivo llamado data.csv que se encuentra en `s3://examplebucket/input/`, es posible que vea una cadena de asignación que tenga este aspecto:

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

Si la entrada de la creación del objeto `BatchPrediction` consta de tres archivos llamados data1.csv, data2.csv y data3.csv y si están almacenados en la ubicación `s3://examplebucket/input/` de S3, es posible que vea una cadena de asignación que tenga este aspecto:

```
{"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"}
```

Recuperación de archivos de salida de predicciones por lotes

Puede descargar cada archivo de predicciones por lotes obtenido de la asignación del manifiesto y procesarlo a nivel local. El formato de archivo es CSV, comprimido con el algoritmo gzip. Dentro de dicho archivo, hay una línea por observación de entrada en el archivo de entrada correspondiente.

Para unir las predicciones con el archivo de entrada de la predicción por lotes, puede realizar una simple record-by-record combinación de los dos archivos. El archivo de salida de la predicción por lotes siempre contiene el mismo número de registros que el archivo de entrada de predicciones, en el mismo orden. Si una observación de entrada falla durante el procesamiento y no se puede generar ninguna predicción, el archivo de salida de la predicción por lotes tendrá una línea en blanco en la ubicación correspondiente.

Interpretación del contenido de los archivos de predicciones por lotes para un modelo de ML de clasificación binaria

Las columnas del archivo de predicciones por lotes para un modelo de clasificación binaria se llaman bestAnswer y score.

La columna bestAnswer contiene la etiqueta de predicción ("1" o "0") que se obtiene mediante la evaluación de la puntuación de la predicción frente a la puntuación de corte. Para obtener más información sobre las puntuaciones de corte, consulte Ajuste de la puntuación de corte. Puede definir una puntuación de corte para el modelo de ML mediante la API de Amazon ML o la funcionalidad de la evaluación del modelo de la consola de Amazon ML. Si no establece ninguna puntuación de corte, Amazon ML se utilizará el valor predeterminado de 0,5.

La columna puntuación contiene la puntuación de la predicción bruta asignada por el modelo de ML para esta predicción. Amazon ML utiliza modelos de regresión logística, por lo que esta puntuación intenta modelar la probabilidad de la observación que corresponde a un valor "true" ("1"). El valor de score se expresa en notación científica, por lo que en la primera fila del siguiente ejemplo el valor 8,7642E-3 es igual a 0,0087642.

Por ejemplo, si la puntuación de corte del modelo de ML es de 0,75, el contenido del archivo de salida de predicciones por lotes para un modelo de clasificación binaria puede tener el siguiente aspecto:

```
bestAnswer,score

0,8.7642E-3

1,7.899012E-1


0,6.323061E-3

0,2.143189E-2


1,8.944209E-1
```

La segunda y la quinta observaciones en el archivo de entrada han recibido puntuaciones de predicciones superiores a 0,75, por lo que la columna bestAnswer de estas observaciones indica el valor "1", mientras que otras observaciones tienen el valor "0".

Interpretación del contenido de los archivos de predicciones por lotes para un modelo de ML de clasificación multiclase

El archivo de predicciones por lotes para un modelo multiclase contiene una columna para cada clase de los datos de entrenamiento. Los nombres de columna aparecen en la línea del encabezado del archivo de predicciones por lotes.

Al solicitar predicciones desde un modelo multiclase, Amazon ML genera varias puntuaciones de predicciones para cada una de las observaciones del archivo de entrada, una para cada una de las clases definidas en el conjunto de datos de entrada. Es equivalente a la pregunta: "¿Cuál es la probabilidad (medida entre 0 y 1) de que esta observación pase a formar parte de esta clase, en lugar de pasar a formar parte de cualquiera de las otras clases?" Cada puntuación puede interpretarse como una "probabilidad de que la observación pertenezca a esta clase". Dado que las puntuaciones de predicción influyen en las probabilidades subyacentes de que la observación pertenezca a una clase u otra, la suma de todas las puntuaciones de predicción en una fila es 1. Debe elegir una clase como clase predicha para el modelo. Lo más común es que elija la clase que tenga la mayor probabilidad como mejor respuesta.

Por ejemplo, imagine que intenta predecir la calificación del cliente de un producto, basándose en una escala de 1 a 5 estrellas. Si las clases se llaman `1_star`, `2_stars`, `3_stars`, `4_stars` y `5_stars`, el archivo de salida de la predicción multiclase puede tener un aspecto similar al siguiente:

```
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
```

En este ejemplo, la primera observación tiene la mayor puntuación de predicción para la clase `3_stars` (puntuación de predicción = 4.77781E-1), por lo que interpretaría los resultados de manera que la clase `3_stars` es la mejor respuesta para esta observación. Las puntuaciones de predicción se muestran en notaciones científicas, por lo que una puntuación de predicción de 4.77781E-1 es igual a 0.477781.

Puede haber circunstancias en las que no desea elegir la clase con la mayor probabilidad. Por ejemplo, es posible que quiera establecer un umbral mínimo por debajo del cual no considerará a una clase como la mejor respuesta aunque tenga la mayor puntuación de predicción. Supongamos que está clasificando películas por géneros y que desea que la puntuación de predicción sea al menos de 5E-1 antes de determinar que un género es la mejor respuesta. Obtiene una puntuación de predicción de 3E-1 para comedias, 2.5E-1 para dramas, 2.5E-1 para documentales y 2E-1 para películas de acción. En este caso, el modelo de ML predice que la comedia es la elección más probable, pero usted decide no elegirla como la mejor respuesta. Dado que ninguna de las puntuaciones de predicción ha superado su puntuación de predicción de referencia de 5E-1, decide que la predicción no es suficiente para predecir el género de forma segura y elegir otra cosa. La aplicación podría tratar el campo del género de esta película como "unknown".

Interpretación del contenido de los archivos de predicciones por lotes para un modelo de ML de regresión

El archivo de predicciones por lotes para un modelo de regresión contiene una única columna llamada score. Esta columna contiene la predicción numérica raíz de cada observación de los

datos de entrada. Los valores se expresan en notación científica, por lo que el valor de score de -1.526385E1 es igual a -15.26835 en la primera fila del siguiente ejemplo.

Este ejemplo muestra un archivo de salida de una predicción por lotes realizada en un modelo de regresión:

```
score

-1.526385E1

-6.188034E0

-1.271108E1

-2.200578E1

8.359159E0
```

Solicitud de predicciones en tiempo real

Una predicción en tiempo real es una llamada sincrónica a Amazon Machine Learning (Amazon ML). La predicción se lleva a cabo cuando Amazon ML obtiene la solicitud y la respuesta se devuelve inmediatamente. Las predicciones en tiempo real se utilizan generalmente para habilitar capacidades predictivas en aplicaciones web, móviles o aplicaciones de escritorio. Puede consultar un modelo de ML creado con Amazon ML para obtener predicciones en tiempo real a través de la API `Predict` de baja latencia. La operación `Predict` acepta una única observación de entrada en la solicitud de carga y devuelve la predicción de la respuesta de forma síncrona. Esto la diferencia de la API de predicciones por lotes, que se invoca con el ID de un objeto de fuente de datos de Amazon ML que apunta a la ubicación de las observaciones de entrada y devuelve de forma asíncrona una URI a un archivo que contiene predicciones para todas estas observaciones. Amazon ML responde a la mayoría de solicitudes de predicciones en tiempo real en un plazo de 100 milisegundos.

Puede probar las predicciones en tiempo real sin incurrir en gastos en la consola de Amazon ML. Si decide utilizar las predicciones en tiempo real, primero debe crear un punto de enlace para la generación de predicciones en tiempo real. Puede hacerlo en la consola de Amazon ML o utilizando la API `CreateRealtimeEndpoint`. Cuando tenga un punto de enlace, utilice la API de predicciones en tiempo real para generar predicciones en tiempo real.

> **Note**
>
> Después de crear un punto de enlace en tiempo real para el modelo, podrá comenzar a incurrir en una carga de reserva de capacidad que se basa en el tamaño del modelo. Para obtener más información, consulte Precios. Si crea el punto de enlace en tiempo real en la consola, la consola mostrará un desglose de los cargos estimados que el punto de enlace acumulará de forma continua. Para parar de incurrir en cargos cuando ya no necesita obtener las predicciones en tiempo real a partir de ese modelo, elimine el punto de enlace en tiempo real a través de la consola o la operación `DeleteRealtimeEndpoint`.

Para ver ejemplos de `Predict` solicitudes y respuestas, consulte Predecir en la referencia de la API de Amazon Machine Learning. Para ver un ejemplo del formato exacto de respuesta que utiliza su modelo, consulte Pruebas con las predicciones en tiempo real.

Temas

- Pruebas con las predicciones en tiempo real
- Creación de un punto de enlace en tiempo real
- Ubicación del punto de enlace de predicciones en tiempo real (consola)
- Ubicación del punto de enlace de predicciones en tiempo real (API)
- Creación de una solicitud de predicciones en tiempo real
- Eliminación de un punto de enlace en tiempo real

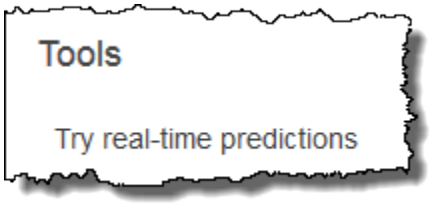
Pruebas con las predicciones en tiempo real

Para ayudarle a decidir si desea habilitar las predicciones en tiempo real, Amazon ML le permite probar la generación de predicciones en registros de datos únicos sin incurrir en cargos adicionales asociados con la configuración de un punto de conexión de predicciones en tiempo real. Para probar las predicciones en tiempo real, debe contar con un modelo de ML. Para crear predicciones en tiempo real a mayor escala, utilice la API Predecir en la Referencia de la API de Amazon Machine Learning.

Pruebas con predicciones en tiempo real

1. Inicie sesión en la consola Amazon Machine Learning AWS Management Console y ábrala en https://console.aws.amazon.com/machinelearning/.

2. En la barra de navegación, en el desplegable Amazon Machine Learning, elija ML models (Modelos ML).
3. Elija el modelo que desea utilizar para probar las predicciones en tiempo real, como el `Subscription propensity model` del tutorial.
4. En la página del informe del modelo de ML, en Predictions (Predicciones), seleccione Summary (Resumen) y, a continuación, elija Try real-time predictions (Predicciones en tiempo real).



   Amazon ML muestra una lista de las variables que componen los registros de datos que Amazon ML utilizó para entrenar el modelo.
5. Puede continuar escribiendo datos en cada uno de los campos en el formulario o pegar un solo registro de datos, en formato CSV, en el cuadro de texto.

   Para utilizar el formulario, para cada campo Value (Valor), escriba los datos que desea utilizar para probar sus predicciones en tiempo real. Si el registro de datos que ha introducido no contiene valores correspondientes a uno o varios atributos de datos, deje los campos de entrada en blanco.

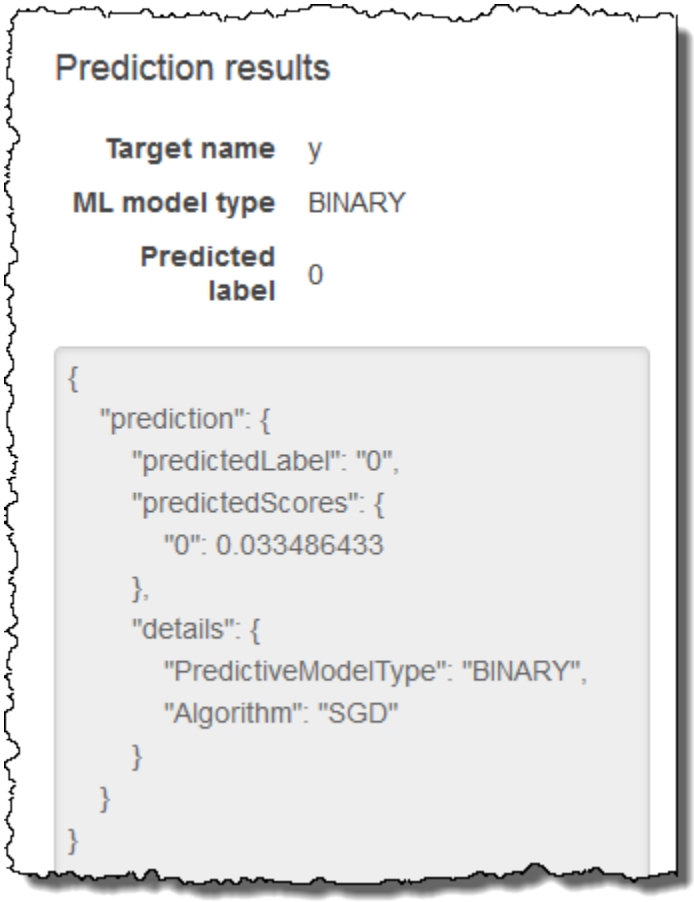
   Para proporcionar un registro de datos, seleccione Paste a record (Pegar un registro). Pegue una sola fila de datos con formato CSV en el campo de texto y elija Enviar. Amazon ML rellena automáticamente los campos Valor.

   > **Note**
   >
   > Los datos en el registro de datos deben tener el mismo número de columnas que los datos de entrenamiento y estar organizados en el mismo orden. La única excepción es que debería omitir el valor de destino. Si incluye un valor de destino, Amazon ML no lo tiene en cuenta.

6. En la parte inferior de la página, elija Create prediction. Amazon ML devuelve la predicción de forma inmediata.

   En el panel Prediction results (Resultados de predicción), verá el objeto de predicciones que devuelve la llamada a la API `Predict`, junto con el tipo de modelo de ML, el nombre de la

variable de destino y la clase o el valor predichos. Para obtener información acerca de cómo interpretar los resultados, consulte Interpretación del contenido de los archivos de predicciones por lotes para un modelo de ML de clasificación binaria.



Creación de un punto de enlace en tiempo real

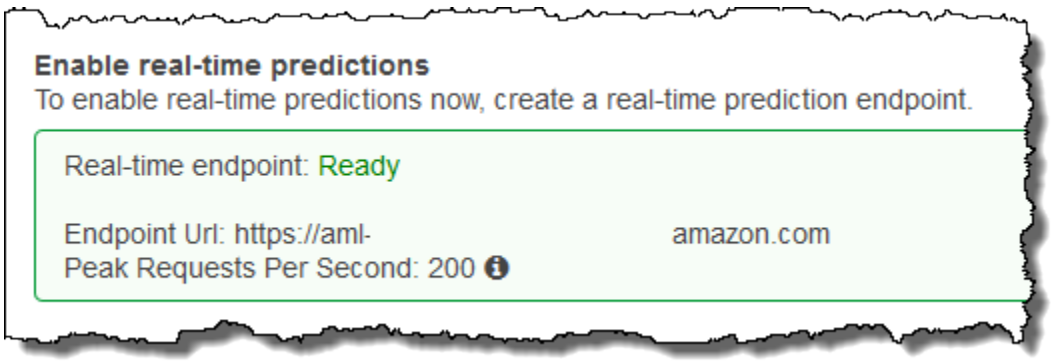
Para generar predicciones en tiempo real, debe crear un punto de enlace en tiempo real. Para crear un punto de enlace en tiempo real, ya debe tener un modelo de ML para el que desea generar predicciones en tiempo real. Puede crear un punto de conexión en tiempo real a través de la consola de Amazon ML o mediante una llamada a la API `CreateRealtimeEndpoint`. Para obtener más información sobre el uso de la API `CreateRealtimeEndpoint`, consulte https://docs.aws.amazon.com/machine-learning/latest/APIReference/API_CreateRealtimeEndpoint.html en la referencia de la API de Amazon Machine Learning.

Creación de un punto de enlace en tiempo real

1. Inicie sesión en la consola Amazon Machine Learning AWS Management Console y ábrala en https://console.aws.amazon.com/machinelearning/.
2. En la barra de navegación, en el desplegable Amazon Machine Learning, elija ML models (Modelos ML).
3. Elija el modelo por el cual desea generar predicciones en tiempo real.
4. En la página ML model summary (Resumen de modelo de ML), en Predictions (Predicciones), seleccione Create real-time endpoint (Crear punto de enlace en tiempo real).

   Aparecerá un cuadro de diálogo que explica el precio que tienen las predicciones en tiempo real.
5. Seleccione Crear. La solicitud del punto de conexión en tiempo real se envía a Amazon ML y se introduce en una cola. El estado del punto de enlace en tiempo real es Updating (Actualizando).

6. Cuando el punto de conexión en tiempo real esté listo, el estado cambia a Listo y Amazon ML muestra la dirección URL del punto de conexión. Utilice la dirección URL del punto de enlace para crear solicitudes de predicciones en tiempo real con la API `Predict`. Para obtener más información sobre el uso de la API `Predict`, consulte https://docs.aws.amazon.com/machine-learning/latest/APIReference/API_Predict.html en la Referencia de API de Amazon Machine Learning.



Ubicación del punto de enlace de predicciones en tiempo real (consola)

Para utilizar la consola de Amazon ML para encontrar la dirección URL del punto de conexión de un modelo de ML, consulte la página Resumen de modelo de ML.

Ubicación de la dirección URL de un punto de enlace en tiempo real

1. Inicie sesión en la consola Amazon Machine Learning AWS Management Console y ábrala en https://console.aws.amazon.com/machinelearning/.
2. En la barra de navegación, en el desplegable Amazon Machine Learning, elija ML models (Modelos ML).
3. Elija el modelo por el cual desea generar predicciones en tiempo real.
4. En la página ML model summary (Resumen de modelo de ML), desplácese hacia abajo hasta que vea la sección Predictions (Predicciones).
5. La dirección URL del punto de enlace del modelo se muestra en Real-time prediction (Predicción en tiempo real). Utilice la dirección URL como dirección URL de Endpoint Url (URL de punto de enlace) para sus llamadas de predicciones en tiempo real. Para obtener información acerca de cómo utilizar el punto de conexión para generar predicciones, consulte https://docs.aws.amazon.com/machine-learning/latest/APIReference/API_Predict.html en la Referencia de API de Amazon Machine Learning.

Ubicación del punto de enlace de predicciones en tiempo real (API)

Cuando crea un punto de conexión en tiempo real mediante la utilización de la operación `CreateRealtimeEndpoint`, la dirección URL y el estado del punto de enlace se le devuelven en la respuesta. Si creó el punto de enlace en tiempo real a través de la consola o si desea recuperar la dirección URL y el estado de un punto de enlace que ha creado anteriormente, llame a la operación `GetMLModel` con el ID del modelo del que desea consultar las predicciones en tiempo real. La información del punto de enlace está en la sección `EndpointInfo` de la respuesta. Para un modelo que tiene un punto de enlace en tiempo real asociado, `EndpointInfo` podría devolver lo siguiente:

```
"EndpointInfo":{
    "CreatedAt": 1427864874.227,
    "EndpointStatus": "READY",
    "EndpointUrl": "https://endpointUrl",
    "PeakRequestsPerSecond": 200
}
```

Un modelo sin un punto de enlace en tiempo real devolverá lo siguiente:

```
EndpointInfo":{
    "EndpointStatus": "NONE",
    "PeakRequestsPerSecond": 0
}
```

Creación de una solicitud de predicciones en tiempo real

Una carga de solicitud `Predict` podría devolver lo siguiente:

```
{
    "MLModelId": "model-id",
    "Record":{
        "key1": "value1",
        "key2": "value2"
    },
    "PredictEndpoint": "https://endpointUrl"
}
```

El campo `PredictEndpoint` debe corresponderse con el campo `EndpointUrl` de la estructura `EndpointInfo`. Amazon ML utiliza este campo para dirigir la solicitud a los servidores adecuados en la flota de predicciones en tiempo real.

El `MLModelId` es el identificador de un modelo entrenado previamente con un punto de enlace en tiempo real.

Un `Record` es un mapa de nombres de variables a valores variables. Cada par representa una observación. El mapa `Record` contiene las entradas de su modelo de Amazon ML. Es similar a una sola fila de datos en el conjunto de datos de entrenamiento, sin la variable de destino. Con independencia del tipo de valores de los datos de entrenamiento, `Record` contiene un string-to-string mapeo.

> **Note**
>
> Puede omitir variables para las que no tiene un valor, aunque esto podría reducir la exactitud de su predicción. Cuantas más variables pueda incluir, más preciso será su modelo.

El formato de la respuesta que devuelven las solicitudes `Predict` depende del tipo de modelo que se utiliza para las consultas de predicción. En todos los casos, el campo `details` contiene información acerca de la solicitud de predicciones, incluyendo de manera notable el campo `PredictiveModelType` con el tipo de modelo.

El siguiente ejemplo muestra una respuesta para un modelo binario:

```
{
    "Prediction":{
        "details":{
            "PredictiveModelType": "BINARY"
        },
        "predictedLabel": "0",
        "predictedScores":{
            "0": 0.47380468249320984
        }
    }
}
```

Tenga en cuenta el campo `predictedLabel` que contiene la etiqueta prevista, que en este caso es 0. Amazon ML procesa la etiqueta predicha comparando la puntuación de predicción con el corte de clasificación:

- Puede obtener el corte de clasificación que está asociado actualmente con un modelo de ML inspeccionando el campo `ScoreThreshold` en la respuesta de la operación `GetMLModel` o visualizando la información del modelo en la consola de Amazon ML. Si no establece un umbral de corte, Amazon ML utiliza el valor predeterminado de 0,5.
- Puede obtener la puntuación de predicciones exacta para un modelo de clasificación binaria inspeccionando el mapa `predictedScores`. Dentro de este mapa, la etiqueta predicha se empareja con la puntuación de predicciones exacta.

Para obtener más información sobre las predicciones binarias, consulte Interpretación de las predicciones.

El siguiente ejemplo muestra una respuesta para un modelo de regresión. Tenga en cuenta que el valor numérico predicho se encuentra en el campo `predictedValue`:

```
{
    "Prediction":{
```

```
        "details":{
            "PredictiveModelType": "REGRESSION"
        },
        "predictedValue": 15.508452415466309
    }
}
```

El siguiente ejemplo muestra una respuesta para un modelo multiclase:

```
{
    "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
        }
    }
}
```

De forma similar que los modelos de clasificación binaria, la etiqueta o clase predicha se encuentra en el campo `predictedLabel`. Puede comprender mejor cómo de fuertemente está relacionada la predicción con cada clase consultando el mapa `predictedScores`. Cuanto mayor sea la puntuación de una clase dentro de este mapa, más fuertemente está relacionada la predicción con dicha clase. El valor máximo quede seleccionado como `predictedLabel`.

Para obtener más información sobre las predicciones multiclase, consulte Información del modelo multiclase.

Eliminación de un punto de enlace en tiempo real

Cuando haya completado sus predicciones en tiempo real, elimine el punto de enlace en tiempo real para evitar incurrir en cargos adicionales. Se detiene la acumulación de cargos en cuanto elimina el punto de enlace.

Eliminación de un punto de enlace en tiempo real

1. Inicie sesión en la consola Amazon Machine Learning AWS Management Console y ábrala en https://console.aws.amazon.com/machinelearning/.
2. En la barra de navegación, en el desplegable Amazon Machine Learning, elija ML models (Modelos ML).
3. Elija el modelo que ya no requiere predicciones en tiempo real.
4. En la página del informe del modelo de ML, en Predictions (Predicciones), elija Summary (Resumen).
5. Seleccione Delete real-time endpoint (Eliminar punto de enlace en tiempo real).
6. En el cuadro de diálogo Delete real-time endpoint (Eliminar puerto de enlace en tiempo real), elija Delete (Eliminar).

Administración de objetos de Amazon ML

Amazon ML proporciona cuatro objetos que se pueden administrar a través de la consola de Amazon ML o la API de Amazon ML:

- Fuentes de datos
- Modelos de ML
- Evaluaciones
- Predicciones por lotes

Cada objeto tiene una finalidad distinta en el ciclo de vida de la creación de una aplicación de aprendizaje automático y cada uno de ellos tiene atributos específicos y funciones que se aplican únicamente a ese objeto. A pesar de estas diferencias, los objetos se administran de forma parecida. Por ejemplo, utiliza procesos casi idénticos para enumerar objetos, recuperar sus descripciones y actualizarlos o eliminarlos.

Las siguientes secciones describen las operaciones de administración que son comunes para los cuatro objetos y destacan las diferencias.

Temas

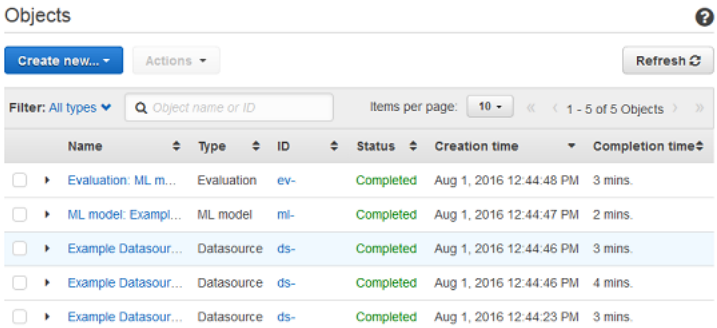
- Listas de objetos
- Recuperación de descripciones de objetos
- Actualización de objetos
- Eliminación de objetos

Listas de objetos

Para obtener información detallada sobre sus orígenes de datos de Amazon Machine Learning (Amazon ML), modelos de ML, evaluaciones y predicciones por lotes, inclúyalos en una lista. Para cada uno de los objetos, verá su nombre, tipo, ID, código de estado y el momento de su creación. También puede consultar información específica de un determinado tipo de objeto. Por ejemplo, puede ver la información de datos de una fuente de datos.

Listas de objetos (consola)

Para ver una lista de los últimos 1 000 objetos que haya creado. abra el panel Objetos en la consola de Amazon ML. Para visualizar el panel Objetos, inicie sesión en la consola de Amazon ML.



Para ver más detalles sobre un objeto, incluidos los detalles concretos del tipo de dicho objeto, elija el ID o el nombre del objeto. Por ejemplo, para ver la propiedad Data insights (Información de datos) de un origen de datos, elija el nombre del origen de datos.

Las columnas en el panel Objects (Objetos) muestran la siguiente información sobre cada uno de los objetos.

Nombre

El nombre del objeto.

Tipo

El tipo del objeto. Los valores válidos son Datasource (Origen de datos), ML model (Modelo de ML), Evaluation (Evaluación) y Batch prediction (Predicción por lotes).

> **Note**
>
> Para ver si el modelo está configurado para admitir predicciones en tiempo real, vaya a la página ML model summary (Resumen de modelos ML) eligiendo el ID o el nombre del modelo.

ID

El ID del objeto.

Estado

El estado del objeto. Los valores incluyen Pending (Pendiente), In Progress (En curso), Completed (Completado) y Failed (Error). Si el estado es Failed (Error), compruebe los datos y vuelva a intentarlo.

Hora de creación

La fecha y la hora en las que Amazon ML terminó la creación de este objeto.

Completion time

El tiempo que necesitó Amazon ML para crear este objeto. Puede utilizar el "Completion time" (tiempo para completar la creación) de un modelo para estimar el tiempo de entrenamiento necesario para un modelo nuevo.

Datasource ID

En el caso de los objetos que se han creado utilizando una fuente de datos, como por ejemplo modelos y evaluaciones, el ID de la fuente de datos. Si elimina el origen de datos, ya no podrá utilizar los modelos de ML creados con ese origen de datos para crear predicciones.

Ordene por cualquier columna eligiendo el icono de doble triángulo junto al encabezado de columna.

Listas de objetos (API)

En la API de Amazon ML, puede enumerar objetos, por tipo, utilizando las siguientes operaciones:

- `DescribeDataSources`
- `DescribeMLModels`
- `DescribeEvaluations`
- `DescribeBatchPredictions`

Cada operación incluye parámetros para filtrar, clasificar y paginar a lo largo de una lista larga de objetos. No existe ningún límite en cuanto al número de objetos a los que puede obtener acceso a través de la API. Para limitar el tamaño de la lista, utilice el parámetro `Limit`, que puede tener un valor máximo de 100.

La respuesta de la API a un comando `Describe*` incluye un token de paginación (`nextPageToken`), si procede, y descripciones breves de cada uno de los objetos. Las

descripciones del objeto incluyen la misma información para cada uno de los tipos de objeto que se muestran en la consola, incluida la información específica de un objeto.

> **Note**
>
> Incluso si la respuesta incluye menos objetos que el límite especificado, puede incluir un `nextPageToken` que indica que hay más resultados disponibles. Incluso una respuesta que contiene 0 elementos podría contener un `nextPageToken`.

Para obtener más información, consulte la Referencia de las API de Amazon ML.

Recuperación de descripciones de objetos

Puede ver descripciones detalladas de cualquier objeto a través de la consola o a través de la API.

Descripciones detalladas en la consola

Para ver descripciones en la consola, vaya a la lista y seleccione un tipo de objeto específico (fuente de datos, modelo de ML, evaluación o predicción por lotes). A continuación, coloque la fila en la tabla que corresponde al objeto, ya sea navegando por la lista o buscando su nombre o ID.

Descripciones detalladas de la API

Cada tipo de objeto tiene una operación que recupera todos los detalles de un objeto de Amazon ML:

- GetDataSource
- Obtenga MLModel
- GetEvaluation
- GetBatchPrediction

Cada operación utiliza exactamente dos parámetros: el ID del objeto y un marcador booleano denominado Verbose. Las llamadas con Verbose establecido en "true" incluirán detalles adicionales sobre el objeto, lo que se traduce en latencias más altas y respuestas más grandes. Para saber qué campos se incluyen ajustando el marcador Verbose, consulte la Referencia de la API de Amazon ML.

Actualización de objetos

Cada tipo de objeto tiene una operación que actualiza los detalles de un objeto de Amazon ML (consulte Referencia de la API de Amazon ML):

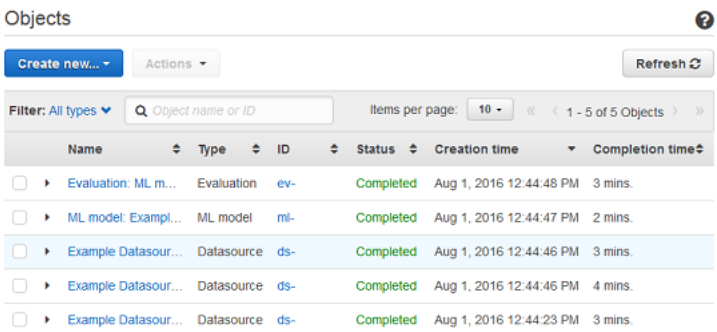
- UpdateDataSource
- Actualización MLModel
- UpdateEvaluation
- UpdateBatchPrediction

Cada operación requiere el ID del objeto para especificar qué objeto se está actualizando. Puede actualizar los nombres de todos los objetos. No se puede actualizar cualquier otra propiedad de objetos para fuentes de datos, evaluaciones y predicciones por lote. En el caso de los modelos de aprendizaje automático, puede actualizar el ScoreThreshold campo, siempre que el modelo de aprendizaje automático no tenga asociado un punto final de predicción en tiempo real.

Eliminación de objetos

Cuando ya no necesite sus fuentes de datos, modelos de ML, evaluaciones y predicciones por lotes, puede eliminarlos. Aunque no existe ningún costo adicional para mantener objetos de Amazon ML que no sean predicciones por lotes después de que haya terminado con ellos, eliminar objetos mantiene su espacio de trabajo despejado y más fácil de administrar. Puede eliminar uno o varios objetos mediante la consola de Amazon Machine Learning (Amazon ML) o mediante la API.

> **Warning**
>
> Cuando elimine objetos de Amazon ML, el efecto es inmediato, permanente e irreversible.



Eliminación de objetos (consola)

Puede utilizar la consola de Amazon ML para eliminar objetos, incluidos los modelos. El procedimiento que utilice para eliminar un modelo depende de si está utilizando el modelo para generar predicciones en tiempo real o no. Para eliminar un modelo que se utiliza para generar predicciones en tiempo real, elimine primero el punto de enlace en tiempo real.

Eliminación de objetos de Amazon ML (consola)

1. Inicie sesión en la consola Amazon Machine Learning AWS Management Console y ábrala en https://console.aws.amazon.com/machinelearning/.
2. Seleccione los objetos de Amazon ML que desea eliminar. Para seleccionar más de un objeto, utilice la tecla Mayús. Para anular la selección de todos los objetos seleccionados, utilice los botones

   o

   .
3. En Actions (Acciones), seleccione Delete (Eliminar).
4. En el cuadro de diálogo, seleccione Delete (Eliminar) para eliminar el modelo.

Eliminación de un modelo de Amazon ML con un punto de conexión en tiempo real (consola)

1. Inicie sesión en la consola Amazon Machine Learning AWS Management Console y ábrala en https://console.aws.amazon.com/machinelearning/.
2. Seleccione el modelo que desea eliminar.
3. Para Actions, seleccione Delete real-time endpoint (Eliminar punto de enlace en tiempo real).
4. Seleccione Delete (Eliminar) para eliminar el punto de enlace.
5. Seleccione el modelo de nuevo.
6. En Actions (Acciones), seleccione Delete (Eliminar).
7. Seleccione Eliminar para eliminar el modelo.

Eliminación de objetos (API)

Puede eliminar objetos de Amazon ML con las siguientes llamadas a la API:

- `DeleteDataSource` usa el parámetro `DataSourceId`.
- `DeleteMLModel` usa el parámetro `MLModelId`.
- `DeleteEvaluation` usa el parámetro `EvaluationId`.
- `DeleteBatchPrediction` usa el parámetro `BatchPredictionId`.

Para obtener más información, consulte Referencia de la API de Amazon Machine Learning.

Supervisión de Amazon ML con Amazon CloudWatch Metrics

Amazon ML envía automáticamente las métricas a Amazon para que CloudWatch pueda recopilar y analizar las estadísticas de uso de sus modelos de aprendizaje automático. Por ejemplo, para realizar un seguimiento de las predicciones por lotes y en tiempo real, puede supervisar la PredictCount métrica en función de la RequestMode dimensión. Las estadísticas se recopilan automáticamente y se envían a Amazon CloudWatch cada cinco minutos. Puede supervisar estas métricas mediante la CloudWatch consola de Amazon, la AWS CLI o AWS SDKs.

No se cobran cargos por las métricas de Amazon ML que se publican a través de ellas CloudWatch. Si configuras alarmas en las estadísticas, se te facturarán según CloudWatch las tarifas estándar.

Para obtener más información, consulte la lista de métricas de Amazon ML en Amazon CloudWatch Namespaces, Dimensions and Metrics Reference en la Guía para CloudWatch desarrolladores de Amazon.

Registrar llamadas a la API de Amazon ML con AWS CloudTrail

Amazon Machine Learning (Amazon ML) está integrado AWS CloudTrail con un servicio que proporciona un registro de las acciones realizadas por un usuario, un rol o AWS un servicio en Amazon ML. CloudTrail captura todas las llamadas a la API de Amazon ML como eventos. Las llamadas capturadas incluyen las llamadas desde la consola de Amazon ML y las llamadas desde el código a las operaciones de la API de Amazon ML. Si crea una ruta, puede habilitar la entrega continua de CloudTrail eventos a un bucket de Amazon S3, incluidos los eventos de Amazon ML. Si no configura una ruta, podrá ver los eventos más recientes en la CloudTrail consola, en el historial de eventos. Con la información recopilada por CloudTrail, puede determinar la solicitud que se realizó a Amazon ML, la dirección IP desde la que se realizó la solicitud, quién la hizo, cuándo se realizó y detalles adicionales.

Para obtener más información CloudTrail, incluido cómo configurarlo y habilitarlo, consulte la Guía del AWS CloudTrail usuario.

Información de Amazon ML en CloudTrail

CloudTrail está activado en su AWS cuenta al crear la cuenta. Cuando se produce una actividad de eventos admitida en Amazon ML, esa actividad se registra en un CloudTrail evento junto con otros eventos de AWS servicio en el historial de eventos. Puede ver, buscar y descargar los eventos recientes en su AWS cuenta. Para obtener más información, consulte Visualización de eventos con el historial de CloudTrail eventos.

Para obtener un registro continuo de los eventos de su AWS cuenta, incluidos los eventos de Amazon ML, cree una ruta. Un rastro permite CloudTrail entregar archivos de registro a un bucket de Amazon S3. De forma predeterminada, cuando se crea un registro de seguimiento en la consola, el registro de seguimiento se aplica a todas las regiones de AWS. La ruta registra los eventos de todas las regiones de la AWS partición y envía los archivos de registro al bucket de Amazon S3 que especifique. Además, puede configurar otros AWS servicios para analizar más a fondo los datos de eventos recopilados en los CloudTrail registros y actuar en función de ellos. Para más información, consulte los siguientes temas:

- Introducción a la creación de registros de seguimiento
- CloudTrail Integraciones y servicios compatibles

- Configuración de las notificaciones de Amazon SNS para CloudTrail
- Recibir archivos de CloudTrail registro de varias regiones y recibir archivos de CloudTrail registro de varias cuentas

Amazon ML admite el registro de las siguientes acciones como eventos en los archivos de CloudTrail registro:

- AddTags
- CreateBatchPrediction
- CreateDataSourceFromRDS
- CreateDataSourceFromRedshift
- CreateDataSourceFromS3
- CreateEvaluation
- CrearMLModel
- CreateRealtimeEndpoint
- DeleteBatchPrediction
- DeleteDataSource
- DeleteEvaluation
- DeleteMLModel
- DeleteRealtimeEndpoint
- DeleteTags
- DescribeTags
- UpdateBatchPrediction
- UpdateDataSource
- UpdateEvaluation
- ActualizaciónMLModel

Las siguientes operaciones de Amazon ML utilizan parámetros de solicitud que contienen credenciales. Antes de enviar estas solicitudes CloudTrail, las credenciales se sustituyen por tres asteriscos («***»):

- CreateDataSourceFromRDS

- CreateDataSourceFromRedshift

Cuando las siguientes operaciones de Amazon ML se realizan con la consola de Amazon ML, el atributo no `ComputeStatistics` se incluye en el `RequestParameters` componente del CloudTrail registro:

- CreateDataSourceFromRedshift
- CreateDataSourceFromS3

Cada entrada de registro o evento contiene información sobre quién generó la solicitud. La información de identidad del usuario le ayuda a determinar lo siguiente:

- Si la solicitud se realizó con credenciales de usuario root o AWS Identity and Access Management (IAM).
- Si la solicitud se realizó con credenciales de seguridad temporales de un rol o fue un usuario federado.
- Si la solicitud la realizó otro AWS servicio.

Para obtener más información, consulte el Elemento userIdentity de CloudTrail .

Ejemplo: entradas del archivo de registro de Amazon ML

Un rastro es una configuración que permite la entrega de eventos como archivos de registro a un bucket de Amazon S3 que usted especifique. CloudTrail Los archivos de registro contienen una o más entradas de registro. Un evento representa una solicitud única de cualquier fuente e incluye información sobre la acción solicitada, la fecha y la hora de la acción, los parámetros de la solicitud, etc. CloudTrail Los archivos de registro no son un registro ordenado de las llamadas a la API pública, por lo que no aparecen en ningún orden específico.

En el siguiente ejemplo, se muestra una entrada de CloudTrail registro que demuestra la acción.

```
{
    "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"
        }
    ]
}
```

Etiquetado de sus objetos Amazon ML

Organice y administre sus objetos de Amazon Machine Learning (Amazon ML) asignándoles metadatos con etiquetas. Una etiqueta es un par clave-valor que define para un objeto.

Además de usar etiquetas para organizar y administrar sus objetos de Amazon ML, puede utilizarlas para clasificar en categorías y realizar un seguimiento de los costos de AWS. Cuando se aplican etiquetas a los objetos de AWS, incluidos los modelos de ML, el informe de asignación de costos de AWS incluye el uso y los costos agregados por etiquetas. Al aplicar etiquetas que representen categorías de negocio (por ejemplo, centros de costos, nombres de aplicación o propietarios), puede organizar los costos entre diferentes servicios. Para obtener más información, consulte Utilizar etiquetas de asignación de costos para informes de facturación personalizados en la Guía del usuario de AWS Billing .

Contenido

- Conceptos básicos de etiquetas
- Restricciones de las etiquetas
- Etiquetado de objetos de Amazon ML (consola)
- Etiquetado de objetos de Amazon ML (API)

Conceptos básicos de etiquetas

Utilice las etiquetas para categorizar los objetos y facilitar su administración. Por ejemplo, puede clasificar en categorías los objetos por finalidad, propietario o entorno. A continuación, podría definir un conjunto de etiquetas que le ayude a realizar un seguimiento de los modelos por propietario y aplicaciones asociadas. Estos son algunos ejemplos:

- Proyecto: nombre del proyecto
- Propietario: nombre
- Finalidad: predicciones de marketing
- Aplicación: nombre de aplicación
- Entorno: producción

Puede utilizar la consola de Amazon ML o la API para realizar las tareas siguientes:

- Añadir etiquetas a un objeto
- Ver las etiquetas de sus objetos
- Editar las etiquetas de sus objetos
- Eliminar etiquetas de un objeto

De forma predeterminada, las etiquetas aplicadas a un objeto Amazon ML se copian en los objetos creados utilizando ese objeto. Por ejemplo, si un origen de datos de Amazon Simple Storage Service (Amazon S3) tiene la etiqueta "Costo de marketing: campaña de marketing focalizada", un modelo creado con ese origen de datos también tendría la etiqueta "Costo de marketing: campaña de marketing focalizada", al igual que la evaluación del modelo. Esto le permite utilizar etiquetas para realizar un seguimiento de los objetos relacionados, como todos los objetos utilizados para una campaña de marketing. En caso de conflicto entre las fuentes de etiquetas, como un modelo con la etiqueta "Costo de marketing: campaña de marketing focalizada" y una fuente de datos con la etiqueta "Costo de marketing: campaña de marketing focalizada", Amazon ML aplica la etiqueta del modelo.

Restricciones de las etiquetas

Se aplican las siguientes restricciones a las etiquetas.

Restricciones básicas:

- El número máximo de etiquetas por objeto es 50.
- Las claves y los valores de las etiquetas distinguen entre mayúsculas y minúsculas.
- No se pueden cambiar ni editar etiquetas de un objeto eliminado.

Limitaciones de clave de etiqueta:

- Cada clave de etiqueta debe ser única. Si añade una etiqueta con una clave que ya está en uso, la nueva etiqueta sobrescribe el par clave-valor existente para ese objeto.
- No se puede iniciar una clave de etiqueta con `aws:` porque este prefijo está reservado para su uso con AWS. AWS crea etiquetas que comienzan con este prefijo en su nombre, pero no puede editarlas o eliminarlas.
- Las claves de etiqueta deben tener entre 1 y 128 caracteres Unicode de longitud.

- Las claves de etiquetas deben constar de los siguientes caracteres: letras Unicode, números, espacios en blanco y los siguientes caracteres especiales: `_ . / = + - @`.

Restricciones de valor de etiqueta:

- Los valores de etiqueta deben tener entre 0 y 255 caracteres Unicode de longitud.
- Los valores de etiqueta pueden estar en blanco. De lo contrario, deben constar de los siguientes caracteres: letras Unicode, números, espacios en blanco y cualquiera de los siguientes caracteres especiales: `_ . / = + - @`.

Etiquetado de objetos de Amazon ML (consola)

Puede ver, añadir, editar y eliminar etiquetas con la consola Amazon ML.

Para ver las etiquetas de un objeto (consola)

1. Inicie sesión en la consola Amazon Machine Learning AWS Management Console y ábrala en https://console.aws.amazon.com/machinelearning/.
2. En la barra de navegación, expanda el selector de regiones y elija una región.
3. En la página Objects (Objetos), elija un objeto.
4. Desplácese hasta la sección Tags (Etiquetas) del objeto elegido. Las etiquetas de ese objeto se muestran en la parte inferior de la sección.

Para añadir una etiqueta a un objeto (consola)

1. Inicie sesión en la consola Amazon Machine Learning AWS Management Console y ábrala en https://console.aws.amazon.com/machinelearning/.
2. En la barra de navegación, expanda el selector de regiones y elija una región.
3. En la página Objects (Objetos), elija un objeto.
4. Desplácese hasta la sección Tags (Etiquetas) del objeto elegido. Las etiquetas de ese objeto se muestran en la parte inferior de la sección.
5. Elija Add or edit tags (Añadir o editar etiquetas).
6. En Add Tag (Añadir etiqueta), especifique la clave de etiqueta en el campo Key (Clave), especifique opcionalmente un valor de etiqueta en el campo Value (Valor) y, a continuación, elija Apply changes (Aplicar cambios).

Si el botón Apply changes (Aplicar cambios) no está habilitado, la clave o el valor de etiqueta que ha especificado no cumple las restricciones de etiquetas. Para obtener más información, consulte Restricciones de las etiquetas.

7. Para ver la nueva etiqueta en la lista de la sección Tags (Etiquetas), actualice la página.

Para editar una etiqueta (consola)

1. Inicie sesión en la consola Amazon Machine Learning AWS Management Console y ábrala en https://console.aws.amazon.com/machinelearning/.
2. En la barra de navegación, expanda el selector de regiones y seleccione una región.
3. En la página Objects (Objetos), elija un objeto.
4. Desplácese hasta la sección Tags (Etiquetas) del objeto elegido. Las etiquetas de ese objeto se muestran en la parte inferior de la sección.
5. Elija Add or edit tags (Añadir o editar etiquetas).
6. En Applied tags (Etiquetas aplicadas), edite el valor de una etiqueta en el campo Value (Valor) y, a continuación, elija Apply changes (Aplicar cambios).

   Si el botón Apply changes (Aplicar cambios) no está habilitado, el valor de etiqueta que ha especificado no cumple las restricciones de etiquetas. Para obtener más información, consulte Restricciones de las etiquetas.
7. Para ver la etiqueta actualizada en la lista de la sección Tags (Etiquetas), actualice la página.

Para eliminar una etiqueta de un objeto (consola)

1. Inicie sesión en la consola Amazon Machine Learning AWS Management Console y ábrala en https://console.aws.amazon.com/machinelearning/.
2. En la barra de navegación, expanda el selector de regiones y elija una región.
3. En la página Objects (Objetos), elija un objeto.
4. Desplácese hasta la sección Tags (Etiquetas) del objeto elegido. Las etiquetas de ese objeto se muestran en la parte inferior de la sección.
5. Elija Add or edit tags (Añadir o editar etiquetas).
6. En Applied tags (Etiquetas aplicadas), elija la etiqueta que desee eliminar y, a continuación, elija Apply changes (Aplicar cambios).

Etiquetado de objetos de Amazon ML (API)

Puede añadir, enumerar y eliminar etiquetas con la API de Amazon ML. Para ver ejemplos, consulte la documentación siguiente:

AddTags

Añade o edita etiquetas del objeto especificado.

DescribeTags

Enumera las etiquetas del objeto especificado.

DeleteTags

Elimina etiquetas del objeto especificado.

Referencia de Amazon Machine Learning

Temas

- Concesión de permisos de Amazon ML para la lectura de datos desde Amazon S3
- Concesión de permisos a Amazon ML para enviar predicciones a Amazon S3
- Control de acceso a los recursos de Amazon ML con IAM
- Prevención de la sustitución confusa entre servicios
- Administración de la dependencia de operaciones asíncronas
- Comprobación del estado de la solicitud
- Límites del sistema
- Nombres y IDs para todos los objetos
- Object Lifetimes

Concesión de permisos de Amazon ML para la lectura de datos desde Amazon S3

Para crear un objeto de una fuente de datos desde los datos de entrada de Amazon S3, debe conceder a Amazon ML los siguientes permisos para la ubicación de S3 donde se almacenan los datos de entrada:

- GetObjectpermiso en el bucket y el prefijo de S3.
- ListBucketpermiso en el depósito de S3. A diferencia de otras acciones, se ListBucketdeben conceder permisos para todo el depósito (y no para el prefijo). Sin embargo, puede conceder el permiso a un prefijo específico mediante una cláusula de Condición.

Si utiliza la consola de Amazon ML para crear la fuente de datos, la aplicación puede añadir estos permisos al bucket en su nombre. Se le pedirá que confirme si desea añadirlos a medida que complete los pasos del asistente. La siguiente política de ejemplo muestra cómo conceder permiso a Amazon ML para leer datos de la ubicación de muestra s3://*examplebucket*/y, al mismo tiempo*exampleprefix*, limitar el ListBucketpermiso únicamente a la ruta de entrada. *exampleprefix*

```
{
```

```
    "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:*" }
        }
    }]
}
```

Para aplicar esta política a sus datos, debe editar la instrucción de política asociada con el bucket de S3 donde están almacenados sus datos.

Edición de la política de permisos para un bucket de S3 (usando la consola anterior)

1. Inicie sesión en la consola de Amazon S3 AWS Management Console y ábrala en https://console.aws.amazon.com/s3/.
2. Seleccione el nombre del bucket donde se encuentran sus datos.
3. Seleccione Propiedades.
4. Elija Edit bucket policy (Editar política de buckets).
5. Introduzca la política que se ha mostrado anteriormente, personalícela para adaptarla a sus necesidades y, a continuación, seleccione Save (Guardar).
6. Seleccione Save.

Edición de la política de permisos para un bucket de S3 (usando la consola nueva)

1. Inicie sesión en la consola de Amazon S3 AWS Management Console y ábrala en https://console.aws.amazon.com/s3/.
2. Elija el nombre del bucket y seleccione Permissions (Permisos).
3. Elija Política de bucket.
4. Escriba la política que se ha mostrado anteriormente y personalícela para adaptarla a sus necesidades.
5. Seleccione Save.

Concesión de permisos a Amazon ML para enviar predicciones a Amazon S3

Para enviar los resultados de la operación de predicción por lotes a Amazon S3, debe conceder los siguientes permisos a Amazon ML en la ubicación de salida, que se proporciona como entrada a la operación de creación de predicción por lotes (Create Batch Prediction):

- GetObjectpermiso en su bucket y prefijo de S3.
- PutObjectpermiso en su bucket y prefijo de S3.
- PutObjectAclen su bucket y prefijo de S3.
  - Amazon ML necesita este permiso para garantizar que puede conceder el bucket-owner-full-control permiso de ACL predefinido a su cuenta de AWS una vez creados los objetos.
- ListBucketpermiso en el bucket de S3. A diferencia de otras acciones, se ListBucketdeben conceder permisos para todo el depósito (y no para el prefijo). Sin embargo, puede conceder el permiso a un prefijo específico mediante una cláusula Condition.

Si utiliza la consola de Amazon ML para crear la solicitud de predicción por lotes, la aplicación puede añadir estos permisos al bucket por usted. Se le pedirá que confirme si desea añadirlos a medida que complete los pasos del asistente.

El siguiente ejemplo de política muestra cómo conceder permiso a Amazon ML para escribir datos en la ubicación de muestra s3://examplebucket/exampleprefix, limitando el ListBucketpermiso únicamente a la ruta de entrada exampleprefix y concediendo el permiso a Amazon ML para establecer el objeto put ACLs en el prefijo de salida:

```
{
    "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:*" }
        }
    }]
}
```

Para aplicar esta política a sus datos, debe editar la instrucción de política asociada con el bucket de S3 donde están almacenados sus datos.

Edición de la política de permisos para un bucket de S3 (usando la consola anterior)

1. Inicie sesión en la consola de Amazon S3 AWS Management Console y ábrala en https://console.aws.amazon.com/s3/.
2. Seleccione el nombre del bucket donde se encuentran sus datos.
3. Seleccione Propiedades.
4. Elija Edit bucket policy (Editar política de buckets).
5. Introduzca la política que se ha mostrado anteriormente, personalícela para adaptarla a sus necesidades y, a continuación, seleccione Save (Guardar).
6. Seleccione Save.

Edición de la política de permisos para un bucket de S3 (usando la consola nueva)

1. Inicie sesión en la consola de Amazon S3 AWS Management Console y ábrala en https://console.aws.amazon.com/s3/.
2. Elija el nombre del bucket y seleccione Permissions (Permisos).
3. Elija Política de bucket.
4. Escriba la política que se ha mostrado anteriormente y personalícela para adaptarla a sus necesidades.
5. Seleccione Save.

Control de acceso a los recursos de Amazon ML con IAM

AWS Identity and Access Management (IAM) le permite controlar de forma segura el acceso de sus usuarios a los servicios y recursos de AWS. Con IAM, puede crear y administrar usuarios, grupos y roles de AWS, y utilizar permisos para permitir y denegar su acceso a los recursos de AWS. Al utilizar IAM con Amazon Machine Learning (Amazon ML), puede controlar si los usuarios de su organización pueden utilizan recursos específicos de AWS y si pueden realizar una tarea mediante acciones específicas de la API de Amazon ML.

IAM; le permite:

- Crear usuarios y grupos en su cuenta de AWS.
- Asignar credenciales de seguridad únicas a cada usuario en su cuenta de AWS
- Controlar los permisos de cada usuario para realizar tareas mediante recursos de AWS
- Compartir fácilmente sus recursos de AWS con los usuarios de su cuenta de AWS.
- Crear roles para su cuenta de AWS y administrar sus permisos para definir los usuarios o servicios que pueden asumir dichos roles.
- Puede crear roles en IAM y administrar los permisos para controlar qué operaciones podrá ejecutar la entidad, o el servicio de AWS, que asuma ese rol. También puede definir a qué entidad se le permite asumir la función.

Si su organización ya tiene identidades de IAM, puede utilizarlas para conceder permisos de realización de tareas usando los recursos de AWS.

Para obtener más información acerca de IAM, consulte la guía del usuario de IAM.

Sintaxis de la política de IAM

Una política de IAM es un documento JSON que contiene una o varias instrucciones. Cada instrucción tiene la siguiente estructura:

```
{
    "Statement":[{
        "Effect":"effect",
        "Action":"action",
        "Resource":"arn",
        "Condition":{
            "condition operator":{
                "key":"value"
            }
        }
    }]
}
```

Una instrucción de política incluye los siguientes elementos:

- El Efecto controla el permiso para utilizar los recursos y acciones de la API que especifique más tarde en la instrucción. Los valores válidos son `Allow` y `Deny`. De forma predeterminada, los usuarios de IAM no tienen permiso para utilizar los recursos y las acciones de la API, por lo que se

deniegan todas las solicitudes. Un valor `Allow` explícito anula la opción predeterminada. Un valor `Deny` explícito anula cualquier valor `Allows` explícito.
- Con el término acción nos referimos a la acción o acciones de la API específicas a las que concede o deniega permiso.
- Resource: el recurso al que afecta la acción. Para especificar un recurso en la instrucción se utiliza el nombre de recurso de Amazon (ARN).
- La condición (opcional) controla cuándo entrará en vigor la política.

Para simplificar la creación y administración de políticas de IAM, puede utilizar el generador de políticas de AWS (AWS Policy Generator) y el generador de políticas de IAM (simulador de política de IAM).

Especificación de las acciones políticas de IAM para Amazon ML MLAmazon

En una instrucción de política de IAM, puede especificar una acción de API para cualquier servicio que tenga soporte para IAM. Cuando crea una instrucción de política para acciones de la API de Amazon ML añada `machinelearning:` al nombre de la acción de la API tal y como se muestra en los ejemplos siguientes:

- `machinelearning:CreateDataSourceFromS3`
- `machinelearning:DescribeDataSources`
- `machinelearning:DeleteDataSource`
- `machinelearning:GetDataSource`

Para especificar varias acciones en una única instrucción, sepárelas con comas:

```
"Action": ["machinelearning:action1", "machinelearning:action2"]
```

También puede utilizar caracteres comodín para especificar varias acciones. Por ejemplo, puede especificar todas las acciones cuyo nombre comience por la palabra "Get":

```
"Action": "machinelearning:Get*"
```

Para especificar todas las acciones de Amazon ML, use el carácter comodín * del siguiente modo:

```
"Action": "machinelearning:*"
```

Para obtener la lista completa de acciones de la API de Amazon ML, consulte la Referencia de la API de Amazon Machine Learning.

Especificación ARNs de los recursos de Amazon ML en las políticas de IAM

Las instrucciones de política de IAM se aplican a uno o más recursos. Los recursos para sus políticas se especifican por ellos. ARNs

Para especificar los ARNs recursos de Amazon ML, utilice el siguiente formato:

"Recurso": `arn:aws:machinelearning:region:account:resource-type/identifier`

Los siguientes ejemplos muestran cómo especificar common ARNs.

ID de la fuente de datos: `my-s3-datasource-id`

```
"Resource":
arn:aws:machinelearning:<region>:<your-account-id>:datasource/my-s3-datasource-id
```

ID del modelo de ML: `my-ml-model-id`

```
"Resource":
arn:aws:machinelearning:<region>:<your-account-id>:mlmodel/my-ml-model-id
```

ID de la predicción por lotes: `my-batchprediction-id`

```
"Resource":
arn:aws:machinelearning:<region>:<your-account-id>:batchprediction/my-batchprediction-
id
```

ID de la evaluación: `my-evaluation-id`

```
"Resource": arn:aws:machinelearning:<region>:<your-account-id>:evaluation/my-
evaluation-id
```

Ejemplos de políticas para Amazon MLs

Ejemplo 1: permitir que los usuarios lean metadatos de recursos de aprendizaje automático

La siguiente política permite a un usuario o grupo leer los metadatos de las fuentes de datos, los modelos de aprendizaje automático, las predicciones por lotes y las evaluaciones realizando acciones de tipo Describe DescribeDataSourcesMLModelsDescribeBatchPredictionsDescribeEvaluationsGetDataSource,MLModel, GetBatchPrediction, Get y otras GetEvaluationacciones en los recursos especificados. No se pueden restringir los permisos para las operaciones "Describe *" a un recurso concreto.

```
{
    "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": [
            "*"
        ]
    }]
}
```

Ejemplo 2: permitir que los usuarios creen recursos de aprendizaje automático

La política siguiente permite que un usuario o grupo cree fuentes de datos de aprendizaje automático, modelos de ML, predicciones por lotes y evaluaciones llevando a cabo las acciones `CreateDataSourceFromS3`, `CreateDataSourceFromRedshift`, `CreateDataSourceFromRDS`, `CreateMLModel`, `CreateBatchPrediction` y `CreateEvaluation`. No puede restringir los permisos para estas acciones a un recurso específico.

```
{
    "Version": "2012-10-17",
    "Statement": [{
        "Effect": "Allow",
        "Action": [
            "machinelearning:CreateDataSourceFrom*",
            "machinelearning:CreateMLModel",
            "machinelearning:CreateBatchPrediction",
            "machinelearning:CreateEvaluation"
        ],
        "Resource": [
            "*"
        ]
    }]
}
```

Ejemplo 3: permitir que los usuarios creen y eliminen puntos de enlace en tiempo real y realicen predicciones en tiempo real en un modelo de ML

La política siguiente permite que los usuarios o grupos creen y eliminen puntos de enlace en tiempo real y realicen predicciones en tiempo real para un modelo de ML específico llevando a cabo las acciones `CreateRealtimeEndpoint`, `DeleteRealtimeEndpoint` y `Predict` en ese modelo.

```
{
    "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"
```

```
        ]
    }]
}
```

Ejemplo 4: permitir que los usuarios actualicen y eliminen recursos específicos

La política siguiente permite que un usuario o grupo actualice y elimine recursos específicos en su cuenta de AWS dándoles permiso para llevar a cabo las acciones `UpdateDataSource`, `UpdateMLModel`, `UpdateBatchPrediction`, `UpdateEvaluation`, `DeleteDataSource`, `DeleteMLModel`, `DeleteBatchPrediction` y `DeleteEvaluation` en esos recursos de su cuenta.

```
{
    "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"
        ]
    }]
}
```

Ejemplo 5: Permitir cualquier Amazon MLaction

La política siguiente permite que un usuario o grupo lleve a cabo cualquier acción de Amazon ML. Dado que esta política concede acceso completo a todos sus recursos de aprendizaje automático, debe restringirla solo a los administradores.

```
{
```

```
    "Version": "2012-10-17",
    "Statement": [{
        "Effect": "Allow",
        "Action": [
            "machinelearning:*"
        ],
        "Resource": [
            "*"
        ]
    }]
}
```

Prevención de la sustitución confusa entre servicios

El problema de la sustitución confusa es un problema de seguridad en el que una entidad que no tiene permiso para realizar una acción puede obligar a una entidad con más privilegios a realizar la acción. En AWS, la suplantación de identidad entre servicios puede provocar el confuso problema de un diputado. La suplantación entre servicios puedes producirse cuando un servicio (el servicio que lleva a cabo las llamadas) llama a otro servicio (el servicio al que se llama). El servicio que lleva a cabo las llamadas se puedes manipular para utilizar sus permisos a fin de actuar en función de los recursos de otro cliente de una manera en la que no debe tener permiso para acceder. Para evitarlo, AWS proporciona herramientas que lo ayudan a proteger sus datos para todos los servicios con entidades principales de servicio a las que se les ha dado acceso a los recursos de su cuenta.

Se recomienda utilizar las claves de contexto de condición global `aws:SourceArn` y `aws:SourceAccount` en las políticas de recursos para limitar los permisos que Amazon Machine Learning concede a otro servicio para el recurso. Si el valor de `aws:SourceArn` no contiene el ID de cuenta, como un ARN de bucket de Amazon S3, debe utilizar ambas claves de contexto de condición global para limitar los permisos. Si utiliza claves de contexto de condición global y el valor de `aws:SourceArn` contiene el ID de cuenta, el valor de `aws:SourceAccount` y la cuenta en el valor de `aws:SourceArn` deben utilizar el mismo ID de cuenta cuando se utiliza en la misma instrucción de política. Utiliza `aws:SourceArn` si desea que solo se asocie un recurso al acceso entre servicios. Utiliza `aws:SourceAccount` si quiere permitir que cualquier recurso de esa cuenta se asocie al uso entre servicios.

La forma más eficaz de protegerse contra el problema de la sustitución confusa es utilizar la clave de contexto de condición global de `aws:SourceArn` con el ARN completo del recurso. Si no conoce el ARN completo del recurso o si especifica varios recursos, utiliza la clave de condición de contexto

global `aws:SourceArn` con comodines (`*`) para las partes desconocidas del ARN. Por ejemplo, `arn:aws:servicename:*:123456789012:*`.

En el ejemplo siguiente, se muestra cómo se pueden utilizar las claves de contexto de condición global de `aws:SourceArn` y `aws:SourceAccount` en Amazon ML para evitar el problema adjunto confuso al leer datos de un bucket de 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:*" }
        }
    }]
}
```

Administración de la dependencia de operaciones asíncronas

Las operaciones por lotes de Amazon ML dependen de otras operaciones para poder completarse correctamente. Para administrar estas dependencias, Amazon ML identifica las solicitudes con dependencias y verifica que las operaciones se hayan completado. Si las operaciones no se han

completado, Amazon ML descarta las peticiones iniciales hasta que las operaciones de las que dependen se hayan completado.

Existen algunas dependencias entre las operaciones por lotes. Por ejemplo, antes de poder crear un modelo de ML, debe crear una fuente de datos con la que pueda entrenar el modelo de ML. Amazon ML no puede entrenar un modelo de ML si no hay ningún tipo de fuente de datos disponible.

Sin embargo, Amazon ML permite administrar las dependencias de las operaciones asíncronas. Por ejemplo, no es necesario esperar hasta que se calculen las estadísticas de datos para poder enviar una solicitud para entrenar un modelo de ML mediante una fuente de datos. De hecho, puede enviar una solicitud para entrenar un modelo de ML mediante una fuente de datos tan pronto como se cree la fuente de datos. Amazon ML no comenzará la operación de entrenamiento hasta que se calculen las estadísticas de la fuente de datos. La MLModel solicitud de creación se pone en cola hasta que se calculen las estadísticas; una vez hecho esto, Amazon ML intenta ejecutar inmediatamente la MLModel operación de creación. Del mismo modo, puede enviar solicitudes de evaluación y predicción por lotes de modelos de ML que no hayan terminado su entrenamiento.

La siguiente tabla muestra los requisitos para continuar con diferentes acciones de Amazon ML.

Para...	Debe tener...
Cree un modelo de aprendizaje automático (crearMLModel)	Una fuente de datos con estadísticas de datos computadas
Crear una predicción por lotes (createBatchPrediction)	Una fuente de datos modelo de ML
Crear una evaluación de lote (createBatchEvaluation)	Una fuente de datos modelo de ML

Comprobación del estado de la solicitud

Cuando envía una solicitud, puede verificar su estado con la API de Amazon Machine Learning (Amazon ML). Por ejemplo, si envía una solicitud `createMLModel`, puede comprobar el estado a través de la llamada `describeMLModel`. Amazon ML responde con uno de los siguientes estados.

Estado	Definición
PENDIENTE	Amazon ML está validando la solicitud. OR Amazon ML está a la espera de que los recursos informáticos estén disponibles antes de ejecutar la solicitud. Esto puede ocurrir cuando la cuenta ha superado el número máximo de solicitudes de operaciones por lotes que estén ejecutándose simultáneamente. Si este es el caso, el estado pasa a InProgresscuando otras solicitud es en ejecución se hayan completado o se hayan cancelado. OR Amazon ML está a la espera de una operación por lotes de la que depende su solicitud para completarse.
INPROGRESS	La solicitud se está ejecutando.
COMPLETED	La solicitud ha finalizado y el objeto está listo para usarse (modelos de ML y fuentes de datos) o visualizarse (predicciones por lotes y evaluaciones).
ERROR	Hay algún problema con los datos que ha proporcionado o ha cancelado la operación. Por ejemplo, si intenta calcular las estadísti cas de datos de un origen de datos que no se ha completado, es posible que reciba un mensaje de estado Invalid o Failed. El mensaje de error explica por qué la operación no se ha completado correctamente.
DELETED	Ya se ha eliminado el objeto.

Amazon ML también ofrece información sobre un objeto, como cuándo terminó Amazon ML de crear dicho objeto. Para obtener más información, consulte Listas de objetos.

Límites del sistema

Con el fin de ofrecer un servicio sólido y de confianza, Amazon ML impone determinados límites en las solicitudes que realice en el sistema. La mayoría de problemas de ML se adaptan fácilmente a estas limitaciones. No obstante, si cree que el uso que hace de Amazon ML está siendo restringido por estos límites, puede ponerse en contacto con el servicio de atención al cliente de AWS y solicitar que se aumente un límite. Por ejemplo, puede tener un límite de cinco trabajos que puede ejecutar de forma simultánea. Si detecta que termina teniendo trabajos en la cola pendientes de recursos debido a este límite, probablemente tenga sentido aumentar ese límite para su cuenta.

La siguiente tabla muestra los límites predeterminados por cuenta en Amazon ML. El servicio de atención al cliente de AWS no está en disposición de aumentar todos estos límites.

Tipo de límite	Límite del sistema
Tamaño de cada una de las observaciones	100 KB
Tamaño de datos de entrenamiento *	100 GB
Tamaño de la entrada de predicción por lotes	1 TB
Tamaño de la entrada de predicción por lotes (número de registros)	100 millones
Número de variables de un archivo de datos (esquema)	1 000
Complejidad de receta (número de variables de salida procesadas)	10 000
TPS para cada punto de enlace de predicción en tiempo real	200
Total de TPS de todos los puntos de conexión de predicción en tiempo real	10 000
Total de RAM de todos los puntos de enlace de predicción en tiempo real	10 GB
Número de trabajos simultáneos	25
Tiempo de ejecución máximo para cualquier trabajo	7 días
Número de clases para modelos de ML multiclase	100

Tipo de límite	Límite del sistema
Tamaño de modelo de ML	Mínimo de 1 MB, máximo de 2 GB
Número de etiquetas por objeto	50

- El tamaño de los archivos de datos se limita para asegurarse de que los trabajos finalicen puntualmente. Los trabajos que llevan ejecutándose más de siete días se finalizarán automáticamente, lo que dará lugar al estado FAILED.

Nombres y IDs para todos los objetos

Todos los objetos de Amazon ML deben tener un identificador (ID). La consola de Amazon ML genera valores de ID por usted. Si utiliza la API, debe generar sus propios ID. Cada ID debe ser único entre todos los objetos de Amazon ML del mismo tipo en su cuenta de AWS. Es decir, no puede tener dos evaluaciones con el mismo ID. Es posible tener una evaluación y una fuente de datos con el mismo ID, aunque no se recomienda.

Le recomendamos que utilice identificadores generadas de forma aleatoria para sus objetos, prefijados con una cadena breve para identificar su tipo. Por ejemplo, cuando la consola de Amazon ML genera una fuente de datos, le asigna un identificador único y aleatorio, como «DS-zsc F». WIu WiOx Este ID es suficientemente aleatorio para evitar conflictos a cualquier usuario y también es compacto y legible. El prefijo "ds-" es por comodidad y claridad, pero no es necesario. Si no está seguro sobre qué utilizar para sus ID de cadenas, le recomendamos que utilice valores hexadecimales de identificador único universal (UUID) como 28b1e915-57e5-4e6c-a7bd-6fb4e729cb23, que se encuentran disponibles en cualquier entorno de programación moderno.

Las cadenas de ID pueden contener letras en codificación ASCII, números, guiones y guiones bajos y pueden ser de hasta 64 caracteres. Es posible, y seguramente resulte también cómodo, codificar los metadatos en una cadena de ID. Sin embargo, no se recomienda porque, una vez que se haya creado un objeto, su ID no podrá modificarse.

Los nombres de objeto proporcionan una forma sencilla para asociar metadatos intuitivos con cada uno de los objetos. Puede actualizar los nombres después de que se haya creado un objeto. Esto permite que el nombre de objeto refleje algunos aspectos de su flujo de trabajo de ML. Por ejemplo, es posible que asigne inicialmente el nombre "experimento #3" a un modelo de ML y que luego

cambie el nombre del modelo a "modelo de producción final". Los nombres puede ser cualquier cadena que desee con hasta 1 024 caracteres.

Object Lifetimes

Cualquier fuente de datos, modelo de ML, evaluación u objeto de predicción por lotes que cree con Amazon ML estará disponible para su uso durante al menos dos años después de su creación. Amazon ML podría eliminar objetos automáticamente a los que no se ha accedido o que no se han utilizado durante más de dos años.

Recursos

Los recursos relacionados siguientes pueden serle de ayuda cuando trabaje con este servicio.

- Información del producto de Amazon ML: reúne toda la información pertinente del producto sobre Amazon ML en una ubicación centralizada.
- Amazon ML FAQs: cubre las principales preguntas que los desarrolladores se han hecho sobre este producto.
- Código de ejemplo de Amazon ML: aplicaciones de ejemplo que utilizan Amazon ML. Puede utilizar el código de muestra como punto de partida para crear sus propias aplicaciones de ML.
- Referencia de la API de Amazon ML: describe de forma detallada todas las operaciones de las APIs de Amazon ML. También incluye ejemplos de solicitudes y respuestas de los protocolos de servicios web aceptados.
- Centro de recursos para desarrolladores de AWS: proporciona un punto de inicio central para buscar documentación, ejemplos de código, notas de la versión y otra información para ayudarle a crear aplicaciones innovadoras con AWS.
- Formación y cursos de AWS: enlaces a cursos basados en roles y especializados, así como a laboratorios autoguiados para ayudarle a desarrollar sus conocimientos de AWS y obtener experiencia práctica.
- Herramientas para desarrolladores de AWS: enlaces a herramientas y recursos para desarrolladores que proporcionan documentación, ejemplos de código, notas de la versión y otra información para ayudarle a crear aplicaciones innovadoras con AWS.
- Centro de AWS Support: centro para crear y administrar sus casos de AWS Support. También incluye enlaces a otros recursos útiles, como foros, información técnica FAQs, estado del servicio y AWS Trusted Advisor.
- AWS Support: la página web principal con información sobre AWS Support one-on-one, un canal de soporte de respuesta rápida que le ayuda a crear y ejecutar aplicaciones en la nube.
- Contacto: un punto central de contacto para las consultas relacionadas con la facturación de AWS, cuentas, eventos, uso indebido y otros problemas.
- Términos del sitio de AWS: información detallada sobre nuestros derechos de autor y marca comercial, su cuenta, licencia y acceso al sitio, entre otros temas.

Historial de documentos

En la siguiente tabla se describen los cambios importantes que se han realizado en la documentación de esta versión de Amazon Machine Learning (Amazon ML).

- Versión de API: 2015-04-09
- Última actualización de la documentación: 02-08-2016

Cambio	Descripción	Fecha de modificación
Se han añadido métricas	Esta versión de Amazon ML añade métricas nuevas para objetos de Amazon ML. Para obtener más información, consulte Listas de objetos.	2 de agosto de 2016
Eliminar varios objetos	Esta versión de Amazon ML añade la posibilidad de eliminar varios objetos de . Para obtener más información, consulte Eliminación de objetos.	20 de julio de 2016
Se han añadido etiquetas	Esta versión de Amazon ML añade la posibilidad de aplicar etiquetas a objetos de . Para obtener más información, consulte Etiquetado de sus objetos Amazon ML.	23 de junio de 2016
Copiar fuentes de datos de Amazon Redshift	Esta versión de Amazon ML añade la posibilidad de copiar la configuración del origen de datos de Amazon Redshift a otro origen de datos de Amazon Redshift nuevo. Para obtener más información acerca de cómo usar la copia de la configuración del origen de datos de Amazon Redshift, consulte Copiar una fuente de datos (consola).	11 de abril de 2016
Se ha añadido la mezcla	Esta versión de Amazon ML añade la posibilidad de mezclar los datos de entrada.	5 de abril de 2016

Cambio	Descripción	Fecha de modificación
	Para obtener más información acerca de cómo utilizar el parámetro Shuffle type (Tipo de mezcla), consulte Tipo de mezcla para los datos de entrenamiento.	
Se ha mejorado la creación de orígenes de datos con Amazon Redshift	Esta versión de Amazon ML añade la posibilidad de probar su configuración de Amazon Redshift al crear un origen de datos de Amazon ML en la consola para comprobar que la conexión funciona. Para obtener más información, consulte Creación de una fuente de datos con datos de Amazon Redshift (consola).	21 de marzo de 2016
Se ha mejorado la conversión de esquemas de datos de Amazon Redshift	Esta versión de Amazon ML mejora la conversión de los esquemas de datos de Amazon Redshift (Amazon Redshift) en esquemas de datos de Amazon ML. Para obtener más información acerca del uso de Amazon Redshift con Amazon ML, consulte Creación de una fuente de datos de Amazon ML a partir de datos de Amazon Redshift.	9 de febrero de 2016
CloudTrail registro agregado	Esta versión de Amazon ML añade la posibilidad de registrar las solicitudes mediante AWS CloudTrail (CloudTrail). Para obtener más información sobre el uso del CloudTrail registro, consulteRegistrar llamadas a la API de Amazon ML con AWS CloudTrail.	10 de diciembre de 2015
Se han añadido DataRearrangement opciones adicionales	Esta versión de Amazon ML añade la posibilidad de dividir los datos de entrada aleatoriamente y crear fuentes de datos complementarias. Para obtener más información sobre cómo usar el parámetro `DataRearrangement` , consulte Reorganización de datos. Para obtener información acerca de cómo utilizar las opciones nuevas para la validación cruzada, consulte Validación cruzada.	3 de diciembre de 2015

Cambio	Descripción	Fecha de modificación
Pruebas con las predicciones en tiempo real	Esta versión de Amazon ML añade la posibilidad de probar las predicciones en tiempo real en la consola de servicio. Para obtener más información acerca de las pruebas con predicciones en tiempo real, consulte Solicitud de predicciones en tiempo real en la Guía para desarrolladores de machine learning de Amazon.	19 de noviembre de 2015
Nueva región de	Esta versión de Amazon ML añade soporte para la región UE (Irlanda). Para obtener más información sobre Amazon ML en la región de la UE (Irlanda), consulte Regiones y puntos de conexión en la Guía para desarrolladores de Amazon Machine Learning.	20 de agosto de 2015
Versión inicial	Esta es la primera versión de la Guía para desarrolladores de Amazon ML.	9 de abril de 2015