aws

사용자 가이드

Amazon Fraud Detector

버전 latest

Amazon Fraud Detector: 사용자 가이드

Table of Contents

Amazon Fraud Detector란 무엇입니까?

Amazon Fraud Detector는 온라인에서 잠재적 사기 행위를 자동으로 탐지하는 완전 관리형 사기 탐지 서비스입니다. 이러한 활동에는 무단 거래 및 가짜 계정 생성이 포함됩니다. Amazon Fraud Detector는 기계 학습을 사용하여 데이터를 분석하는 방식으로 작동합니다. Amazon에서 20년 이상 사기 행위를 탐지해 온 노련한 전문 지식을 바탕으로 이를 수행합니다.

Amazon Fraud Detector를 사용하여 사용자 지정된 사기 탐지 모델을 구축하고, 모델의 사기 평가를 해석하는 의사 결정 로직을 추가하고, 가능한 각 사기 평가에 대해 통과 또는 검토용 전송과 같은 결과를 할당할 수 있습니다. Amazon Fraud Detector를 사용하면 기계 학습 전문 지식이 없어도 사기 행위를 탐지할 수 있습니다.

시작하려면 조직에서 수집한 사기 데이터를 수집하고 준비하십시오. 그러면 Amazon Fraud Detector가 이 데이터를 사용하여 사용자 지정 사기 탐지 모델을 교육, 테스트 및 배포합니다. 이 프로세스의 일환으로 Amazon Fraud Detector는 Amazon의 자체 사기 전문 지식을 통해 사기 패턴을 학습한 기계 학습 모델을 사용하여 사기 데이터를 평가하고 모델 점수 및 모델 성능 데이터를 생성합니다. AWS 모델의 점수를 해석하고 각 사기 평가를 처리하는 방법에 대한 결과를 할당하도록 의사 결정 로직을 구성합니다.

이점

Amazon Fraud Detector는 다음과 같은 이점을 제공합니다. 이러한 이점을 통해 기존에 사기 관리 시스템을 구축하고 유지 관리하는 데 필요했던 시간과 리소스를 투자하지 않고도 사기를 빠르게 탐지할 수 있습니다.

자동화된 사기 모델 생성

Amazon Fraud Detector의 사기 탐지 모델은 특정 비즈니스 요구 사항에 맞게 사용자 지정된 완전 자동화된 기계 학습 모델입니다. Amazon Fraud Detector 모델을 사용하여 신규 계정 생성, 온라인 결제, 게스트 체크아웃과 같은 모든 온라인 거래에서 잠재적 사기를 식별할 수 있습니다.

사기 모델은 자동화된 프로세스를 통해 생성되므로 모델 생성 및 교육과 관련된 많은 단계를 생략할 수 있습니다. 이러한 단계에는 데이터 검증 및 강화, 기능 엔지니어링, 알고리즘 선택, 하이퍼파라미터 조정, 모델 배포 등이 포함됩니다.

Amazon Fraud Detector를 사용하여 사기 탐지 모델을 생성하려면 회사의 과거 사기 데이터 세트를 업로드하고 모델 유형만 선택하면 됩니다. 그러면 Amazon Fraud Detector가 사용 사례에 가장 적합한 사

기 탐지 알고리즘을 자동으로 찾아 모델을 생성합니다. 부정 행위 탐지 모델을 생성하기 위해 코딩에 대한 지식이나 기계 학습 전문 지식이 없어도 됩니다.

진화하고 학습하는 사기 모델

사기 탐지 모델은 변화하는 사기 환경에 발맞추어 끊임없이 진화해야 합니다. Amazon Fraud Detector는 계정 기간, 마지막 활동 이후 시간, 활동 횟수 등의 정보를 계산하여 이를 자동으로 수행합니다. 결과적으로 모델은 거래를 자주 하는 신뢰할 수 있는 고객과 일반적인 사기꾼의 지속적인 시도 간의 차이를 학습합니다. 이렇게 하면 재교육 세션 사이에 모델의 성능을 더 오래 유지할 수 있습니다.

사기 모델 성능 시각화

제공된 데이터를 사용하여 모델을 학습한 후 Amazon Fraud Detector는 모델 성능을 검증합니다. 또한 성능을 평가할 수 있는 시각적 도구도 제공합니다. 학습한 각 모델에 대해 모델 성능 점수, 점수 분포 그래프, 오차 행렬, 임계값 테이블 및 제공한 모든 입력이 모델 성능에 미치는 영향을 기준으로 순위가 매겨진 것을 볼 수 있습니다. 이러한 성능 도구를 사용하면 모델이 어떻게 작동하고 어떤 입력이 모델 성능을 주도하는지 알 수 있습니다. 필요한 경우 모델을 조정하여 전반적인 성능을 개선할 수 있습니다.

사기 예측

Amazon Fraud Detector는 조직의 비즈니스 활동에 대한 사기 예측을 생성합니다. 사기 예측은 비즈니스 활동을 대상으로 사기 위험을 평가하는 것입니다. Amazon Fraud Detector는 활동과 관련된 데이터와 함께 예측 로직을 사용하여 예측을 생성합니다. 사기 탐지 모델을 생성할 때 이 데이터를 제공했습니다. 단일 활동에 대한 사기 예측을 실시간으로 얻거나 오프라인에서 일련의 활동에 대한 사기 예측을 얻을 수 있습니다.

사기 예측 설명 시각화

Amazon Fraud Detector는 사기 예측 프로세스의 일부로 예측 설명을 생성합니다. 예측 설명은 모델 학습에 사용된 각 데이터 요소가 모델의 사기 예측 점수에 어떤 영향을 미쳤는지에 대한 통찰력을 제공합니다. 예측 설명은 표 및 그래프와 같은 시각적 도구를 사용하여 제공됩니다. 이러한 도구를 사용하여 각 데이터 요소가 예측 점수에 미치는 영향을 시각적으로 식별할 수 있습니다. 그런 다음 이 정보를 사용하여 데이터 세트 전반의 사기 패턴을 분석하고 편향이 있는 경우 이를 찾아낼 수 있습니다. 마지막으로 예측 설명을 사용하여 수동 사기 조사 프로세스 중에 주요 위험 지표를 식별할 수도 있습니다. 이를 통해 오탐지 예측으로 이어지는 근본 원인을 좁힐 수 있습니다.

규칙 기반 조치

사기 탐지 모델을 학습한 후에는 평가된 데이터에 대해 데이터 수락, 검토용 데이터 전송, 추가 데이터 수집 등의 조치를 취하는 규칙을 추가할 수 있습니다. 규칙은 Amazon Fraud Detector에 사기 예측 중에 데이터를 해석하는 방법을 알려주는 조건입니다. 예를 들어, 의심스러운 고객 계정을 플래그로 지정

하여 검토하도록 하는 규칙을 생성할 수 있습니다. 탐지된 모델 점수가 모두 미리 정해진 기준치를 초과하고 계정 결제의 승인 코드 (AUTH_CODE) 가 유효하지 않은 경우 이 규칙이 시작되도록 설정할 수 있습니다.

핵심 개념 및 용어

Amazon Fraud Detector에서 사용되는 핵심 개념 및 용어 목록은 다음과 같습니다.

Event

이벤트는 사기 위험 여부를 평가한 조직의 비즈니스 활동입니다. Amazon Fraud Detector는 이벤트에 대한 사기 예측을 생성합니다.

레이블

라벨은 단일 이벤트를 사기 또는 합법적인 이벤트로 분류합니다. 레이블은 Amazon Fraud Detector에서 기계 학습 모델을 학습시키는 데 사용됩니다.

개체

엔터티는 이벤트를 수행 중인 사용자를 나타냅니다. 회사 사기 데이터의 일부로 개체 ID를 제공하여 이벤트를 수행한 특정 주체를 나타냅니다.

이벤트 유형

이벤트 유형은 Amazon Fraud Detector로 전송되는 이벤트의 구조를 정의합니다. 여기에는 이벤트의 일부로 전송된 데이터, 이벤트를 수행하는 주체 (예: 고객), 이벤트를 분류하는 레이블이 포함됩니다. 이벤트 유형의 예로는 온라인 결제 거래, 계정 등록, 인증 등이 있습니다.

엔터티 유형

엔터티 유형에 따라 엔터티가 분류됩니다. 분류의 예로는 고객, 판매자 또는 계정이 있습니다.

이벤트 데이터셋

이벤트 데이터세트는 특정 비즈니스 활동이나 이벤트에 대한 회사의 과거 데이터입니다. 예를 들어 회사의 이벤트는 온라인 계정 등록일 수 있습니다. 단일 이벤트 (등록) 의 데이터에는 관련 IP 주소, 이메일 주소, 청구 주소, 이벤트 타임스탬프가 포함될 수 있습니다. Amazon Fraud Detector에 이벤트 데이터세트를 제공하여 사기 탐지 모델을 생성하고 교육할 수 있습니다.

모델

모델은 기계 학습 알고리즘의 결과물입니다. 이러한 알고리즘은 코드로 구현되며 사용자가 제공한 이벤트 데이터를 기반으로 실행됩니다.

모델 유형

모델 유형은 모델 학습 중에 사용되는 알고리즘, 강화 및 기능 변환을 정의합니다. 또한 모델을 학습시키는 데 필요한 데이터 요구 사항도 정의합니다. 이러한 정의는 특정 유형의 사기에 맞게 모델을 최적화하는 역할을 합니다. 모델을 생성할 때 사용할 모델 유형을 지정합니다.

모델 훈련

모델 학습은 제공된 이벤트 데이터세트를 사용하여 사기 이벤트를 예측할 수 있는 모델을 만드는 프로세스입니다. 모델 학습 프로세스의 모든 단계는 완전히 자동화되어 있습니다. 이러한 단계에는 데이터 검증, 데이터 변환, 기능 엔지니어링, 알고리즘 선택, 모델 최적화가 포함됩니다.

모델 점수

모델 점수는 회사의 과거 사기 데이터를 평가한 결과입니다. 모델 교육 프로세스 중에 Amazon Fraud Detector는 데이터 세트의 사기 행위를 평가하여 0에서 1000 사이의 점수를 생성합니다. 이 점수에서 0은 낮은 사기 위험을 나타내고 1000은 가장 높은 사기 위험을 나타냅니다. 점수 자체는 가양성률 (FPR) 과 직접적인 관련이 있습니다.

모델 버전

모델 버전은 모델 학습의 결과입니다.

모델 배포

모델 배포는 모델 버전을 활성화하여 사기 예측을 생성하는 데 사용할 수 있도록 하는 프로세스입니다.

Amazon SageMaker 모델 엔드포인트

Amazon Fraud Detector를 사용하여 모델을 구축하는 것 외에도 Amazon Fraud Detector SageMaker 평가에서 선택적으로 호스팅된 모델 엔드포인트를 사용할 수 있습니다.

에서 SageMaker 모델을 구축하는 방법에 대한 자세한 내용은 다음을 사용하여 모델 학습을 참조하십시오. Amazon SageMaker

감지기

탐지기에는 사기 여부를 평가하려는 특정 이벤트에 대한 모델 및 규칙과 같은 탐지 로직이 포함되어 있습니다. 모델 버전을 사용하여 탐지기를 생성합니다.

Detector 버전

탐지기는 여러 버전을 가질 수 있으며 각 버전의 상태는 `DraftActive`, 또는 `Inactive` 입니다. 한 번에 하나의 검출기 버전만 `Active` 상태를 유지할 수 있습니다.

변수

변수는 사기 예측에 사용하려는 이벤트와 관련된 데이터 요소를 나타냅니다. 변수는 사기 예측의 일부로 이벤트와 함께 전송되거나 Amazon Fraud Detector 모델의 출력과 같이 파생될 수 Amazon SageMaker 있습니다.

규칙

규칙은 Amazon Fraud Detector에서 사기 예측 중에 변수 값을 해석하는 방법을 알려주는 조건입니다. 규칙은 하나 이상의 변수, 논리 표현식 및 하나 이상의 결과로 구성됩니다. 규칙에 사용되는 변수는 탐지기가 평가하는 이벤트 데이터셋의 일부여야 합니다. 또한 각 탐지기에는 관련된 규칙이 하나 이상 있어야 합니다.

결과

이는 사기 예측의 결과 또는 결과입니다. 사기 예측에 사용되는 각 규칙은 하나 이상의 결과를 지정해야 합니다.

사기 예측

사기 예측은 단일 이벤트 또는 일련의 이벤트에 대한 사기 사례를 평가하는 것입니다. Amazon Fraud Detector는 규칙에 따라 모델 점수와 결과를 동시에 제공하여 단일 온라인 이벤트에 대한 사기 예측을 실시간으로 생성합니다. Amazon Fraud Detector는 오프라인에서 발생한 일련의 이벤트에 대한 사기 예측을 생성합니다. 예측을 사용하여 오프라인으로 수행하거나 시간별 proof-of-concept, 일별 또는 주별로 소급적으로 사기 위험을 평가할 수 있습니다.

사기 예측 설명

사기 예측 설명은 각 변수가 모델의 사기 예측 점수에 어떤 영향을 미쳤는지에 대한 통찰력을 제공합니다. 각 변수가 규모 (0~5, 최고 범위 5) 및 방향 (점수 상승 또는 하락) 측면에서 각 변수가 위험 점수에 미치는 영향에 대한 정보를 제공합니다.

아마존 Fraud Detoret에 대한 작동 방식

Amazon Fraud Detector는 비즈니스 내 잠재적인 온라인 사기 활동을 탐지하도록 사용자 지정된 기계 학습 모델을 구축합니다. 시작하려면 비즈니스 사용 사례를 제공합니다. 비즈니스 사용 사례에 따라 Amazon Fraud Detector에서는 사기 탐지 모델을 생성하는 데 사용할 모델 유형을 권장합니다. 또한 비즈니스 기록 데이터의 일부로 제공해야 하는 데이터 요소에 대한 통찰력도 제공합니다. Amazon Fraud Detector는 과거 데이터 세트를 사용하여 사용자 지정 모델을 자동으로 생성하고 교육합니다.

자동 모델 교육 프로세스에는 특정 비즈니스 사용 사례에 맞게 사기를 탐지하는 기계 학습 알고리즘을 선택하고, 제공한 데이터를 검증하고, 데이터 조작을 수행하여 모델 성능을 개선하는 작업이 포함됩니

다. 모델을 학습시킨 후 Amazon Fraud Detector는 모델 점수 및 기타 모델 성능 지표를 생성합니다. 점수와 성능 지표를 사용하여 모델 성능을 평가할 수 있습니다. 필요한 경우 학습용으로 제공한 데이터셋에서 데이터 요소를 추가하거나 제거하고 모델을 재학습하여 모델 점수를 높일 수 있습니다.

모델을 만들고, 학습하고, 활성화한 후에는 비즈니스에서 생성된 데이터를 해석하는 방법을 모델에 지시하고 각 활동의 해석을 처리하는 방법에 대한 결과를 할당하는 의사 결정 로직 (규칙이라고도 함) 을 구성해야 합니다. 결과는 활동 승인 또는 검토와 같은 조치를 나타내거나 고위험, 중위험, 저위험과 같은 활동의 위험 수준을 나타낼 수 있습니다.

검출기는 모델 및 관련 규칙을 보관하는 컨테이너입니다. 탐지기를 만들고 테스트하여 프로덕션 환경에 배포해야 합니다.

프로덕션 환경에 배포된 탐지기는 비즈니스 애플리케이션에 사기 탐지 기능을 제공합니다. 사기 평가를 수행하기 위해 모델은 비즈니스 활동에서 들어오는 모든 데이터를 비즈니스의 과거 데이터와 비교하고 정교한 기계 학습 알고리즘을 사용자가 만든 규칙과 함께 사용하여 결과를 분석하고 결과를 할당합니다. Amazon Fraud Detector를 사용하면 단일 비즈니스 활동의 데이터를 실시간으로 평가하거나 여러 비즈니스 활동의 데이터를 오프라인으로 평가할 수 있습니다.

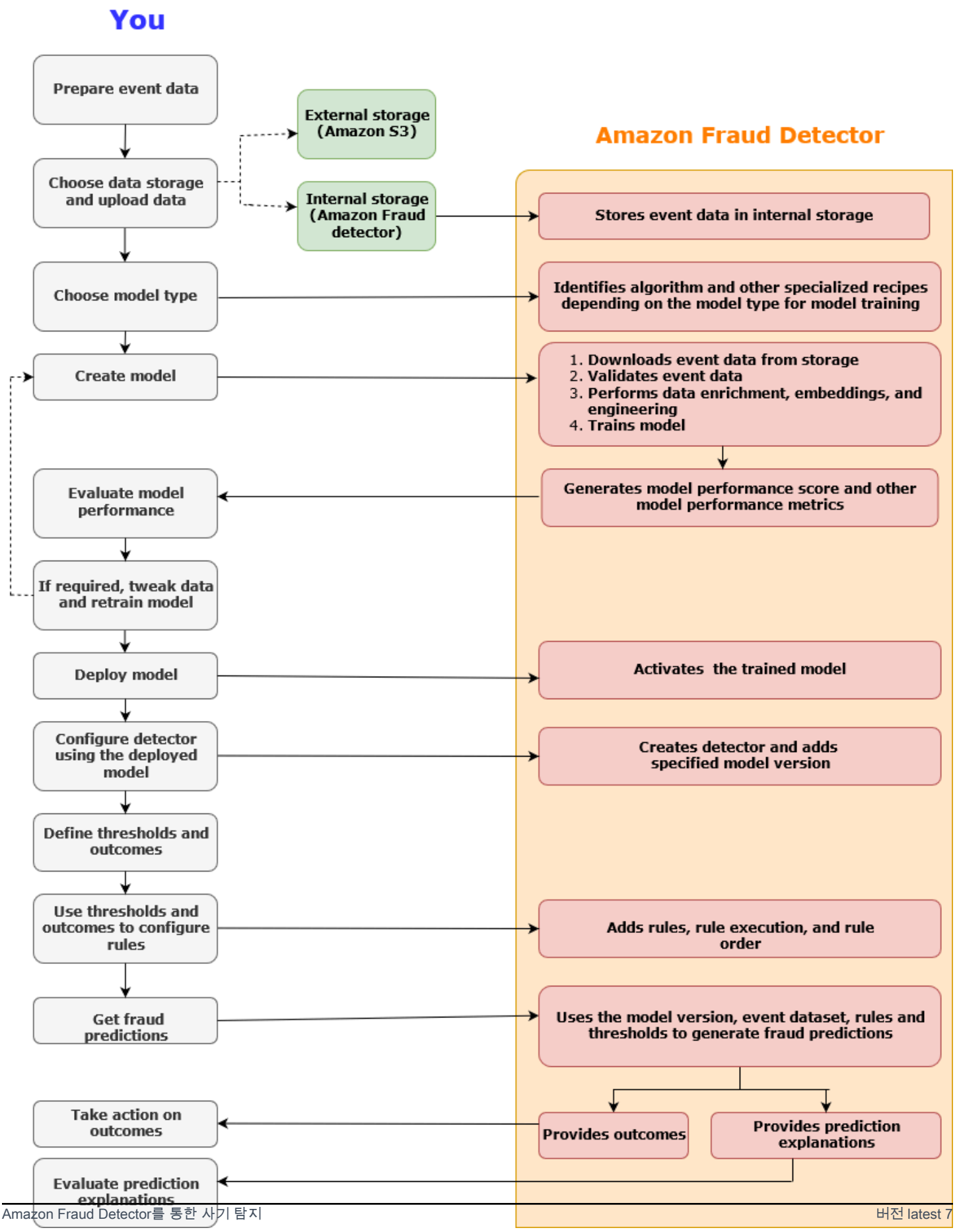
온라인 자금 이체를 활동 중 하나로 하는 기업이 있다고 가정해 보겠습니다. Amazon Fraud Detector를 사용하여 허위 자금 이체 요청을 실시간으로 탐지하려고 합니다. 시작하려면 먼저 Amazon Fraud Detector에 과거 자금 이체 요청의 데이터를 제공해야 합니다. Amazon Fraud Detector는 이 데이터를 사용하여 사기성 자금 이체 요청을 탐지하도록 사용자 지정된 모델을 만들고 교육합니다. 그런 다음 모델을 추가하고 모델이 데이터를 해석하도록 규칙을 구성하여 탐지기를 생성합니다. 온라인 자금 이체 활동에 대한 규칙의 예로는 자금 이체 요청이 다음과 같은 경우를 들 수 있습니다.xyz@example.com이메일 주소, 검토 요청을 보내주세요. 비즈니스 프로덕션 환경에서 자금 이체 요청이 들어오면 모델은 요청과 함께 제공된 데이터를 분석하고 규칙을 사용하여 결과를 할당합니다. 그런 다음 할당된 결과에 따라 요청에 대한 조치를 취할 수 있습니다.

Amazon Fraud Detector는 교육 데이터 세트, 모델, 탐지기, 규칙 및 결과와 같은 구성 요소를 사용하여 비즈니스에 사기 평가 로직을 제공합니다.

Amazon Fraud Detector를 사용하여 사기를 탐지하는 데 사용할 워크플로에 대한 자세한 내용은 을 참조하십시오.Amazon Fraud Detector를 통한 사기 탐지

Amazon Fraud Detector를 통한 사기 탐지

이 섹션에서는 Amazon Fraud Detector를 사용하여 사기를 탐지하는 일반적인 워크플로를 설명합니다. 또한 이러한 작업을 수행할 수 있는 방법도 요약되어 있습니다. 다음 다이어그램은 Amazon Fraud Detector를 사용하여 사기를 탐지하는 워크플로를 개괄적으로 보여줍니다.


You
Prepare event data
Choose data storage and upload data
External storage (Amazon S3)
Internal storage (Amazon Fraud detector)
Choose model type
Create model
Evaluate model performance
If required, tweak data and retrain model
Deploy model
Configure detector using the deployed model
Define thresholds and outcomes
Use thresholds and outcomes to configure rules
Get fraud predictions
Take action on outcomes
Evaluate prediction explanations
Amazon Fraud Detector
Stores event data in internal storage
Identifies algorithm and other specialized recipes depending on the model type for model training
1. Downloads event data from storage
2. Validates event data
3. Performs data enrichment, embeddings, and engineering
4. Trains model
Generates model performance score and other model performance metrics
Activates the trained model
Creates detector and adds specified model version
Adds rules, rule execution, and rule order
Uses the model version, event dataset, rules and thresholds to generate fraud predictions
Provides outcomes
Provides prediction explanations

사기 탐지는 지속적인 프로세스입니다. 모델을 배포한 후에는 예측 설명을 기반으로 성능 점수와 지표를 평가해야 합니다. 이렇게 하면 주요 위험 지표를 식별하고, 오탐으로 이어지는 근본 원인을 좁히고, 데이터세트 전반의 사기 패턴을 분석하고 편향이 있는 경우 이를 찾아낼 수 있습니다. 예측의 정확도를 높이려면 새 데이터나 수정된 데이터를 포함하도록 데이터세트를 조정할 수 있습니다. 그런 다음 업데이트된 데이터셋으로 모델을 재학습할 수 있습니다. 더 많은 데이터를 사용할 수 있게 되면 정확도를 높이기 위해 모델을 계속 재학습시킵니다.

Amazon Fraud Detector에 액세스하기

Amazon Fraud Detector는 여러 AWS 리전 가지로 제공되며 AWS 인터페이스를 사용하여 액세스할 수 있습니다.

가용성

Amazon Fraud Detector는 미국 동부 (버지니아 북부), 미국 동부 (오하이오), 미국 서부 (오레곤), 유럽 (아일랜드), 아시아 태평양 (싱가포르) 및 아시아 태평양 (시드니) 에서 사용할 수 있습니다. AWS 리전

인터페이스

다음 인터페이스 중 하나를 사용하여 사기 탐지 모델 및 탐지기를 생성, 교육, 배포, 테스트, 실행 및 관리할 수 있습니다.

AWS Management Console- Amazon Fraud Detector는 웹 기반 사용자 인터페이스인 Amazon Fraud Detector 콘솔을 제공합니다. 가입한 AWS 계정 경우 Amazon Fraud Detector 콘솔에 액세스할 수 있습니다. 자세한 내용은 Amazon Fraud Detector 설정을 참조하십시오.

AWS Command Line Interface(AWS CLI) - 명령줄 셸의 AWS 서비스 명령을 사용하여 Amazon Fraud Detector를 비롯한 다양한 범위와 상호 작용하는 데 사용할 수 있는 인터페이스를 제공합니다. AWS CLIAmazon Fraud Detector의 명령은 Amazon Fraud Detector 콘솔에서 제공하는 것과 동일한 기능을 구현합니다.

AWSSDK - 언어별 API를 제공하고 서명 계산, 요청 재시도 처리, 오류 처리와 같은 많은 연결 세부 정보를 관리합니다. 자세한 내용을 보려면 빌드 도구 AWS 페이지로 이동하여 SDK 섹션까지 아래로 스크롤한 다음 더하기 (+) 기호를 선택하여 섹션을 확장하십시오.

AWS CloudFormation- Amazon Fraud Detector 리소스 및 속성을 정의하는 데 사용할 수 있는 템플릿을 제공합니다. 자세한 내용은 AWS CloudFormation 사용 설명서의 Amazon Fraud Detector 리소스 유형 참조를 참조하십시오.

요금

Amazon Fraud Detector를 사용하면 사용한 만큼만 비용을 지불하면 됩니다. 최소 요금이나 사전 약정은 없습니다. 모델 교육 및 호스팅에 사용된 컴퓨팅 시간, 사용한 스토리지 용량, 부정 행위 예측의 양을 기준으로 요금이 부과됩니다. 자세한 내용은 Amazon Fraud Detector 요금을 참조하십시오.

Amazon Fraud Detector 설정

Amazon Fraud Detector를 사용하려면 먼저 Amazon Web Services (AWS) 계정이 있어야 하며, 그런 다음 모든 인터페이스에 AWS 계정 대한 액세스 권한을 부여하는 권한을 설정해야 합니다. 나중에 Amazon Fraud Detector 리소스를 생성하기 시작하면 Amazon Fraud Detector가 사용자 계정에 액세스하여 사용자를 대신하여 작업을 수행하고 사용자가 소유한 리소스에 액세스할 수 있도록 권한을 부여해야 합니다.

Amazon Fraud Detector를 사용하도록 설정하려면 이 섹션의 다음 작업을 완료하십시오.

- 가입하세요 AWS.
- Amazon Fraud Detector AWS 계정 인터페이스에 액세스할 수 있는 권한을 설정합니다.
- Amazon Fraud Detector에 액세스하는 데 사용할 인터페이스를 설정합니다.

이 단계를 완료한 후 Amazon Fraud Amazon Fraud Detector와 함께 시작하기 Detector를 계속 시작하려면 을 참조하십시오.

가입하십시오. AWS

Amazon Web Services (AWS) 에 가입하면 Amazon Fraud Detector를 AWS포함한 모든 서비스에 자동으로 AWS 계정 가입됩니다. 사용한 서비스에 대해서만 청구됩니다. 이미 계정이 AWS 계정있다면 다음 작업으로 건너뛰십시오.

등록해 보세요 AWS 계정

계정이 없는 경우 다음 단계를 완료하여 계정을 만드세요. AWS 계정

가입하려면 AWS 계정

1. https://portal.aws.amazon.com/billing/signup을 여세요.
2. 온라인 지시 사항을 따르세요.

   등록 절차 중에는 전화를 받고 키패드로 인증 코드를 입력하는 과정이 있습니다.

   에 AWS 계정가입하면 AWS 계정 루트 사용자a가 생성됩니다. 루트 사용자에게는 계정의 모든 AWS 서비스 및 리소스 액세스 권한이 있습니다. 보안 모범 사례는 사용자에게 관리 액세스 권한

을 할당하고, 루트 사용자만 사용하여 루트 사용자 액세스 권한이 필요한 작업을 수행하는 것입니다.

AWS 가입 절차가 완료된 후 확인 이메일을 보냅니다. 언제든지 https://aws.amazon.com/으로 가서 내 계정(My Account)을 선택하여 현재 계정 활동을 보고 계정을 관리할 수 있습니다.

관리자 액세스 권한이 있는 사용자 생성

등록한 AWS 계정후에는 일상적인 작업에 루트 사용자를 사용하지 않도록 관리 사용자를 보호하고 AWS IAM Identity Center활성화하고 생성하십시오 AWS 계정 루트 사용자.

보안을 유지하세요 AWS 계정 루트 사용자

1. Root user를 선택하고 AWS 계정 이메일 주소를 입력하여 계정 AWS Management Console소유자로 로그인합니다. 다음 페이지에서 비밀번호를 입력합니다.

   루트 사용자를 사용하여 로그인하는 데 도움이 필요하면AWS 로그인 사용 설명서의 루트 사용자로 로그인을 참조하세요.
2. 루트 사용자의 다중 인증(MFA)을 활성화합니다.

   지침은 IAM 사용 설명서의 AWS 계정 루트 사용자 (콘솔) 에 대한 가상 MFA 디바이스 활성화를 참조하십시오.

관리자 액세스 권한이 있는 사용자 생성

1. IAM Identity Center를 활성화합니다.

   지침은 AWS IAM Identity Center 사용 설명서의 AWS IAM Identity Center설정을 참조하세요.
2. IAM Identity Center에서 사용자에게 관리 액세스 권한을 부여합니다.

   를 ID 소스로 사용하는 방법에 대한 자습서는 사용 설명서의 기본값으로 IAM Identity Center 디렉터리사용자 액세스 구성을 참조하십시오. IAM Identity Center 디렉터리 AWS IAM Identity Center

관리 액세스 권한이 있는 사용자로 로그인

- IAM IDentity Center 사용자로 로그인하려면 IAM IDentity Center 사용자를 생성할 때 이메일 주소로 전송된 로그인 URL을 사용합니다.

IAM Identity Center 사용자를 사용하여 로그인하는 데 도움이 필요하면 사용 설명서의 AWS 액세스 포털에 로그인을 참조하십시오.AWS 로그인

추가 사용자에게 액세스 권한 할당

1. IAM Identity Center에서 최소 권한 적용 모범 사례를 따르는 권한 세트를 생성합니다.

   지침은AWS IAM Identity Center 사용 설명서의 Create a permission set를 참조하세요.
2. 사용자를 그룹에 할당하고, 그룹에 Single Sign-On 액세스 권한을 할당합니다.

   지침은AWS IAM Identity Center 사용 설명서의 Add groups를 참조하세요.

Amazon Fraud Detector 인터페이스에 액세스할 수 있는 권한을 설정합니다.

Amazon Fraud Detector를 사용하려면 Amazon Fraud Detector 콘솔 및 API 작업에 액세스할 수 있는 권한을 설정하십시오.

보안 모범 사례에 따라 Amazon Fraud Detector 작업에 대한 액세스가 제한되고 필요한 권한을 가진 AWS Identity and Access Management (IAM) 사용자를 생성합니다. 필요하다면 그 밖의 권한을 추가할 수 있습니다.

다음 정책은 Amazon Fraud Detector를 사용하는 데 필요한 권한을 제공합니다.

- `AmazonFraudDetectorFullAccessPolicy`

  다음 작업을 수행할 수 있습니다.
  - 모든 Amazon Fraud Detector 리소스에 액세스
  - 모든 모델 엔드포인트를 나열하고 설명하십시오. SageMaker
  - 계정의 모든 IAM 역할을 나열하십시오.
  - 모든 Amazon S3 버킷을 나열합니다.
  - IAM 전달 역할이 Amazon Fraud Detector에 역할을 전달하도록 허용
- `AmazonS3FullAccess`

  에 대한 전체 액세스를 Amazon Simple Storage Service허용합니다. 이는 Amazon S3에 교육 데이터 세트를 업로드해야 하는 경우 필요합니다.

다음은 IAM 사용자를 생성하고 필요한 권한을 할당하는 방법을 설명합니다.

사용자를 생성하고 필요한 권한을 할당하려면

1. https://console.aws.amazon.com/iam/ 에서 AWS Management Console 로그인하고 IAM 콘솔을 엽니다.
2. 탐색 창에서 사용자(Users)와 사용자 추가(Add user)를 차례로 선택합니다.
3. 사용자 이름(User name)에 **AmazonFraudDetectorUser**를 입력합니다.
4. AWS 관리 콘솔 액세스 확인란을 선택한 다음 사용자 암호를 구성합니다.
5. (선택 사항) 기본적으로 새 사용자가 처음 로그인할 때 새 암호를 생성해야 합니다. AWS 사용자가 다음에 로그인할 때 새 암호를 생성해야 합니다(User must create a new password at next sign-in) 옆에 있는 확인란의 선택을 취소하면 새 사용자가 로그인한 후 암호를 재설정할 수 있습니다.
6. 다음: 권한(Next: Permissions)을 선택합니다.
7. 그룹 생성을 선택합니다.
8. 그룹 이름에는 를 입력합니다. **AmazonFraudDetectorGroup**
9. 정책 목록에서 AmazonFraudDetectorFullAccessPolicy및 FullAccessAmazonS3의 확인란을 선택합니다. 그룹 생성을 선택합니다.
10. 그룹 목록에서 새로운 그룹의 확인란을 선택합니다. 목록에 그룹이 보이지 않으면 [Refresh] 를 선택합니다.
11. 다음: 태그(Next: Tags)를 선택합니다.
12. (선택 사항) 태그를 키 값 페어로 연결하여 메타데이터를 사용자에게 추가합니다. IAM에서 태그를 사용하는 방법에 대한 지침은 IAM 사용자 및 역할 태그 지정을 참조하십시오.
13. 새 사용자에 대한 사용자 세부 정보 및 권한 요약을 보려면 Next: Review를 선택합니다. 계속 진행할 준비가 되었으면 사용자 생성을 선택합니다.

Amazon Fraud Detector에 액세스하기 위한 인터페이스를 설정합니다.

Amazon Fraud Detector 콘솔 또는 AWS SDK를 사용하여 Amazon Fraud Detector에 액세스할 수 있습니다. AWS CLI사용하기 전에 먼저 AWS CLI 및 AWS SDK를 설정해야 합니다.

Amazon Fraud Detector 콘솔에 액세스

를 통해 Amazon Fraud Detector 콘솔 및 기타 AWS 서비스에 액세스할 수 AWS Management Console 있습니다. AWS 계정 Your는 에 대한 액세스 권한을 부여합니다 AWS Management Console.

Amazon Fraud Detector 콘솔에 액세스하려면

1. 으로 https://console.aws.amazon.com/ 이동하여 로그인하십시오 AWS 계정.
2. Amazon Fraud Detector로 이동합니다.

Amazon Fraud Detector 콘솔을 사용하면 모델 및 사기 탐지 리소스 (예: 탐지기, 변수, 이벤트, 개체, 레이블 및 결과) 를 생성하고 관리할 수 있습니다. 예측을 생성하고 모델의 성능 및 예측을 평가할 수 있습니다.

설정 AWS CLI

명령줄 셸에서 명령을 실행하여 AWS Command Line Interface (AWS CLI) 를 사용하여 Amazon Fraud Detector와 상호 작용할 수 있습니다. 최소한의 구성으로 터미널의 명령 프롬프트에서 Amazon Fraud Detector 콘솔에서 제공하는 것과 유사한 기능을 위한 명령을 실행하는 AWS CLI 데 사용할 수 있습니다.

설정하려면 AWS CLI

AWS CLI를 다운로드하고 구성합니다. 지침은 사용 설명서의 다음 항목을 참조하십시오. AWS Command Line Interface

- AWS 명령줄 인터페이스를 사용하여 설정하기
- AWS 명령줄 인터페이스 구성

Amazon Fraud Detector 명령에 대한 자세한 내용은 사용 가능한 명령을 참조하십시오.

AWS SDK 설정

AWS SDK를 사용하여 사기 탐지 리소스를 생성 및 관리하고 사기 예측을 얻기 위한 코드를 작성할 수 있습니다. AWS SDK는 Python (Boto3) 의 JavaScriptAmazon Fraud Detector를 지원합니다.

설정하려면 AWS SDK for Python (Boto3)

를 AWS SDK for Python (Boto3) 사용하여 AWS 서비스를 생성, 구성 및 관리할 수 있습니다. Boto를 설치하는 방법에 대한 지침은 파이썬용AWS SDK (Boto3) 를 참조하십시오. Boto3 SDK 버전 1.14.29 이상을 사용하고 있는지 확인하세요.

설치 AWS SDK for Python (Boto3)후 다음 Python 예제를 실행하여 환경이 올바르게 구성되었는지 확인합니다. 올바르게 구성된 경우 응답에 감지기 목록이 포함됩니다. 탐지기가 생성되지 않은 경우 목록은 비어 있습니다.

```
import boto3
fraudDetector = boto3.client('frauddetector')

response = fraudDetector.get_detectors()
print(response)
```

AWS Java용 SDK를 설정하려면

설치 및 로드 방법에 대한 지침은 SDK 설정을 참조하십시오. AWS SDK for JavaScript JavaScript

Amazon Fraud Detector와 함께 시작하기

시작하기 전에 관련 단계를Amazon Fraud Detector를 통한 사기 탐지 읽고 완료했는지 확인하세요Amazon Fraud Detector 설정.

이 단원의 실습용 자습서를 사용하여 Fraud Detector를 사용하여 Fraud Detector를 사용하여 Fraud Detector를 사용하여 Fraud Detector를 사용하여 Fraud Detector를 사용하여 Fraud Detector를 사용하는 방법을 알아볼 수 있습니다. 이 자습서에서는 기계 학습 모델을 사용하여 새 계정 등록의 사기 여부를 예측하는 사기 분석가의 역할을 맡게 됩니다. 모델은 계정 등록 데이터를 사용하여 학습되어야 합니다. Amazon Fraud Detector는 이 자습서의 예제 계정 등록 데이터 세트를 제공합니다. 튜토리얼을 시작하기 전에 예제 데이터세트를 업로드해야 합니다.

다음 인터페이스 중 하나를 사용하여 Amazon Fraud Detector를 시작할 수 있습니다. 자습서를 시작하기 전에 다음 지침을 따르십시오.예제 데이터세트 가져오기 및 업로드

- 자습서: Amazon Fraud Detector 콘솔 사용 시작하기
- 자습서: 사용 시작하기AWS SDK for Python (Boto3)

예제 데이터세트 가져오기 및 업로드

이 자습서에서 사용하는 예제 데이터셋은 온라인 계정 등록에 대한 세부 정보를 제공합니다. 데이터세트는 UTF-8 형식의 쉼표로 구분된 값 (CSV) 을 사용하는 텍스트 파일에 있습니다. CSV 데이터셋 파일의 첫 번째 행에는 헤더가 포함됩니다. 헤더 행 다음에는 여러 데이터 행이 옵니다. 각 행은 단일 계정 등록의 데이터 요소로 구성됩니다. 데이터는 사용자의 편의를 위해 레이블이 지정되어 있습니다. 데이터세트의 열에는 계정 등록이 사기인지 여부가 나와 있습니다.

예제 데이터세트를 가져오고 업로드하려면

1. 샘플로 이동합니다.

   온라인 계정 등록 데이터가 있는 두 개의 데이터 파일 (registration_data_20K_minimum.csv 및 registration_data_20K_full.csv) 이 있습니다. `registration_data_20K_minimum`파일에는 ip_address 및 email_address 라는 두 개의 변수만 포함됩니다. 파일에는 다른 변수가`registration_data_20K_full` 포함되어 있습니다. 이러한 변수는 각 이벤트에 대한 변수이며 청구_주소, 전화_번호 및 사용자_에이전트를 포함합니다. 두 데이터 파일 모두 두 개의 필수 필드도 포함합니다.

   - EVENT_TIMESTAMP — 이벤트가 발생한 시기를 정의합니다.

- EVENT_LABEL — 이벤트가 시기성인지 합법적인지를 분류합니다.

이 자습서에서는 두 파일 중 하나를 사용할 수 있습니다. 사용할 데이터 파일을 다운로드합니다.

2. Amazon Simple Storage Service(Amazon S3) 버킷을 생성합니다.

   이 단계에서는 데이터세트를 저장할 외부 스토리지를 생성합니다. 이 외부 스토리지는 Amazon S3 버킷입니다. Amazon S3에 대한 자세한 내용은 Amazon S3란 무엇입니까? 단원을 참조하세요.

   a. AWS Management Console에 로그인한 후 https://console.aws.amazon.com/s3/에서 Amazon S3 콘솔을 엽니다.
   b. 버킷에서 버킷 생성을 선택합니다.
   c. 버킷 이름(Bucket Name)에 버킷 이름을 입력합니다. 콘솔의 버킷 이름 지정 규칙을 따르고 전역적으로 고유한 이름을 입력해야 합니다. 버킷의 용도를 설명하는 이름을 사용하는 것이 좋습니다.
   d. 의 AWS 리전경우 버킷을 만들려는AWS 리전 위치를 선택합니다. 선택한 지역은 Amazon Fraud Detector 지원해야 합니다. 지연 시간을 줄이려면 지리적 위치와 가장 가까운 곳을 선택하세요.AWS 리전 Amazon Fraud Detector를 지원하는 지역 목록은 글로벌 인프라 가이드의 지역 표를 참조하십시오.
   e. 객체 소유권, 퍼블릭 액세스 차단을 위한 버킷 설정, 버킷 버전 관리 및 태그에 대한 기본 설정은 이 자습서에 그대로 두십시오.
   f. 이 자습서에서는 [기본 암호화] 에서 [사용 안 함] 을 선택합니다.
   g. 버킷 구성을 검토한 다음 버킷 생성을 선택합니다.

3. Amazon S3 버킷에 예제 데이터 파일을 업로드합니다.

   이제 버킷이 생겼으니 이전에 다운로드한 예제 파일 중 하나를 방금 생성한 Amazon S3 버킷에 업로드하십시오.

   a. 버킷에는 버킷 이름이 나열됩니다. 버킷을 선택합니다.
   b. 업로드를 선택합니다.
   c. 파일 및 폴더에서 파일 추가를 선택합니다.
   d. 컴퓨터에 다운로드한 예제 데이터 파일 중 하나를 선택한 다음 [Open] 을 선택합니다.
   e. 대상, 권한 및 속성에 대한 기본 설정을 그대로 유지합니다.
   f. 구성을 검토한 다음 업로드를 선택합니다.

g. 예제 데이터 파일이 Amazon S3 버킷에 업로드됩니다. 버킷의 위치를 기록해 둡니다. 객체에서 방금 업로드한 예제 데이터 파일을 선택합니다.

h. 객체 개요에서 S3 URI 아래의 위치를 복사합니다. 예제 데이터 파일의 Amazon S3 위치입니다. 나중에 사용합니다. S3 버킷의 Amazon 리소스 이름 (ARN) 을 추가로 복사하여 저장할 수 있습니다.

자습서: Amazon Fraud Detector 콘솔 사용 시작하기

이 튜토리얼은 두 부분으로 구성되어 있습니다. 첫 번째 부분에서는 사기 탐지 모델을 구축, 교육 및 배포하는 방법을 설명합니다. 두 번째 부분에서는 모델을 사용하여 실시간으로 사기 예측을 생성하는 방법을 다룹니다. 모델은 S3 버킷에 업로드한 예제 데이터 파일을 사용하여 학습됩니다. 이 자습서를 마치면 다음 작업을 완료합니다.

- Amazon Fraud Detector 모델 구축 및 교육
- 실시간 Fraud Detector 생성

> **Important**
>
> 계속하기 전에 다음 지침을 따랐는지 확인하십시오.예제 데이터세트 가져오기 및 업로드

파트 A: Amazon Fraud Detector 탐지 모델 구축, 교육 및 배포

파트 A에서는 비즈니스 사용 사례 정의, 이벤트 정의, 모델 구축, 모델 교육, 모델 성능 평가 및 모델 배포를 진행합니다.

1단계: 비즈니스 사용 사례 선택

- 이 단계에서는 데이터 모델 탐색기를 사용하여 비즈니스 사용 사례를 Amazon Fraud Detector에서 지원하는 사기 탐지 모델 유형과 일치시킵니다. 데이터 모델 탐색기는 Amazon Fraud Detector 콘솔과 통합된 도구로, 비즈니스 사용 사례에 맞는 사기 탐지 모델을 만들고 학습하는 데 사용할 모델 유형을 권장합니다. 또한 데이터 모델 탐색기는 데이터세트에 포함해야 하는 필수, 권장 및 선택 데이터 요소에 대한 인사이트를 제공합니다. 데이터세트는 사기 탐지 모델을 만들고 교육하는 데 사용됩니다.

  이 자습서의 목적상 비즈니스 사용 사례는 신규 계정 등록입니다. 비즈니스 사용 사례를 지정하면 데이터 모델 탐색기가 사기 탐지 모델을 만들기 위한 모델 유형을 추천하고 데이터세트를 만드는

데 필요한 데이터 요소 목록도 제공합니다. 새 계정 등록 데이터가 포함된 샘플 데이터세트를 이미 업로드했으므로 새 데이터세트를 만들지 않아도 됩니다.

a. AWS관리 콘솔을 열고 계정에 로그인합니다. Amazon Fraud Detector로 이동합니다.

b. 왼쪽 탐색 창에서 데이터 모델 탐색기를 선택합니다.

c. 데이터 모델 탐색기 페이지의 비즈니스 사용 사례에서 새 계정 사기를 선택합니다.

d. Amazon Fraud Detector는 선택한 비즈니스 사용 사례에 대한 사기 탐지 모델을 생성하는 데 사용할 권장 모델 유형을 표시합니다. 모델 유형은 Amazon Fraud Detector가 사기 탐지 모델을 학습하는 데 사용할 알고리즘, 강화 및 변환을 정의합니다.

   권장 모델 유형을 기록해 둡니다. 나중에 모형을 생성할 때 이 정보가 필요합니다.

e. 데이터 모델 인사이트 패널은 사기 탐지 모델을 만들고 교육하는 데 필요한 필수 및 권장 데이터 요소에 대한 통찰력을 제공합니다.

   다운로드한 샘플 데이터세트를 살펴보고 표에 나열된 필수 및 일부 권장 데이터 요소가 모두 포함되어 있는지 확인하세요.

   나중에 특정 비즈니스 사용 사례에 맞는 모델을 만들 때 제공된 통찰력을 사용하여 데이터세트를 만들게 됩니다.

2단계: 이벤트 유형 생성

- 이 단계에서는 사기 여부를 평가할 비즈니스 활동 (이벤트) 을 정의합니다. 이벤트를 정의하려면 데이터세트에 있는 변수, 엔터티 시작 이벤트 및 이벤트를 분류하는 레이블을 설정해야 합니다. 이 자습서에서는 계정 등록 이벤트를 정의합니다.

  a. AWS관리 콘솔을 열고 계정에 로그인합니다. Amazon Fraud Detector로 이동합니다.

  b. 왼쪽 탐색 창에서 Events를 선택합니다.

  c. 이벤트 유형 페이지에서 생성을 선택합니다.

  d. 이벤트 유형 세부 정보에서 이벤트 유형 이름을 입력하고 필요에 따라 이벤트 설명을 입력합니다.`sample_registration`

  e. 엔티티에서 엔티티 생성을 선택합니다.

  f. 엔티티 생성 페이지에서 엔티티 유형 이름을 입력합니다`sample_customer`. 필요에 따라 개체 유형에 대한 설명을 입력합니다.

  g. 엔티티 생성을 선택합니다.

h. 이벤트 변수에서 이 이벤트 변수를 정의하는 방법 선택에서 훈련 데이터셋에서 변수 선택을 선택합니다.

i. IAM 역할에서 IAM 역할 생성을 선택합니다.

j. IAM 역할 생성 페이지에서 예제 데이터를 업로드한 S3 버킷의 이름을 입력하고 Create role (Create role) 을 선택합니다.

k. 데이터 위치에 예제 데이터의 경로를 입력합니다. 예제 데이터를 업로드한 후 저장한S3 URI 경로입니다. 경로는 다음과 비슷합니다`S3://your-bucket-name/example dataset filename.csv`.

l. 업로드를 선택합니다.

   Amazon Fraud Detector는 예제 데이터 파일에서 헤더를 추출하여 변수 유형으로 매핑합니다. 매핑이 콘솔에 표시됩니다.

m. 레이블 - 선택 사항에서 레이블에서 새 레이블 만들기를 선택합니다.

n. 라벨 생성 페이지에서 이름을 입력합니다`fraud`. 이 레이블은 예제 데이터셋의 사기성 계정 등록을 나타내는 값에 해당합니다.

o. 라벨 생성을 선택합니다.

p. 두 번째 레이블을 만든 다음 이름을 입력합니다`legit`. 이 레이블은 예제 데이터셋의 합법적인 계정 등록을 나타내는 값에 해당합니다.

q. 이벤트 유형 생성을 선택합니다.

3단계: 모델 생성

1. 모델 페이지에서 모델 추가를 선택한 다음 모델 생성을 선택합니다.
2. 1단계 — 모델 세부 정보 정의의 경우 모델 이름으로 를 입력합니다`sample_fraud_detection_model`. 선택적으로 모델에 대한 설명을 추가합니다.
3. 모델 유형에서 온라인 사기 인사이트 모델을 선택합니다.
4. 이벤트 유형에서 샘플_등록을 선택합니다. 1단계에서 생성한 이벤트 유형입니다.
5. 과거 이벤트 데이터에서

   a. 이벤트 데이터 소스에서 S3에 저장된 이벤트 데이터를 선택합니다.

   b. IAM 역할의 경우, 1단계에서 생성한 역할을 선택합니다.

   c. 교육 데이터 위치에서 예제 데이터 파일의 S3 URI 경로를 입력합니다.

6. Next(다음)를 선택합니다.

4단계: 모델 생성

1. 모델 입력에서 모든 체크박스를 선택된 상태로 둡니다. 기본적으로 Amazon Fraud Detector는 이전 이벤트 데이터 세트의 모든 변수를 모델 입력으로 사용합니다.
2. 레이블 분류에서 Fraud 레이블의 경우 Fraud를 선택하십시오. 이 레이블은 예제 데이터세트의 사기 이벤트를 나타내는 값과 일치하기 때문입니다. 합법적인 레이블의 경우 이 레이블이 예제 데이터세트에서 합법적인 이벤트를 나타내는 값과 일치하므로 합법적인 레이블을 선택하십시오.
3. 레이블이 지정되지 않은 이벤트 처리의 경우 이 예제 데이터세트에 대해 기본 선택인 레이블이 지정되지 않은 이벤트 무시를 유지합니다.
4. Next(다음)를 선택합니다.
5. 검토한 후 모델 생성 및 학습을 선택합니다. Amazon Fraud Detector는 모델을 생성하고 모델의 새 버전을 교육하기 시작합니다.

   모델 버전에서 Status 열은 모델 학습 상태를 나타냅니다. 예제 데이터세트를 사용하는 모델 학습은 완료하는 데 약 45분이 걸립니다. 모델 학습이 완료되면 상태가 배포 준비 완료로 변경됩니다.

5단계: 모델 성능 검토

Amazon Fraud Detector를 사용할 때 중요한 단계는 모델 점수 및 성능 메트릭을 사용하여 모델의 정확도를 평가하는 것입니다. 모델 교육이 완료되면 Amazon Fraud Detector는 모델 학습에 사용되지 않은 데이터의 15% 를 사용하여 모델 성능을 검증하고 모델 성능 점수 및 기타 성능 지표를 생성합니다.

1. 모델의 성능을 보려면

   a. Amazon Fraud Detector 콘솔의 왼쪽 탐색 창에서 모형을 선택합니다.

   b. 모델 페이지에서 방금 학습시킨 모델 (sample_fraud_detection_model) 을 선택한 다음 1.0을 선택합니다. 이 버전은 귀하의 모델에서 생성한 Amazon Fraud Detector 버전입니다.
2. 모델 성능 전체 점수와 Amazon Fraud Detector가 이 모델에 대해 생성한 기타 모든 메트릭을 살펴보십시오.

   이 페이지의 모델 성능 점수 및 성능 메트릭에 대해 자세히 알아보려면모델 점수 및 을 참조하십시오모델 성능 지표.

   학습된 모든 Amazon Fraud Detector 모델에는 이 자습서의 모델에 대한 성능 지표와 유사한 실제 사기 탐지 성능 지표가 있을 것으로 예상할 수 있습니다.

6단계: 모델 배포

학습된 모델의 성과 지표를 검토하고 이를 사용하여 사기 예측을 생성할 준비가 되었으면 모델을 배포할 수 있습니다.

1. Amazon Fraud Detector 콘솔의 왼쪽 탐색 창에서 모형을 선택합니다.
2. 모델 페이지에서 sample_fraud_detection_model을 선택한 다음 배포하려는 특정 모델 버전을 선택합니다. 이 자습서에서는 1.0을 선택합니다.
3. 모델 버전 페이지에서 작업을 선택한 다음 모델 버전 배포를 선택합니다.
4. 모델 버전에서 Status는 배포 상태를 보여줍니다. 배포가 완료되면 상태가 Active로 변경됩니다. 이는 모델 버전이 활성화되어 사기 예측을 생성할 수 있음을 나타냅니다. 계속해서 사기 예측 생성 단계를 완료하세요.파트 B: 사기 예측 생성

파트 B: 사기 예측 생성

사기 예측은 비즈니스 활동 (이벤트) 에 대한 사기에 대한 평가입니다. Amazon 사기 탐지기는 탐지기를 사용하여 사기 예측을 생성합니다. 탐지기에는 사기 여부를 평가하려는 특정 이벤트에 대한 탐지 로직 (예: 모델 및 규칙) 이 포함되어 있습니다. 탐지 로직은 규칙을 사용하여 Amazon Fraud Detector에게 모델과 관련된 데이터를 해석하는 방법을 알려줍니다. 이 자습서에서는 이전에 업로드한 계정 등록 예제 데이터세트를 사용하여 계정 등록 이벤트를 평가합니다.

파트 A에서는 모델을 만들고, 학습하고, 배포했습니다. 파트 B에서는`sample_registration` 이벤트 유형에 맞는 탐지기를 구축하고, 배포된 모델을 추가하고, 규칙과 규칙 실행 순서를 만든 다음, 사기 예측을 생성하는 데 사용할 탐지기 버전을 만들고 활성화합니다.

1단계: Detector 생성

검출기를 만들려면

1. Amazon Fraud Detector 콘솔의 왼쪽 탐색 창에서 Detector를 선택합니다.
2. 감지기 생성을 선택합니다.
3. 탐지기 세부 정보 정의 페이지에서 검출기 이름을`sample_detector` 입력합니다. 선택적으로, 검출기에 대한 설명 (예:) 을 입력합니다`my sample fraud detector`.
4. 이벤트 유형에서 샘플_등록을 선택합니다. 이 이벤트는 이 자습서의 파트 A에서 만든 이벤트입니다.
5. Next(다음)를 선택합니다.

2단계: 모델 추가

이 자습서의 파트 A를 완료했다면 탐지기에 추가할 수 있는 Amazon Fraud Detector 모델이 이미 있을 것입니다. 아직 모델을 만들지 않았다면 파트 A로 이동하여 모델 생성, 학습 및 배포 단계를 완료한 다음 파트 B를 계속 진행하세요.

1. 모델 추가 - 선택 사항에서 모델 추가를 선택합니다.
2. 모델 추가 페이지의 모델 선택에서 이전에 배포한 Amazon Fraud Detector 모델 이름을 선택합니다. 버전 선택에서 배포된 모델의 모델 버전을 선택합니다.
3. 모델 추가를 선택합니다.
4. Next(다음)를 선택합니다.

3단계: 규칙 추가

규칙은 Fraud Detector에 Fraud Detector에 Fraud Detector에 평가 시 모델 성능 점수를 해석하는 방법을 알려주는 조건입니다. 이 자습서에서는`high_fraud_risk`, `medium_fraud_risk`, 및 라는 세 가지 규칙을 만듭니다`low_fraud_risk`.

1. 규칙 추가 페이지의 규칙 정의에서 규칙 이름을 입력하고`high_fraud_risk` 설명 - 선택 사항에서 규칙에 대한 설명을 입력합니다**`This rule captures events with a high ML model score`**.
2. 표현식에 Amazon Fraud Detector의 단순화된 규칙 표현식 언어를 사용하여 다음 규칙 표현식을 입력합니다.

   ```
   $sample_fraud_detection_model_insightscore > 900
   ```
3. 결과에서 새 결과 만들기를 선택합니다. 결과는 사기 예측의 결과이며 평가 중에 규칙이 일치하면 반환됩니다.
4. 새 결과 만들기에서`verify_customer` 결과 이름을 입력합니다. 설명을 입력할 수도 있습니다.
5. 결과 저장을 선택합니다.
6. 규칙 추가를 선택하여 규칙 검증 검사기를 실행하고 규칙을 저장합니다. Amazon Fraud Detector는 규칙을 생성한 후 해당 규칙을 탐지기에서 사용할 수 있도록 합니다.
7. 다른 규칙 추가를 선택한 다음 규칙 만들기 탭을 선택합니다.
8. 이 프로세스를 두 번 더 반복하여 다음`low_fraud_risk` 규칙 세부 정보를 사용하여`medium_fraud_risk` 및 규칙을 생성합니다.

   - 중간의_사기_위험

규칙 이름:`medium_fraud_risk`

결과:`review`

표현식:

```
$sample_fraud_detection_model_insightscore <= 900 and

$sample_fraud_detection_model_insightscore > 700
```

- 낮은 사기 위험

  규칙 이름:`low_fraud_risk`

  결과:`approve`

  표현식:

```
$sample_fraud_detection_model_insightscore <= 700
```

이 값은 이 튜토리얼에서 사용된 예제입니다. 자체 검출기에 대한 규칙을 만들 때는 모델 및 사용 사례에 적합한 값을 사용하십시오.

9. 세 가지 규칙을 모두 생성한 후 [Next] 를 선택합니다.

   규칙 작성 및 작성에 대한 자세한 내용은규칙 및 을 참조하십시오규칙 언어 참조.

4단계: 규칙 실행 및 규칙 순서 구성

탐지기에 포함된 규칙의 규칙 실행 모드는 정의한 모든 규칙을 평가할지 또는 일치하는 첫 번째 규칙에서 규칙 평가가 중지되는지를 결정합니다. 그리고 규칙 순서에 따라 규칙을 실행할 순서가 결정됩니다.

기본 규칙 실행 모드는 입니다`FIRST_MATCHED`.

첫 매치

첫 번째 일치 규칙 실행 모드는 정의된 규칙 순서에 따라 첫 번째 일치 규칙의 결과를 반환합니다. `FIRST_MATCHED`를 지정하면 Amazon Fraud Detector는 처음부터 마지막까지 순차적으로 규칙을 평가하고 처음 일치하는 규칙에서 중지합니다. 그러면 Amazon Fraud Detector가 해당 단일 규칙에 대한 결과를 제공합니다.

규칙을 실행하는 순서는 부정 행위 예측 결과에 영향을 미칠 수 있습니다. 규칙을 생성한 후에는 다음 단계에 따라 원하는 순서로 규칙을 실행하도록 규칙을 재정렬하십시오.

high_fraud_risk규칙이 아직 규칙 목록 상단에 없는 경우 순서를 선택한 다음 1을 선택합니다. 그러면 첫 번째high_fraud_risk 위치로 이동합니다.

이 과정을 반복하여medium_fraud_risk 규칙이 두 번째 위치에 있고low_fraud_risk 규칙이 세 번째 위치에 오도록 합니다.

모두 일치함

일치하는 모든 규칙 실행 모드는 규칙 순서에 관계없이 일치하는 모든 규칙에 대한 결과를 반환합니다. ALL_MATCHED지정하면 Amazon Fraud Detector가 모든 규칙을 평가하고 일치하는 모든 규칙에 대한 결과를 반환합니다.

이 자습서를 선택하고 다음을 선택합니다.FIRST_MATCHED

5단계: Detector 버전 검토 및 생성

탐지기 버전은 사기 예측을 생성하는 데 사용되는 특정 모델 및 규칙을 정의합니다.

1. 검토 및 생성 페이지에서 구성한 감지기 세부 정보, 모델 및 규칙을 검토하십시오. 변경이 필요한 경우 해당 섹션 옆의 편집을 선택합니다.
2. 감지기 생성을 선택합니다. 검출기가 생성되면 검출기의 첫 번째 버전이Draft 상태와 함께 Detector 버전 테이블에 나타납니다.

   Draft 버전을 사용하여 탐지기를 테스트할 수 있습니다.

6단계: Detector 버전 테스트 및 활성화

Amazon Fraud Detector 콘솔에서는 테스트 실행 기능이 있는 모의 데이터를 사용하여 탐지기의 로직을 테스트할 수 있습니다. 이 자습서에서는 예제 데이터세트의 계정 등록 데이터를 사용할 수 있습니다.

1. Detector 버전 세부 정보 페이지 하단에서 테스트 실행으로 스크롤합니다.
2. 이벤트 메타데이터에는 이벤트가 발생한 시점의 타임스탬프를 입력하고 이벤트를 수행하는 개체의 고유 식별자를 입력합니다. 이 자습서에서는 타임스탬프에 대한 날짜 선택기에서 날짜를 선택하고 엔티티 ID로 “1234"를 입력합니다.
3. 이벤트 변수에 테스트하려는 변수 값을 입력합니다. 이 자습서에서는ip_address 및email_address 필드만 필요합니다. 이는 Amazon Fraud Detector 모델을 학습하는 데 사용되

는 입력이기 때문입니다. 다음 예제 값을 사용할 수 있습니다. 여기서는 제안된 변수 이름을 사용했다고 가정합니다.

- 아이피_주소:205.251.233.178
- 이메일_주소:johndoe@exampledomain.com

4. 테스트 실행을 선택합니다.
5. Amazon Fraud Detector는 규칙 실행 모드를 기반으로 사기 예측 결과를 반환합니다. 규칙 실행 모드가FIRST_MATCHED 인 경우 반환된 결과는 일치하는 첫 번째 규칙에 해당합니다. 첫 번째 규칙은 우선 순위가 가장 높은 규칙입니다. 사실로 평가되면 일치하는 것으로 간주됩니다. 규칙 실행 모드가ALL_MATCHED 인 경우 반환된 결과는 일치하는 모든 규칙에 해당합니다. 즉, 모두 사실로 평가된다는 뜻입니다. 또한 Amazon Fraud Detector는 탐지기에 추가된 모든 모델의 모델 점수를 반환합니다.

   입력을 변경하고 몇 가지 테스트를 실행하여 다양한 결과를 확인할 수 있습니다. 예제 데이터세트의 ip_address 및 email_address 값을 테스트에 사용하고 결과가 예상과 같은지 확인할 수 있습니다.
6. 검출기의 작동 방식에 만족하면 에서Draft 로 승격시키세요Active. 이렇게 하면 탐지기를 실시간 사기 탐지에 사용할 수 있습니다.

   Detector 버전 세부 정보 페이지에서 작업, 게시, 게시 버전을 선택합니다. 그러면 감지기 상태가 드래프트에서 활성으로 변경됩니다.

   이제 모델 및 관련 탐지 로직은 Amazon Fraud DetectorGetEventPrediction API를 사용하여 사기 행위에 대한 온라인 활동을 실시간으로 평가할 준비가 되었습니다. CSV 입력 파일과CreateBatchPredictionJob API를 사용하여 이벤트를 오프라인으로 평가할 수도 있습니다. Fraud Detector에 대한 자세한 내용은 단원을 참조하세요.사기 예측

이 자습서를 완료하면 다음 작업을 수행했습니다.

- Amazon S3에 예제 이벤트 데이터 세트를 업로드했습니다.
- 예제 데이터 세트를 사용하여 Amazon Fraud Detector 사기 탐지 모델을 만들고 교육했습니다.
- Amazon Fraud Detector가 생성한 모델 성능 점수 및 기타 성능 지표를 확인했습니다.
- 사기 탐지 모델을 배포했습니다.
- 탐지기를 만들고 배포된 모델을 추가했습니다.
- 탐지기에 규칙, 규칙 실행 순서 및 결과를 추가했습니다.

- 다양한 입력을 제공하고 규칙 및 규칙 실행 순서가 예상대로 작동하는지 확인하여 탐지기를 테스트했습니다.
- 탐지기를 게시하여 활성화했습니다.

자습서: 사용 시작하기AWS SDK for Python (Boto3)

이 자습서에서는 Amazon Fraud Detector 모델을 구축 및 교육한 다음 이 모델을 사용하여 실시간 사기 예측을 생성하는 방법을 설명합니다AWS SDK for Python (Boto3). 모델은 Amazon S3 버킷에 업로드한 계정 등록 예제 데이터 파일을 사용하여 학습됩니다.

이 자습서를 마치면 다음 작업을 완료합니다.

- Amazon Fraud Detector 모델 구축 및 교육
- 실시간 사기 예측 생성

사전 조건

다음은 이 자습서의 필수 단계입니다.

- 완료Amazon Fraud Detector 설정.

  이미AWS SDK 설정 사용하고 있다면 Boto3 SDK 버전 1.14.29 이상을 사용하고 있는지 확인하세요.
- 지침에 따라 이 자습서에 필요한예제 데이터세트 가져오기 및 업로드 파일을 제출했습니다.

시작하기

1단계: Pyon을 설정하고 확인합니다

Boto는 파이썬용 Amazon Web Services (AWS) SDK입니다. 이를 사용하여 작성, 구성 및 관리할 수 AWS 서비스 있습니다. Boto3를 설치하는 방법에 대한 지침은 Boto3 를 참조하십시오.

설치AWS SDK for Python (Boto3) 후 다음 Python 예제 명령을 실행하여 환경이 올바르게 구성되었는지 확인합니다. 환경이 올바르게 구성된 경우 응답에 탐지기 목록이 포함됩니다. 감지기를 생성하지 않은 경우 목록이 비어 있습니다.

```
import boto3
fraudDetector = boto3.client('frauddetector')
```

```
response = fraudDetector.get_detectors()
print(response)
```

2단계: 변수, 개체 유형 및 레이블 만들기

이 단계에서는 모델, 이벤트 및 규칙을 정의하는 데 사용되는 리소스를 생성합니다.

변수 생성

변수는 이벤트 유형, 모델 및 규칙을 만드는 데 사용하려는 데이터세트의 데이터 요소입니다.

다음 예에서는 CreateVariableAPI를 사용하여 두 개의 변수를 만듭니다. 변수는email_address 및ip_address 입니다. 해당 변수 유형 (EMAIL_ADDRESS및) 에IP_ADDRESS 할당합니다. 이러한 변수는 업로드한 예제 데이터세트의 일부입니다. 변수 유형을 지정하면 Amazon Fraud Detector는 모델 학습 중 및 예측을 받을 때 변수를 해석합니다. 관련 변수 유형이 있는 변수만 모델 학습에 사용할 수 있습니다.

```
import boto3
fraudDetector = boto3.client('frauddetector')

 #Create variable email_address
fraudDetector.create_variable(
    name = 'email_address',
    variableType = 'EMAIL_ADDRESS',
    dataSource = 'EVENT',
    dataType = 'STRING',
    defaultValue = '<unknown>'
)

#Create variable ip_address
fraudDetector.create_variable(
    name = 'ip_address',
    variableType = 'IP_ADDRESS',
    dataSource = 'EVENT',
    dataType = 'STRING',
    defaultValue = '<unknown>'
)
```

엔티티 유형 생성

엔터티는 이벤트를 수행 중인 사용자를 나타내고 엔터티 유형은 엔터티를 분류합니다. 분류의 예로는 고객, 판매자 또는 계정이 있습니다.

다음 예에서는 PutEntityTypeAPI를 사용하여sample_customer 엔티티 유형을 만듭니다.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.put_entity_type(
   name = 'sample_customer',
   description = 'sample customer entity type'
)
```

라벨 생성

레이블은 이벤트가 시기성인지 합법적인지를 분류하고 사기 탐지 모델을 훈련하는 데 사용됩니다. 모델은 이러한 레이블 값을 사용하여 이벤트를 분류하는 방법을 학습합니다.

다음 예제에서는 Putlabel API를 사용하여 두 개의 레이블fraud 및 를 생성합니다legit.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.put_label(
    name = 'fraud',
    description = 'label for fraud events'
)

fraudDetector.put_label(
    name = 'legit',
    description = 'label for legitimate events'
)
```

3단계: 이벤트 유형 생성

Amazon Fraud Detector를 사용하면 위험을 평가하고 개별 이벤트에 대한 사기 예측을 생성하는 모델을 구축할 수 있습니다. 이벤트 유형은 개별 이벤트의 구조를 정의합니다.

다음 예에서는 PutEventTypeAPI를 사용하여 이벤트 유형을 만듭니다sample_registration. 이전 단계에서 만든 변수 (email_address,ip_address), 개체 유형 (sample_customer) 및 레이블 (fraud,legit) 을 지정하여 이벤트 유형을 정의합니다.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.put_event_type (
     name = 'sample_registration',
     eventVariables = ['ip_address', 'email_address'],
     labels = ['legit', 'fraud'],
     entityTypes = ['sample_customer'])
```

4단계: 모델 생성, 학습 및 배포

Amazon Fraud Detector는 모델을 학습시켜 특정 이벤트 유형의 사기 행위를 탐지하는 방법을 학습시킵니다. 이전 단계에서 이벤트 유형을 생성했습니다. 이 단계에서는 이벤트 유형에 맞는 모델을 생성하고 훈련합니다. 모델은 모델 버전의 컨테이너 역할을 합니다. 모델을 트레이닝할 때마다 새 버전이 생성됩니다.

다음 예제 코드를 사용하여 온라인 사기 인사이트 모델을 만들고 학습시키십시오. 이 모델을`sample_fraud_detection_model` 호출합니다. Amazon S3에 업로드한 계정 등록 예제 데이터 세트를`sample_registration` 사용하는 이벤트 유형용입니다.

Amazon Fraud Detector가 지원하는 다양한 모델 유형에 대한 자세한 내용은 을 참조하십시오모델 유형을 선택하세요..

모델 생성

다음 예에서는 CreateModelAPI를 사용하여 모델을 생성합니다.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.create_model (
       modelId = 'sample_fraud_detection_model',
       eventTypeName = 'sample_registration',
       modelType = 'ONLINE_FRAUD_INSIGHTS')
```

모델 트레이닝

다음 예에서는 CreateModelVersionAPI를 사용하여 모델 훈련에 사용됩니다. 예제 데이터 세트를 저장한 Amazon S3 위치와 Amazon S3 RoleArn버킷의 위치를`'EXTERNAL_EVENTS'` 지정하십시오`externalEventsDetail.trainingDataSource` `trainingDataSchema`파라미터로 Amazon

Fraud Detector가 예제 데이터를 해석하는 방법을 지정하십시오. 보다 구체적으로 말하자면, 포함할 변수와 이벤트 레이블을 분류하는 방법을 지정하십시오.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.create_model_version (
         modelId = 'sample_fraud_detection_model',
         modelType = 'ONLINE_FRAUD_INSIGHTS',
         trainingDataSource = 'EXTERNAL_EVENTS',
         trainingDataSchema = {
            'modelVariables' : ['ip_address', 'email_address'],
            'labelSchema' : {
               'labelMapper' : {
                   'FRAUD' : ['fraud'],
                   'LEGIT' : ['legit']
        }
    }
},
         externalEventsDetail = {
              'dataLocation' : 's3://your-S3-bucket-name/your-example-data-
filename.csv',
              'dataAccessRoleArn' : 'role_arn'
}
)
```

모델을 여러 번 트레이닝할 수 있습니다. 모델을 학습시킬 때마다 새 버전이 생성됩니다. 모델 학습이 완료되면 모델 버전 상태가 로TRAINING_COMPLETE 업데이트됩니다. 모델 성능 점수 및 기타 모델 성능 지표를 검토할 수 있습니다.

모델 성능 검토

Amazon Fraud Detector를 사용할 때 중요한 단계는 모델 점수 및 성능 지표를 사용하여 모델의 정확도를 평가하는 것입니다. 모델 학습이 완료되면 Amazon Fraud Detector는 모델 학습에 사용되지 않은 데이터의 15% 를 사용하여 모델 성능을 검증합니다. 모델 성능 점수 및 기타 성능 메트릭을 생성합니다.

DescribeModelVersionsAPI를 사용하여 모델 성능을 검토하세요. 이 모델에 대해 Amazon Fraud Detector에서 생성한 모델 성능 전체 점수와 기타 모든 지표를 살펴보십시오.

모델 성능 점수 및 성능 지표에 대한 자세한 내용은모델 점수 및 을 참조하십시오모델 성능 지표.

학습된 모든 Amazon Fraud Detector 모델에는 이 자습서의 지표와 유사한 실제 사기 탐지 성능 지표가 있을 것으로 예상할 수 있습니다.

모델 배포

학습된 모델의 성능 지표를 검토한 후 모델을 배포하고 Amazon Fraud Detector에서 이를 사용하여 사기 예측을 생성할 수 있도록 하십시오. 학습된 모델을 배포하려면 UpdateModelVersionStatusAPI를 사용하십시오. 다음 예에서는 모델 버전 상태를 ACTIVE로 업데이트하는 데 사용됩니다.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.update_model_version_status (
     modelId = 'sample_fraud_detection_model',
     modelType = 'ONLINE_FRAUD_INSIGHTS',
     modelVersionNumber = '1.00',
     status = 'ACTIVE'
)
```

5단계: 탐지기, 결과, 규칙 및 탐지기 버전 생성

검출기에는 모델 및 규칙과 같은 탐지 로직이 포함됩니다. 이 로직은 사기 여부를 평가하려는 특정 이벤트에 대한 것입니다. 규칙은 예측 중 변수 값을 해석하는 방법을 Amazon Fraud Detector에 알려주기 위해 지정하는 조건입니다. 결과는 사기 예측의 결과입니다. 감지기에는 여러 버전이 있을 수 있으며 각 버전에는 드래프트, 활성 또는 비활성 상태가 있습니다. 탐지기 버전에는 적어도 하나의 탐지기 버전과 관련된 규칙이 있어야 합니다.

다음 예제 코드를 사용하여 탐지기, 규칙, 결과를 생성하고 탐지기를 게시할 수 있습니다.

탐지기 생성

다음 예에서는 PutDetectorAPI를 사용하여`sample_registration` 이벤트 유형에 대한`sample_detector` 감지기를 만듭니다.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.put_detector (
     detectorId = 'sample_detector',
     eventTypeName = 'sample_registration'
)
```

성과 창출

가능한 각 사기 예측 결과에 대한 결과가 생성됩니다. 다음 예에서는 PutOutcomeAPI를 사용하여 세 가지 결과 (verify_customerreview, 및) 를 생성합니다approve. 이러한 결과는 나중에 규칙에 할당됩니다.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.put_outcome(
     name = 'verify_customer',
     description = 'this outcome initiates a verification workflow'
    )

fraudDetector.put_outcome(
     name = 'review',
     description = 'this outcome sidelines event for review'
    )

fraudDetector.put_outcome(
     name = 'approve',
     description = 'this outcome approves the event'
)
```

규칙 생성

규칙은 데이터세트의 하나 이상의 변수, 논리 표현식 및 하나 이상의 결과로 구성됩니다.

다음 예에서는 CreateRuleAPI를 사용하여 세 가지 다른 규칙 (high_riskmedium_risk, 및) 을 생성합니다low_risk. 규칙 표현식을 생성하여 모델 성능 점수sample_fraud_detection_model_insightscore 값을 다양한 임계값과 비교합니다. 이는 이벤트의 위험 수준을 결정하고 이전 단계에서 정의한 결과를 할당하기 위한 것입니다.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.create_rule(
     ruleId = 'high_fraud_risk',
     detectorId = 'sample_detector',
     expression = '$sample_fraud_detection_model_insightscore > 900',
     language = 'DETECTORPL',
     outcomes = ['verify_customer']
```

```
     )

fraudDetector.create_rule(
     ruleId = 'medium_fraud_risk',
     detectorId = 'sample_detector',
     expression = '$sample_fraud_detection_model_insightscore <= 900 and
 $sample_fraud_detection_model_insightscore > 700',
     language = 'DETECTORPL',
     outcomes = ['review']
     )

fraudDetector.create_rule(
     ruleId = 'low_fraud_risk',
     detectorId = 'sample_detector',
     expression = '$sample_fraud_detection_model_insightscore <= 700',
     language = 'DETECTORPL',
     outcomes = ['approve']
     )
```

탐지기 버전 생성

탐지기 버전은 사기 예측에 사용되는 모델 및 규칙을 정의합니다.

다음 예에서는 CreateDetectorVersionAPI를 사용하여 감지기 버전을 생성합니다. 이를 위해 모델 버전 세부 정보, 규칙 및 규칙 실행 모드 FIRST_MATCHED를 제공합니다. 규칙 실행 모드는 규칙 평가 순서를 지정합니다. 규칙 실행 모드 FIRST_MATCHED 는 처음부터 마지막까지 순차적으로 규칙을 평가하고 처음 일치하는 규칙에서 중지합니다.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.create_detector_version(
      detectorId = 'sample_detector',
      rules = [{
          'detectorId' : 'sample_detector',
          'ruleId' : 'high_fraud_risk',
          'ruleVersion' : '1'
},
{
          'detectorId' : 'sample_detector',
          'ruleId' : 'medium_fraud_risk',
```

```
          'ruleVersion' : '1'
},
{
          'detectorId' : 'sample_detector',
          'ruleId' : 'low_fraud_risk',
          'ruleVersion' : '1'
}
],
      modelVersions = [{
          'modelId' : 'sample_fraud_detection_model',
          'modelType': 'ONLINE_FRAUD_INSIGHTS',
          'modelVersionNumber' : '1.00'
}      ],
      ruleExecutionMode = 'FIRST_MATCHED'
)
```

6단계: 사기 예측 생성

이 자습서의 마지막 단계에서는 이전 단계에서sample_detector 만든 탐지기를 사용하여sample_registration 이벤트 유형에 대한 사기 예측을 실시간으로 생성합니다. 탐지기는 Amazon S3에 업로드된 예제 데이터를 평가합니다. 응답에는 모델 성과 점수와 일치하는 규칙과 관련된 모든 결과가 포함됩니다.

다음 예에서는 GetEventPredictionAPI를 사용하여 각 요청과 함께 단일 계정 등록의 데이터를 제공합니다. 이 자습서에서는 계정 등록 예제 데이터 파일에서 데이터 (이메일_주소 및 ip_address) 를 가져옵니다. 상단 헤더 줄 뒤의 각 줄 (행) 은 단일 계정 등록 이벤트의 데이터를 나타냅니다.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.get_event_prediction(
      detectorId = 'sample_detector',
      eventId = '802454d3-f7d8-482d-97e8-c4b6db9a0428',
      eventTypeName = 'sample_registration',
      eventTimestamp = '2020-07-13T23:18:21Z',
      entities = [{'entityType':'sample_customer', 'entityId':'12345'}],
 eventVariables = {
      'email_address': 'johndoe@exampledomain.com',
      'ip_address': '1.2.3.4'
}
)
```

이 자습서를 완료한 후 다음 작업을 수행했습니다.

- Amazon S33에 예제 이벤트 데이터 세트를 업로드했습니다.
- 모델을 만들고 학습하는 데 사용되는 변수, 개체 및 레이블을 생성했습니다.
- 예제 데이터세트를 사용하여 모델을 만들고 학습했습니다.
- Amazon Fraud Detector가 생성한 모델 성능 점수 및 기타 성능 지표를 확인했습니다.
- 사기 탐지 모델을 배포했습니다.
- 탐지기를 만들고 배포된 모델을 추가했습니다.
- 탐지기에 규칙, 규칙 실행 순서 및 결과를 추가했습니다.
- 탐지기 버전을 생성했습니다.
- 다양한 입력을 제공하고 규칙 및 규칙 실행 순서가 예상대로 작동하는지 확인하여 탐지기를 테스트했습니다.

(선택 사항) Jupyter (iPython) 노트북으로 아마존 Fraud Detector API를 살펴보세요.

Amazon Fraud Detector API를 사용하는 방법에 대한 추가 예제는 aws-fraud-detector-samples GitHub 리포지토리를 참조하십시오. 노트북에서 다루는 주제에는 Amazon Fraud Detector API를 사용한 모델 및 탐지기 구축과`GetEventPrediction` API를 사용한 일괄 사기 예측 요청이 포함됩니다.

다음 단계

이제 모델과 탐지기를 만들었으니 더 자세히 살펴보고 모델 및 탐지기를 만들고 사기 예측을 생성할 수 있습니다.

Amazon Fraud Detector 사용 설명서의 다음 섹션에서는 비즈니스 또는 조직에서 Amazon Fraud Detector를 사용하여 사기를 탐지하는 방법을 설명합니다.

- 모델 훈련을 위한 이벤트 데이터세트를 준비하고 생성합니다.
- 이벤트 유형 생성
- 모델 생성
- 감지기 생성
- 사기 예측 받기

- Amazon Fraud Detector 리소스 (특히 변수, 항목, 결과 및 레이블) 를 관리합니다.
- 보안 및 규정 준수 목표에 맞게 Amazon Fraud Detector를 구성합니다.
- 아마존 Fraud Detector 모니터링 및 아마존 Fraud Detector API 호출 기록
- Amazon Fraud Detector에 대한 문제 해결

이벤트 데이터 세트

이벤트 데이터셋은 회사의 과거 부정 행위 데이터입니다. 이 데이터를 Amazon Fraud Detector에 제공하여 사기 탐지 모델을 생성합니다.

Amazon Fraud Detector는 기계 학습 모델을 사용하여 사기 예측을 생성합니다. 각 모델은 모델 유형을 사용하여 학습됩니다. 모델 유형은 모델 학습에 사용되는 알고리즘 및 변환을 지정합니다. 모델 트레이닝은 사용자가 제공한 데이터세트를 사용하여 사기 이벤트를 예측할 수 있는 모델을 만드는 프로세스입니다. 자세한 내용은 Amazon Fraud Detector의 작동 방식을 참조하세요.

사기 탐지 모델을 만드는 데 사용되는 데이터셋은 이벤트의 세부 정보를 제공합니다. 이벤트는 사기 위험에 대한 평가가 이루어지는 비즈니스 활동입니다. 예를 들어, 계정 등록은 이벤트일 수 있습니다. 계정 등록 이벤트와 관련된 데이터는 이벤트 데이터셋일 수 있습니다. Amazon Fraud Detector는 이 데이터 세트를 사용하여 계정 등록 사기를 평가합니다.

모델을 생성하기 위해 Amazon Fraud Detector에 데이터 세트를 제공하기 전에 모델 생성 목표를 정의해야 합니다. 또한 모델을 어떻게 사용할지 결정하고 특정 요구 사항에 따라 모델의 성능을 평가하기 위한 메트릭을 정의해야 합니다.

예를 들어, 계정 등록 사기를 평가하는 사기 탐지 모델을 만들기 위한 목표는 다음과 같을 수 있습니다.

- 합법적인 등록을 자동 승인하기 위해서입니다.
- 나중에 조사할 수 있도록 허위 등록을 캡처하기 위해서입니다.

목표를 결정했으면 다음 단계는 모델을 어떻게 사용할지 결정하는 것입니다. 사기 탐지 모델을 사용하여 등록 사기를 평가하는 몇 가지 예는 다음과 같습니다.

- 각 계정 등록에 대한 실시간 사기 탐지를 위한 것입니다.
- 매시간 모든 계정 등록에 대한 오프라인 평가용.

모델의 성능을 측정하는 데 사용할 수 있는 몇 가지 지표의 예는 다음과 같습니다.

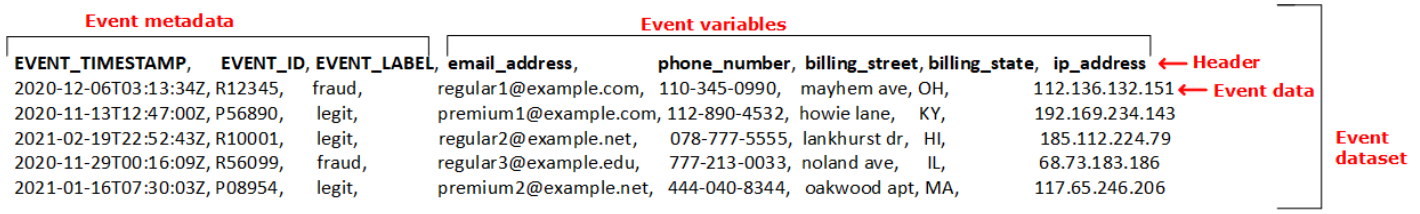
- 현재 생산 기준선보다 지속적으로 더 나은 성능을 발휘합니다.
- Y%의 오탐률로 X% 사기 등록을 캡처합니다.
- 자동 승인된 허위 등록의 최대 5% 를 수락합니다.

이벤트 데이터 세트 구조

Amazon Fraud Detector를 사용하려면 UTF-8 형식의 쉼표로 구분된 값 (CSV) 을 사용하는 텍스트 파일로 이벤트 데이터 세트를 제공해야 합니다. CSV 데이터세트 파일의 첫 번째 줄에는 파일 헤더가 포함되어야 합니다. 파일 헤더는 이벤트와 관련된 각 데이터 요소를 설명하는 이벤트 메타데이터와 이벤트 변수로 구성됩니다. 헤더 다음에는 이벤트 데이터가 옵니다. 각 라인은 단일 이벤트의 데이터 요소로 구성됩니다.

- 이벤트 메타데이터 - 이벤트에 대한 정보를 제공합니다. 예를 들어 EVENT_TIMESTAMP는 이벤트가 발생한 시간을 지정하는 이벤트 메타데이터입니다. 비즈니스 사용 사례와 사기 탐지 모델을 만들고 학습하는 데 사용한 모델 유형에 따라 Amazon Fraud Detector에서는 특정 이벤트 메타데이터를 제공해야 합니다. CSV 파일 헤더에 이벤트 메타데이터를 지정할 때는 Amazon Fraud Detector에서 지정한 것과 동일한 이벤트 메타데이터 이름을 사용하고 대문자만 사용하십시오.
- 이벤트 변수 - 사기 탐지 모델을 만들고 학습하는 데 사용할 이벤트와 관련된 데이터 요소를 나타냅니다. 비즈니스 사용 사례와 사기 탐지 모델을 만들고 학습하는 데 사용한 모델 유형에 따라 Amazon Fraud Detector에서는 특정 이벤트 변수를 제공하도록 요구하거나 권장할 수 있습니다. 모델 학습에 포함하려는 이벤트의 다른 이벤트 변수를 선택적으로 제공할 수도 있습니다. 온라인 등록 이벤트에 대한 이벤트 변수의 예로는 이메일 주소, IP 주소 및 전화번호가 있습니다. CSV 파일 헤더에 이벤트 변수 이름을 지정할 때는 원하는 변수 이름을 사용하고 소문자만 사용하십시오.
- 이벤트 데이터 - 실제 이벤트에서 수집된 데이터를 나타냅니다. CSV 파일에서 파일 헤더 뒤의 각 행은 단일 이벤트의 데이터 요소로 구성됩니다. 예를 들어, 온라인 등록 이벤트 데이터 파일의 각 행에는 단일 등록의 데이터가 포함됩니다. 행의 각 데이터 요소는 해당 이벤트 메타데이터 또는 이벤트 변수와 일치해야 합니다.

다음은 계정 등록 이벤트의 데이터를 포함하는 CSV 파일의 예입니다. 헤더 행에는 이벤트 메타데이터가 대문자로 표시되고 이벤트 변수가 소문자로 표시되고 그 뒤에 이벤트 데이터가 표시됩니다. 데이터 세트의 각 행에는 단일 계정 등록과 관련된 데이터 요소가 포함되며 각 데이터 요소는 헤더에 해당합니다.



데이터 모델 탐색기를 사용하여 이벤트 데이터세트 요구 사항 가져오기

모델을 생성하기 위해 선택한 모델 유형에 따라 데이터세트의 요구 사항이 정의됩니다. Amazon Fraud Detector는 사용자가 제공한 데이터 세트를 사용하여 사기 탐지 모델을 생성하고 교육합니다. Amazon Fraud Detector는 모델 생성을 시작하기 전에 데이터세트가 크기, 형식 및 기타 요구 사항을 충족하는지 확인합니다. 데이터세트가 요구 사항을 충족하지 않으면 모델 생성 및 학습이 실패합니다. 데이터 모델 탐색기를 사용하여 비즈니스 사용 사례에 사용할 모델 유형을 식별하고 식별된 모델 유형에 대한 데이터세트 요구 사항을 파악할 수 있습니다.

데이터 모델 탐색기

데이터 모델 탐색기는 Amazon Fraud Detector 콘솔의 도구로, Amazon Fraud Detector에서 지원하는 모델 유형에 맞게 비즈니스 사용 사례를 조정합니다. 또한 데이터 모델 탐색기는 Amazon Fraud Detector가 사기 탐지 모델을 생성하는 데 필요한 데이터 요소에 대한 통찰력을 제공합니다. 이벤트 데이터세트를 준비하기 전에 데이터 모델 탐색기를 사용하여 Amazon Fraud Detector가 업무용으로 권장하는 모델 유형을 파악하고 데이터세트를 생성하는 데 필요한 필수, 권장 및 선택 데이터 요소 목록을 확인하십시오.

데이터 모델 탐색기를 사용하려면

1. AWS관리 콘솔을 열고 계정에 로그인합니다. Amazon Fraud Detector로 이동합니다.
2. 왼쪽 탐색 창에서 데이터 모델 탐색기를 선택합니다.
3. 데이터 모델 탐색기 페이지의 비즈니스 사용 사례에서 사기 위험을 평가하려는 비즈니스 사용 사례를 선택합니다.
4. Amazon Fraud Detector는 비즈니스 사용 사례에 맞는 권장 모델 유형을 표시합니다. 모델 유형은 Amazon Fraud Detector가 사기 탐지 모델을 학습하는 데 사용할 알고리즘, 강화 및 변환을 정의합니다.

   권장 모델 유형을 기록해 둡니다. 나중에 모형을 생성할 때 이 정보가 필요합니다.

> **Note**
>
> 비즈니스 사용 사례를 찾을 수 없는 경우 설명에 있는 문의하기 링크를 사용하여 비즈니스 사용 사례의 세부 정보를 제공하세요. 비즈니스 사용 사례에 맞는 사기 탐지 모델을 만드는 데 사용할 모델 유형을 권장합니다.

5. 데이터 모델 인사이트 패널은 비즈니스 사용 사례에 맞는 사기 탐지 모델을 만들고 학습하는 데 필요한 필수, 권장 및 선택 데이터 요소에 대한 통찰력을 제공합니다. 인사이트 패널의 정보를 사용하여 이벤트 데이터를 수집하고 데이터 세트를 만들 수 있습니다.

이벤트 데이터 수집

이벤트 데이터 수집은 모델 생성의 중요한 단계입니다. 사기 예측 모델의 성능은 데이터세트의 품질에 따라 달라지기 때문입니다. 이벤트 데이터 수집을 시작할 때는 데이터 모델 탐색기가 데이터세트를 만들기 위해 제공한 데이터 요소 목록을 염두에 두십시오. 모든 필수 (이벤트 메타데이터) 데이터를 수집하고 모델 생성 목표에 따라 포함할 권장 및 선택적 데이터 요소 (이벤트 변수) 를 결정해야 합니다. 포함하려는 각 이벤트 변수의 형식과 데이터셋의 전체 크기를 결정하는 것도 중요합니다.

이벤트 데이터세트 품질

모델에 대해 고품질 데이터 세트를 수집하려면 다음을 권장합니다.

- 성숙한 데이터 수집 - 최신 데이터를 사용하면 가장 최근의 사기 패턴을 식별하는 데 도움이 됩니다. 하지만 사기 사용 사례를 탐지하려면 데이터가 완성되도록 허용해야 합니다. 만기 기간은 비즈니스에 따라 다르며 2주에서 3개월까지 걸릴 수 있습니다. 예를 들어, 이벤트에 신용 카드 거래가 포함된 경우 신용 카드의 지불 거절 기간 또는 조사자가 결정을 내리는 데 걸린 시간에 따라 데이터 만기가 결정될 수 있습니다.

  모델을 학습하는 데 사용된 데이터세트가 비즈니스에 맞게 성숙할 수 있는 충분한 시간을 확보했는지 확인하세요.
- 데이터 분포가 크게 달라지지 않도록 주의하세요. Amazon Fraud Detector 모델 학습 프로세스는 EVENT_TIMESTAMP를 기반으로 데이터세트를 샘플링하고 파티셔닝합니다. 예를 들어 데이터세트가 지난 6개월 동안 수집된 사기 이벤트로 구성되어 있지만 합법적인 이벤트의 마지막 달만 포함된 경우 데이터 분포는 변동하고 불안정한 것으로 간주됩니다. 데이터 세트가 불안정하면 모델 성능 평가에 편향이 생길 수 있습니다. 데이터 분포가 크게 변동하는 경우 현재 데이터 분포와 유사한 데이터를 수집하여 데이터세트의 균형을 맞추는 것을 고려해 보십시오.
- 데이터세트가 모델이 구현/테스트된 사용 사례를 대표하는지 확인하십시오. 그렇지 않으면 예상 성능이 편향될 수 있습니다. 모든 실내 지원자를 자동으로 거절하는 모델을 사용하고 있는데 이전에 승인된 과거 데이터/라벨이 있는 데이터세트로 모델을 학습시켰다고 가정해 보겠습니다. 그러면 평가가 거부된 지원자의 표현이 없는 데이터세트를 기반으로 하기 때문에 모델 평가가 정확하지 않을 수 있습니다.

이벤트 데이터 형식

Amazon Fraud Detector는 모델 학습 프로세스의 일환으로 대부분의 데이터를 필요한 형식으로 변환합니다. 하지만 나중에 Amazon Fraud Detector가 데이터 세트를 검증할 때 문제가 발생하지 않도록 데이터를 제공하는 데 쉽게 사용할 수 있는 몇 가지 표준 형식이 있습니다. 다음 표는 권장 이벤트 메타데이터를 제공하기 위한 형식에 대한 지침을 제공합니다.

> **Note**
>
> CSV 파일을 만들 때는 아래 나열된 대로 이벤트 메타데이터 이름을 대문자로 입력해야 합니다.

메타데이터 이름	형식	필수
이벤트_ID	제공된 경우 다음 요구 사항을 충족해야 합니다. • 이 이벤트에서만 볼 수 있는 특별한 이벤트입니다. • 이는 비즈니스에 의미 있는 정보를 나타냅니다. • 정규 표현식 패턴을 따릅니다 (예:`^[0-9a-z_-]+$.`) • 위의 요구 사항 외에도 EVENT_ID에 타임스탬프를 추가하지 않는 것이 좋습니다. 이렇게 하면 이벤트를 업데이트할 때 문제가 발생할 수 있습니다. 이렇게 하려면 정확히 동일한 EVENT_ID를 제공해야 하기 때문입니다.	모델 유형에 따라 다름
이벤트_타임스탬프	• 다음 형식 중 하나로 지정해야 합니다. • %yyy-%mm-%DDT%HH:%mm: %ssZ (밀리초 없이 UTC로만 표시되는 ISO 8601 표준)	예

메타데이터 이름	형식	필수
	예: 2019-11-30T13:01:01Z • %yyy/%mm/%dd %hh:%mm: %s (오전/오후) 예: 2019/11/30 오후 1:01:01 또는 2019/11/30 13:01:01 • %mm/%dd/%yyyy %hh:%mm: %s 예: 2019년 11월 30일 오후 1:01:01, 2019년 11월 30일 13:01:01 • %mm/%dd/%yy %hh:%mm: %ss 예: 11/30/19 오후 1:01:01, 11/30/19 13:01:01 • Amazon Fraud Detector는 이벤트 타임스탬프의 날짜/타임스탬프 형식을 분석할 때 다음과 같은 가정을 합니다. • ISO 8601 표준을 사용하는 경우 이전 사양과 정확히 일치해야 합니다. • 다른 형식 중 하나를 사용하는 경우 유연성이 더욱 향상됩니다. • 월 및 일 단위로 한 자리 또는 두 자리 숫자를 제공할 수 있습니다. 예를	

메타데이터 이름	형식	필수
	들어, 2019년 1월 12일은 유효한 날짜입니다. • hh:mm:ss가 없으면 포함하지 않아도 됩니다 (즉, 날짜를 입력하기만 하면 됩니다). 시간과 분의 하위 집합 (예: hh:mm) 을 제공할 수도 있습니다. 시간만 제공하는 것은 지원되지 않습니다. 밀리초도 지원되지 않습니다. • AM/PM 레이블을 제공하는 경우 12시간 시간을 기준으로 합니다. AM/PM 정보가 없는 경우 24시간 시계를 사용하는 것으로 가정합니다. • 날짜 요소의 구분 기호로 “/” 또는 “-”를 사용할 수 있습니다. 타임스탬프 요소에는 “:”가 사용됩니다.	
엔티티_아이디	• 정규 표현식 패턴을 따라야 합니다`^[0-9A-Za-z_.@+-]+$` . • 평가 시 개체 ID를 사용할 수 없는 경우 개체 ID를 알 수 없음으로 지정하십시오.	모델 유형에 따라 다름
엔티티_유형	모든 문자열을 사용할 수 있습니다.	모델 유형에 따라 다름

메타데이터 이름	형식	필수
이벤트_라벨	“사기”, “합법적”, “1" 또는 “0"과 같은 모든 레이블을 사용할 수 있습니다.	LABEL_타임스탬프가 포함된 경우 필수
레이블_타임스탬프	타임스탬프 형식을 따라야 합니다.	이벤트_라벨이 포함된 경우 필수

이벤트 변수에 대한 자세한 내용은 변수를 참조하십시오.

> **Important**
>
> 계정 인사이트 (ATI) 모델을 생성하는 경우 데이터 준비 및 선택에데이터 준비 대한 자세한 내용은 을 참조하십시오.

값이 없거나 누락된 값

EVENT_TIMESTAMP 및 EVENT_LABEL 변수에는 null이나 누락된 값이 포함되어서는 안 됩니다. 다른 변수에는 null 또는 누락된 값이 있을 수 있습니다. 그러나 이러한 변수에는 적은 수의 null만 사용하는 것이 좋습니다. Amazon Fraud Detector는 이벤트 변수에 대해 null 또는 누락된 값이 너무 많다고 판단하면 모델에서 변수를 자동으로 생략합니다.

최소 변수

모델을 만들 때 데이터셋에는 필수 이벤트 메타데이터 외에 두 개 이상의 이벤트 변수가 포함되어야 합니다. 두 이벤트 변수는 유효성 검사를 통과해야 합니다.

이벤트 데이터세트 크기

필수

성공적인 모델 학습을 위해서는 데이터세트가 다음과 같은 기본 요구 사항을 충족해야 합니다.

- 최소 100개 이벤트의 데이터
- 데이터세트에는 사기로 분류된 이벤트 (행) 가 50개 이상 포함되어야 합니다.

권장

성공적인 모델 학습과 우수한 모델 성능을 위해서는 데이터세트에 다음을 포함하는 것이 좋습니다.

- 최소 3주 분량의 과거 데이터를 포함하되, 최대 6개월 분량의 데이터를 포함해야 합니다.
- 최소 10,000개의 총 이벤트 데이터를 포함해야 합니다.
- 사기성 이벤트로 분류된 최소 400개의 이벤트 (행) 와 합법적인 것으로 분류된 400개 이상의 이벤트 (행) 를 포함합니다.
- 모델 유형에 ENTITY_ID가 필요한 경우 100개 이상의 고유 엔티티를 포함하십시오.

데이터 세트 검증

Amazon Fraud Detector는 모델 생성을 시작하기 전에 모델 학습용 데이터세트에 포함된 변수가 크기, 형식 및 기타 요구 사항을 충족하는지 확인합니다. 데이터세트가 검증을 통과하지 못하면 모델이 생성되지 않습니다. 모델을 생성하기 전에 검증을 통과하지 못한 변수를 먼저 수정해야 합니다. Amazon Fraud Detector는 모델 학습을 시작하기 전에 데이터 세트와 관련된 문제를 식별하고 해결하는 데 사용할 수 있는 데이터 프로파일러를 제공합니다.

데이터 프로파일러

Amazon Fraud Detector는 모델 교육을 위해 데이터를 프로파일링하고 준비할 수 있는 오픈 소스 도구를 제공합니다. 이 자동화된 데이터 프로파일러를 사용하면 일반적인 데이터 준비 오류를 방지하고 잘못 매핑된 변수 유형과 같이 모델 성능에 부정적인 영향을 미칠 수 있는 잠재적 문제를 식별할 수 있습니다. 프로파일러는 변수 통계, 라벨 분포, 범주형 및 수치 분석, 변수 및 레이블 상관관계를 포함하여 데이터세트에 대한 직관적이고 포괄적인 보고서를 생성합니다. 변수 유형에 대한 지침과 데이터 세트를 Amazon Fraud Detector에 필요한 형식으로 변환하는 옵션을 제공합니다.

데이터 프로파일러 사용

자동화된 데이터 프로파일러는 몇 번의 클릭만으로 쉽게 실행할 수 있는AWS CloudFormation 스택으로 구축됩니다. 모든 코드는 Github에서 사용할 수 있습니다. 데이터 프로파일러 사용 방법에 대한 자세한 내용은 블로그의 지침을 따르십시오. Amazon Fraud Detector용 자동 데이터 프로파일러로 모델을 더 빠르게 교육하세요.

일반적인 이벤트 데이터세트 오류

다음은 Amazon Fraud Detector가 이벤트 데이터 세트를 검증할 때 발생하는 몇 가지 일반적인 문제입니다. 데이터 프로파일러를 실행한 후 모델을 생성하기 전에 이 목록을 사용하여 데이터세트에 오류가 있는지 확인하십시오.

- CSV 파일은 UTF-8 형식이 아닙니다.

- 데이터셋의 이벤트 수가 100개 미만입니다.
- 사기 또는 합법적인 것으로 확인된 이벤트의 수는 50개 미만입니다.
- 사기 이벤트와 관련된 고유 개체의 수는 100개 미만입니다.
- EVENT_TIMESTAMP의 0.1% 이상의 값에 지원되는 날짜/타임스탬프 형식이 아닌 다른 값이나 null이 포함되어 있습니다.
- EVENT_LABEL 값의 1% 이상에 이벤트 유형에 정의된 것과 다른 값이나 null이 포함되어 있습니다.
- 모델 학습에 사용할 수 있는 변수는 두 개 미만입니다.

데이터 세트 스토리지에 대해

데이터 세트를 수집한 후에는 Amazon Simple Storage Service (Amazon S3) 을 사용하여 데이터 세트를 내부적으로 저장합니다. 사기 예측을 생성하는 데 사용하는 모델을 기반으로 데이터세트를 저장할 위치를 선택하는 것이 좋습니다. 모델 유형에 대한 자세한 내용은 모델 유형 선택을 참조하십시오. 데이터세트 저장에 대한 자세한 내용은 을 참조하십시오이벤트 데이터 스토리지.

이벤트 유형

Amazon 사기 탐지기를 사용하면 이벤트에 대한 사기 예측을 생성할 수 있습니다. 이벤트 유형은 Amazon Fraud Detector로 전송되는 개별 이벤트의 구조를 정의합니다. 정의한 후에는 특정 이벤트 유형에 대한 위험을 평가하는 모델 및 감지기를 구축할 수 있습니다.

이벤트 구조에는 다음이 포함됩니다.

- 개체 유형: 이벤트를 수행하는 사람을 분류합니다. 예측 중에 개체 유형과 개체 ID를 지정하여 이벤트를 수행한 사람을 정의합니다.
- 변수: 이벤트의 일부로 전송할 수 있는 변수를 정의합니다. 변수는 모델 및 규칙에서 사기 위험을 평가하는 데 사용됩니다. 추가한 후에는 이벤트 유형에서 변수를 제거할 수 없습니다.
- 레이블: 이벤트를 사기 또는 합법적인 것으로 분류합니다. 모델 학습 중에 사용됩니다. 추가한 후에는 이벤트 유형에서 레이블을 제거할 수 없습니다.

이벤트 유형 생성

사기 탐지 모델을 만들기 전에 먼저 이벤트 유형을 만들어야 합니다. 이벤트 유형을 만들려면 사기 여부를 평가할 비즈니스 활동 (이벤트) 을 정의해야 합니다. 이벤트를 정의하려면 데이터 세트에서 사기 평가에 포함할 이벤트 변수를 식별하고 이벤트를 시작하는 주체를 지정하고 이벤트를 분류하는 레이블을 지정해야 합니다.

이벤트 유형 생성을 위한 전제 조건

이벤트 유형 생성을 시작하기 전에 다음을 완료했는지 확인하세요.

- 데이터 모델 탐색기도구를 사용하여 Amazon Fraud Detector에서 사기 탐지 모델을 만드는 데 필요한 데이터 요소에 대한 통찰력을 얻었습니다.
- 데이터 모델 탐색기에서 얻은 통찰력을 사용하여 이벤트 데이터 세트를 만들고 데이터 세트를 Amazon S3 버킷에 업로드했습니다.
- Variables생성되었으며 Labels Amazon Fraud Detector에서 이 이벤트에 대한 부정 행위 탐지 모델을 만드는 데 사용하기를 원합니다. 엔터티 생성한 변수, 개체 유형 및 레이블이 이벤트 데이터세트에 포함되어 있는지 확인하세요.

Amazon Fraud Detector 콘솔에서 APIAWS CLI, 또는 AWS SDK를 사용하여 이벤트 유형을 생성할 수 있습니다.

Amazon 사기 탐지기 콘솔에서 이벤트 유형 생성

이벤트 유형을 만들려면

1. AWS관리 콘솔을 열고 계정에 로그인합니다. 아마존 사기 탐지기로 이동합니다.
2. 왼쪽 탐색 창에서 [Events] 를 선택합니다.
3. 이벤트 유형 페이지에서 생성을 선택합니다.
4. 이벤트 유형 세부 정보에서

   a. 이름에 이벤트 이름을 입력합니다.

   b. 설명에 설명을 입력할 수도 있습니다.

   c. 엔티티에서 이벤트용으로 생성한 엔티티 유형을 선택합니다.
5. 이벤트 변수에서

   - 이 이벤트의 변수를 정의하는 방법 선택에서

     - 이 이벤트에 대한 이벤트 변수를 이미 만든 경우 변수 목록에서 변수 선택을 선택하고 변수에서 이 이벤트에 대해 만든 변수를 선택합니다.
     - 이 이벤트에 대한 변수를 만들지 않았으면 훈련 데이터셋에서 변수 선택을 선택하십시오.
       - IAM 역할에서 Amazon Fraud Detector가 데이터 세트가 포함된 Amazon S3 버킷에 액세스하는 데 사용할 IAM 역할을 선택합니다.
       - 데이터 위치에서 데이터세트 위치의 경로를 입력합니다. 다음과 비슷한 `S3 URI` 경로를 사용하십시오`S3://your-bucket-name/example dataset filename.csv`.
       - 업로드를 선택합니다.
       - 변수 아래에는 Amazon Fraud Detector가 데이터세트 파일에서 추출한 모든 이벤트 변수 이름이 표시됩니다.

         사기 탐지를 위해 변수를 포함시키려면 변수 유형에서 변수 유형을 선택합니다. 부정행위 탐지에 포함되는 변수를 제거하려면 제거를 선택합니다. 목록의 각 변수에 대해 이 단계를 반복합니다.
6. 레이블 (선택 사항) 의 레이블에서 이 이벤트에 대해 만든 레이블을 선택합니다. 사기 및 적법한 이벤트의 라벨을 각각 하나씩 선택해야 합니다.
7. 이 이벤트에 대한 자동 다운스트림 처리를 설정하려면 Amazon을 사용한 이벤트 오케스트레이션 EventBridge - 선택 사항에서 Amazon을 통한 이벤트 오케스트레이션 활성화를 켜십시오. EventBridge 이벤트 오케스트레이션에 대한 자세한 내용은 을 참조하십시오이벤트 오케스트레이션.

> **Note**
>
> 이벤트 유형을 만든 후 나중에 이벤트 오케스트레이션을 활성화할 수도 있습니다.

8. 이벤트 유형 생성을 선택합니다.

를 사용하여 이벤트 유형 만들기 AWS SDK for Python (Boto3)

다음 예제는 `PutEventType` API에 대한 샘플 요청을 보여줍니다. 이 예제에서는 변수`email_address`, 레이블 `ip_address` `legit` 및 `fraud` 엔티티 유형을 `sample_customer` 생성했다고 가정합니다. 이러한 리소스를 만드는 방법에 대한 자세한 내용은 을 참조하십시오리소스.

> **Note**
>
> 변수, 엔티티 유형 및 레이블을 이벤트 유형에 추가하기 전에 먼저 만들어야 합니다.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.put_event_type (
name = 'sample_registration',
eventVariables = ['ip_address', 'email_address'],
labels = ['legit', 'fraud'],
entityTypes = ['sample_customer'])
```

이벤트 또는 이벤트 유형 삭제

이벤트를 삭제하면 Amazon Fraud Detector는 해당 이벤트를 영구적으로 삭제하며 이벤트와 관련된 데이터는 더 이상 Amazon Fraud Detector에 저장되지 않습니다.

Amazon Fraud Detector가 **`GetEventPrediction`** API를 통해 평가한 이벤트를 삭제하려면

1. https://console.aws.amazon.com/frauddetector 에서 로그인하여 아마존 사기 탐지기 콘솔을 여십시오. AWS Management Console
2. 콘솔의 왼쪽 탐색 창에서 과거 예측 검색을 선택합니다.
3. 삭제하려는 이벤트를 선택합니다.

4. 작업을 선택한 다음 이벤트 삭제를 선택합니다.
5. 입력한 **delete** 다음 이벤트 삭제를 선택합니다.

> **Note**
>
> 이렇게 하면 작업에 전송된 이벤트 데이터 및 SendEvent 작업을 통해 생성된 예측 데이터를 포함하여 해당 이벤트 ID와 관련된 모든 레코드가 삭제됩니다GetEventPrediction.

Amazon Fraud Detector에 저장되어 있지만 평가되지 않은 이벤트 (즉, SendEvent 작업을 통해 저장된 이벤트) 를 삭제하려면 DeleteEvent 요청하고 이벤트 ID 및 이벤트 유형 ID를 지정해야 합니다. 이벤트 및 이벤트와 관련된 예측 기록을 모두 삭제하려면 deleteAuditHistory 파라미터 값을 “true”로 설정하십시오. deleteAuditHistory파라미터를 “true”로 설정하면 삭제 작업이 완료된 후 최대 30초 동안 검색을 통해 이벤트 데이터를 사용할 수 있습니다.

이벤트 유형과 관련된 모든 이벤트를 삭제하려면

1. 콘솔의 왼쪽 탐색 창에서 이벤트 유형을 선택합니다.
2. 모든 이벤트를 삭제하려는 이벤트 유형을 선택합니다.
3. 저장된 이벤트 탭으로 이동하여 저장된 이벤트 삭제를 선택합니다.

이벤트 유형에 대해 저장된 이벤트 수에 따라 저장된 모든 이벤트를 삭제하는 데 다소 시간이 걸릴 수 있습니다. 예를 들어 1GB 데이터 세트 (일반 고객의 경우 약 1~200만 개의 이벤트) 를 삭제하는 데 약 2시간이 걸립니다. 이 기간 동안 Amazon Fraud Detector로 전송한 이 이벤트 유형의 새 이벤트는 저장되지 않지만 GetEventPrediction 작업을 통해 계속해서 부정 행위 예측을 생성할 수 있습니다.

이벤트 유형을 삭제하려면

감지기 또는 모델에서 사용되거나 저장된 이벤트와 연관된 이벤트 유형은 삭제할 수 없습니다. 이벤트 유형을 삭제하려면 먼저 해당 이벤트 유형과 관련된 모든 이벤트를 삭제해야 합니다.

이벤트 유형을 삭제하면 Amazon Fraud Detector는 해당 이벤트 유형을 영구적으로 삭제하며 데이터는 더 이상 Amazon Fraud Detector에 저장되지 않습니다.

1. Amazon Fraud Detector 콘솔의 왼쪽 탐색 창에서 리소스를 선택한 다음 이벤트를 선택합니다.
2. 삭제하려는 이벤트 유형을 선택합니다.
3. 작업을 선택한 다음 이벤트 유형 삭제를 선택합니다.

4. 이벤트 유형 이름을 입력한 다음 이벤트 유형 삭제를 선택합니다.

이벤트 데이터 스토리지

데이터 세트를 수집한 후에는 Amazon Storage Service (Amazon S3) 를 사용하여 내부적으로 저장하거나 Amazon Storage Service (Amazon S3) 를 사용하여 외부에 저장합니다. 사기 예측을 생성하는 데 사용하는 모델을 기반으로 데이터세트를 저장할 위치를 선택하는 것이 좋습니다. 다음은 이 두 가지 스토리지 옵션에 대한 자세한 설명입니다.

- 내부 스토리지 - 데이터 세트는 Amazon Fraud Detector 저장됩니다. 이벤트와 관련된 모든 이벤트 데이터는 함께 저장됩니다. Amazon Fraud Detector에 저장된 이벤트 데이터 세트를 언제든지 업로드할 수 있습니다. 이벤트를 한 번에 하나씩 Amazon Fraud Detector API로 스트리밍하거나 일괄 가져오기 기능을 사용하여 대용량 데이터 세트 (최대 1GB) 를 가져올 수 있습니다. Amazon Fraud Detector에 저장된 데이터세트를 사용하여 모델을 학습시키는 경우 시간 범위를 지정하여 데이터세트의 크기를 제한할 수 있습니다.
- 외부 스토리지 - 데이터세트는 Amazon Fraud Detector가 아닌 외부 데이터 소스에 저장됩니다. 현재 Amazon Simple Storage Service (Amazon S3) 를 사용할 수 있도록 지원합니다. 모델이 Amazon S3에 업로드된 파일에 있는 경우 해당 파일은 5GB를 초과할 수 없는 비압축 데이터일 수 있습니다. 그 이상이라면 데이터세트의 시간 범위를 줄여야 합니다.

다음 표에는 모델 유형 및 모델이 지원하는 데이터 소스에 대한 세부 정보가 나와 있습니다.

모델 유형	호환 가능한 교육 데이터 소스
온라인 Fraud Detector	외장 스토리지, 내부 스토리지
Fraud Detector	내부 스토리지
계정 탈취 도우미용	내부 스토리지

Amazon Simple Storage Service를 사용하여 데이터 세트를 외부에 저장하는 방법에 대한 자세한 내용은 을 참조하십시오Amazon S3를 사용하여 이벤트 데이터를 외부에 저장 . Amazon Fraud Detector를 사용하여 내부적으로 데이터 세트를 저장하는 방법에 대한 자세한 내용은 을 참조하십시오Amazon Fraud Detector를 사용하여 이벤트 데이터를 내부적으로 저장하십시오..

Amazon S3를 사용하여 이벤트 데이터를 외부에 저장

온라인 사기 인사이트 모델을 트레이닝하는 경우 Amazon S3를 통해 이벤트 데이터를 외부에 저장하도록 선택할 수 있습니다. Amazon S3에 이벤트 데이터를 저장하려면 먼저 CSV 형식의 텍스트 파일을 만들고 이벤트 데이터를 추가한 다음 CSV 파일을 Amazon S3 버킷에 업로드해야 합니다.

> **Note**
>
> 거래 사기 인사이트 및 계정 탈취 인사이트 모델 유형은 Amazon S3와 함께 외부에 저장된 데이터 세트를 지원하지 않습니다.

CSV 파일 생성

Amazon Fraud Detector를 사용하려면 CSV 파일의 첫 번째 행에 열 헤더가 있어야 합니다. CSV 파일의 열 헤더는 이벤트 유형에 정의된 변수에 매핑되어야 합니다. 예제 데이터셋은 을 참조하십시오.예제 데이터세트 가져오기 및 업로드

Online Fraud Insights 모델에는 최소 2개의 변수와 최대 100개의 변수가 있는 학습 데이터세트가 필요합니다. 이벤트 변수 외에도 훈련 데이터셋에는 다음과 같은 헤더가 포함되어야 합니다.

- EVENT_타임스탬프 - 이벤트가 발생한 시점을 정의합니다.
- EVENT_LABEL - 이벤트가 시기성인지 합법적인지를 분류합니다. 열의 값은 이벤트 유형에 정의된 값과 일치해야 합니다.

다음 샘플 CSV 데이터는 온라인 판매자의 과거 등록 이벤트를 나타냅니다.

```
EVENT_TIMESTAMP,EVENT_LABEL,ip_address,email_address
4/10/2019 11:05,fraud,209.146.137.48,fake_burtonlinda@example.net
12/20/2018 20:04,legit,203.0.112.189,fake_davidbutler@example.org
3/14/2019 10:56,legit,169.255.33.54,fake_shelby76@example.net
1/3/2019 8:38,legit,192.119.44.26,fake_curtis40@example.com
9/25/2019 3:12,legit,192.169.85.29,fake_rmiranda@example.org
```

> **Note**
>
> CSV 데이터 파일에는 데이터의 일부로 큰따옴표와 쉼표가 포함될 수 있습니다.

해당 이벤트 유형의 단순화된 버전이 아래에 나와 있습니다. 이벤트 변수는 CSV 파일의 헤더에 해당하고 값은 레이블 목록의 값에EVENT_LABEL 해당합니다.

```
(
name = 'sample_registration',
eventVariables = ['ip_address', 'email_address'],
labels = ['legit', 'fraud'],
entityTypes = ['sample_customer']
)
```

이벤트 타임스탬프 형식

이벤트 타임스탬프가 필수 형식인지 확인하십시오. 모델 구축 프로세스의 일부로서 Online Fraud Insights 모델 유형은 이벤트 타임스탬프를 기반으로 데이터를 정렬하고 학습 및 테스트 목적으로 데이터를 분할합니다. 성능에 대한 공정한 추정치를 얻기 위해 모델은 먼저 훈련 데이터세트를 기반으로 훈련한 다음 테스트 데이터세트에서 이 모델을 테스트합니다.

Amazon Fraud Detector는 모델 학습EVENT_TIMESTAMP 중에 입력된 값에 대해 다음과 같은 날짜/타임스탬프 형식을 지원합니다.

- %yyy-%mm-%DDT%HH: %mm: %ssZ (밀리초 없이 UTC로만 표시되는 ISO 8601 표준)

  예: 2019-11-30T13:01:01 Z
- %yyy/%mm/%dd %hh: %mm: %s (오전/오후)

  예: 2019/11/30 오후 1:01:01 또는 2019/11/30 13:01:01
- %mm/%dd/%yyyy %hh: %mm: %s

  예: 2019년 11월 30일 오후 1:01:01, 2019년 11월 30일 13:01:01
- %mm/%dd/%yy %hh: %mm: %ss

  예: 11/30/19 오후 1:01:01, 11/30/19 13:01:01

Amazon Fraud Detector는 이벤트 타임스탬프의 날짜/타임스탬프 형식을 분석할 때 다음과 같은 가정을 합니다.

- ISO 8601 표준을 사용하는 경우 이전 사양과 정확히 일치해야 합니다.
- 다른 형식 중 하나를 사용하는 경우 유연성이 더욱 향상됩니다.

- 월 및 일 단위로 한 자리 또는 두 자리 숫자를 제공할 수 있습니다. 예를 들어, 2019년 1월 12일은 유효한 날짜입니다.
- hh:mm:ss가 없으면 포함하지 않아도 됩니다. 즉, 날짜를 입력하기만 하면 됩니다. 시간과 분의 하위 집합 (예: hh:mm) 을 제공할 수도 있습니다. 시간만 제공하는 것은 지원되지 않습니다. 밀리초도 지원되지 않습니다.
- AM/PM 레이블을 제공하는 경우 12시간 시간을 기준으로 합니다. AM/PM 정보가 없는 경우 24시간 시계를 사용하는 것으로 가정합니다.
- 날짜 요소의 구분 기호로 “/” 또는 “-”를 사용할 수 있습니다. 타임스탬프 요소에는 “:”가 사용됩니다.

시간에 따른 데이터세트 샘플링

동일한 시간대의 사기 사례와 합법적인 샘플을 제공하는 것이 좋습니다. 예를 들어, 지난 6개월간의 사기 이벤트를 제공하는 경우 동일한 기간에 균등하게 발생하는 합법적인 이벤트도 제공해야 합니다. 데이터세트에 사기 및 합법적인 이벤트가 고르지 않게 분포되어 있는 경우 다음과 같은 오류 메시지가 표시될 수 있습니다. “시간의 흐름에 따른 사기 분포는 용납할 수 없을 정도로 변동적입니다. 데이터세트를 제대로 분할할 수 없습니다.” 일반적으로 이 오류를 해결하는 가장 쉬운 방법은 사기 이벤트와 합법적인 이벤트가 동일한 기간 동안 균등하게 샘플링되도록 하는 것입니다. 또한 짧은 기간 내에 사기가 급증한 경우 데이터를 제거해야 할 수도 있습니다.

균등하게 분산된 데이터세트를 만들기에 충분한 데이터를 생성할 수 없는 경우 한 가지 방법은 이벤트의 EVENT_TIMESTAMP를 무작위로 지정하여 균등하게 분산되도록 하는 것입니다. 그러나 Amazon Fraud Detector는 EVENT_TIMESTAMP를 사용하여 데이터세트의 적절한 이벤트 하위 집합에 대한 모델을 평가하기 때문에 성능 지표가 비현실적으로 표시되는 경우가 많습니다.

0값 및 누락된 값

아마존 Fraud Detector null 값과 누락된 값을 처리합니다. 그러나 변수에 대한 null 백분율은 제한되어야 합니다. EVENT_TIMESTAMP 및 EVENT_LABEL 열에는 누락된 값이 없어야 합니다.

파일 검증

다음 조건 중 하나가 트리거되는 경우 Amazon Fraud Detector (Amazon Fraud Detector) 가 모델을 학습시키는 데 실패합니다.

- CSV를 파싱할 수 없는 경우
- 열의 데이터 유형이 잘못된 경우

Amazon S3 버킷에 이벤트 데이터를 업로드합니다.

이벤트 데이터가 포함된 CSV 파일을 생성한 다음 Amazon S3 버킷에 파일을 업로드합니다.

Amazon S3 버킷에 업로드하려면

1. AWS Management Console에 로그인한 후 https://console.aws.amazon.com/s3/에서 Amazon S3 콘솔을 엽니다.
2. 버킷 만들기를 선택합니다.

   버킷 만들기 마법사가 열립니다.
3. 버킷 이름에 버킷의 DNS 호환 이름을 입력합니다.

   버킷 이름은 다음과 같아야 합니다.

   - 모든 Amazon S3에서 고유해야 합니다.
   - 3~63자 이내여야 합니다.
   - 대문자가 없어야 합니다.
   - 소문자 또는 숫자로 시작해야 합니다.

   버킷을 생성한 후에는 해당 이름을 변경할 수 없습니다. 버킷 이름 지정에 대한 자세한 내용은 Amazon Storage Service 사용 설명서의 버킷 이름 지정 규칙을 참조하십시오.

   > **Important**
   >
   > 버킷 이름에 계정 번호와 같은 중요한 정보를 포함하지 마세요. 버킷 이름은 버킷의 객체를 가리키는 URL에 표시됩니다.
4. 리전에서 버킷이 속할 AWS 리전을 선택합니다. 미국 동부 (버지니아 북부), 미국 동부 (오하이오), 미국 서부 (오레곤), 유럽 (아일랜드), 아시아 태평양 (싱가포르) 또는 아시아 태평양 (시드니), 아시아 태평양 (시드니), 아시아 태평양 (시드니), 아시아 태평양 (시드니), 아시아 태평양 (시드니)
5. Bucket settings for Block Public Access(퍼블릭 액세스 차단을 위한 버킷 설정)에서 버킷에 적용할 퍼블릭 액세스 차단 설정을 선택합니다.

   모든 설정을 활성화시켜 두는 것이 좋습니다. 퍼블릭 액세스 차단사용을. Amazon Storage Service 사용자 안내서의 Amazon Storage Service 사용자 안내서의 Amazon Storage Service 사용자 안내서의 Amazon S3 Storage Service 사용자 안내서의 Amazon Storage Service 사용자 안내서의 Amazon

6. 버킷 만들기를 선택합니다.
7. Amazon S3 버킷에 교육 데이터 파일을 업로드합니다. 학습 파일 (예: s3://bucketname/object.csv)의 Amazon S3 위치 경로를 기록해 둡니다.

Amazon Fraud Detector를 사용하여 이벤트 데이터를 내부적으로 저장하십시오.

Amazon Fraud Detector에 이벤트 데이터를 저장하도록 선택하고 나중에 저장된 데이터를 사용하여 모델을 학습시킬 수 있습니다. Amazon Fraud Detector에 이벤트 데이터를 저장하면 자동 계산된 변수를 사용하는 모델을 학습시켜 성능을 개선하고, 모델 재교육을 간소화하고, 사기 레이블을 업데이트하여 기계 학습 피드백 루프를 닫을 수 있습니다. 이벤트는 Event Type 리소스 수준에서 저장되므로 동일한 이벤트 유형의 모든 이벤트가 단일 이벤트 유형 데이터세트에 함께 저장됩니다. 이벤트 유형을 정의하는 과정에서 Amazon Fraud Detector 콘솔에서 이벤트 통합 설정을 전환하여 해당 이벤트 유형에 대한 이벤트를 저장할지 여부를 선택적으로 지정할 수 있습니다.

Amazon Fraud Detector에 단일 이벤트를 저장하거나 많은 수의 이벤트 데이터 세트를 가져올 수 있습니다. GetEventPredictionAPI 또는 API를 사용하여 단일 이벤트를 스트리밍할 수 있습니다. SendEvent Amazon Fraud Detector 콘솔의 일괄 가져오기 기능 또는 CreateBatchImportJobAPI를 사용하여 대용량 데이터 세트를 Amazon Fraud Detector로 빠르고 쉽게 가져올 수 있습니다.

언제든지 Amazon Fraud Detector 콘솔을 사용하여 각 이벤트 유형에 대해 이미 저장된 이벤트 수를 확인할 수 있습니다.

저장을 위한 이벤트 데이터 준비

Amazon Fraud Detector를 통해 내부적으로 저장되는 이벤트 데이터는`Event Type` 리소스 수준에서 저장됩니다. 따라서 동일한 이벤트의 모든 이벤트 데이터가 단일 이벤트에 저장됩니다`Event Type`. 저장된 이벤트는 나중에 새 모델을 트레이닝하거나 기존 모델을 재트레이닝하는 데 사용할 수 있습니다. 저장된 이벤트 데이터를 사용하여 모델을 훈련할 때 선택적으로 이벤트의 시간 범위를 지정하여 훈련 데이터세트의 크기를 제한할 수 있습니다.

Amazon Fraud Detector 콘솔,`SendEvent` API 또는 API를 사용하여 Amazon Fraud Detector에 데이터를 저장할 때마다 Amazon Fraud Detector는 데이터를 저장하기 전에 데이터를 검증합니다.`CreateBatchImportJob` 데이터가 검증에 실패하면 이벤트 데이터는 저장되지 않습니다.

Amazon Fraud Detector를 사용하여 내부적으로 데이터를 저장하기 위한 사전 요구 사항

- 이벤트 데이터가 검증을 통과하고 데이터세트가 성공적으로 저장되도록 하려면 데이터 모델 탐색기에서 제공하는 인사이트를 사용하여 데이터세트를 준비해야 합니다.
- Amazon Fraud Detector에 저장하려는 이벤트 데이터에 대한 이벤트 유형을 생성했습니다. 아직 하지 않았다면 지침에 따라 이벤트 유형을 생성하세요.

스마트 데이터 유효성 검사

일괄 가져오기를 위해 Amazon Fraud Detector 콘솔에 데이터 세트를 업로드하면 Amazon Fraud Detector는 스마트 데이터 검증 (SDV) 을 사용하여 데이터를 가져오기 전에 데이터 세트를 검증합니다. SV는 업로드된 데이터 파일을 스캔하여 누락된 데이터, 잘못된 형식 또는 데이터 유형과 같은 문제를 식별합니다. SV는 데이터세트를 검증하는 것 외에도 식별된 모든 문제를 나열하고 가장 큰 영향을 미치는 문제를 해결하기 위한 조치를 제안하는 검증 보고서를 제공합니다. SV로 식별된 일부 문제는 심각할 수 있으므로 Amazon Fraud Detector가 데이터 세트를 성공적으로 가져오려면 먼저 해결해야 합니다. 자세한 정보는 스마트 데이터 검증 보고서을 참조하세요.

SV는 파일 수준 및 데이터 (행) 수준에서 데이터세트를 검증합니다. 파일 수준에서 SV는 데이터 파일을 스캔하여 부적절한 파일 액세스 권한, 잘못된 파일 크기, 파일 형식 및 헤더 (이벤트 메타데이터 및 이벤트 변수) 와 같은 문제를 식별합니다. 데이터 수준에서 SV는 각 이벤트 데이터 (행) 를 스캔하고 잘못된 데이터 형식, 데이터 길이, 타임스탬프 형식 및 null 값과 같은 문제를 식별합니다.

스마트 데이터 검증은 현재 Amazon Fraud Detector 콘솔에서만 사용할 수 있으며 검증은 기본적으로 켜져 있습니다. Amazon Fraud Detector가 데이터세트를 가져오기 전에 스마트 데이터 검증을 사용하지 않도록 하려면 데이터세트를 업로드할 때 Amazon Fraud Detector 콘솔에서 검증을 끄십시오.

API 또는AWS SDK 사용 시 저장된 데이터 검증

`SendEventGetEventPrediction`, 또는`CreateBatchImportJob` API 작업을 통해 이벤트를 업로드할 때 Amazon Fraud Detector는 다음을 검증합니다.

- 해당 이벤트 유형의 EventIngestion 설정은 ENABLED입니다.
- 이벤트 타임스탬프는 업데이트할 수 없습니다. 이벤트 ID가 반복되고 EVENT_TIMESTAMP가 다른 이벤트는 오류로 처리됩니다.
- 변수 이름 및 값은 예상 형식과 일치합니다. 자세한 내용은 변수 만들기 단원을 참조하세요.
- 필수 변수는 값으로 채워집니다.
- 모든 이벤트 타임스탬프는 18개월을 넘지 않았으며 future 날짜도 아닙니다.

일괄 가져오기를 사용하여 이벤트 데이터 저장

일괄 가져오기 기능을 사용하면 콘솔, API 또는 AWS SDK를 사용하여 Amazon Fraud Detector에 대규모 기간별 이벤트 데이터 세트를 빠르고 쉽게 업로드할 수 있습니다. 일괄 가져오기를 사용하려면 모든 이벤트 데이터가 포함된 CSV 형식의 입력 파일을 생성하고 CSV 파일을 Amazon S3 버킷에 업로드한 다음 가져오기 작업을 시작하십시오. Amazon Fraud Detector는 먼저 이벤트 유형을 기반으로 데이터를 검증한 다음 전체 데이터 세트를 자동으로 가져옵니다. 데이터를 가져오면 새 모델을 학습시키거나 기존 모델을 재학습하는 데 사용할 수 있습니다.

입력 및 출력 파일

입력 CSV 파일에는 관련 이벤트 유형에 정의된 변수와 일치하는 헤더와 4개의 필수 변수가 포함되어야 합니다. 자세한 정보는 저장을 위한 이벤트 데이터 준비 섹션을 참조하세요. 입력 데이터 파일의 최대 크기는 20기가바이트 (GB) 또는 약 5천만 개의 이벤트입니다. 이벤트 수는 이벤트 규모에 따라 달라집니다. 가져오기 작업이 성공하면 출력 파일이 비어 있습니다. 가져오기에 실패한 경우 출력 파일에 오류 로그가 포함됩니다.

CSV 파일 생성

Amazon Fraud Detector (쉼표로 분리된 값) 형식의 파일에서만 데이터를 가져옵니다. CSV 파일의 첫 번째 행에는 연결된 이벤트 유형에 정의된 변수와 정확히 일치하는 열 헤더와 4개의 필수 변수 (EVENT_ID, EVENT_TIMESTAMP, ENTITY_ID 및 ENTITY_TYPE) 가 포함되어야 합니다. EVENT_LABEL 및 LABEL_TIMESTAMP를 선택적으로 포함할 수도 있습니다 (EVENT_LABEL이 포함된 경우 LABEL_TIMESTAMP가 필요함).

필수 변수 정의

필수 변수는 이벤트 메타데이터로 간주되며 대문자로 지정해야 합니다. 이벤트 메타데이터는 모델 학습에 자동으로 포함됩니다. 다음 표에는 필수 변수, 각 변수에 대한 설명 및 변수에 필요한 형식이 나와 있습니다.

이름	설명	요구 사항
이벤트_ID	이벤트의 식별자입니다. 예를 들어, 이벤트가 온라인 거래인 경우 EVENT_ID는 고객에게 제공된 거래 참조 번호일 수 있습니다.	• EVENT_ID는 일괄 가져오기 작업에 필요합니다. • 해당 이벤트에 대해 고유해야 합니다.

이름	설명	요구 사항
		• 비즈니스에 의미 있는 정보를 나타내야 합니다. • 정규 표현식 패턴을 충족해야 합니다 (예:^[0-9a-z_-]+$.) • EVENT_ID에 타임스탬프를 추가하지 않는 것이 좋습니다. 이렇게 하면 이벤트를 업데이트할 때 문제가 발생할 수 있습니다. 이렇게 하려면 정확히 동일한 EVENT_ID를 제공해야 하기 때문입니다.

이름	설명	요구 사항

이벤트_타임스탬프	이벤트가 발생한 시점의 타임스탬프입니다. 타임스탬프는 ISO 8601 표준 (UTC) 이어야 합니다.	• EVENT_TIMESTAMP는 일괄 가져오기 작업에 필요합니다.
		• 다음 형식 중 하나로 지정해야 합니다.
		  • %yyy-%mm-%DDT%HH: %mm: %ssZ (밀리초 없이 UTC로만 표시되는 ISO 8601 표준)
		  예: 2019-11-30T13:01:01Z
		  • %yyy/%mm/%dd %hh: %mm: %s (오전/오후)
		  예: 2019/11/30 오후 1:01:01 또는 2019/11/30 13:01:01
		  • %mm/%dd/%yyyy %hh: %mm: %s
		  예: 2019년 11월 30일 오후 1:01:01, 2019년 11월 30일 13:01:01
		  • %mm/%dd/%yy %hh: %mm: %ss
		  예: 11/30/19 오후 1:01:01, 11/30/19 13:01:01
		• Amazon Fraud Detector는 이벤트 타임스탬프의 날짜/타임스탬프 형식을 분석할 때 다음과 같은 가정을 합니다.

이름	설명	요구 사항

		• ISO 8601 표준을 사용하는 경우 이전 사양과 정확히 일치해야 합니다.
		• 다른 형식 중 하나를 사용하는 경우 유연성이 더욱 향상됩니다.
		  • 월 및 일 단위로 한 자리 또는 두 자리 숫자를 제공할 수 있습니다. 예를 들어, 2019년 1월 12일은 유효한 날짜입니다.
		  • hh:mm:ss가 없으면 포함하지 않아도 됩니다 (즉, 날짜를 입력하기만 하면 됩니다). 시간과 분의 하위 집합 (예: hh:mm) 을 제공할 수도 있습니다. 시간만 제공하는 것은 지원되지 않습니다. 밀리초도 지원되지 않습니다.
		  • AM/PM 레이블을 제공하는 경우 12시간 시간을 기준으로 합니다. AM/PM 정보가 없는 경우 24시간 시계를 사용하는 것으로 가정합니다.
		  • 날짜 요소의 구분 기호로 “/” 또는 “-”를 사용할 수 있습니다. 타임스탬프 요소에는 “:”가 사용됩니다.

이름	설명	요구 사항
엔티티_아이디	이벤트를 수행하는 엔터티의 식별자입니다.	• 일괄 가져오기 작업에는 ENTITY_ID가 필요합니다. • 정규 표현식 패턴을 따라야 합니다^[0-9A-Za-z_.@+-]+$. • 평가 시 개체 ID를 사용할 수 없는 경우 개체 ID를 알 수 없음으로 지정하십시오.
엔티티_유형	이벤트를 수행하는 주체 (예: 판매자 또는 고객)	일괄 가져오기 작업에는 ENTITY_TYPE이 필요합니다.
이벤트_라벨	이벤트를 다음과 같이fraudulent 분류합니다.legitimate	LABEL_타임스탬프가 포함된 경우 EVENT_LABEL이 필요합니다.
레이블_타임스탬프	이벤트 레이블이 마지막으로 채워지거나 업데이트된 타임스탬프입니다.	• EVENT_LABEL이 포함된 경우 LABEL_타임스탬프가 필요합니다. • 타임스탬프 형식을 따라야 합니다.

배치 가져오기를 위해 Amazon S3에 CSV 파일을 업로드합니다.

데이터를 사용하여 CSV 파일을 생성한 다음 Amazon Storage Simple Storage Service (Amazon S3) 버킷에 파일을 업로드합니다.

Amazon S3 버킷에 이벤트 데이터를 업로드하려면

1. AWS Management Console에 로그인한 후 https://console.aws.amazon.com/s3/에서 Amazon S3 콘솔을 엽니다.
2. 버킷 만들기를 선택합니다.

   버킷 만들기 마법사가 열립니다.
3. 버킷 이름에 버킷의 DNS 호환 이름을 입력합니다.

버킷 이름은 다음과 같아야 합니다.

- 모든 Amazon S3에서 고유해야 합니다.
- 3~63자 이내여야 합니다.
- 대문자가 없어야 합니다.
- 소문자 또는 숫자로 시작해야 합니다.

버킷을 생성한 후에는 해당 이름을 변경할 수 없습니다. 버킷 이름 지정에 대한 자세한 내용은 Amazon Storage Service 사용 설명서의 버킷 이름 지정 규칙을 참조하십시오.

> **Important**
>
> 버킷 이름에 계정 번호와 같은 중요한 정보를 포함하지 마세요. 버킷 이름은 버킷의 객체를 가리키는 URL에 표시됩니다.

4. 리전에서 버킷이 속할 AWS 리전을 선택합니다. 미국 동부 (버지니아 북부), 미국 동부 (오하이오), 미국 서부 (오레곤), 유럽 (아일랜드), 아시아 태평양 (싱가포르) 또는 아시아 태평양 (시드니), 아시아 태평양 (시드니), 아시아 태평양 (시드니), 아시아 태평양 (시드니), 아시아 태평양 (시드니)
5. Bucket settings for Block Public Access(퍼블릭 액세스 차단을 위한 버킷 설정)에서 버킷에 적용할 퍼블릭 액세스 차단 설정을 선택합니다.

   모든 설정을 활성화시켜 두는 것이 좋습니다. 퍼블릭 액세스 차단사용을. Amazon Storage Service 사용자 안내서의 Amazon Storage Service 사용자 안내서의 Amazon Storage Service 사용자 안내서의 Amazon S3 Storage Service 사용자 안내서의 Amazon Storage Service 사용자 안내서의 Amazon
6. 버킷 만들기를 선택합니다.
7. Amazon S3 버킷에 교육 데이터 파일을 업로드합니다. 학습 파일 (예: s3://bucketname/object.csv)의 Amazon S3 위치 경로를 기록해 둡니다.

Amazon Fraud Detector 콘솔에서 이벤트 데이터를 Batch 가져오기

`CreateBatchImportJob`API를 사용하거나 AWS SDK를 사용하여 Amazon Fraud Detector 콘솔에서 대량의 이벤트 데이터 세트를 쉽게 가져올 수 있습니다. 진행하기 전에 지침에 따라 데이터세트를 CSV 파일로 준비했는지 확인하세요. Amazon S3 버킷에 CSV 파일도 업로드했는지 확인하십시오.

Amazon Fraud Detector 콘솔 사용

콘솔에서 이벤트 데이터를 일괄적으로 가져오려면

1. AWS 콘솔을 열고 계정에 로그인한 다음 Amazon Fraud Detector로 이동합니다.
2. 왼쪽 탐색 창에서 이벤트를 선택합니다.
3. 이벤트 유형을 선택합니다.
4. 저장된 이벤트 탭을 선택합니다.
5. 저장된 이벤트 세부 정보 창에서 이벤트 수집이 켜져 있는지 확인합니다.
6. 이벤트 가져오기 데이터 창에서 새로 만들기 가져오기를 선택합니다.
7. 새 이벤트 가져오기 페이지에서 다음 정보를 제공합니다.

   - [권장] 이 데이터세트에 대해 스마트 데이터 유효성 검사 활성화를 기본 설정으로 새로 설정한 상태로 두십시오.
   - 데이터용 IAM 역할에서 가져오려는 CSV 파일이 들어 있는 Amazon S3 버킷에 대해 생성한 IAM 역할을 선택합니다.
   - 입력 데이터 위치에 CSV 파일이 있는 S3 위치를 입력합니다.
   - 가져오기 결과를 저장할 별도의 위치를 지정하려면 입력 및 결과를 위한 별도의 데이터 위치 버튼을 클릭하고 유효한 Amazon S3 버킷 위치를 제공하십시오.

   > **Important**
   >
   > 선택한 IAM 역할에 입력 Amazon S3 버킷에 대한 읽기 권한과 출력 Amazon S3 버킷에 대한 쓰기 권한이 있는지 확인하십시오.

8. 시작을 선택합니다.
9. 이벤트 가져오기 데이터 패널의 상태 열에는 검증 및 가져오기 작업의 상태가 표시됩니다. 상단의 배너는 데이터세트가 먼저 유효성 검사를 거친 다음 가져오기를 거치므로 상태에 대한 자세한 설명을 제공합니다.
10. 에 제공된 지침을 따르십시오데이터세트 유효성 검사 진행 상황 모니터링 및 가져오기 작업.

데이터세트 유효성 검사 진행 상황 모니터링 및 가져오기 작업

Amazon Fraud Detector 콘솔을 사용하여 일괄 가져오기 작업을 수행하는 경우 기본적으로 Amazon Fraud Detector는 가져오기 전에 데이터 세트를 검증합니다. Amazon Fraud Detector 콘솔의 새 이벤트 가져오기 페이지에서 검증 및 가져오기 작업의 진행 상황 및 상태를 모니터링할 수 있습니다. 페이지 상단에 있는 배너는 유효성 검사 결과 및 가져오기 작업의 상태에 대한 간략한 설명을 제공합니다.

검증 결과 및 가져오기 작업의 상태에 따라 데이터세트를 성공적으로 검증하고 가져오기 위한 조치를 취해야 할 수 있습니다.

다음 표에는 검증 및 가져오기 작업의 결과에 따라 수행해야 하는 작업에 대한 세부 정보가 나와 있습니다.

배너 메시지	상태	의미	어떻게 해야 하나요
데이터 유효성 검사 시작	유효성 검사 진행 중	SV가 데이터세트 검증을 시작했습니다	상태가 바뀔 때까지 기다리세요
데이터세트의 오류로 인해 데이터 검증을 진행할 수 없습니다. 데이터 파일의 오류를 수정하고 새 가져오기 작업을 시작합니다. 자세한 내용은 검증 보고서를 참조하십시오.	유효성 검사 실패	SV가 데이터 파일에서 문제를 식별했습니다. 데이터세트를 성공적으로 가져오려면 이러한 문제를 해결해야 합니다.	이벤트 가져오기 데이터 창에서 Job ID를 선택하고 검증 보고서를 확인합니다. 보고서의 권장 사항에 따라 나열된 모든 오류를 해결하십시오. 자세한 정보는 검증 보고서 사용을 참조하세요.
데이터 가져오기가 시작되었습니다. 유효성 검사가 성공적으로 완료되었습니다.	가져오기 진행 중	데이터세트가 검증을 통과했습니다. AFD가 데이터세트를 가져오기 시작했습니다	상태가 바뀔 때까지 기다리세요
유효성 검사가 완료되고 경고가 표시됩니다. 데이터 가져오기가 시작되었습니다	가져오기 진행 중	데이터세트의 일부 데이터가 검증에 실패했습니다. 하지만 유효성 검사	배너에서 메시지를 모니터링하고 상태가 변경될 때까지 기다립니다.

배너 메시지	상태	의미	어떻게 해야 하나요
		를 통과한 데이터는 가져오기에 필요한 최소 데이터 크기 요구 사항을 충족합니다.	
데이터를 부분적으로 가져왔습니다. 일부 데이터가 검증에 실패하여 가져오지 못했습니다. 자세한 내용은 유효성 검사 보고서를 참조하세요.	수입산. 상태에는 경고 아이콘이 표시됩니다.	데이터 파일에서 검증에 실패한 일부 데이터를 가져오지 못했습니다. 검증을 통과한 나머지 데이터를 가져왔습니다.	이벤트 가져오기 데이터 창에서 Job ID를 선택하고 검증 보고서를 확인합니다. 데이터 수준 경고 표의 권장 사항을 따라 나열된 경고를 해결하십시오. 모든 경고를 해결할 필요는 없습니다. 하지만 성공적인 가져오기를 위해서는 데이터 세트에 검증을 통과한 데이터가 50% 이상 포함되어 있어야 합니다. 경고를 해결한 후 새 가져오기 작업을 시작합니다. 자세한 정보는 검증 보고서 사용을 참조하세요.
처리 오류로 인해 데이터 가져오기에 실패했습니다. 새 데이터 가져오기 작업 시작	가져오기 실패	일시적인 런타임 오류로 인해 가져오기에 실패했습니다.	새 가져오기 작업 시작
데이터를 성공적으로 가져왔습니다	수입산	검증 및 가져오기가 모두 성공적으로 완료되었습니다.	가져오기 작업의 Job ID를 선택하여 세부 정보를 확인한 다음 모델 학습을 진행합니다.

> **Note**
>
> Amazon Fraud Detector로 데이터 세트를 성공적으로 가져온 후 시스템에서 데이터를 완전히 수집할 수 있도록 10분간 기다리는 것이 좋습니다.

스마트 데이터 검증 보고서

스마트 데이터 검증은 검증이 완료된 후 검증 보고서를 생성합니다. 검증 보고서는 SV가 데이터세트에서 식별한 모든 문제에 대한 세부 정보와 가장 영향력 있는 문제를 해결하기 위한 권장 조치를 제공합니다. 검증 보고서를 사용하여 문제가 무엇인지, 데이터세트에서 문제가 있는 위치, 문제의 심각도 및 해결 방법을 확인할 수 있습니다. 유효성 검사가 성공적으로 완료된 경우에도 검증 보고서가 생성됩니다. 이 경우 보고서를 보고 나열된 문제가 있는지 확인하고 문제가 있는 경우 해당 문제를 해결할지 결정할 수 있습니다.

> **Note**
>
> 현재 버전의 SDV는 데이터세트를 스캔하여 배치 가져오기에 실패할 수 있는 문제를 찾아냅니다. 검증 및 일괄 가져오기에 성공하더라도 데이터세트에 여전히 모델 학습이 실패할 수 있는 문제가 있을 수 있습니다. 검증 및 가져오기가 성공했더라도 검증 보고서를 확인하고 성공적인 모델 학습을 위해 보고서에 나열된 모든 문제를 해결하는 것이 좋습니다. 문제를 해결한 후 새 일괄 가져오기 작업을 생성하십시오.

검증 보고서 액세스

검증이 완료된 후 언제든지 다음 옵션 중 하나를 사용하여 검증 보고서에 액세스할 수 있습니다.

1. 검증이 완료되고 가져오기 작업이 진행되는 동안 상단 배너에서 검증 보고서 보기를 선택합니다.
2. 가져오기 Job 완료된 후 이벤트 가져오기 데이터 창에서 방금 완료된 가져오기 작업의 작업 ID를 선택합니다.

검증 보고서 사용

가져오기 작업의 검증 보고서 페이지에서는 이 가져오기 작업의 세부 정보, 심각한 오류 (발견된 경우) 목록, 데이터셋의 특정 이벤트 (행) 에 대한 경고 목록 (발견된 경우), 유효하지 않은 값 및 각 변수의 누락된 값과 같은 정보가 포함된 데이터세트의 간략한 요약을 제공합니다.

- 작업 세부 정보 가져오기

가져오기 작업의 세부 정보를 제공합니다. 가져오기 작업이 실패했거나 데이터세트를 부분적으로 가져온 경우 결과 파일로 이동을 선택하면 가져오기에 실패한 이벤트의 오류 로그를 볼 수 있습니다.

- 심각한 오류

  SV에서 식별한 데이터세트에서 가장 영향력 있는 문제에 대한 세부 정보를 제공합니다. 이 창에 나열된 모든 문제는 중요하므로 가져오기를 진행하기 전에 문제를 해결해야 합니다. 중요한 문제를 해결하지 않고 데이터세트를 가져오려고 하면 가져오기 작업이 실패할 수 있습니다.

  중요한 문제를 해결하려면 각 경고에 제공된 권장 사항을 따르십시오. 심각한 오류 창에 나열된 문제를 모두 해결한 후 새 일괄 가져오기 작업을 생성하십시오.

- 데이터 수준 경고

  데이터세트의 특정 이벤트 (행) 에 대한 경고 요약을 제공합니다. 데이터 수준 경고 패널이 채워진 경우 데이터세트의 일부 이벤트가 검증에 실패하여 가져오지 못한 것입니다.

  각 경고의 설명 열에는 문제가 있는 이벤트 수가 표시됩니다. 또한 샘플 이벤트 ID는 문제가 있는 나머지 이벤트를 찾기 위한 출발점으로 사용할 수 있는 샘플 이벤트 ID의 일부 목록을 제공합니다. 경고와 관련하여 제공된 권장 사항을 사용하여 문제를 해결하십시오. 또한 출력 파일의 오류 로그를 사용하여 문제에 대한 추가 정보를 확인하십시오. 일괄 가져오기에 실패한 모든 이벤트에 대해 오류 로그가 생성됩니다. 오류 로그에 액세스하려면 작업 세부 정보 가져오기 창에서 결과 파일로 이동을 선택합니다.

  > **Note**
  >
  > 데이터세트의 이벤트 (행) 중 50% 이상이 검증에 실패하면 가져오기 작업도 실패합니다. 이 경우 새 가져오기 작업을 시작하기 전에 데이터를 수정해야 합니다.

- 데이터세트 요약

  데이터세트의 검증 보고서 요약을 제공합니다. 경고 수 열에 0개 이상의 경고가 표시되면 해당 경고를 수정해야 하는지 결정하십시오. 경고 횟수 열에 0이 표시되면 모델을 계속 훈련시키십시오.

Python용 AWS SDK (Boto3) 를 사용하여 이벤트 데이터를 Batch 가져오기

다음 예제에서는 CreateBatchImportJobAPI에 대한 샘플 요청을 보여 줍니다. 일괄 가져오기 작업에는 작업 ID, 입력 경로, 출력 경로 등이 포함되어야 iamRoleArn합니다. eventTypeName 작업이 CREATE_FAILED 상태에 있지 않는 한 JobID에는 이전 작업의 동일한 ID가 포함될 수 없습니다. 입력 경로와 출력 경로는 유효한 S3 경로여야 합니다. OutputPath에서 파일 이름을 지정하지 않도록 선택할

수 있지만 여전히 유효한 S3 버킷 위치를 제공해야 합니다. eventTypeName 및 가 iamRoleArn 존재해야 합니다. IAM 역할은 Amazon S3 버킷을 입력하기 위한 읽기 권한과 Amazon S3 버킷을 출력하기 위한 쓰기 권한을 부여해야 합니다.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.create_batch_import_job (
jobId = 'sample_batch_import',
inputPath = 's3://bucket_name/input_file_name.csv',
outputPath = 's3://bucket_name/',
eventTypeName = 'sample_registration',
iamRoleArn: 'arn:aws:iam::*************:role/service-role/AmazonFraudDetector-
DataAccessRole-*************'
)
```

일괄 가져오기 작업 취소

Amazon Fraud Detector 콘솔에서CancelBatchImportJob API 또는 AWS SDK를 사용하여 언제든지 진행 중인 배치 가져오기 작업을 취소할 수 있습니다.

콘솔에서 배치 가져오기 작업을 취소하려면

1. AWS 콘솔을 열고 계정에 로그인한 다음 Amazon Fraud Detector로 이동합니다.
2. 왼쪽 탐색 창에서 이벤트를 선택합니다.
3. 이벤트 유형을 선택합니다.
4. 저장된 이벤트 탭을 선택합니다.
5. 이벤트 가져오기 데이터 창에서 취소하려는 진행 중인 가져오기 작업의 작업 ID를 선택합니다.
6. 이벤트 작업 페이지에서 작업을 클릭하고 이벤트 가져오기 취소를 선택합니다.
7. 이벤트 가져오기 중지를 선택하여 일괄 가져오기 작업을 취소합니다.

Python용 AWS SDK (Boto3) 를 사용하여 배치 가져오기 작업을 취소하는 중입니다.

다음 예제에서는CancelBatchImportJob API에 대한 샘플 요청을 보여 줍니다. 가져오기 취소 작업에는 진행 중인 일괄 가져오기 작업의 작업 ID가 포함되어야 합니다.

```
import boto3
```

```
fraudDetector = boto3.client('frauddetector')
fraudDetector.cancel_batch_import_job (
    jobId = 'sample_batch'
)
```

GetEventPredictions API 작업을 사용하여 이벤트 데이터 저장

기본적으로 평가를 위해GetEventPrediction API로 전송되는 모든 이벤트는 Amazon Fraud Detector에 저장됩니다. 즉, Amazon Fraud Detector는 사용자가 예측을 생성할 때 자동으로 이벤트 데이터를 저장하고 해당 데이터를 사용하여 계산된 변수를 거의 실시간으로 업데이트합니다. Amazon Fraud Detector 콘솔에서 이벤트 유형으로 이동하여 이벤트 수집을 OFF로 설정하거나PutEventType API 작업을 사용하여 EventIngestion 값을 DISABLED로 업데이트하여 데이터 스토리지를 비활성화할 수 있습니다. GetEventPredictionAPI 작업에 대한 자세한 내용은 을 참조하십시오사기 예측.

> **Important**
>
> 이벤트 유형에 대해 이벤트 통합을 활성화한 후에는 활성화된 상태로 유지하는 것이 좋습니다. 동일한 Event 유형에 대해 Event 수집을 비활성화한 다음 예측을 생성하면 동작이 일관되지 않을 수 있습니다.

SendEvent API 작업을 사용하여 이벤트 데이터 저장

SendEventAPI 작업을 사용하면 해당 이벤트에 대한 사기 예측을 생성하지 않고도 Amazon Fraud Detector에 이벤트를 저장할 수 있습니다. 예를 들어,SendEvent 작업을 통해 과거 데이터세트를 업로드하여 나중에 모델을 훈련하는 데 사용할 수 있습니다.

SendEvent API용 이벤트 타임스탬프 형식

SendEventAPI를 사용하여 이벤트 데이터를 저장할 때는 이벤트 타임스탬프가 필수 형식인지 확인해야 합니다. Amazon Fraud Detector 다음과 같은 날짜/타임스탬프 형식을 지원합니다.

- %yyy-%mm-%DDT%HH: %mm: %ssZ (밀리초 없이 UTC로만 표시되는 ISO 8601 표준)

  예: 2019-11-30T13:01:01 Z
- %yyy/%mm/%dd %hh: %mm: %s (오전/오후)

예: 2019/11/30 오후 1:01:01 또는 2019/11/30 13:01:01

- %mm/%dd/%yyyy %hh: %mm: %s

  예: 2019년 11월 30일 오후 1:01:01, 2019년 11월 30일 13:01:01

- %mm/%dd/%yy %hh: %mm: %ss

  예: 11/30/19 오후 1:01:01, 11/30/19 13:01:01

Amazon Fraud Detector는 이벤트 타임스탬프의 날짜/타임스탬프 형식을 분석할 때 다음과 같은 가정을 합니다.

- ISO 8601 표준을 사용하는 경우 이전 사양과 정확히 일치해야 합니다.
- 다른 형식 중 하나를 사용하는 경우 유연성이 더욱 향상됩니다.
  - 월 및 일 단위로 한 자리 또는 두 자리 숫자를 제공할 수 있습니다. 예를 들어, 2019년 1월 12일은 유효한 날짜입니다.
  - hh:mm:ss가 없으면 포함하지 않아도 됩니다 (즉, 날짜를 입력하기만 하면 됩니다). 시간과 분의 하위 집합 (예: hh:mm) 을 제공할 수도 있습니다. 시간만 제공하는 것은 지원되지 않습니다. 밀리초도 지원되지 않습니다.
  - AM/PM 레이블을 제공하는 경우 12시간 시간을 기준으로 합니다. AM/PM 정보가 없는 경우 24시간 시계를 사용하는 것으로 가정합니다.
  - 날짜 요소의 구분 기호로 “/” 또는 “-”를 사용할 수 있습니다. 타임스탬프 요소에는 “:”가 사용됩니다.

다음은 예제SendEvent API 호출입니다.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.send_event(
            eventId        = '802454d3-f7d8-482d-97e8-c4b6db9a0428',
            eventTypeName  = 'sample_registration',
            eventTimestamp = '2020-07-13T23:18:21Z',
            eventVariables =  {
       'email_address' : 'johndoe@exampledomain.com',
       'ip_address' : '1.2.3.4'},
            assignedLabel  = ‘legit’,
            labelTimestamp = '2020-07-13T23:18:21Z',
```

```
            entities       = [{'entityType':'sample_customer', 'entityId':'12345'}],

)
```

저장된 이벤트 데이터의 세부 정보 가져오기

Amazon Fraud Detector에 이벤트 데이터를 저장한 후 GetEventAPI를 사용하여 이벤트에 대해 저장된 최신 데이터를 확인할 수 있습니다. 다음 예제 코드는`sample_registration` 이벤트에 대해 저장된 최신 데이터를 확인합니다.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.get_event(
            eventId        = '802454d3-f7d8-482d-97e8-c4b6db9a0428',
            eventTypeName  = 'sample_registration'
)
```

저장된 이벤트 데이터세트의 지표 보기

각 이벤트 유형에 대해 Amazon Fraud Detector 콘솔에서 저장된 이벤트 수, 저장된 이벤트의 총 크기, 가장 먼저 저장된 이벤트와 가장 최근에 저장된 이벤트의 타임스탬프와 같은 지표를 볼 수 있습니다.

이벤트 유형의 저장된 이벤트 지표를 보려면

1. AWS콘솔을 열고 계정에 로그인합니다. Amazon Fraud Detector로 이동합니다.
2. 왼쪽 탐색 창에서 이벤트를 선택합니다.
3. 이벤트 유형을 선택합니다.
4. 저장된 이벤트 탭을 선택합니다.
5. 저장된 이벤트 세부 정보 창에는 지표가 표시됩니다. 이러한 지표는 매일 한 번 자동으로 업데이트 됩니다.
6. 원하는 경우 이벤트 지표 새로 고침을 클릭하여 지표를 수동으로 업데이트할 수 있습니다.

> **Note**
>
> 방금 데이터를 가져온 경우 데이터 가져오기를 완료한 후 5~10분 정도 기다려 지표를 새로 고치고 확인하는 것이 좋습니다.

이벤트 오케스트레이션

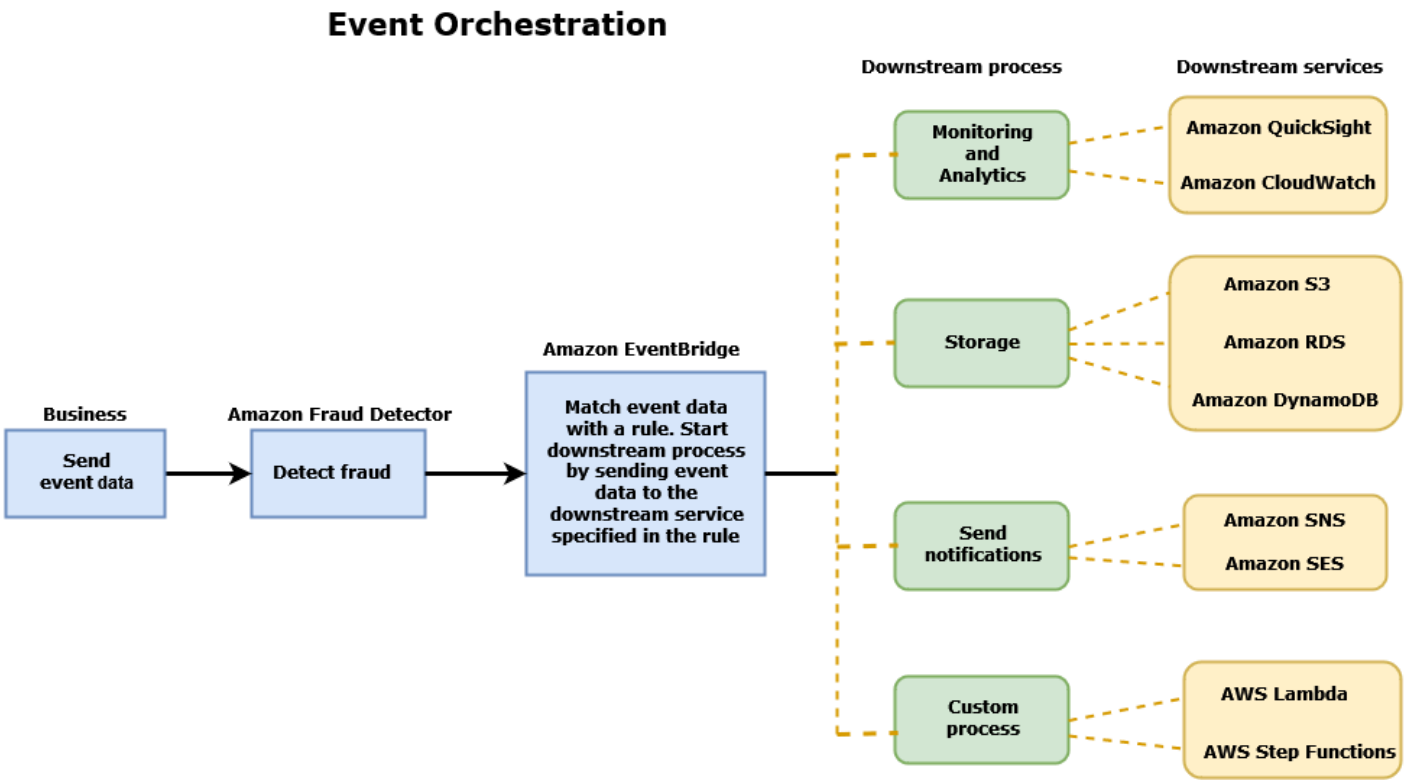
이벤트 오케스트레이션을 사용하면 Amazon을 사용하여 다운스트림 처리를 AWS 서비스 위해 이벤트를 쉽게 전송할 수 있습니다. EventBridge Amazon Fraud Detector는 사기 탐지 후 이벤트 처리를 자동화하는 데 사용할 수 있는 간단한 규칙을 제공합니다. 이벤트 오케스트레이션을 사용하면 이벤트를 대시보드로 전송하여 이벤트 데이터로부터 통찰력을 얻고, 사기 탐지 결과를 기반으로 알림을 생성하고, 사기 탐지에서 학습한 내용을 기반으로 레이블을 사용하여 이벤트를 업데이트하는 등의 다운스트림 이벤트 프로세스를 자동화할 수 있습니다.

이벤트 오케스트레이션을 사용하면 EventBridge Amazon을 통해 AWS 환경의 서비스에 쉽게 액세스할 수 있습니다. API 대상을 사용하여 이벤트를 직접 전송하거나 간접적으로 EventBridge 전송하도록 AWS 서비스 Amazon을 구성할 수 있습니다. 다운스트림 프로세스를 오케스트레이션하는 데 사용하는 AWS 서비스 것을 대상이라고도 합니다. 다운스트림 처리를 오케스트레이션하는 데 사용할 수 있는 일부 대상은 다음과 같습니다.

- 모니터링 및 분석용 — 아마존 QuickSight, 아마존 CloudWatch
- 스토리지용 — 아마존 S3, 아마존 RDS, 아마존 다이나모DB
- 알림 전송용 — 아마존 SNS, 아마존 SES
- 사용자 지정 프로세싱의 경우 — AWS Lambda, AWS Step Functions

Amazon에서 지원하는 오케스트레이션 대상에 대한 자세한 내용은 Amazon EventBridge EventBridge 대상을 참조하십시오.

다음 다이어그램은 이벤트 오케스트레이션의 작동 방식을 개괄적으로 보여줍니다.



이벤트 오케스트레이션 설정

이벤트에 대한 이벤트 오케스트레이션을 설정하려면 대상 서비스에서 프로세스를 설정하고, 이벤트 데이터를 수신 및 EventBridge 전송하도록 Amazon을 구성하고, 다운스트림 프로세스를 시작하기 위한 조건을 EventBridge 지정하는 규칙을 Amazon에서 생성해야 합니다. 이벤트 오케스트레이션을 설정하려면 다음 단계를 완료하십시오.

이벤트 오케스트레이션을 설정하려면

1. Amazon 사용 EventBridge 설명서로 이동하여 Amazon 사용 방법을 알아보십시오 EventBridge. Amazon에서 사용 사례에 EventBridge 맞는 규칙을 생성하는 방법을 알아보십시오.
2. 의 지침을 따르십시오Amazon Fraud Detector에서 이벤트 오케스트레이션을 활성화합니다..

> **Note**
>
> 이벤트의 이벤트 오케스트레이션은 기본적으로 비활성화되어 있습니다.

3. 대상 서비스가 이벤트 데이터를 수신하고 처리하도록 설정하십시오. 예를 들어, 다운스트림 프로세스에 알림 전송이 포함되고 Amazon SNS를 사용하려는 경우 Amazon SNS 콘솔로 이동하여 SNS 주제를 생성한 다음 해당 주제에 대한 엔드포인트를 구독하십시오.

4. 지침에 따라 Amazon EventBridge 규칙을 생성하십시오.

> **Important**
>
> EventBridgeAmazon에서 이벤트 패턴을 작성할 때는 소스 필드와 `Event Prediction Result Returned` 세부 정보 유형 필드를 제공해야 *`aws.frauddetector`* 합니다.

Amazon Fraud Detector에서 이벤트 오케스트레이션을 활성화합니다.

이벤트 유형을 생성할 때 또는 이벤트 유형을 생성한 후에 이벤트에 대한 이벤트 오케스트레이션을 활성화할 수 있습니다. Amazon Fraud Detector 콘솔에서 `put-event-type` 명령, `PutEventType` API 또는 를 사용하여 이벤트 오케스트레이션을 활성화할 수 있습니다. AWS SDK for Python (Boto3)

Amazon Fraud Detector 콘솔에서 이벤트 오케스트레이션을 활성화합니다.

이 예제는 이미 생성된 이벤트 유형에 대한 이벤트 오케스트레이션을 활성화합니다. 새 이벤트 유형을 만들고 오케스트레이션을 활성화하려면 의 지침을 따르십시오. 이벤트 유형 생성

이벤트 오케스트레이션을 활성화하려면

1. AWS관리 콘솔을 열고 계정에 로그인합니다. Amazon Fraud Detector로 이동합니다.
2. 왼쪽 탐색 창에서 이벤트를 선택합니다.
3. 이벤트 유형 페이지에서 이벤트 유형을 선택합니다.
4. EventBridgeAmazon에서 이벤트 오케스트레이션 활성화를 켜십시오.
5. 에 대한 이벤트 오케스트레이션 설정 3단계 지침을 계속 진행하십시오.

를 사용하여 이벤트 오케스트레이션을 활성화합니다. AWS SDK for Python (Boto3)

다음 예는 이벤트 오케스트레이션을 `sample_registration` 활성화하기 위해 이벤트 유형을 업데이트하기 위한 샘플 요청을 보여줍니다. 이 예제에서는 `PutEventType` API를 사용하며 변수`email_address`, 레이블 `ip_address` `legit` 및 `fraud` 엔티티 유형을 생성했다고 가정합니다. `sample_customer` 이러한 리소스를 만드는 방법에 대한 자세한 내용은 리소스를 참조하십시오.

```
import boto3
fraudDetector = boto3.client('frauddetector')
fraud_detector.put_event_type(
  name = 'sample_registration',
  eventVariables = ['ip_address', 'email_address'],
  eventOrchestration = {'eventBridgeEnabled': True},
  labels = ['legit', 'fraud'],
  entityTypes = ['sample_customer'])
```

Amazon Fraud Detector에서 이벤트 오케스트레이션을 비활성화합니다

Amazon Fraud Detector 콘솔에서 `put-event-type` 명령, `PutEventType` API 또는 를 사용하여 언제든지 이벤트에 대한 이벤트 오케스트레이션을 비활성화할 수 있습니다. AWS SDK for Python (Boto3)

Amazon Fraud Detector 콘솔에서 이벤트 오케스트레이션을 비활성화합니다.

이벤트 오케스트레이션을 비활성화하려면

1. AWS관리 콘솔을 열고 계정에 로그인합니다. Amazon Fraud Detector로 이동합니다.
2. 왼쪽 탐색 창에서 이벤트를 선택합니다.
3. 이벤트 유형 페이지에서 이벤트 유형을 선택합니다.
4. EventBridgeAmazon에서 이벤트 오케스트레이션 활성화를 끕니다.

를 사용하여 이벤트 오케스트레이션을 비활성화합니다. AWS SDK for Python (Boto3)

다음 예제는 API를 사용하여 이벤트 오케스트레이션을 `sample_registration` 비활성화하도록 이벤트 유형을 업데이트하기 위한 샘플 요청을 보여줍니다. `PutEventType`

```
import boto3
fraudDetector = boto3.client('frauddetector')
fraud_detector.put_event_type(
  name = 'sample_registration',
```

```
  eventVariables = ['ip_address', 'email_address'],
  eventOrchestration = {'eventBridgeEnabled': False},
  entityTypes = ['sample_customer'])
```

모델

Amazon Fraud Detector는 기계 학습 모델을 사용하여 사기 예측을 생성합니다. 각 모델은 모델 유형을 사용하여 학습됩니다. 모델 유형은 모델 학습에 사용되는 알고리즘과 변환을 지정합니다. 모델 학습은 제공된 데이터세트를 사용하여 사기 이벤트를 예측할 수 있는 모델을 만드는 프로세스입니다.

모델을 만들려면 먼저 모델 유형을 선택한 다음 모델 학습에 사용할 데이터를 준비하여 제공해야 합니다.

모델 유형을 선택하세요.

Amazon Fraud Detector에서는 다음과 같은 모델 유형을 사용할 수 있습니다. 사용 사례에 맞는 모델 유형을 선택하십시오.

- 온라인 사기 인사이트

  Online Fraud Insights 모델 유형은 평가 대상 기업에 대한 과거 데이터가 거의 없을 때 사기를 탐지하도록 최적화되어 있습니다 (예: 새 계정을 온라인으로 등록하는 신규 고객).
- 거래 사기 인사이트

  거래 사기 인사이트 모델 유형은 평가 대상 주체가 예측 정확도를 높이기 위해 분석할 수 있는 상호작용 기록을 가지고 있을 수 있는 사기 사용 사례를 탐지하는 데 가장 적합합니다 (예: 과거 구매 내역이 있는 기존 고객).
- 계정 탈취 인사이트

  계정 탈취 인사이트 모델 유형은 계정이 피싱이나 다른 유형의 공격으로 인해 손상되었는지 감지합니다. 침해된 계정 (예: 로그인 시 사용한 브라우저 및 장치) 의 로그인 데이터는 해당 계정과 관련된 과거 로그인 데이터와 다릅니다.

온라인 사기 인사이트

Online Fraud Insights는 감독형 기계 학습 모델입니다. 즉, 사기 및 합법적인 거래의 과거 사례를 사용하여 모델을 학습시킵니다. Online Fraud Insights 모델은 소량의 과거 데이터를 기반으로 사기를 탐지할 수 있습니다. 모델 입력은 유연하므로 가짜 리뷰, 프로모션 악용, 게스트 체크아웃 사기 등 다양한 사기 위험을 탐지하도록 모델을 조정할 수 있습니다.

Online Fraud Insights 모델은 다양한 기계 학습 알고리즘을 사용하여 데이터 강화, 변환 및 사기 분류를 수행합니다. 모델 교육 프로세스의 일환으로 Online Fraud Insights는 IP 주소 또는 신용 카드 발급

은행의 지리적 위치와 같은 타사 데이터로 IP 주소 및 BIN 번호와 같은 원시 데이터 요소를 보강합니다. Online Fraud Insights는 타사 데이터 외에도 Amazon에서 발견된 사기 패턴을 고려하는 딥 러닝 알고리즘을 사용합니다AWS. 이러한 사기 패턴은 그래디언트 트리 부스팅 알고리즘을 사용하여 모델의 입력 기능이 됩니다.

성능을 높이기 위해 Online Fraud Insights는 베이지안 최적화 프로세스를 통해 그래디언트 트리 부스팅 알고리즘의 하이퍼 파라미터를 최적화합니다. 다양한 모델 매개변수 (예: 나무 수, 나무 깊이, 잎당 샘플 수) 를 사용하여 수십 개의 서로 다른 모델을 순차적으로 학습시킵니다. 또한 매우 낮은 사기율을 처리하기 위해 소수 부정 행위 집단을 늘리는 등 다양한 최적화 전략을 사용합니다.

데이터 소스 선택

온라인 사기 인사이트 모델을 교육할 때는 외부 (Amazon Fraud Detector 외부) 에 저장되거나 Amazon Fraud Detector 내에 저장되는 이벤트 데이터를 기반으로 모델을 학습하도록 선택할 수 있습니다. 현재 Amazon Fraud Detector에서 지원하는 외부 스토리지는 아마존 심플 스토리지 서비스 (Amazon S3) 입니다. 외부 스토리지를 사용하는 경우 이벤트 데이터세트를 쉼표로 구분된 값 (CSV) 형식으로 Amazon S3 버킷에 업로드해야 합니다. 모델 교육 구성 내에서 이러한 데이터 스토리지 옵션을 EXTERNAL_EVENTS (외부 스토리지용) 및 INGESTED_EVENTS (내부 스토리지용) 라고 합니다. 사용 가능한 데이터 원본 및 해당 데이터 원본에 데이터를 저장하는 방법에 대한 자세한 내용은 을 참조하십시오. 이벤트 데이터 스토리지

데이터 준비

이벤트 데이터를 어디에 저장하든 (Amazon S3 또는 Amazon Fraud Detector), 온라인 사기 인사이트 모델 유형에 대한 요구 사항은 동일합니다.

데이터세트에는 EVENT_LABEL 열 헤더가 포함되어야 합니다. 이 변수는 이벤트를 사기 또는 합법적 이벤트로 분류합니다. CSV 파일 (외부 저장소) 을 사용하는 경우 파일의 각 이벤트에 대해 EVENT_LABEL을 포함해야 합니다. 내부 저장소의 경우 EVENT_LABEL 필드는 선택 사항이지만 교육 데이터세트에 포함되려면 모든 이벤트에 레이블을 지정해야 합니다. 모델 학습을 구성할 때 레이블이 지정되지 않은 이벤트를 무시할지, 레이블이 지정되지 않은 이벤트에 대해 합법적인 레이블로 가정할지, 레이블이 지정되지 않은 모든 이벤트에 대해 허위 레이블을 적용할지 선택할 수 있습니다.

데이터 선택

온라인 사기 인사이트 모델 교육을 위한 데이터 선택에 대한 자세한 내용은 이벤트 데이터 수집을 참조하십시오.

온라인 사기 인사이트 교육에서는 EVENT_TIMESTAMP를 기반으로 이전 데이터를 샘플링하고 분할합니다. 데이터를 수동으로 샘플링할 필요가 없으며 이렇게 하면 모델 결과에 부정적인 영향을 미칠 수 있습니다.

이벤트 변수

Online Fraud Insights 모델에는 필수 이벤트 메타데이터 외에 모델 학습을 위한 데이터 검증을 통과하고 모델당 최대 100개의 변수를 허용하는 변수가 두 개 이상 필요합니다. 일반적으로 더 많은 변수를 제공할수록 모델이 사기 사건과 합법적인 사건을 더 잘 구분할 수 있습니다. Online Fraud Insights 모델은 사용자 지정 변수를 포함하여 수십 개의 변수를 지원할 수 있지만 IP 주소와 이메일 주소를 포함하는 것이 좋습니다. 이러한 변수는 일반적으로 평가 대상 개체를 식별하는 데 가장 효과적이기 때문입니다.

데이터 검증

교육 프로세스의 일환으로 Online Fraud Insights는 모델 학습에 영향을 미칠 수 있는 데이터 품질 문제가 있는지 데이터세트를 검증합니다. Amazon Fraud Detector는 데이터를 검증한 후 적절한 조치를 취하여 가능한 최상의 모델을 구축합니다. 여기에는 잠재적 데이터 품질 문제에 대한 경고 발행, 데이터 품질 문제가 있는 변수 자동 제거, 오류 발생 및 모델 교육 프로세스 중단이 포함됩니다. 자세한 내용은 데이터세트 검증을 참조하세요.

거래 사기 인사이트

거래 사기 인사이트 모델 유형은 온라인 또는 card-not-present 거래 사기를 탐지하도록 설계되었습니다. Transaction Fraud Insights는 감독형 기계 학습 모델입니다. 즉, 사기 및 합법적인 거래의 과거 사례를 사용하여 모델을 학습시킵니다.

Transaction Fraud Insights 모델은 다양한 기계 학습 알고리즘을 사용하여 데이터 강화, 변환 및 사기 분류를 수행합니다. 기능 엔지니어링 엔진을 활용하여 개체 수준 및 이벤트 수준 집계를 생성합니다. 모델 교육 프로세스의 일환으로 Transaction Fraud Insights는 IP 주소 및 BIN 번호와 같은 원시 데이터 요소를 IP 주소 또는 신용 카드 발급 은행의 지리적 위치와 같은 타사 데이터로 보강합니다. Transaction Fraud Insights는 타사 데이터 외에도 Amazon에서 발견된 사기 패턴을 고려하는 딥 러닝 알고리즘을 사용합니다. AWS 이러한 사기 패턴은 그래디언트 트리 부스팅 알고리즘을 사용하여 모델의 입력 기능이 됩니다.

성능을 높이기 위해 Transaction Fraud Insights는 베이지안 최적화 프로세스를 통해 그래디언트 트리 부스팅 알고리즘의 하이퍼 파라미터를 최적화하고, 다양한 모델 매개변수 (예: 트리 수, 트리 깊이, 리프당 샘플 수) 를 사용하여 수십 개의 서로 다른 모델을 순차적으로 트레이닝하고, 소수 부정 행위 집단의 가중치를 높여 매우 낮은 사기율을 처리하는 등의 다양한 최적화 전략을 사용합니다.

모델 교육 프로세스의 일환으로 Transaction Fraud 모델의 기능 엔지니어링 엔진은 교육 데이터 세트 내의 각 고유 엔티티에 대한 값을 계산하여 사기 예측을 개선하는 데 도움이 됩니다. 예를 들어, 교육 프로세스 중에 Amazon Fraud Detector는 엔티티가 마지막으로 구매한 시간을 계산 및 저장하고, `GetEventPrediction` 또는 `SendEvent` API를 호출할 때마다 이 값을 동적으로 업데이트합니다. 사기 예측 중에는 이벤트 변수를 다른 개체 및 이벤트 메타데이터와 결합하여 거래가 부정 거래인지 여부를 예측합니다.

데이터 소스 선택

거래 사기 인사이트 모델은 Amazon Fraud Detector (INGESTED_EVENTS) 를 사용하여 내부적으로 저장된 데이터 세트에 대해서만 학습됩니다. 이를 통해 Amazon Fraud Detector는 평가 중인 항목에 대한 계산된 값을 지속적으로 업데이트할 수 있습니다. 사용 가능한 데이터 소스에 대한 자세한 내용은 을 참조하십시오. 이벤트 데이터 스토리지

데이터 준비

거래 사기 인사이트 모델을 학습시키기 전에 이벤트 데이터세트 준비에 설명된 대로 데이터 파일에 모든 헤더가 포함되어 있는지 확인하세요. Transaction Fraud Insights 모델은 새로 수신된 엔티티를 데이터세트에 있는 사기성 및 합법적인 엔티티의 예와 비교하므로 각 엔티티에 대해 많은 예를 제공하는 것이 좋습니다.

Amazon Fraud Detector는 저장된 이벤트 데이터 세트를 교육에 적합한 형식으로 자동 변환합니다. 모델 학습이 완료되면 성능 지표를 검토하고 교육 데이터 세트에 항목을 추가해야 하는지 여부를 결정할 수 있습니다.

데이터 선택

기본적으로 거래 사기 인사이트는 선택한 이벤트 유형에 대해 저장된 전체 데이터세트를 기반으로 학습합니다. 선택적으로 시간 범위를 설정하여 모델 학습에 사용되는 이벤트를 줄일 수 있습니다. 시간 범위를 설정할 때는 모델 학습에 사용되는 레코드가 완성되기까지 충분한 시간이 있었는지 확인하십시오. 즉, 합법적인 기록과 사기 기록을 정확히 식별할 수 있을 만큼 충분한 시간이 지났습니다. 예를 들어 차지백 사기의 경우 사기 사건을 정확히 식별하는 데 60일 이상이 걸리는 경우가 많습니다. 최상의 모델 성능을 위해서는 교육 데이터세트의 모든 기록이 완전한지 확인하세요.

이상적인 사기율을 나타내는 시간 범위를 선택할 필요는 없습니다. Amazon Fraud Detector는 데이터를 자동으로 샘플링하여 사기율, 시간 범위 및 개체 수 간의 균형을 유지합니다.

Amazon Fraud Detector는 모델 학습에 필요한 이벤트가 충분하지 않은 시간 범위를 선택하면 모델 교육 중에 검증 오류를 반환합니다. 저장된 데이터 세트의 경우 EVENT_LABEL 필드는 선택 사항이지만

교육 데이터 세트에 포함되려면 이벤트에 레이블을 지정해야 합니다. 모델 학습을 구성할 때 레이블이 지정되지 않은 이벤트를 무시할지, 레이블이 지정되지 않은 이벤트에 대해 합법적인 레이블로 가정할지, 레이블이 지정되지 않은 이벤트에 대해 허위 레이블로 가정할지 선택할 수 있습니다.

이벤트 변수

모델 학습에 사용되는 이벤트 유형에는 필수 이벤트 메타데이터를 제외하고 데이터 검증을 통과하고 최대 100개의 변수를 포함할 수 있는 변수가 2개 이상 포함되어야 합니다. 일반적으로 더 많은 변수를 제공할수록 모델이 사기 이벤트와 합법적인 이벤트를 더 잘 구분할 수 있습니다. Transaction Fraud Insight 모델은 사용자 지정 변수를 포함하여 수십 개의 변수를 지원할 수 있지만 IP 주소, 이메일 주소, 결제 수단 유형, 주문 가격 및 카드 BIN을 포함하는 것이 좋습니다.

데이터 검증

교육 프로세스의 일환으로 Transaction Fraud Insights는 모델 학습에 영향을 미칠 수 있는 데이터 품질 문제가 있는지 교육 데이터 세트를 검증합니다. Amazon Fraud Detector는 데이터를 검증한 후 적절한 조치를 취하여 가능한 최상의 모델을 구축합니다. 여기에는 잠재적 데이터 품질 문제에 대한 경고 발행, 데이터 품질 문제가 있는 변수 자동 제거, 오류 발생 및 모델 교육 프로세스 중단이 포함됩니다. 자세한 내용은 데이터세트 검증을 참조하세요.

Amazon Fraud Detector는 고유 개체 수가 1,500개 미만인 경우 교육 데이터의 품질에 영향을 줄 수 있으므로 경고를 발행하지만 모델을 계속 학습시킵니다. 경고를 받으면 성능 지표를 검토하십시오.

계정 탈취 인사이트

ATI (Account Takeover Insights) 모델 유형은 계정이 악의적인 도용, 피싱 또는 자격 증명 도용을 통해 손상되었는지 탐지하여 온라인 사기 활동을 식별합니다. Account Takeover Insights는 온라인 비즈니스의 로그인 이벤트를 사용하여 모델을 학습시키는 기계 학습 모델입니다.

실시간 로그인 흐름에 훈련된 Account Takeover Insights 모델을 내장하여 계정 침해 여부를 감지할 수 있습니다. 이 모델은 다양한 인증 및 로그인 유형을 평가합니다. 여기에는 웹 애플리케이션 로그인, API 기반 인증 및 (SSO) 가 포함됩니다. single-sign-on 계정 테이크오버 인사이트 모델을 사용하려면 유효한 로그인 자격 증명을 제시한 후 GetEventPredictionAPI를 호출하십시오. API는 계정 도용 위험을 수치화하는 점수를 생성합니다. Amazon Fraud Detector는 사용자가 정의한 점수와 규칙을 사용하여 로그인 이벤트에 대해 하나 이상의 결과를 반환합니다. 결과는 사용자가 구성한 결과입니다. 수신한 결과에 따라 각 로그인에 대해 적절한 조치를 취할 수 있습니다. 즉, 로그인을 위해 제공된 자격 증명을 승인하거나 이의를 제기할 수 있습니다. 예를 들어 추가 인증 수단으로 계정 PIN을 요청하여 자격 증명에 이의를 제기할 수 있습니다.

또한 계정 탈취 인사이트 모델을 사용하여 비동기적으로 계정 로그인을 평가하고 고위험 계정에 조치를 취할 수 있습니다. 예를 들어 검토자가 계정 일시 중지와 같은 추가 조치가 필요한지 판단할 수 있도록 고위험 계정을 조사 대기열에 추가할 수 있습니다.

계정 탈취 인사이트 모델은 비즈니스의 과거 로그인 이벤트가 포함된 데이터세트를 사용하여 학습됩니다. 이 데이터를 제공합니다. 필요에 따라 계정을 합법적이거나 사기성 계정으로 표시할 수 있습니다. 하지만 모델을 학습시키는 데 반드시 필요한 것은 아닙니다. 계정 탈취 인사이트 모델은 계정의 성공적인 로그인 기록을 기반으로 이상 현상을 탐지합니다. 또한 악의적인 계정 도용 이벤트의 위험 증가를 시사하는 사용자 행동의 이상을 탐지하는 방법도 학습합니다. 일반적으로 동일한 장치 및 IP 주소를 사용하여 로그인하는 사용자를 예로 들 수 있습니다. 사기범은 일반적으로 다른 장치 및 지리적 위치에서 로그인합니다. 이 기법은 비정상적인 활동에 대한 위험 점수를 산출하는데, 이는 일반적으로 악의적인 계정 도용의 주요 특징입니다.

계정 탈취 인사이트 모델을 교육하기 전에 Amazon Fraud Detector는 여러 기계 학습 기술을 조합하여 데이터 강화, 데이터 집계 및 데이터 변환을 수행합니다. 그런 다음, Amazon Fraud Detector는 교육 프로세스 중에 사용자가 제공하는 원시 데이터 요소를 보강합니다. 원시 데이터 요소의 예로는 IP 주소 및 사용자 에이전트가 있습니다. Amazon Fraud Detector는 이러한 요소를 사용하여 로그인 데이터를 설명하는 추가 입력을 생성합니다. 이러한 입력에는 디바이스, 브라우저 및 지리적 위치 입력이 포함됩니다. 또한 Amazon Fraud Detector는 사용자가 제공한 로그인 데이터를 사용하여 과거 사용자 행동을 설명하는 집계된 변수를 지속적으로 계산합니다. 사용자 행동의 예로는 사용자가 특정 IP 주소에서 로그인한 횟수 등이 있습니다. Amazon Fraud Detector는 이러한 추가 보강 및 집계를 사용하여 로그인 이벤트의 작은 입력 집합에서 강력한 모델 성능을 생성할 수 있습니다.

Account Takeover Insights 모델은 악의적인 공격자가 사람인지 로봇인지에 관계없이 악의적인 공격자가 합법적인 계정에 액세스하는 사례를 탐지합니다. 이 모델은 계정 침해의 상대적 위험을 나타내는 단일 점수를 산출합니다. 침해되었을 수 있는 계정은 고위험 계정으로 플래그가 지정됩니다. 두 가지 방법 중 하나로 고위험 계정을 처리할 수 있습니다. 어느 쪽이든 추가 신원 확인을 시행할 수 있습니다. 또는 계정을 대기열로 보내 수동 조사를 진행할 수도 있습니다.

데이터 소스 선택

계정 탈취 인사이트 모델은 Amazon Fraud Detector에 내부적으로 저장된 데이터 세트를 기반으로 학습됩니다. Amazon Fraud Detector에 로그인 이벤트 데이터를 저장하려면 사용자의 로그인 이벤트가 포함된 CSV 파일을 생성하십시오. 각 이벤트에 이벤트 타임스탬프, 사용자 ID, IP 주소, 사용자 에이전트, 로그인 데이터의 유효 여부와 같은 로그인 데이터를 포함하십시오. CSV 파일을 생성한 후 먼저 Amazon Fraud Detector에 파일을 업로드한 다음 가져오기 기능을 사용하여 데이터를 저장합니다. 그런 다음 저장된 데이터를 사용하여 모델을 학습시킬 수 있습니다. Amazon Fraud Detector를 사용하여 이벤트 데이터 세트를 저장하는 방법에 대한 자세한 내용은 다음을 참조하십시오. Amazon Fraud Detector를 사용하여 이벤트 데이터를 내부적으로 저장하십시오.

데이터 준비

Amazon Fraud Detector에서는 UTF-8 형식으로 인코딩된 CSV (쉼표로 구분된 값) 파일로 사용자 계정 로그인 데이터를 제공해야 합니다. CSV 파일의 첫 줄에는 파일 헤더가 포함되어야 합니다. 파일 헤더는 이벤트 메타데이터와 각 데이터 요소를 설명하는 이벤트 변수로 구성됩니다. 이벤트 데이터는 헤더 뒤에 옵니다. 이벤트 데이터의 각 줄은 단일 로그인 이벤트의 데이터로 구성됩니다.

Account Takeover Insights 모델의 경우 CSV 파일의 헤더 라인에 다음과 같은 이벤트 메타데이터와 이벤트 변수를 제공해야 합니다.

이벤트 메타데이터

CSV 파일 헤더에 다음 메타데이터를 제공하는 것이 좋습니다. 이벤트 메타데이터는 대문자여야 합니다.

- EVENT_ID - 로그인 이벤트의 고유 식별자입니다.
- ENTITY_TYPE - 로그인 이벤트를 수행하는 주체 (예: 판매자 또는 고객).
- ENTITY_ID - 로그인 이벤트를 수행하는 엔티티의 식별자입니다.
- EVENT_TIMESTAMP - 로그인 이벤트가 발생한 시점의 타임스탬프입니다. 타임스탬프는 ISO 8601 표준 (UTC) 을 준수해야 합니다.
- EVENT_LABEL (권장) - 이벤트를 사기 또는 합법적인 이벤트로 분류하는 라벨입니다. “사기”, “합법적”, “1" 또는 “0"과 같은 모든 레이블을 사용할 수 있습니다.

> **Note**
>
> - 이벤트 메타데이터는 대문자여야 합니다. 대소문자를 구분합니다.
> - 로그인 이벤트에는 레이블이 필요하지 않습니다. 하지만 EVENT_LABEL 메타데이터를 포함하고 로그인 이벤트의 레이블을 제공하는 것이 좋습니다. 레이블이 불완전하거나 산발적이어도 괜찮습니다. 라벨을 제공하면 Amazon Fraud Detector가 라벨을 사용하여 계정 도용 발견률을 자동으로 계산하고 모델 성과 차트와 표에 표시합니다.

이벤트 변수

계정 인수 인사이트 모델의 경우 반드시 제공해야 하는 필수 (필수) 변수와 선택적 변수가 모두 있습니다. 변수를 만들 때는 변수를 올바른 변수 유형에 할당해야 합니다. 모델 교육 프로세스의 일환으로

Amazon Fraud Detector는 변수와 연결된 변수 유형을 사용하여 변수 강화 및 기능 엔지니어링을 수행합니다.

> **Note**
>
> 이벤트 변수 이름은 소문자여야 합니다. 대소문자를 구분합니다.

필수 변수

계정 인수 인사이트 모델을 학습하려면 다음 변수가 필요합니다.

범주	변수 유형	설명
IP 주소	IP_ADDRESS	로그인 이벤트에 사용된 IP 주소
브라우저 및 장치	유저 에이전트	로그인 이벤트에 사용된 브라우저, 디바이스, OS
유효한 자격 증명	유효성이 검증됨	로그인에 사용된 자격 증명이 유효한지 여부를 나타냅니다.

선택적 변수

다음 변수는 계정 인수 인사이트 모델 교육을 위한 선택 사항입니다.

범주	유형	설명
브라우저 및 기기	지문	브라우저 또는 기기 핑거프린트의 고유 식별자
세션 ID	SESSION_ID	인증 세션의 식별자입니다.
레이블	이벤트_라벨	이벤트를 사기 또는 합법적인 이벤트로 분류하는 라벨입니다. “사기”, “합법적”, “1" 또는 “0"과 같은 모든 레이블을 사용할 수 있습니다.

범주	유형	설명
Timestamp	라벨_타임스탬프	라벨이 마지막으로 업데이트된 시점의 타임스탬프입니다. EVENT_LABEL이 제공되는 경우 이는 필수입니다.

> **Note**
>
> - 두 필수 변수 (선택 변수) 에 원하는 변수 이름을 제공할 수 있습니다. 각 필수 및 선택적 변수를 올바른 변수 유형에 할당하는 것이 중요합니다.
> - 추가 변수를 제공할 수 있습니다. 하지만 Amazon Fraud Detector에는 계정 탈취 인사이트 모델 교육을 위한 이러한 변수가 포함되지 않습니다.

데이터 선택

데이터 수집은 계정 탈취 인사이트 모델을 만드는 중요한 단계입니다. 로그인 데이터를 수집하기 시작할 때 다음 요구 사항 및 권장 사항을 고려하십시오.

필수

- 1,500개 이상의 사용자 계정 예시를 제공하고 각 예시마다 최소 2개의 관련 로그인 이벤트를 포함하십시오.
- 데이터세트는 최소 30일간의 로그인 이벤트를 포함해야 합니다. 나중에 모델 학습에 사용할 이벤트의 특정 시간 범위를 지정할 수 있습니다.

권장

- 데이터셋에는 실패한 로그인 이벤트의 예가 포함되어 있습니다. 이러한 실패한 로그인에 '사기' 또는 '합법적'이라는 라벨을 붙일 수도 있습니다.
- 6개월 이상의 로그인 이벤트와 함께 10만 개의 엔티티를 포함하여 과거 데이터를 준비하세요.

최소 요구 사항을 이미 충족하는 데이터 세트가 없는 경우 SendEventAPI 작업을 호출하여 Amazon Fraud Detector로 이벤트 데이터를 스트리밍하는 것을 고려해 보십시오.

데이터 검증

계정 탈취 인사이트 모델을 생성하기 전에 Amazon Fraud Detector는 모델 교육을 위해 데이터 세트에 포함시킨 메타데이터와 변수가 크기 및 형식 요구 사항을 충족하는지 확인합니다. 자세히 알아보려면 데이터 세트 검증의 내용을 참조하세요. 또한 다른 요구 사항도 확인합니다. 데이터셋이 검증을 통과하지 못하면 모델이 생성되지 않습니다. 모델을 성공적으로 만들려면 다시 훈련하기 전에 검증을 통과하지 못한 데이터를 수정해야 합니다.

일반적인 데이터세트 오류

계정 탈취 인사이트 모델 교육을 위해 데이터 세트를 검증할 때 Amazon Fraud Detector는 이러한 문제와 기타 문제를 스캔한 후 하나 이상의 문제가 발생하면 오류를 발생시킵니다.

- CSV 파일은 UTF-8 형식이 아닙니다.
- CSV 파일 헤더에는`EVENT_ID`,`ENTITY_ID`, 또는 메타데이터 중 하나 이상이 포함되어 있지 않습니다. `EVENT_TIMESTAMP`
- CSV 파일 헤더에는 `IP_ADDRESS`,`USERAGENT`, 또는 변수 유형 중 하나 이상의 변수가 포함되어 있지 않습니다. `VALIDCRED`
- 동일한 변수 유형에 연결된 변수가 두 개 이상 있습니다.
- 의 값 중 0.1% 이상이 지원되는 날짜 및 타임스탬프 형식이 아닌 null이나 값을 `EVENT_TIMESTAMP` 포함합니다.
- 첫 번째 이벤트와 마지막 이벤트 사이의 일 수는 30일 미만입니다.
- 변수 유형의 `IP_ADDRESS` 변수 중 10% 이상이 유효하지 않거나 null입니다.
- 변수 유형의 `USERAGENT` 변수 중 50% 이상이 null을 포함합니다.
- `VALIDCRED`변수 유형의 모든 변수는 로 설정됩니다. `false`

모델 빌드

Amazon Fraud Detector 모델은 특정 이벤트 유형에 대한 사기를 탐지하는 방법을 학습합니다. Amazon Fraud Detector에서는 먼저 모델 버전의 컨테이너 역할을 하는 모델을 생성합니다. 모델을 학습시킬 때마다 새 버전이 생성됩니다. AWS콘솔을 사용하여 모델을 만들고 학습시키는 방법에 대한 자세한 내용은 을 참조하십시오3단계: 모델 생성.

각 모델에는 해당하는 모델 점수 변수가 있습니다. Amazon Fraud Detector는 모델을 생성할 때 사용자를 대신하여 이 변수를 생성합니다. 규칙 표현식에서 이 변수를 사용하여 사기 평가 중에 모델 점수를 해석할 수 있습니다.

를 사용하여 모델을 훈련하고 배포하십시오. AWS SDK for Python (Boto3)

CreateModel및 CreateModelVersion 작업을 호출하여 모델 버전을 생성합니다. CreateModel모델 버전의 컨테이너 역할을 하는 모델을 시작합니다. CreateModelVersion학습 프로세스를 시작하여 모델의 특정 버전을 생성합니다. CreateModelVersion을 호출할 때마다 새 솔루션 버전이 생성됩니다.

다음 예제는 CreateModel API에 대한 샘플 요청을 보여줍니다. 이 예제에서는 Online Fraud Insights 모델 유형을 생성하고 이벤트 유형을 sample_registration 생성했다고 가정합니다. 이벤트 유형 생성에 대한 자세한 내용은 을 참조하십시오이벤트 유형 생성.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.create_model (
modelId = 'sample_fraud_detection_model',
eventTypeName = 'sample_registration',
modelType = 'ONLINE_FRAUD_INSIGHTS')
```

CreateModelVersionAPI를 사용하여 첫 번째 버전을 학습시키십시오. 의 TrainingDataSource 경우 교육 데이터 세트의 소스 및 Amazon S3 위치를 ExternalEventsDetail 지정하십시오. 를 위해 Amazon Fraud Detector가 교육 데이터를 해석하는 방법, 특히 포함할 이벤트 변수와 이벤트 레이블을 분류하는 방법을 TrainingDataSchema 지정하십시오. 기본적으로 Amazon Fraud Detector는 레이블이 지정되지 않은 이벤트는 무시합니다. 이 예제 코드는 unlabeledEventsTreatment for를 사용하여 AUTO Amazon Fraud Detector가 레이블이 지정되지 않은 이벤트의 사용 방법을 결정하도록 지정합니다.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.create_model_version (
modelId = 'sample_fraud_detection_model',
modelType = 'ONLINE_FRAUD_INSIGHTS',
trainingDataSource = 'EXTERNAL_EVENTS',
trainingDataSchema = {
    'modelVariables' : ['ip_address', 'email_address'],
    'labelSchema' : {
        'labelMapper' : {
            'FRAUD' : ['fraud'],
            'LEGIT' : ['legit']
```

```
        }
       unlabeledEventsTreatment = 'AUTO'
    }
},
externalEventsDetail = {
    'dataLocation' : 's3://bucket/file.csv',
    'dataAccessRoleArn' : 'role_arn'
}
)
```

요청이 성공하면 상태가 포함된 새 모델 버전이 생성됩니다. TRAINING_IN_PROGRESS 교육 중 언제든지 전화를 걸어 상태를 로 UpdateModelVersionStatus 업데이트하여 교육을 취소할 수 TRAINING_CANCELLED 있습니다. 교육이 완료되면 모델 버전 상태가 로 업데이트됩니다TRAINING_COMPLETE. Amazon Fraud Detector 콘솔을 사용하거나 전화를 걸어 모델 성능을 검토할 수 DescribeModelVersions 있습니다. 모델 점수 및 성능을 해석하는 방법에 대한 자세한 내용은 및 을 참조하십시오모델 점수. 모델 성능 지표

모델 성능을 검토한 후 Detectors가 실시간 사기 예측에 사용할 수 있도록 모델을 활성화하십시오. Amazon Fraud Detector는 자동 크기 조정 기능을 활성화한 상태에서 중복성을 위해 여러 가용 영역에 모델을 배포하여 사기 예측 횟수에 맞게 모델을 확장할 수 있도록 합니다. 모델을 활성화하려면 UpdateModelVersionStatus API를 호출하고 상태를 로 업데이트하십시오. ACTIVE

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.update_model_version_status (
modelId = 'sample_fraud_detection_model',
modelType = 'ONLINE_FRAUD_INSIGHTS',
modelVersionNumber = '1.00',
status = 'ACTIVE'
)
```

모델 점수

Amazon Fraud Detector는 모델 유형별로 모델 점수를 다르게 생성합니다.

계정 탈취 인사이트 (ATI) 모델의 경우 Amazon Fraud Detector는 집계된 값 (원시 변수 세트를 조합하여 계산한 값) 만 사용하여 모델 점수를 생성합니다. 새 개체의 첫 번째 이벤트에 대해 -1점의 점수가 생성되며, 이는 알려지지 않은 위험을 나타냅니다. 이는 새 엔티티의 경우 집계 계산에 사용되는 값이 0 또는 null이기 때문입니다. ATI (Account Takeover Insights) 모델은 동일한 엔티티와 기존 엔티티에 대

해 모든 후속 이벤트에 대해 0에서 1000 사이의 모델 점수를 생성합니다. 여기서 0은 낮은 사기 위험을 나타내고 1000은 높은 사기 위험을 나타냅니다. ATI 모델의 경우 모델 점수는 챌린지 비율 (CR) 과 직접적인 관련이 있습니다. 예를 들어, 500점은 예상 5% 의 챌린지 비율에 해당하는 반면, 900점은 예상 0.1% 의 챌린지 비율에 해당합니다.

온라인 사기 인사이트 (OFI) 및 거래 사기 인사이트 (TFI) 모델의 경우 Amazon Fraud Detector는 집계된 값 (원시 변수 세트를 조합하여 계산한 값) 과 원시 값 (변수에 제공된 값) 을 모두 사용하여 모델 점수를 생성합니다. 모델 점수는 0에서 1000 사이일 수 있으며, 여기서 0은 낮은 사기 위험을 나타내고 1000은 높은 사기 위험을 나타냅니다. OFI 및 TFI 모델의 경우 모델 점수는 거짓양성률 (FPR) 과 직접적인 관련이 있습니다. 예를 들어 600점은 추정 10% 의 오탐지율에 해당하는 반면, 900점은 추정 2% 의 오탐지율에 해당합니다. 다음 표에는 특정 모델 점수가 추정 오탐률과 어떤 상관관계가 있는지에 대한 세부 정보가 나와 있습니다.

모델 점수는	예상 FPR
975	0.50%
950	1%
900	2%
860	3%
775	5%
700	7%
600	10%

모델 성능 지표

모델 교육이 완료되면 Amazon Fraud Detector는 모델 학습에 사용되지 않은 데이터의 15% 를 사용하여 모델 성능을 검증합니다. 학습된 Amazon Fraud Detector 모델은 검증 성능 지표와 유사한 실제 사기 탐지 성능을 제공할 것으로 예상할 수 있습니다.

기업에서는 더 많은 사기를 탐지하는 것과 합법적인 고객을 상대로 마찰을 가중시키는 것 사이에서 균형을 잡아야 합니다. 적절한 균형을 선택하는 데 도움이 되도록 Amazon Fraud Detector는 모델 성능을 평가하는 다음 도구를 제공합니다.

- 점수 분포도 — 모델 점수 분포 히스토그램은 100,000개의 이벤트로 구성된 예제 모집단을 가정합니다. 왼쪽 Y축은 합법적인 이벤트를 나타내고 오른쪽 Y축은 사기 사건을 나타냅니다. 차트 영역을 클릭하여 특정 모델 임계값을 선택할 수 있습니다. 그러면 혼동행렬 및 ROC 차트의 해당 뷰가 업데이트됩니다.
- 혼동 매트릭스 — 모델 예측과 실제 결과를 비교하여 주어진 점수 임계값에 대한 모델 정확도를 요약합니다. Amazon Fraud Detector는 예시 이벤트 모집단이 100,000개라고 가정합니다. 사기 및 합법적인 이벤트가 배포되면 기업의 사기 발생률을 시뮬레이션할 수 있습니다.
  - 진실 — 이 모델은 사기를 예측하는데, 이 모델은 실제로 사기에 해당합니다.
  - 오탐지 (False positive) — 이 모델은 사기를 예측하지만 사기는 실제로 정당합니다.
  - 진정한 단점 — 모델은 합법적인 사건을 예측하고 실제로 사건이 합법적이라고 예측합니다.
  - 거짓 네거티브 — 모델은 합법적인 사건을 예측하지만 실제로는 사기에 해당합니다.
  - 트루 포지티브 비율 (TPR) — 전체 사기 중 모델이 탐지한 비율입니다. 캡처율이라고도 합니다.
  - 거짓 양성률 (FPR) — 사기로 잘못 예측되는 총 합법적 사건의 비율입니다.
- Receiver Operator Curve (ROC) - 가능한 모든 모델 점수 임계값에서 참양성률을 오양성률의 함수로 표시합니다. 고급 지표를 선택하면 이 차트를 볼 수 있습니다.
- 곡선 아래 면적 (AUC) - 가능한 모든 모델 점수 임계값에 대한 TPR 및 FPR을 요약합니다. 예측력이 없는 모형의 AUC는 0.5점인 반면, 완벽한 모형의 점수는 1.0입니다.
- 불확실성 범위 — 모델에서 기대되는 AUC 범위를 보여줍니다. 범위가 클수록 (AUC의 상한과 하한 차이 > 0.1) 모델 불확실성이 높아집니다. 불확실성 범위가 큰 경우 (>0.1), 레이블이 지정된 이벤트를 더 제공하고 모델을 다시 훈련시키는 것을 고려해 보십시오.

모델 성능 메트릭을 사용하려면

1. 먼저 점수 분포 차트로 시작하여 사기 및 합법적인 사건에 대한 모델 점수 분포를 검토하십시오. 이상적으로는 사기 행위와 합법적인 사건을 명확하게 구분할 수 있어야 합니다. 이는 모델이 어떤 이벤트가 사기이고 어떤 이벤트가 합법적인지 정확하게 식별할 수 있음을 나타냅니다. 차트 영역을 클릭하여 모델 임계값을 선택합니다. 모델 점수 임계값 조정이 참양성률과 거짓양성률에 어떤 영향을 미치는지 확인할 수 있습니다.

   > **Note**
   >
   > 점수 분포 차트는 사기 사건과 합법적인 사건을 서로 다른 두 Y축에 표시합니다. 왼쪽 Y축은 합법적인 이벤트를 나타내고 오른쪽 Y축은 사기 사건을 나타냅니다.

2. 혼란 매트릭스를 검토하십시오. 선택한 모델 점수 임계값에 따라 100,000개의 이벤트 샘플을 기반으로 시뮬레이션된 영향을 확인할 수 있습니다. 사기 및 합법적인 이벤트의 분포를 보면 비즈니스의 사기 발생률을 시뮬레이션할 수 있습니다. 이 정보를 사용하여 참양성률과 거짓양성률 사이의 적절한 균형을 찾아보십시오.
3. 자세한 내용을 보려면 고급 지표를 선택하십시오. ROC 차트를 사용하여 모든 모델 점수 임계값에 대한 참양성률과 거짓양성률 간의 관계를 파악할 수 있습니다. ROC 곡선은 참양성률과 거짓양성률 간의 균형을 세밀하게 조정하는 데 도움이 될 수 있습니다.

> **Note**
>
> 표를 선택하여 표 형식의 지표를 검토할 수도 있습니다.
> 테이블 뷰에는 지표 정밀도도 표시됩니다. 정확도는 사기로 예측된 모든 이벤트와 비교하여 사기로 올바르게 예측된 사기 이벤트의 비율입니다.

4. 성능 지표를 사용하여 목표 및 사기 탐지 사용 사례를 기반으로 비즈니스에 가장 적합한 모델 임계값을 결정하십시오. 예를 들어 모델을 사용하여 신규 계정 등록을 위험도가 높음, 중간 또는 낮음으로 분류하려는 경우 다음과 같이 세 가지 규칙 조건의 초안을 작성할 수 있도록 두 임계값 점수를 식별해야 합니다.

   - 점수 > X는 위험도가 높습니다.
   - 점수 < X but > Y는 중간 위험도입니다.
   - 점수가 Y 미만이면 위험도가 낮습니다.

모델 변수 중요도

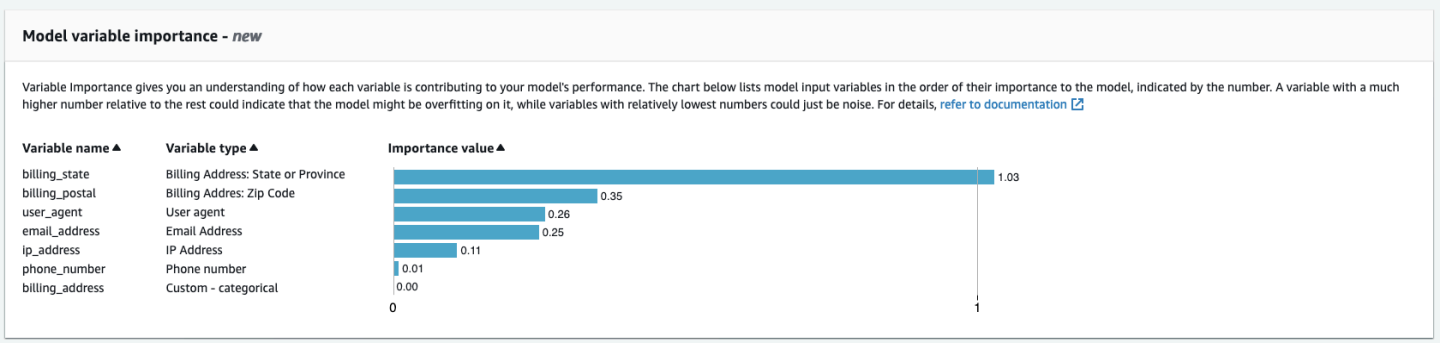
모델 변수 중요도는 모델 버전 내에서 모델 변수의 순위를 매기는 Amazon Fraud Detector의 기능입니다. 각 모델 변수에는 모델의 전체 성능에 대한 상대적 중요도를 기반으로 한 값이 제공됩니다. 값이 가장 높은 모델 변수는 해당 모델 버전의 데이터셋에 있는 다른 모델 변수보다 모델에 더 중요하며 기본적으로 상단에 나열됩니다. 마찬가지로, 값이 가장 낮은 모델 변수는 기본적으로 맨 아래에 나열되며 다른 모델 변수에 비해 중요도가 가장 낮습니다. 모델 변수 중요도 값을 사용하면 어떤 입력이 모델 성능을 좌우하는지 파악할 수 있습니다.

Amazon Fraud Detector 콘솔에서 또는 DescribeModelVersionAPI를 사용하여 학습된 모델 버전의 모델 변수 중요도 값을 볼 수 있습니다.

모델 변수 중요도는 모델 버전을 학습하는 데 사용되는 각 변수에 대해 다음과 같은 값 세트를 제공합니다.

- 변수 유형: 변수 유형 (예: IP 주소 또는 이메일). 자세히 알아보려면 변수 유형의 내용을 참조하세요. 계정 인수 인사이트 (ATI) 모델의 경우 Amazon Fraud Detector는 원시 및 집계 변수 유형 모두에 변수 중요도 값을 제공합니다. 원시 변수 유형은 사용자가 제공하는 변수에 할당됩니다. 집계 변수 유형은 Amazon Fraud Detector가 집계된 중요도 값을 계산하기 위해 결합한 원시 변수 세트에 할당됩니다.
- 변수 이름: 모델 버전을 학습시키는 데 사용된 이벤트 변수의 이름 (예:, `ip_addressemail_address,are_creadentials_valid`). 집계 변수 유형의 경우 집계된 변수 중요도 값을 계산하는 데 사용된 모든 변수의 이름이 나열됩니다.
- 변수 중요도 값: 모델 성능에 대한 원시 또는 집계 변수의 상대적 중요도를 나타내는 숫자입니다. 일반적인 범위: 0~10

Amazon Fraud Detector 콘솔에서는 온라인 사기 인사이트 (OFI) 또는 거래 사기 인사이트 (TFI) 모델에 대한 모델 변수 중요도 값이 다음과 같이 표시됩니다. ATI (계정 탈취 인사이트) 모델은 원시 변수의 중요도 값 외에도 집계된 변수 중요도 값을 제공합니다. 시각적 차트를 사용하면 가장 순위가 높은 변수의 중요도 값을 참조할 수 있는 세로 점선을 통해 변수 간의 상대적 중요도를 쉽게 확인할 수 있습니다.



Amazon Fraud Detector는 추가 비용 없이 모든 Fraud Detector 모델 버전에 대해 가변 중요도 값을 생성합니다.

> **Important**
>
> 2021년 7월 9일 이전에 생성된 모델 버전에는 가변 중요도 값이 없습니다. 새 버전의 모델을 학습시켜 모델 변수 중요도 값을 생성해야 합니다.

모델 변수 중요도 값 사용

모델 변수 중요도 값을 사용하여 모델 성능을 높이거나 낮추는 요인과 가장 큰 기여를 하는 변수를 파악할 수 있습니다. 그런 다음 모델을 수정하여 전반적인 성능을 개선하세요.

좀 더 구체적으로 설명하자면, 모델 성능을 개선하려면 변수 중요도 값을 도메인 지식과 비교하여 검토하고 학습 데이터에서 문제를 디버깅하십시오. 예를 들어, 계정 ID가 모델에 대한 입력으로 사용되었고 모델이 맨 위에 나열되어 있다면 변수 중요도 값을 살펴보세요. 변수 중요도 값이 나머지 값보다 훨씬 높으면 모델이 특정 사기 패턴에 과도하게 적합할 수 있습니다 (예: 모든 사기 이벤트가 동일한 계정 ID에서 발생한 경우). 그러나 변수가 사기 레이블에 따라 달라지는 경우 레이블 유출이 발생할 수도 있습니다. 도메인 지식을 기반으로 한 분석 결과에 따라 변수를 제거하고 더 다양한 데이터셋으로 훈련시키거나 모델을 그대로 유지하는 것이 좋습니다.

마찬가지로, 가장 늦게 순위가 매겨진 변수를 살펴보세요. 변수 중요도 값이 나머지 값보다 현저히 낮으면 이 모델 변수는 모델 훈련에 그다지 중요하지 않을 수 있습니다. 변수를 제거하여 더 간단한 모델 버전을 학습시키는 것을 고려해 볼 수 있습니다. 모델에 변수가 거의 없는 경우 (예: 변수가 두 개뿐인 경우) Amazon Fraud Detector는 여전히 변수 중요도 값을 제공하고 변수의 순위를 지정합니다. 하지만 이 경우 통찰력이 제한될 수 있습니다.

> **Important**
>
> 1. 모델 변수 중요도 차트에서 누락된 변수를 발견했다면 다음 이유 중 하나 때문일 수 있습니다. 데이터셋의 변수를 수정하고 모델을 다시 훈련해 보세요.
>   - 훈련 데이터셋에 있는 변수의 고유 값 개수는 100개 미만입니다.
>   - 훈련 데이터 세트에서 누락된 변수 값이 0.9% 를 넘습니다.
> 2. 모델의 입력 변수를 조정할 때마다 새 모델 버전을 학습시켜야 합니다.

모델 변수 중요도 값 평가

모델 변수 중요도 값을 평가할 때는 다음 사항을 고려하는 것이 좋습니다.

- 변수 중요도 값은 항상 도메인 지식과 조합하여 평가해야 합니다.
- 모델 버전 내 다른 변수의 변수 중요도 값과 비교하여 변수의 변수 중요도 값을 검토하십시오. 단일 변수에 대한 변수 중요도 값을 독립적으로 고려하지 마십시오.
- 동일한 모델 버전 내 변수의 변수 중요도 값을 비교하십시오. 모델 버전에 있는 변수의 변수 중요도 값이 다른 모델 버전의 동일한 변수 값과 다를 수 있으므로 모델 버전 간에 동일한 변수의 변수 중요

도 값을 비교하지 마십시오. 동일한 변수와 데이터셋을 사용하여 서로 다른 모델 버전을 학습시키는 경우 반드시 동일한 변수 중요도 값이 생성되는 것은 아닙니다.

모델 변수 중요도 순위 보기

모델 교육이 완료되면 Amazon Fraud Detector 콘솔에서 또는 DescribeModelVersionAPI를 사용하여 학습된 모델 버전의 모델 변수 중요도 순위를 확인할 수 있습니다.

콘솔을 사용하여 모델 변수 중요도 순위를 보려면

1. AWS콘솔을 열고 계정에 로그인합니다. Amazon Fraud Detector로 이동합니다.
2. 왼쪽 탐색 창에서 모델을 선택합니다.
3. 모델을 선택한 다음 모델 버전을 선택합니다.
4. 개요 탭이 선택되어 있는지 확인하십시오.
5. 아래로 스크롤하여 모델 변수 중요도 창을 확인합니다.

모델 변수 중요도 값 계산 방법 이해

Amazon Fraud Detector는 각 모델 버전 교육을 완료하면 모델 변수 중요도 값과 모델의 성능 지표를 자동으로 생성합니다. 이를 위해 Amazon Fraud Detector는 섀플리 부가적 설명 (SHAP) 을 사용합니다. SHAP는 기본적으로 모든 모델 변수의 가능한 모든 조합을 고려한 후의 모델 변수의 평균 예상 기여도입니다.

SHAP는 먼저 이벤트 예측을 위해 각 모델 변수의 기여도를 할당합니다. 그런 다음 이러한 예측을 집계하여 모델 수준에서 변수 순위를 생성합니다. 각 모델 변수의 예측 기여도를 지정하기 위해 SHAP는 가능한 모든 변수 조합 간의 모델 출력 차이를 고려합니다. SHAP는 특정 변수 세트를 포함하거나 제거하여 모델 결과를 생성할 수 있는 모든 가능성을 포함함으로써 각 모델 변수의 중요도에 정확하게 접근할 수 있습니다. 이는 모델 변수가 서로 높은 상관관계를 가질 때 특히 중요합니다.

대부분의 경우 ML 모델에서는 변수를 제거할 수 없습니다. 대신 모델에서 제거되거나 누락된 변수를 하나 이상의 기준선에 있는 해당 변수 값으로 바꿀 수 있습니다 (예: 사기 행위가 아닌 이벤트). 적절한 기준 인스턴스를 선택하는 것은 어려울 수 있지만 Amazon Fraud Detector에서는 이 기준을 인구 평균으로 설정하여 쉽게 선택할 수 있습니다.

모델 가져오기 SageMaker

선택적으로 SageMaker 호스팅된 모델을 Amazon Fraud Detector로 가져올 수 있습니다. SageMaker 모델과 마찬가지로 API를 사용하여 탐지기에 모델을 추가하고 사기 예측을 생성할 수 있습니다. `GetEventPrediction` `GetEventPrediction`요청의 일부로 Amazon Fraud Detector는 SageMaker 엔드포인트를 호출하고 결과를 규칙에 전달합니다.

`GetEventPrediction`요청의 일부로 전송된 이벤트 변수를 사용하도록 Amazon Fraud Detector를 구성할 수 있습니다. 이벤트 변수를 사용하기로 선택한 경우 입력 템플릿을 제공해야 합니다. Amazon Fraud Detector는 이 템플릿을 사용하여 이벤트 변수를 엔드포인트를 호출하는 데 필요한 입력 페이로드로 변환합니다 SageMaker . 또는 요청의 일부로 전송되는 ByteBuffer를 사용하도록 SageMaker 모델을 구성할 수 있습니다. `GetEventPrediction`

Amazon Fraud Detector는 JSON 또는 CSV 입력 형식과 JSON 또는 CSV 출력 형식을 사용하는 가져오기 SageMaker 알고리즘을 지원합니다. 지원되는 SageMaker 알고리즘의 예로는 XGBoost, 선형 학습기, 랜덤 컷 포레스트 등이 있습니다.

를 SageMaker 사용하여 모델 가져오기 AWS SDK for Python (Boto3)

SageMaker 모델을 가져오려면 `PutExternalModel` API를 사용하세요. 다음 예제에서는 SageMaker `sagemaker-transaction-model` 엔드포인트가 배포되었고, `InService` 상태이며, XGBoost 알고리즘을 사용한다고 가정합니다.

입력 구성은 이벤트 변수를 사용하여 모델 입력을 구성하도록 지정합니다 (`useEventVariables`로 설정). `TRUE` 입력 형식은 TEXT_CSV입니다. XGBoost에는 CSV 입력이 필요하다는 점을 감안하면 입력 형식은 TEXT_CSV입니다. 는 csvInputTemplate 요청의 일부로 전송된 변수에서 CSV 입력을 구성하는 방법을 지정합니다. `GetEventPrediction` 이 예제에서는 변수 `order_amt`prev_amt, `hist_amt` 및 를 만들었다고 가정합니다. `payment_type`

출력 구성은 SageMaker 모델의 응답 형식을 지정하고 적절한 CSV 인덱스를 Amazon Fraud Detector 변수에 `sagemaker_output_score` 매핑합니다. 구성한 후에는 규칙에서 출력 변수를 사용할 수 있습니다.

> **Note**
>
> SageMaker 모델의 출력은 소스가 `EXTERNAL_MODEL_SCORE` 있는 변수에 매핑되어야 합니다. 콘솔에서는 변수를 사용하여 이러한 변수를 만들 수 없습니다. 모델 가져오기를 구성할 때 대신 변수를 생성해야 합니다.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.put_external_model (
modelSource = 'SAGEMAKER',
modelEndpoint = 'sagemaker-transaction-model',
invokeModelEndpointRoleArn = 'your_SagemakerExecutionRole_arn',
inputConfiguration = {
    'useEventVariables' : True,
    'eventTypeName' : 'sample_transaction',
    'format' : 'TEXT_CSV',
    'csvInputTemplate' : '{{order_amt}}, {{prev_amt}}, {{hist_amt}}, {{payment_type}}'
},

outputConfiguration = {
    'format' : 'TEXT_CSV',
    'csvIndexToVariableMap' : {
        '0' : 'sagemaker_output_score'
    }
},

modelEndpointStatus = 'ASSOCIATED'
)
```

모델 버전 또는 모델 버전 삭제

탐지기 버전과 연결되지 않은 경우, 모형과 모형 버전을 Amazon Fraud Detector에서 삭제할 수 있습니다. 모델을 삭제하면 Amazon Fraud Detector는 해당 모델을 영구적으로 삭제하며 데이터는 더 이상 Amazon Fraud Detector에 저장되지 않습니다.

탐지기 버전과 연결되지 않은 경우 삭제할 수 있습니다. SageMaker SageMaker 모델을 제거하면 Amazon Fraud Detector와의 연결이 끊어지지만 모델은 에서 계속 사용할 수 있습니다 SageMaker.

모델 버전 삭제하기

`Ready to deploy`상태에 있는 모델 버전만 삭제할 수 있습니다. 모델 버전을 에서`Ready to deploy` 상태로`ACTIVE` 변경하려면 모델 버전을 배포 취소하십시오.

1. 에 로그인한 후 https://console.aws.amazon.com/frauddetector 에서 Amazon Fraud Detector에서 엽니다.AWS Management Console
2. Amazon Fraud Detector에서 선택합니다.

3. 선택합니다.
4. 선택합니다.
5. [Actions]를 선택한 후 [Delete]를 선택합니다.
6. 모델 버전 이름을 입력한 다음 모델 버전 삭제를 선택합니다.

모델 버전을 선택합니다.

모든 탐지기 버전 (ACTIVE,INACTIVE,DRAFT) 에서 사용 중인 모델 버전은 배포를 취소할 수 없습니다. 따라서 탐지기 버전에서 사용 중인 모델 버전을 배포를 취소하려면 먼저 탐지기 버전에서 모델 버전을 제거하십시오.

1. Amazon Fraud Detector에서 선택합니다.
2. 배포를 취소하려는 모델 버전이 포함된 모델을 선택합니다.
3. 선택합니다.
4. 작업을 선택한 다음 모델 버전 배포 해제를 선택합니다.

모형을 삭제하기

모형을 삭제하기 전에

1. Amazon Fraud Detector에서 선택합니다.
2. 선택합니다.
3. [Actions]를 선택한 후 [Delete]를 선택합니다.
4. 모델 이름을 입력한 다음 모델 삭제를 선택합니다.

아마존 SageMaker 모델을 제거하려면

1. Amazon Fraud Detector에서 선택합니다.
2. SageMaker 선택합니다.
3. 동작을 선택한 다음 모델 제거를 선택합니다.
4. 모델 이름을 입력한 다음 SageMaker모델 제거를 선택합니다.

감지기

탐지기는 사기 여부를 평가하려는 특정 비즈니스 이벤트에 대한 부정 행위 탐지 로직 (예: 모델 및 규칙) 을 포함하는 컨테이너입니다. 먼저 이미 정의한 이벤트를 지정하여 탐지기를 생성하고 필요에 따라 Amazon Fraud Detector에서 이미 생성하고 해당 이벤트에 대해 학습시킨 모델 버전을 추가할 수 있습니다.

그런 다음 탐지기에 규칙 및 규칙 실행 순서를 추가하여 탐지기 버전을 생성합니다. 탐지기 버전은 규칙을 정의하며 선택적으로 사기 예측 생성 요청의 일부로 실행할 모델을 정의합니다. 검출기 내에 정의된 모든 규칙을 검출기 버전에 추가할 수 있습니다. 또한 평가된 이벤트 유형에 대해 학습된 모든 모델을 감지기 버전에 추가할 수 있습니다. 탐지기에는 여러 버전이 있을 수 있으며, 각 버전에는 여러 사용 사례를 충족하기 위해 서로 다른 규칙과 규칙 실행 순서가 있습니다.

각 감지기 버전의 상태는 다음과 같아야 합니다.`DRAFT`,`ACTIVE`, 또는`INACTIVE`. 하나의 검출기 버전만 포함될 수 있습니다.`ACTIVE`한 번에 상태. Amazon 사기 탐지기는 다음과 같은 탐지기 버전을 사용합니다.`ACTIVE`사기 예측을 위한 상태

감지기 만들기

이미 정의한 이벤트 유형을 지정하여 감지기를 만들 수 있습니다. Amazon Fraud Detector에서 이미 학습 및 배포한 모델을 선택적으로 추가할 수 있습니다. 모델을 추가하면 규칙을 생성할 때 Amazon Fraud Detector에서 생성한 모델 점수를 규칙 표현식에 사용할 수 있습니다 (예:`$model score < 90`).

Amazon 사기 탐지기 콘솔에서 다음을 사용하여 탐지기를 만들 수 있습니다.PutDetectorAPI, 사용풋 디텍터명령 또는 사용AWSSDK. API, 명령 또는 SDK를 사용하여 탐지기를 만드는 경우 탐지기를 만든 후 지침을 따르십시오.검출기 버전 생성.

Amazon 사기 탐지기 콘솔에서 탐지기 생성

이 예제에서는 이벤트 유형을 만들었으며 사기 예측에 사용할 모델 버전도 만들고 배포했다고 가정합니다.

1단계: 검출기 구축

1. Amazon 사기 탐지기 콘솔의 왼쪽 탐색 창에서 다음을 선택합니다.감지기.
2. 선택해 주세요감지기 생성.

3. 에서검출기 세부 정보 정의페이지, 입력`sample_detector`검출기 이름을 입력합니다. 필요에 따라 다음과 같이 검출기에 대한 설명을 입력합니다.`my sample fraud detector`.
4. ... 에 대한이벤트 유형에서 사기 예측을 위해 생성한 이벤트 유형을 선택합니다.
5. 다음을 선택합니다.

2단계: 배포된 모델 버전 추가

1. 이 단계는 선택 사항이라는 점을 참고하세요. 검출기에 모델을 추가할 필요는 없습니다. 이 단계를 건너뛰려면 다음(Next)을 선택합니다.
2. 에서모델 추가 - 선택 사항, 선택모델 추가.
3. 에서모델 추가페이지, 대상모델 선택에서 이전에 배포한 Amazon 사기 탐지기 모델 이름을 선택합니다. ... 에 대한버전 선택에서 배포된 모델의 모델 버전을 선택합니다.
4. 모델 추가를 선택합니다.
5. 다음을 선택합니다.

3단계: 규칙 추가

규칙은 Amazon Fraud Detector에 부정 행위 예측을 평가할 때 변수 값을 해석하는 방법을 알려주는 조건입니다. 이 예제에서는 모델 점수를 변수 값으로 사용하여 세 가지 규칙을 생성합니다.`high_fraud_risk`,`medium_fraud_risk`, 및`low_fraud_risk`. 자체 규칙, 규칙 표현식, 규칙 실행 순서 및 결과를 만들려면 모델 및 사용 사례에 적합한 값을 사용하십시오.

1. 에서규칙 추가페이지, 아래규칙 정의, 입력`high_fraud_risk`규칙 이름 및 아래설명 - 선택 사항, 입력**`This rule captures events with a high ML model score`**규칙에 대한 설명으로
2. 에서익스프레션Amazon Fraud Detector의 단순화된 규칙 표현 언어를 사용하여 다음 규칙 표현식을 입력하십시오.

   ```
   $sample_fraud_detection_model_insightscore > 900
   ```
3. 에서성과, 선택새 결과 만들기. 결과는 사기 예측의 결과이며 평가 중에 규칙이 일치하면 결과가 반환됩니다.
4. 에서새 결과 만들기, 입력`verify_customer`결과 이름으로. 설명을 입력할 수도 있습니다.
5. 선택해 주세요결과 저장.
6. 선택해 주세요규칙 추가규칙 유효성 검사기를 실행하고 규칙을 저장합니다. Amazon Fraud Detector가 생성되면 해당 규칙을 탐지기에서 사용할 수 있게 됩니다.

7. 선택해 주세요다른 규칙 추가를 선택한 다음규칙 생성탭.
8. 이 과정을 두 번 더 반복하여 다음을 만드세요medium_fraud_risk과low_fraud_risk다음 규칙 세부 정보를 사용하는 규칙:

   - 중간_사기_위험

     규칙 이름:medium_fraud_risk

     결과:review

     표현식:

     ```
     $sample_fraud_detection_model_insightscore <= 900 and

     $sample_fraud_detection_model_insightscore > 700
     ```

   - 낮은 사기 위험

     규칙 이름:low_fraud_risk

     결과:approve

     표현식:

     ```
     $sample_fraud_detection_model_insightscore <= 700
     ```

9. 사용 사례에 대한 모든 규칙을 생성한 후 다음을 선택하십시오.다음.

   규칙 작성 및 작성에 대한 자세한 내용은 을 참조하십시오.규칙과규칙 언어 참조.

4단계: 규칙 실행 및 규칙 순서 구성

탐지기에 포함된 규칙의 규칙 실행 모드에 따라 정의한 모든 규칙이 평가되는지 또는 일치하는 첫 번째 규칙에서 규칙 평가가 중지되는지가 결정됩니다. 그리고 규칙 순서에 따라 규칙을 실행할 순서가 결정됩니다.

기본 규칙 실행 모드는FIRST_MATCHED.

첫 번째 매칭

첫 번째 일치 규칙 실행 모드는 정의된 규칙 순서에 따라 첫 번째 일치 규칙에 대한 결과를 반환합니다. FIRST_MATCHED를 지정하면 Amazon Fraud Detector는 처음부터 마지막까지 순차적으로 규

칙을 평가하고 처음 일치하는 규칙에서 중지합니다. 그러면 Amazon Fraud Detector가 해당 단일 규칙에 대한 결과를 제공합니다.

규칙을 실행하는 순서가 부정 행위 예측 결과에 영향을 미칠 수 있습니다. 규칙을 생성한 후 다음 단계에 따라 규칙을 다시 정렬하여 원하는 순서로 규칙을 실행합니다.

만약 당신의high_fraud_risk규칙이 이미 규칙 목록의 맨 위에 있지 않습니다. 선택하세요주문, 그런 다음 선택1. 이것은 움직입니다high_fraud_risk첫 번째 위치로.

이 과정을 반복하면medium_fraud_risk규칙은 두 번째 위치에 있고 당신의low_fraud_risk규칙은 세 번째 위치에 있습니다.

모두 일치함

일치하는 모든 규칙 실행 모드는 규칙 순서에 관계없이 일치하는 모든 규칙에 대한 결과를 반환합니다. 지정하는 경우ALL_MATCHED, Amazon Fraud Detector는 모든 규칙을 평가하여 일치하는 모든 규칙에 대한 결과를 반환합니다.

선택FIRST_MATCHED이 튜토리얼을 보시고 다음을 선택하십시오다음.

5단계: 검출기 버전 검토 및 생성

탐지기 버전은 사기 예측을 생성하는 데 사용되는 특정 모델 및 규칙을 정의합니다.

1. 에서검토 및 생성페이지에서 구성한 감지기 세부 정보, 모델 및 규칙을 검토하십시오. 변경이 필요한 경우 다음을 선택하십시오.편집해당 섹션 옆에 있습니다.
2. 선택해 주세요감지기 생성. 생성되면 검출기의 첫 번째 버전이 감지기 버전 표에 다음과 같이 표시됩니다.Draft상태.

   당신은 다음을 사용합니다드래프트검출기를 테스트하기 위한 버전.

를 사용하여 검출기 만들기AWS SDK for Python (Boto3)

다음 예제는 에 대한 샘플 요청을 보여줍니다.PutDetectorAPI. 검출기는 검출기 버전의 컨테이너 역할을 합니다. 더PutDetectorAPI는 감지기가 평가할 이벤트 유형을 지정합니다. 다음 예제에서는 이벤트 유형을 생성했다고 가정합니다.sample_registration.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.put_detector (
```

```
detectorId = 'sample_detector',
eventTypeName = 'sample_registration'
)
```

검출기 버전 생성

탐지기 버전은 규칙, 규칙 실행 순서 및 선택적으로 사기 예측 생성 요청의 일부로 사용될 모델 버전을 정의합니다. 검출기 내에 정의된 모든 규칙을 검출기 버전에 추가할 수 있습니다. 또한 평가된 이벤트 유형에 대해 학습된 모든 모델을 추가할 수 있습니다.

각 감지기 버전의 상태는 다음과 같습니다.DRAFT,ACTIVE, 또는INACTIVE. 하나의 검출기 버전만 포함될 수 있습니다.ACTIVE한 번에 상태. ... 동안GetEventPrediction요청, Amazon 사기 탐지기는 다음을 사용합니다.ACTIVE검출기 (아니오)DetectorVersion지정되어 있습니다.

규칙 실행 모드

Amazon 사기 탐지기는 두 가지 규칙 실행 모드를 지원합니다.FIRST_MATCHED과ALL_MATCHED.

- 규칙 실행 모드가 다음과 같은 경우FIRST_MATCHED, Amazon Fraud Detector는 규칙을 처음부터 마지막까지 순차적으로 평가하여 일치하는 첫 번째 규칙부터 중지합니다. 그러면 Amazon Fraud Detector가 해당 단일 규칙에 대한 결과를 제공합니다. 규칙이 false (일치하지 않음) 로 평가되면 목록의 다음 규칙이 평가됩니다.
- 규칙 실행 모드가 다음과 같은 경우ALL_MATCHED그러면 평가의 모든 규칙이 순서에 관계없이 병렬로 실행됩니다. Amazon Fraud Detector는 모든 규칙을 실행하고 일치하는 모든 규칙에 대해 정의된 결과를 반환합니다.

다음을 사용하여 검출기 버전을 생성하십시오.AWS SDK for Python (Boto3)

다음 예제는 에 대한 샘플 요청을 보여줍니다.CreateDetectorVersionAPI. 규칙 실행 모드는 다음과 같이 설정됩니다.FIRST_MATCHED따라서 Amazon Fraud Detector는 규칙을 처음부터 마지막까지 순차적으로 평가하여 일치하는 첫 번째 규칙부터 중단합니다. 그런 다음 Amazon Fraud Detector는 다음 기간 동안 해당 단일 규칙에 대한 결과를 제공합니다.GetEventPrediction response.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.create_detector_version(
detectorId = 'sample_detector',
rules = [{
```

```
    'detectorId' : 'sample_detector',
    'ruleId' : 'high_fraud_risk',
    'ruleVersion' : '1'
},
{
    'detectorId' : 'sample_detector',
    'ruleId' : 'medium_fraud_risk',
    'ruleVersion' : '1'
},
{
    'detectorId' : 'sample_detector',
    'ruleId' : 'low_fraud_risk',
    'ruleVersion' : '1'
}
],
modelVersions = [{
    'modelId' : 'sample_fraud_detection_model',
    'modelType': 'ONLINE_FRAUD_INSIGHTS',
    'modelVersionNumber' : '1.00'
}],
ruleExecutionMode = 'FIRST_MATCHED'
)
```

검출기 버전의 상태를 업데이트하려면 다음을 사용하십시오.UpdateDetectorVersionStatusAPI. 다음 예제에서는 에서 감지기 버전 상태를 업데이트합니다.DRAFT에ACTIVE. 동안GetEventPrediction요청, 탐지기 ID가 지정되지 않은 경우 Amazon Fraud Detector는 다음을 사용합니다.ACTIVE검출기 버전.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.update_detector_version_status(
detectorId = 'sample_detector',
detectorVersionId = '1',
status = 'ACTIVE'
)
```

탐지기, 탐지기 버전 또는 규칙 버전 삭제

Amazon Fraud Fraud Detector 삭제하기 전에 먼저 감지기와 연관된 모든 감지기 버전 및 규칙 버전을 삭제해야 합니다.

탐지기, 탐지기 버전 또는 규칙 버전을 삭제하면 Amazon Fraud Detector는 해당 리소스를 영구적으로 삭제하며 데이터는 더 이상 Amazon Fraud Detector에 저장되지 않습니다.

감지기 버전을 삭제하려면

DRAFT또는INACTIVE 상태인 감지기 버전은 삭제할 수 있습니다.

1. 에 로그인한 후 https://console.aws.amazon.com/frauddetector 에서 Amazon Fraud Fraud Detector 콘솔을 엽니다.AWS Management Console
2. Amazon Fraud Detector 콘솔의 왼쪽 탐색 창에서 탐지기를 선택합니다.
3. 삭제하려는 탐지기 버전이 포함된 탐지기를 선택합니다.
4. 삭제하려는 감지기 버전을 선택합니다.
5. [Actions]를 선택한 후 [Delete]를 선택합니다.
6. **delete**를 입력한 다음 탐지기 삭제를 선택합니다.

규칙 버전을 삭제하려면

어떤ACTIVEINACTIVE 감지기 버전에서도 사용되지 않는 규칙 버전만 규칙 버전을 삭제할 수 있습니다. 필요한 경우 규칙 버전을 삭제하기 전에 먼저ACTIVE 탐지기 버전을 로INACTIVE 이동한 다음INACTIVE 탐지기 버전을 삭제하십시오.

1. Amazon Fraud Detector 콘솔의 왼쪽 탐색 창에서 탐지기를 선택합니다.
2. 삭제하려는 규칙 버전을 포함하는 탐지기를 선택합니다.
3. 연관된 규칙 탭을 선택하고 삭제할 규칙을 선택합니다.
4. 삭제하려는 규칙 버전을 선택합니다.
5. 작업을 선택한 다음 규칙 버전 삭제를 선택합니다.
6. 입력한**delete** 다음 버전 삭제를 선택합니다.

검출기를 삭제하려면

감지기를 삭제하기 전에 먼저 감지기와 연관된 모든 감지기 버전 및 규칙 버전을 삭제해야 합니다.

1. Amazon Fraud Detector 콘솔의 왼쪽 탐색 창에서 탐지기를 선택합니다.
2. 삭제하려는 감지기를 선택합니다.
3. 동작을 선택한 다음 탐지기 삭제를 선택합니다.
4. **delete**를 입력한 다음 탐지기 삭제를 선택합니다.

리소스

모델, 규칙 및 탐지기는 변수, 결과, 레이블, 목록 및 엔티티와 같은 리소스를 사용하여 이벤트의 사기 위험을 평가합니다. 이 섹션에서는 리소스 만들기 및 관리에 대한 정보를 제공합니다.

주제

Variables

변수는 사기 예측에 사용하려는 데이터 요소를 나타냅니다. 이러한 변수는 모델 학습을 위해 준비한 이벤트 데이터 세트, Amazon Fraud Detector 모델의 위험 점수 출력 또는 Amazon 모델에서 가져올 수 있습니다. SageMaker 이벤트 데이터셋에서 가져온 변수에 대한 자세한 내용은 을 참조하십시오데이터 모델 탐색기를 사용하여 이벤트 데이터세트 요구 사항 가져오기.

사기 예측에 사용할 변수를 먼저 생성한 다음 이벤트 유형을 생성할 때 이벤트에 추가해야 합니다. 생성하는 각 변수에는 데이터 유형, 기본값 및 선택적으로 변수 유형을 할당해야 합니다. Amazon Fraud Detector는 IP 주소, 은행 식별 번호 (BIN), 전화 번호 등 사용자가 제공하는 일부 변수를 강화하여 추가 입력을 생성하고 이러한 변수를 사용하는 모델의 성능을 향상시킵니다.

데이터 유형

변수에는 해당 변수가 나타내는 데이터 요소의 데이터 유형이 있어야 하며 선택적으로 사전 변수 유형 정의된 데이터 유형 중 하나를 할당할 수 있습니다. 변수 유형에 할당된 변수의 경우 데이터 유형이 미리 선택됩니다. 가능한 데이터 유형에는 다음 유형이 포함됩니다.

데이터 형식	설명	기본값	예제 값
문자열	문자, 정수 또는 둘 다의 모든 조합	<empty>	ABC, 123, 1D3B
Integer	양수 또는 음수 정수	0	1, -1
부울	참 또는 거짓	False	참, 거짓
DateTime	ISO 8601 표준 UTC 형식으로만 지정된 날짜 및 시간	<empty>	2019-11-30T 13:01:01 Z
Float	소수점이 있는 숫자	0.0	4.01, 0.10

기본값

변수에는 기본값이 있어야 합니다. Amazon Fraud Detector가 부정 행위 예측을 생성할 때 Amazon Fraud Detector가 변수에 대한 값을 받지 못하는 경우 규칙 또는 모델을 실행하는 데 이 기본값이 사용됩니다. 입력한 기본값은 선택한 데이터 유형과 일치해야 합니다. Amazon Fraud Detector는 AWS 콘솔에서 정수, 부울, 부동 소수점, 문자열의 `false` 경우 (비어 있음) 의 `0` 기본값을 할당합니다. `0.0` 이러한 모든 데이터 유형에 대해 사용자 지정 기본값을 설정할 수 있습니다.

변수 유형

변수를 만들 때 선택적으로 변수를 변수 유형에 할당할 수 있습니다. 변수 유형은 모델을 학습시키고 사기 예측을 생성하는 데 사용되는 일반적인 데이터 요소를 나타냅니다. 관련 변수 유형이 있는 변수만 모델 학습에 사용할 수 있습니다. 모델 학습 프로세스의 일부로 Amazon Fraud Detector는 변수와 연결된 변수 유형을 사용하여 변수 강화, 기능 엔지니어링 및 위험 점수를 매깁니다.

Amazon Fraud Detector는 변수에 할당하는 데 사용할 수 있는 다음과 같은 변수 유형을 미리 정의했습니다.

범주	변수 유형	설명	데이터 형식	예
세션	IP_ADDRESS	이벤트 기간 동안 수집된 IP 주소	문자열	192.0.2.0 참고: Amazon 사기 탐지기는 이 데이터를 강화합니다. 자세한 내용은 지리적

범주	변수 유형	설명	데이터 형식	예
				위치 강화 섹션을 참조하세요.
	사용자 에이전트	이벤트 기간 동안 수집되는 사용자 에이전트	문자열	모질라 5.0 (윈도우 NT 10.0, 윈64, x64, rv:68.0) 게코 20100101

범주	변수 유형	설명	데이터 형식	예
	지문	이벤트에 사용된 장치의 고유 식별자	문자열	sadfow987u234
	SESSION_ID	이벤트의 활성 세션의 세션 ID	문자열	sid123456789
	유효한_자격 증명_인증서_유효	이벤트 로그인에 사용된 자격 증명이 유효한지 여부를 나타냅니다.	부울	True
사용자	이메일_주소	이벤트 기간 동안 수집된 이메일 주소	문자열	abc@domain.com

범주	변수 유형	설명	데이터 형식	예
	PHONE_NUMBER	이벤트 기간 동안 수집된 전화번호	문자열	+1 555-0100 참고: Amazon 사기 탐지기는 이 데이터를 강화합니다. 자세한 내용은 전화번

범주	변수 유형	설명	데이터 형식	예
				호강화 섹션을 참조하세요.
결제	청구_이름	청구서 수신 주소와 연결된 이름	문자열	John Doe

범주	변수 유형	설명	데이터 형식	예
	청구_전화	청구서 수신 주소와 연결된 전화번호	문자열	+1 555-0100 참고: Amazon 사기 탐지기는 이 데이터를 강화합니다. 자세한 내용은 전화번

범주	변수 유형	설명	데이터 형식	예
				호강화 섹션을 참조하세요.
	청구_주소_L1	청구서 수신 주소의 첫 번째 줄	문자열	모든 거리
	청구_주소_L2	청구서 수신 주소의 두 번째 줄	문자열	모든 유닛 123
	빌링_시티	청구서 수신 주소에 있는 도시	문자열	모든 도시

범주	변수 유형	설명	데이터 형식	예	
	청구_주	청구서 수신 주소에 있는 주 또는 도	문자열	모든 주 또는 도	

범주	변수 유형	설명	데이터 형식	예
	청구_국가	청구서 수신 주소에 있는 국가	문자열	모든 국가 참고: Amazon 사기 탐지기는 이 데이터를 강화합니다. 자세한 내용은 지

범주	변수 유형	설명	데이터 형식	예	
				리전 위치 강화 섹션을 참조하세요.	

범주	변수 유형	설명	데이터 형식	예
	빌링_ZIP	청구서 수신 주소에 있는 우편번호	문자열	01234 참고: Amazon 사기 탐지기는 이 데이터를 강화합니다. 자세한 내용은 지리적 위

범주	변수 유형	설명	데이터 형식	예
				치강화 섹션을 참조하세요.
배송	배송_이름	배송 주소와 연결된 이름	문자열	John Doe

범주	변수 유형	설명	데이터 형식	예
	배송_전화	배송 주소와 연결된 전화번호	문자열	+1 555-0100 참고: Amazon 사기 탐지기는 이 데이터를 강화합니다. 자세한 내용은 전화번

범주	변수 유형	설명	데이터 형식	예
				호강화 섹션을 참조하세요.
	배송_주소_L1	배송 주소의 첫 번째 줄	문자열	123 Any Street
	배송_주소_L2	배송 주소의 두 번째 줄	문자열	유닛 123
	배송_도시	배송 주소에 있는 도시	문자열	모든 도시
	배송_상태	배송 주소에 있는 주 또는 도	문자열	모든 주

범주	변수 유형	설명	데이터 형식	예
	배송_국가	배송지 주소에 있는 해당 국가	문자열	모든 국가 참고: Amazon 사기 탐지기는 이 데이터를 강화합니다. 자세한 내용은 지

범주	변수 유형	설명	데이터 형식	예
				리전 위치 강화 섹션을 참조하세요.

범주	변수 유형	설명	데이터 형식	예
	배송_ZIP	배송 주소에 있는 우편 번호	문자열	01234 참고: Amazon 사기 탐지기는 이 데이터를 강화합니다. 자세한 내용은 지리적 위

범주	변수 유형	설명	데이터 형식	예
				치강화 섹션을 참조하세요.
결제	주문_ID	거래의 고유 식별자	문자열	LUX60
	가격	총 주문 가격	문자열	560.00
	통화_코드	ISO 4217 통화 코드	문자열	USD
	결제_유형	이벤트 기간 동안 결제에 사용되는 결제 수단	문자열	크레디트카드

범주	변수 유형	설명	데이터 형식	예
	인증_코드	신용 카드 발급사 또는 발급 은행에서 보낸 영숫자 코드	문자열	0000
	AVS	카드 프로세서의 주소 확인 시스템(AVS) 응답 코드	문자열	Y
제품	제품_카테고리	주문 품목의 제품 카테고리	문자열	키친
사용자 지정(Custom)	NUMERIC	실수로 표현할 수 있는 모든 변수	Floa	1.224
	CATEGORICAL	카테고리, 세그먼트 또는 그룹을 설명하는 모든 변수	문자열	라지
	자유_양식_텍스트	이벤트의 일부로 캡처된 모든 자유 형식 텍스트 (예: 고객 리뷰 또는 의견)	문자열	자유 형식 텍스트 입력의 예

변수 유형에 변수 할당

모델을 훈련하는 데 변수를 사용할 계획이라면 변수에 할당할 올바른 변수 유형을 선택하는 것이 중요합니다. 잘못된 변수 유형 할당은 모델 성능에 부정적인 영향을 미칠 수 있습니다. 또한 나중에 할당을 변경하기가 매우 어려워질 수 있습니다. 특히 여러 모델 및 이벤트에서 변수를 사용한 경우에는 더욱 그렇습니다.

변수에 사전 정의된 변수 유형 중 하나 또는 사용자 지정 변수 유형 (FREE_FORM_TEXT,CATEGORICAL, 또는NUMERIC) 중 하나를 할당할 수 있습니다.

변수를 올바른 변수 유형에 할당하기 위한 중요 참고 사항

1. 변수가 사전 정의된 변수 유형 중 하나와 일치하는 경우 해당 변수를 사용하십시오. 변수 유형이 변수와 일치하는지 확인하십시오. 예를 들어 ip_address 변수를 EMAIL_ADDRESS 변수 유형에 할당하면 ip_address 변수에는 ASN, ISP, 지리적 위치 및 위험 점수와 같은 강화 기능이 포함되지 않습니다. 자세한 정보는 가변 인리치먼트을 참조하세요.
2. 변수가 사전 정의된 변수 유형과 일치하지 않는 경우 아래 나열된 권장 사항에 따라 사용자 지정 변수 유형 중 하나를 할당하십시오.
3. 일반적으로 순서가 자연스럽지 않고 범주, 세그먼트 또는 그룹에 넣을 수 있는 변수에 CATEGORICAL 변수 유형을 할당합니다. 모델 학습에 사용하는 데이터세트에는 판매자_ID, campaign_id 또는 policy_id와 같은 ID 변수가 있을 수 있습니다. 이러한 변수는 그룹을 나타냅니다 (예: 동일한 policy_id를 가진 모든 고객이 그룹을 나타냄). 다음 데이터가 있는 변수에는 CATEGORICAL 변수 유형을 할당해야 합니다 -
   - 고객_ID, 세그먼트_ID, 색상_ID, 부서_코드 또는 제품_ID와 같은 데이터를 포함하는 변수.
   - 참, 거짓 또는 null 값을 가진 부울 데이터를 포함하는 변수.
   - 회사 이름, 제품 범주, 카드 유형 또는 추천 매체와 같은 그룹 또는 범주에 넣을 수 있는 변수.

> **Note**
>
> ENTITY_ID아마존 사기 탐지기가 ENTITY_ID 변수에 할당하는 데 사용하는 예약 변수 유형입니다. ENTITY_ID 변수는 평가하려는 작업을 시작하는 개체의 ID입니다. 거래 사기 인사이트 (TFI) 모델 유형을 생성하는 경우 ENTITY_ID 변수를 제공해야 합니다. 데이터의 어떤 변수가 작업을 시작하는 엔티티를 고유하게 식별하는지 결정하여 ENTITY_ID 변수로 전달해야 합니다. 데이터셋의 다른 모든 ID가 존재하고 모델 학습에 사용하는 경우 CATEGORICAL 변수 유형을 할당하십시오. 데이터세트의 개체가 아닌 다른 ID의 예로는 판매자_ID, 정책_ID, 캠페인_ID가 있습니다.

4. 텍스트 블록이 포함된 변수에 FREE_FORM_TEXT 변수 유형을 지정합니다. FREE_FORM_TEXT 변수 유형의 예로는 사용자 리뷰, 댓글, 날짜 및 추천 코드가 있습니다. FREE_FORM_TEXT 데이터에는 구분 기호로 구분된 여러 토큰이 포함됩니다. 구분 기호는 영숫자 및 밑줄 기호 이외의 모든 문자일 수 있습니다. 예를 들어 사용자 리뷰와 댓글은 “공백” 구분 기호로 구분할 수 있으며, 날짜 및 추천 코드는 하이픈을 구분 기호로 사용하여 접두사, 접미사 및 중간 부분을 구분할 수 있습니다. 아마존 사기 탐지기는 구분 기호를 사용하여 FREE_FORM_TEXT 변수에서 데이터를 추출합니다.
5. 실수이고 고유한 순서가 있는 변수에 NUMERIC 변수 유형을 할당합니다. NUMERIC 변수의 예로는 요일, 사고 심각도, 고객 등급 등이 있습니다. 이러한 변수에 CATEGORICAL 변수 유형을 할당할 수 있지만 고유한 순서가 있는 모든 실수 변수를 NUMERIC 변수 유형에 할당하는 것이 좋습니다.

가변 인리치먼트

Amazon Fraud Detector는 IP 주소, 은행 식별 번호 (BIN), 전화 번호 등 사용자가 제공하는 일부 원시 데이터 요소를 강화하여 추가 입력을 생성하고 이러한 데이터 요소를 사용하는 모델의 성능을 향상시킵니다. 강화 기능은 잠재적으로 의심스러운 상황을 식별하고 모델이 더 많은 부정 행위를 포착하는 데 도움이 됩니다.

전화번호 강화

Amazon Fraud Detector는 지리적 위치, 원래 배송사 및 전화번호의 유효성과 관련된 추가 정보로 전화번호 데이터를 강화합니다. 전화번호 강화 기능은 2021년 12월 13일 또는 그 이후에 교육을 받고 전화번호가 국가 코드 (+xxx) 를 포함하는 모든 모델에 대해 자동으로 활성화됩니다. 모델에 전화번호 변수를 포함시켰고 2021년 12월 13일 이전에 학습시킨 경우 모델을 다시 학습시켜 이 강화 기능을 활용할 수 있도록 하십시오.

데이터를 성공적으로 보강하려면 전화번호 변수에 다음 형식을 사용하는 것이 좋습니다.

변수	형식	설명
PHONE_NUMBER	E.164 스탠다드	전화번호에 국가 코드 (+xxx) 를 포함해야 합니다.
청구_전화 및 배송_전화	E.164 스탠다드	전화번호에 국가 코드 (+xxx) 를 포함해야 합니다.

지리적 위치 강화

2022년 2월 8일부터 아마존 사기 탐지기는 이벤트에 제공한 IP_주소, 청구_ZIP 및 배송_ZIP 값 간의 물리적 거리를 계산합니다. 계산된 거리는 사기 탐지 모델의 입력으로 사용됩니다.

지리적 위치 보강을 활성화하려면 이벤트 데이터에 세 가지 변수 (IP_ADDRESS, BILLING_ZIP 또는 SHIPPING_ZIP) 중 2개 이상이 포함되어야 합니다. 또한 각 청구_우편 번호와 배송_우편 번호에는 각각 유효한 청구_국가 코드 및 배송_국가 코드가 있어야 합니다. 2022년 2월 8일 이전에 학습된 모델에 이러한 변수가 포함되어 있는 경우 모델을 다시 학습시켜 지리적 위치 강화를 활성화해야 합니다.

데이터가 유효하지 않아 Amazon Fraud Detector가 이벤트의 IP_ADDRESS, BILLING_ZIP 또는 SHIPPING_ZIP 값과 연결된 위치를 확인할 수 없는 경우 특수 자리 표시자 값이 대신 사용됩니다. 예를 들어 이벤트에 유효한 IP_ADDRESS 및 BILLING_ZIP 값이 있지만 SHIPPING_ZIP 값은 유효하지 않다고 가정해 보겠습니다. 이 경우 IP_ADDRESS—> BILLING_ZIP에 대해서만 인리치먼트가 수행됩니다. IP_주소—>배송_ZIP 및 청구_ZIP—>배송_ZIP에 대해서는 강화가 수행되지 않습니다. 대신 자리 표시자 값이 대신 사용됩니다. 모델에 지리적 위치 보강을 활성화했는지 여부에 관계없이 모델의 성능은 변하지 않습니다.

BILLING_ZIP 및 SHIPPING_ZIP 변수를 CUSTOM_CATEGORICAL 변수 유형에 매핑하여 지리적 위치 강화를 거부할 수 있습니다. 변수 유형을 변경해도 모델의 성능에는 영향을 주지 않습니다.

지리적 위치 변수 형식

위치 데이터를 성공적으로 보강하려면 다음과 같은 지리 위치 변수 형식을 사용하는 것이 좋습니다.

변수	형식	설명
IP_ADDRESS	IPv4 주소	예를 들어 - 1.1.1.1
청구_우편번호 및 배송_ZIP	지정된 국가의 ISO 3166-1 알파-2 우편 번호	자세한 내용은 이 항목의 국가 및 관할 구역 코드 섹션을 참조하십시오.
청구_국가 및 배송_국가	ISO 3166-1 알파-2 두 글자 표준 국가 코드	자세한 내용은 이 항목의 국가 및 관할 구역 코드 섹션을 참조하십시오. Amazon Fraud Detector는 국가 이름의 일반적인 변형을 모두

변수	형식	설명
		ISO 3166-1 두 글자 표준 국가 코드와 일치시키려고 합니다. 그러나 정확히 일치한다는 보장은 할 수 없습니다.

국가 및 관할 구역 코드

다음 표에는 Amazon Fraud Detector가 지리적 위치 강화를 지원하는 국가 및 지역의 전체 목록이 나와 있습니다. 각 국가 및 지역에는 지정된 국가 코드 (특히 ISO 3166-1 alpha-2 두 글자 국가 코드) 와 우편번호가 있습니다.

우편 번호 형식

- 9 - 넘버
- a - 레터
- [X] - X는 선택 사항입니다. 예를 들어, 거스니 “GY9 [9] 9aa”는 “GY9 9aa”와 “GY99 9aa”가 모두 유효하다는 것을 의미합니다. 한 가지 형식을 사용하세요.
- [X/XX] - X 또는 XX 중 하나를 사용할 수 있습니다. 예를 들어 버뮤다의 “aa [aa/99]”는 “aa aa”와 “aa 99"가 모두 유효함을 의미합니다. 다음 형식 중 하나를 사용하되 둘 다 사용하지는 마십시오.
- 일부 국가에는 고정 접두사가 있습니다. 예를 들어, 안도라의 우편 번호는 AD999 입니다. 즉, 국가 코드는 문자 AD로 시작하고 그 뒤에 숫자 3개가 와야 합니다.

코드	이름	우편 번호
광고	안도라	AD999
AR	네덜란드령 안틸 제도	9999
AT	오스트리아	9999
AU	호주	9999
아즈	아제르바이잔	AZ 9999

코드	이름	우편 번호
BD	방글라데시	9999
있다	벨기에	9999
가방	불가리아	9999
BM	버뮤다	aa [aa/99]
BY	벨로루시	999999
CA	캐나다	a9a 9a9
CH	스위스	9999
CL	칠레	9999999
CO	콜롬비아	999999
CR	코스타리카	99999
싸이	사이프러스	9999
CZ	체코	999 99
DE	독일	99999
DK	덴마크	9999
DO	도미니카 공화국	99999
DZ	알제리	99999
EE	에스토니아	99999
ES	스페인	99999
FI	핀란드	99999
FM	미크로네시아 연방	99999

코드	이름	우편 번호
FO	페로 제도	999
FR	프랑스	99999
GB	영국	[a] [a/9] 9aa
GG	건지	GY9 [9] 9aa
GL	그린란드	9999
GP	과들루프	99999
GT	과테말라	99999
총	괌	99999
HR	크로아티아	99999
휴	헝가리	9999
IE	아일랜드	a99 [a/9] [a/9] [a/9] [a/9]
임	맨 섬	IM9 [9] 9aa
IN	인도	999999
IS	아이슬란드	999
그것을	이탈리아	99999
JE	저지	JE9 [9] 9aa
JP	일본	999-9999
KR	대한민국	99999
리	리히텐슈타인	9999
LK	스리랑카	99999

코드	이름	우편 번호
LT	리투아니아	99999
루	룩셈부르크	L-9999
LV	라트비아	LV-9999
MC	모나코	99999
MD	몰도바 공화국	9999
MH	마셜 제도	99999
MK	북 마케도니아	9999
지도	북마리아나 제도	99999
MQ	마티니크	99999
산	몰타	aaa 999
MX	멕시코	99999
내	말레이시아	99999
NL	네덜란드	999 a
아니요	노르웨이	9999
NZ	뉴질랜드	9999
PH	필리핀	9999
PK	파키스탄	99999
PL	폴란드	99-999
PR	푸에르토리코	99999
PT	포르투갈	9999-999

코드	이름	우편 번호
PW	팔라우	99999
다시	레위니옹	99999
RO	루마니아	999999
RU	러시아 연방	999999
사용	스웨덴	999 99
SG	싱가포르	999999
시	슬로베니아	9999
SK	슬로바키아	999 99
SM	산마리노	99999
TH	태국	99999
TR	터키	99999
UA	우크라이나	99999
US	미국	99999
사세요	우루과이	99999
VI	미국령 버진 제도	99999
WF	월리스 푸투나	99999
YT	마요트	99999
ZA	남아프리카공화국	9999

사용자 에이전트 강화

ATI (어카운트 테이크오버 인사이트) 모델을 만드는 경우 데이터셋에 useragent 변수 유형의 변수를 제공해야 합니다. 이 변수에는 로그인 이벤트의 브라우저, 장치 및 OS 데이터가 포함됩니다. Amazon Fraud Detector는, 등의 user_agent_family OS_family 추가 정보를 사용하여 사용자 에이전트 데이터를 강화합니다. device_family

변수 만들기

변수 생성 명령을 사용하거나 를 사용하거나 Amazon Fraud Detector 콘솔에서 변수를 생성할 수 있습니다. CreateVariableAWS SDK for Python (Boto3)

Amazon 사기 탐지기 콘솔을 사용하여 변수 생성

이 예제에서는 두 개의 변수 및 를 만들어 해당 변수 유형 (EMAIL_ADDRESS및IP_ADDRESS) 에 할당합니다. email_address ip_address 이러한 변수가 예제로 사용됩니다. 모델 학습에 사용할 변수를 만들려면 데이터셋에서 사용 사례에 적합한 변수를 사용하세요. 변수를 만들기 가변 인리치먼트 전에 변수 유형 및 에 대해 읽어보세요.

변수를 만들려면

1. AWS관리 콘솔을 열고 계정에 로그인합니다.
2. Amazon Fraud Detector로 이동하여 왼쪽 탐색 메뉴에서 변수를 선택한 다음 생성을 선택합니다.
3. 새 변수 페이지에서 변수 email_address 이름으로 를 입력합니다. 변수에 대한 설명을 입력할 수도 있습니다.
4. 변수 유형에서 이메일 주소를 선택합니다.
5. 이 변수 유형은 사전 정의되어 있으므로 Amazon Fraud Detector는 이 변수 유형에 대한 데이터 유형을 자동으로 선택합니다. 변수에 변수 유형이 자동으로 할당되지 않는 경우 목록에서 변수 유형을 선택합니다. 자세한 정보는 변수 유형을 참조하세요.
6. 변수에 기본값을 제공하려면 사용자 지정 기본값 정의를 선택하고 변수의 기본값을 입력합니다. 이 예를 따르는 경우 이 단계를 건너뛰십시오.
7. Create(생성)를 선택합니다.
8. email_address 개요 페이지에서 방금 생성한 변수의 세부 정보를 확인합니다.

   업데이트가 필요한 경우 편집을 선택하고 업데이트를 제공하십시오. 변경 사항 저장을 선택합니다.
9. 프로세스를 반복하여 다른 변수를 ip_address 생성하고 변수 유형으로 IP 주소를 선택합니다.

10. 변수 페이지에는 새로 만든 변수가 표시됩니다.

> **Important**
>
> 데이터셋에서 원하는 만큼 변수를 만드는 것이 좋습니다. 나중에 이벤트 유형을 생성할 때 사기를 탐지하고 사기 탐지를 생성하도록 모델을 학습시키는 데 포함할 변수를 결정할 수 있습니다.

를 사용하여 변수 만들기 AWS SDK for Python (Boto3)

다음 예제는 CreateVariableAPI에 대한 요청을 보여줍니다. 이 예제에서는 두 개의 변수 및 를 만들어 해당 변수 유형 (EMAIL_ADDRESS및IP_ADDRESS) 에 할당합니다. email_address ip_address

이러한 변수가 예제로 사용됩니다. 모델 학습에 사용할 변수를 만들려면 데이터셋에서 사용 사례에 적합한 변수를 사용하세요. 변수를 만들기 가변 인리치먼트 전에 변수 유형 및 에 대해 읽어보세요.

반드시 가변 소스를 지정해야 합니다. 변수 값이 파생되는 위치를 식별하는 데 도움이 됩니다. 변수 소스가 EVENT인 경우 변수 값은 GetEventPrediction요청의 일부로 전송됩니다. 변수 값이 인 경우 해당 값은 MODEL_SCORE Amazon 사기 탐지기로 채워집니다. 변수 값이 가져온 SageMaker 모델로 채워지는 경우 EXTERNAL_MODEL_SCORE

```
import boto3
fraudDetector = boto3.client('frauddetector')

 #Create variable email_address
   fraudDetector.create_variable(
     name = 'email_address',
     variableType = 'EMAIL_ADDRESS',
     dataSource = 'EVENT',
     dataType = 'STRING',
     defaultValue = '<unknown>'
     )

#Create variable ip_address
   fraudDetector.create_variable(
     name = 'ip_address',
     variableType = 'IP_ADDRESS',
     dataSource = 'EVENT',
     dataType = 'STRING',
     defaultValue = '<unknown>'
```

```
)
```

변수 삭제

변수를 삭제하면 Amazon Fraud Detector는 해당 변수를 영구적으로 삭제하며 데이터는 더 이상 Amazon Fraud Detector에 저장되지 않습니다.

Amazon Fraud Detector의 이벤트 유형에 포함된 변수는 삭제할 수 없습니다. 먼저 변수와 연결된 이벤트 유형을 삭제한 다음 변수를 삭제해야 합니다.

Amazon Fraud Detector 모델 출력 변수 및 SageMaker 모델 출력 변수는 수동으로 삭제할 수 없습니다. Amazon Fraud Detector는 모델을 삭제하면 모델 출력 변수를 자동으로 삭제합니다.

Amazon Fraud Detector 콘솔에서 변수 삭제 CLI 명령을 사용하거나 DeleteVariableAPI를 사용하거나 다음을 사용하여 변수를 삭제할 수 있습니다. AWS SDK for Python (Boto3)

콘솔을 사용하여 변수 삭제

변수를 삭제하려면

1. https://console.aws.amazon.com/frauddetector 에서 로그인하여 아마존 사기 탐지기 콘솔을 여십시오. AWS Management Console
2. Amazon Fraud Detector 콘솔의 왼쪽 탐색 창에서 리소스를 선택한 다음 변수를 선택합니다.
3. 삭제하려는 변수를 선택합니다.
4. 작업을 선택한 후 삭제를 선택합니다.
5. 변수 이름을 입력한 다음 변수 삭제를 선택합니다.

를 사용하여 변수 삭제 AWS SDK for Python (Boto3)

다음 코드 샘플은 API를 사용하여 customer_name 변수를 삭제합니다. DeleteVariable

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.delete_variable (

name = 'customer_name'

)
```

Labels

레이블은 이벤트가 시기성인지 합법적인지를 분류합니다. 레이블은 이벤트 유형과 연결되어 Amazon Fraud Detector에서 기계 학습 모델을 훈련하는 데 사용됩니다. 온라인 사기 인사이트 (OFI) 또는 거래 사기 인사이트 (TFI) 모델을 트레이닝하려는 경우 교육 데이터세트에 있는 최소 400개의 이벤트가 사기 또는 합법적인 것으로 분류되어야 합니다. Fraud, legit, 1 또는 0과 같은 모든 레이블을 사용하여 훈련 데이터세트의 이벤트를 분류할 수 있습니다. 교육이 완료되면 학습된 모델이 이벤트의 사기 여부를 평가하고 이 값을 사용하여 이벤트를 사기 또는 합법적인 이벤트로 분류합니다.

먼저 학습 데이터세트에 사용된 값으로 레이블을 만든 다음 해당 레이블을 사기 탐지 모델을 구축 및 학습하는 데 사용되는 이벤트 유형과 연결해야 합니다.

라벨 생성

Amazon Fraud Detector 콘솔에서 put-label 명령을 사용하거나 PutLabelAPI를 사용하거나 를 사용하여 라벨을 생성할 수AWS SDK for Python (Boto3) 있습니다.

Amazon Fraud Detector 콘솔을 사용하여 라벨을 생성합니다.

라벨을 만들려면

1. AWS관리 콘솔을 열고 계정에 로그인합니다.
2. Amazon Fraud Detector로 이동하여 왼쪽 내비게이션에서 라벨을 선택한 다음 생성을 선택합니다.
3. 라벨 생성 페이지에서 사기 이벤트의 라벨 이름을 라벨 이름으로 입력합니다. 레이블 이름은 학습 데이터세트의 사기 행위를 나타내는 레이블과 일치해야 합니다. 레이블에 대한 선택적 설명을 입력합니다.
4. 라벨 생성을 선택합니다.
5. 두 번째 레이블을 만들고 합법적인 이벤트의 레이블 이름을 입력합니다. 레이블 이름이 훈련 데이터세트의 합법적인 활동을 나타내는 값과 일치하는지 확인하세요.

를 사용하여 레이블 만들기AWS SDK for Python (Boto3)

다음AWS SDK for Python (Boto3) 예제 코드는 PutLabelAPI를 사용하여 두 개의 레이블 (사기, 합법)을 생성합니다. 레이블을 만든 후 이벤트 유형에 레이블을 추가하여 특정 이벤트를 분류할 수 있습니다.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.put_label(
name = 'fraud',
description = 'label for fraud events'
)

fraudDetector.put_label(
name = 'legit',
description = 'label for legitimate events'
)
```

라벨 업데이트

이벤트 데이터세트가 Amazon Fraud Detector에 저장되어 있는 경우, 이벤트에 대한 오프라인 사기 조사를 수행하고 기계 학습 피드백 루프를 닫으려는 경우와 같이 저장된 이벤트의 레이블을 추가하거나 업데이트해야 할 수 있습니다.

update-event-label명령, UpdateEventLabelAPI 또는 다음을 사용하여 저장된 이벤트의 레이블을 추가하거나 업데이트할 수 있습니다.AWS SDK for Python (Boto3)

다음AWS SDK for Python (Boto3) 예제 코드는*UpdateEventLabel* API를 사용한 이벤트 유형 등록과 관련된 라벨 프라우드를 추가합니다.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.update_event_label(
            eventId        = '802454d3-f7d8-482d-97e8-c4b6db9a0428',
            eventTypeName  = 'registration',
            assignedLabel  = 'fraud',
            labelTimestamp = '2020-07-13T23:18:21Z'
)
```

Amazon Fraud Detector에 저장된 이벤트 데이터의 이벤트 라벨 업데이트

이벤트에 대해 오프라인 사기 조사를 수행하고 기계 학습 피드백 루프를 닫으려는 경우와 같이 Amazon Fraud Detector에 이미 저장된 이벤트에 대해 사기 레이블을 추가하거나 업데이트해야 할 수 있습니다. Amazon Fraud Detector에 이미 저장된 이벤트의 라벨을 업데이트하려면`UpdateEventLabel` API 작업을 사용하십시오. 다음 예제에서는 예제 UpdateEventLabel API 호출을 보여줍니다.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.update_event_label(
            eventId        = '802454d3-f7d8-482d-97e8-c4b6db9a0428',
            eventTypeName  = 'sample_registration',
            assignedLabel  = 'fraud',
            labelTimestamp = '2020-07-13T23:18:21Z'
)
```

라벨 삭제

라벨을 삭제하면 Amazon Fraud Detector는 해당 라벨을 영구적으로 삭제하며 데이터는 더 이상 Amazon Fraud Detector에 저장되지 않습니다.

Amazon Fraud Detector에서 이벤트 유형에 포함된 레이블은 삭제할 수 없습니다. 이벤트 ID에 할당된 레이블은 삭제할 수 없습니다. 먼저 관련 이벤트 ID를 삭제해야 합니다.

Amazon Fraud Detector 콘솔에서 delete-label 명령을 사용하거나 DeleteLabelAPI를 사용하거나 다음을 사용하여 라벨을 삭제할 수 있습니다.AWS SDK for Python (Boto3)

콘솔을 사용하여 레이블을 삭제

레이블을 삭제하려면

1. 에 로그인한 후 https://console.aws.amazon.com/frauddetector 에서 Amazon Fraud Detector에서 Amazon Fraud Detector에서 Amazon Fraud DetectorAWS Management Console
2. Amazon Fraud Detector 콘솔의 왼쪽 탐색 창에서 리소스를 선택한 다음 라벨을 선택합니다.
3. 삭제하고 싶은 레이블을 선택합니다.

4. [Actions]를 선택한 후 [Delete]를 선택합니다.
5. 레이블 이름을 입력한 다음 레이블 삭제를 선택합니다.

를 사용하여 라벨을 삭제합니다.AWS SDK for Python (Boto3)

다음AWS SDK for Python (Boto3) 예제 코드는 DeleteLabelAPI를 사용하여 합법적인 라벨을 삭제합니다.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.delete_event_label (
    name = 'legit'
)
```

규칙

규칙은 Amazon Fraud Detector에 사기 예측 중에 변수 값을 해석하는 방법을 알려주는 조건입니다. 규칙은 탐지기 로직의 일부이며 다음 요소로 구성됩니다.

- 변수 또는 목록 — 변수는 부정 행위 예측에 사용하려는 이벤트 데이터 세트의 데이터 요소를 나타냅니다. 목록은 이벤트 데이터셋의 변수에 대한 입력 데이터 요소 집합입니다. 규칙에 사용되는 변수는 평가된 이벤트 유형에 미리 정의되어야 하며 규칙에 사용되는 목록은 변수 유형과 연결되어야 합니다. 자세한 내용은 Variables 및 목록 단원을 참조하세요.
- 표현식 — 규칙의 표현식은 비즈니스 로직을 캡처합니다. 규칙에 변수를 사용하는 경우 변수, 비교 연산자 (예: >, <, <=, >=. ==) 및 값을 사용하여 간단한 규칙 표현식이 구성됩니다. 목록을 사용하는 경우 규칙 표현식은 목록 항목과 목록 이름으로 구성됩니다. `in` 자세한 정보는 규칙 언어 참조을 참조하세요. `and`및 를 사용하여 여러 식을 결합할 수 `or` 있습니다. 모든 표현식은 부울 값 (true 또는 false) 으로 평가되고 길이가 4,000자 미만이어야 합니다. If-else 유형 조건은 지원되지 않습니다.
- 결과 — 결과는 규칙이 일치할 때 Amazon Fraud Detector에서 반환하는 응답입니다. 결과는 사기 예측의 결과를 나타냅니다. 가능한 각 사기 예측에 대한 결과를 생성하여 규칙에 추가할 수 있습니다. 자세한 정보는 결과을 참조하세요.

탐지기에는 하나 이상의 관련 규칙이 있어야 합니다. 규칙은 최대 3개의 목록을 포함할 수 있으며, 탐지기는 최대 30개의 목록을 포함할 수 있습니다. 검출기 생성 프로세스의 일부로 규칙을 생성합니다. 새 규칙을 만들어 기존 탐지기에 연결할 수도 있습니다.

규칙 언어 참조

다음 섹션에서는 Amazon Fraud Detector의 표현 (즉, 규칙 작성) 기능에 대해 간략히 설명합니다.

변수 사용

평가된 이벤트 유형에 정의된 모든 변수를 표현식의 일부로 사용할 수 있습니다. 변수를 나타내려면 달러 기호를 사용하십시오.

```
$example_variable < 100
```

목록 사용

변수 유형과 연관되고 규칙 표현식의 일부로 항목이 채워진 모든 목록을 사용할 수 있습니다. 달러 기호를 사용하여 목록 입력 값을 나타냅니다.

```
$example_list_variable in @list_name
```

비교, 멤버십 및 ID 연산자

Amazon Fraud Detector에는 **>**, **>=**, **<**, **<=**,**!** 같은 비교 연산자가 포함됩니다. **=**, **==**, 인, 인, 인 아님

예를 들면 다음과 같습니다.

예: **<**

```
$variable < 100
```

예: 인, 인 아님

```
$variable in [5, 10, 25, 100]
```

예:**!** **=**

```
$variable != "US"
```

예: **==**

```
$variable == 1000
```

오퍼레이터 테이블

연산자	아마존 사기 탐지기 운영자
같음	==
같지 않음	!=
초과	>
미만	<
다음보다 크거나 같음	>=
작거나 같음	<=
In	in
및	및
또는	또는
아님	!

기본 수학

표현식에 기본 수학 연산자 (예: +, -, *,/) 를 사용할 수 있습니다. 일반적인 사용 사례는 평가 중에 변수를 결합해야 하는 경우입니다.

아래 규칙에서는 를 $variable_1 사용하여 $variable_2 변수를 추가하고 합계가 10보다 작은지 확인합니다.

```
$variable_1 + $variable_2 < 10
```

기본 수학 테이블 데이터

연산자	아마존 사기 탐지기 운영자
플러스	+

연산자	아마존 사기 탐지기 운영자
마이너스	-
Multiply	*
Divide	/
모듈로	%

정규 표현식 (정규식)

정규 표현식을 사용하여 표현식의 일부로 특정 패턴을 검색할 수 있습니다. 이는 변수 중 하나에 대해 특정 문자열이나 숫자 값을 일치시키려는 경우에 특히 유용합니다. Amazon Fraud Detector는 정규 표현식으로 작업할 때만 일치를 지원합니다 (예를 들어, 제공된 문자열이 정규 표현식과 일치하는지 여부에 따라 True/False를 반환합니다). Amazon Fraud Detector의 정규 표현식 지원은 Java의.matches () 를 기반으로 합니다 (RE2J 정규 표현식 라이브러리 사용). 인터넷에는 다양한 정규 표현식 패턴을 테스트하는 데 유용한 여러 웹 사이트가 있습니다.

아래 첫 번째 예에서는 먼저 변수를 소문자로 `email` 변환합니다. 그런 다음 패턴이 `@gmail.com` `email` 변수에 있는지 확인합니다. 두 번째 마침표가 이스케이프 처리되어 문자열을 명시적으로 확인할 수 있습니다. `.com`

```
regex_match(".*@gmail\.com", lowercase($email))
```

두 번째 예에서는 변수에 국가 코드가 `phone_number` `+1` 포함되어 있는지 확인하여 전화번호가 미국에서 온 전화번호인지 확인합니다. 더하기 기호는 이스케이프 처리되므로 문자열을 명시적으로 확인할 수 있습니다. `+1`

```
regex_match(".*\+1", $phone_number)
```

정규식 테이블

연산자	아마존 사기 탐지기 예제
다음으로 시작하는 모든 문자열과 일치합니다.	정규 표현_일치 (“^내 문자열”, $변수)

연산자	아마존 사기 탐지기 예제
전체 문자열을 정확히 일치시킵니다.	정규 표현_일치 (“내 문자열”, $변수)
줄 바꿈을 제외한 모든 문자와 일치	정규 일치 (” . “, $변수)
'mystring' 앞의 새 줄을 제외한 모든 문자 수 일치	정규 일치 (”. *마이스트링”, $변수)
특수 문자 이스케이프	\

누락된 값 확인

때로는 값이 누락되었는지 확인하는 것이 좋습니다. 아마존 사기 탐지기에서는 null로 표시됩니다. 다음 구문을 사용하여 이 작업을 수행할 수 있습니다.

```
$variable != null
```

마찬가지로 값이 없는지 확인하려면 다음과 같이 할 수 있습니다.

```
$variable == null
```

다양한 조건

and및 를 사용하여 여러 식을 결합할 수 or 있습니다. Amazon Fraud Detector는 하나의 참값이 발견되면 OR 표현식에서 멈추고, 하나의 거짓값이 AND 발견되면 표현식에서 멈춥니다.

아래 예에서는 조건을 사용하여 두 가지 조건을 확인하고 있습니다. and 첫 번째 명령문에서는 변수 1이 100보다 작은지 확인합니다. 두 번째 단계에서는 변수 2가 미국이 아닌지 확인합니다.

규칙이 를 사용하므로 전체 조건이 TRUE로 평가되려면 둘 다 TRUE여야 합니다. and

```
$variable_1 < 100 and $variable_2 != "US"
```

다음과 같이 괄호를 사용하여 부울 연산을 그룹화할 수 있습니다.

```
$variable_1 < 100 and $variable_2 != "US" or ($variable_1 * 100.0 > $variable_3)
```

기타 표현식 유형

DateTime기능

함수	설명	예
현재 날짜/시간 가져오기 ()	규칙 실행의 현재 시간을 ISO8601 UTC 형식으로 제공합니다. getepoch 밀리초 (getcurrentdatetime ()) 를 사용하여 추가 작업을 수행할 수 있습니다.	현재 날짜시간 가져오기 () == “2023-03-28T 18:34:02 Z”
이스보레 (DateTime1, DateTime 2)	호출자 DateTime 1이 2보다 이전이면 부울 (참/거짓) 을 반환합니다. DateTime	이전 (현재 날짜 시간 가져오기 (), “2019-11-30T 01:01:01 Z”) == “거짓” 이전 (현재 날짜 시간 가져오기 (), “2050-11-30T 01:05:01 Z”) == “참”
이스턴 (DateTime 1, DateTime 2)	호출자 DateTime 1이 2 이후인 경우 부울 (참/거짓) 을 반환합니다. DateTime	isafter (현재 날짜 시간 가져오기 (), “2019-11-30T 01:01:01 Z”) == “참” isafter (현재 날짜 시간 가져오기 (), “2050-11-30T 01:05:01 Z”) == “거짓”
몇 밀리초 () 를 가져옵니다. DateTime	a를 DateTime 가져와 에포크 밀리초 DateTime 단위로 반환합니다. 날짜에 수학 연산을 수행하는 데 유용합니다.	겟포치 밀리세컨드 (“2019-11-30T 01:01 Z”) == 1575032461

문자열 연산자

연산자	예
문자열을 대문자로 변환	대문자 ($변수)
문자열을 소문자로 변환	소문자 ($ 변수)

기타

연산자	Comment
댓글 추가	# 내 의견

규칙 생성

Amazon Fraud Detector 콘솔에서 create-rule 명령, CreateRuleAPI 또는 를 사용하여 규칙을 생성할 수 있습니다. AWS SDK for Python (Boto3)

각 규칙에는 비즈니스 로직을 캡처하는 단일 표현식이 포함되어야 합니다. 모든 표현식은 부울 값 (true 또는 false) 으로 평가되고 길이가 4,000자 미만이어야 합니다. If-else 유형 조건은 지원되지 않습니다. 표현식에 사용된 모든 변수는 평가된 이벤트 유형에 미리 정의되어 있어야 합니다. 마찬가지로 표현식에 사용되는 모든 목록은 미리 정의되고 변수 유형과 연결되며 항목으로 채워져야 합니다.

다음 예제에서는 기존 감지기에 `high_risk` 대한 규칙을 만듭니다`payments_detector`. 규칙은 표현식과 결과를 `verify_customer` 규칙과 연관시킵니다.

사전 조건

아래에 설명된 단계를 수행하려면 규칙 생성을 진행하기 전에 다음을 완료해야 합니다.

- 감지기 만들기
- 결과 만들기

사용 사례에 대한 탐지기, 규칙 및 결과를 만드는 경우 예제 탐지기 이름, 규칙 이름, 규칙 표현식 및 결과 이름을 사용 사례와 관련된 이름 및 표현식으로 바꾸십시오.

Amazon 사기 탐지기 콘솔에서 새 규칙 생성

1. AWS관리 콘솔을 열고 계정에 로그인합니다. 아마존 사기 탐지기로 이동합니다.
2. 왼쪽 탐색 창에서 탐지기를 선택하고 사용 사례에 맞게 생성한 감지기 (예: payments_detector) 를 선택합니다.
3. payments_detector 페이지에서 관련 규칙 탭을 선택한 다음 규칙 생성을 선택합니다.
4. 새 규칙 페이지에서 다음을 입력합니다.

a. 이름에 규칙 이름을 입력합니다. 예 **high_risk**

b. 설명 - 선택 사항에서 필요에 따라 규칙 설명 (예:) 을 입력합니다. **This rule captures events with a high ML model score**

c. 표현식에서 표현식 빠른 참조 가이드를 사용하여 사용 사례에 맞는 규칙 표현식을 입력합니다. 예: `$sample_fraud_detection_model_insightscore >900`

d. 결과에서 사용 사례에 대해 생성한 결과 (예: verify_customer) 를 선택합니다. 결과는 사기 예측의 결과이며 평가 중에 규칙이 일치하면 결과가 반환됩니다.

5. 저장 규칙을 선택합니다.

검출기에 대한 새 규칙을 만들었습니다. 이는 Amazon Fraud Detector가 자동으로 이를 탐지기가 사용할 수 있도록 하는 규칙의 버전 1입니다.

를 사용하여 규칙 만들기 AWS SDK for Python (Boto3)

다음 예제 코드는 CreateRuleAPI를 사용하여 기존 감지기에 `high_risk` 대한 규칙을 만듭니다`payments_detector`. 또한 예제 코드는 규칙 표현식과 결과를 `verify_customer` 규칙에 추가합니다.

사전 조건

예제 코드를 사용하려면 규칙 생성을 진행하기 전에 다음을 완료해야 합니다.

- 감지기 만들기
- 결과 만들기

사용 사례에 대한 탐지기, 규칙 및 결과를 만드는 경우 예제 감지기 이름, 규칙 이름, 규칙 표현식 및 결과 이름을 사용 사례와 관련된 이름 및 표현식으로 바꾸십시오.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.create_rule(
ruleId = 'high_risk',
detectorId = 'payments_detector',
expression = '$sample_fraud_detection_model_insightscore > 900',
language = 'DETECTORPL',
outcomes = ['verify_customer']
```

```
)
```

Amazon Fraud Detector가 자동으로 해당 규칙을 탐지기에서 사용할 수 있도록 하는 규칙 버전 1을 생성했습니다.

업데이트 규칙

규칙 설명을 추가 또는 업데이트하거나, 규칙 표현식을 업데이트하거나, 규칙에 대한 결과를 추가 또는 제거하여 언제든지 규칙을 업데이트할 수 있습니다. 규칙을 업데이트하면 새 규칙 버전이 생성됩니다.

Amazon Fraud Detector 콘솔에서 update-rule-version명령, UpdateRuleVersionAPI 또는 AWS SDK를 사용하여 규칙을 업데이트할 수 있습니다.

규칙을 업데이트한 후에는 감지기 버전을 업데이트하여 새 규칙 버전을 사용해야 합니다.

Amazon 사기 탐지기 콘솔의 업데이트 규칙

규칙을 업데이트하려면

1. AWS관리 콘솔을 열고 계정에 로그인합니다. 아마존 사기 탐지기로 이동합니다.
2. 왼쪽 탐색 창에서 탐지기를 선택합니다.
3. 탐지기 창에서 업데이트하려는 규칙과 연결된 감지기를 선택합니다.
4. 탐지기 페이지에서 관련 규칙 탭을 선택하고 업데이트하려는 규칙을 선택합니다.
5. 규칙 페이지에서 작업을 선택하고 버전 생성을 선택합니다.
6. 참고로 버전이 변경되었습니다. 업데이트된 설명, 표현 또는 결과를 입력합니다.
7. 새 버전 저장을 선택합니다.

를 사용하여 규칙 업데이트 AWS SDK for Python (Boto3)

다음 예제 코드에서는 UpdateRuleVersionAPI를 사용하여 규칙 임계값을 `high_risk` 900에서 950으로 업데이트합니다. 이 규칙은 감지기와 관련이 `payments_detector` 있습니다.

```
fraudDetector.update_rule_version(
rule = {
    'detectorId' : 'payments_detector',
    'ruleId' : 'high_risk',
    'ruleVersion' : '1'
},
```

```
expression = '$sample_fraud_detection_model_insightscore > 950',
language = 'DETECTORPL',
outcomes = ['verify_customer']
)
```

목록

목록은 이벤트 데이터셋의 변수에 대한 입력 데이터 세트입니다. 검출기와 관련된 규칙의 입력 데이터를 사용합니다. 규칙은 Amazon Fraud Detector에 Fraud Detector에 Fraud Detector에 입력 데이터를 해석하는 방법을 알려주는 조건입니다. 예를 들어 IP 주소 목록을 만든 다음 특정 IP 주소가 목록에 있는 경우 액세스를 거부하는 규칙을 만들 수 있습니다. 목록을 사용하는 규칙은`$ip_address_value in@list_name` 형식으로 표현됩니다.

Amazon Fraud Detector를 사용하면 관련 규칙을 업데이트할 필요 없이 데이터를 추가하거나 제거하여 목록을 관리할 수 있습니다. 목록에 연결된 규칙은 새로 추가되거나 제거된 데이터를 자동으로 통합합니다.

목록은 최대 100,000개의 고유한 항목을 포함할 수 있으며 각 항목의 길이는 최대 320자일 수 있습니다. 규칙에 사용하는 모든 목록은 기본적으로 아마존 사기 탐지기의변수 유형 FREE_FORM_TEXT와 연결됩니다. 언제든지 목록에 변수 유형을 할당할 수 있습니다. 한 규칙에 최대 3개의 목록을 사용할 수 있습니다.

Amazon Fraud Detector 콘솔에서 API를 사용하거나AWS SDK를 사용하여 목록을 만들거나, 목록에 항목을 추가하거나, 목록을 삭제하거나, 목록에서 하나 이상의 항목을 삭제하거나AWS CLI, 목록에 변수 유형을 할당할 수 있습니다.

목록 생성

이벤트 데이터셋에 있는 변수의 입력 데이터 (항목) 가 포함된 목록을 만들고 이 목록을 규칙 표현식에 사용할 수 있습니다. 목록을 사용하는 규칙을 업데이트하지 않고도 목록의 항목을 동적으로 관리할 수 있습니다.

목록을 생성하려면 먼저 이름을 지정한 다음 필요에 따라 해당 목록을 Amazon Fraud Detector에서변수 유형 지원하는 것과 연결해야 합니다. 기본적으로 아마존 Fraud Detector 목록을 FREE_FORM_TEXT 변수 유형이라고 가정합니다.

Amazon Fraud Detector 콘솔에서 API를 사용하거나 SDK를 사용하거나AWS SDK를 사용하여 목록을 생성할 수 있습니다.AWS CLI

Amazon Fraud Detector 콘솔을 사용하여 목록 생성

목록을 생성하려면

1. AWS관리 콘솔을 열고 계정에 로그인합니다. Amazon Fraud Detector로 이동합니다.
2. 왼쪽 탐색 창에서 목록을 선택합니다.
3. 목록 세부 정보에서
   a. 목록 이름에 목록 이름을 입력합니다.
   b. 설명에 설명을 입력할 수도 있습니다.
   c. (선택 사항) 변수 유형에서 목록의 변수 유형을 선택합니다.

> **Important**
> 목록에 IP 주소가 포함된 경우 IP_ADDRESS를 변수 유형으로 선택해야 합니다. 변수 유형을 선택하지 않는 경우 Amazon Fraud Detector는 목록을 FREE_FORM_TEXT 변수 유형으로 간주합니다.

4. 목록 데이터 추가에서 각 줄에 한 항목씩 목록 항목을 추가합니다. 스프레드시트에서 항목을 복사하여 붙여넣을 수도 있습니다.

> **Note**
> 항목이 쉼표로 구분되지 않고 목록에서 고유한지 확인하세요. 동일한 항목을 두 개 입력하면 한 개만 추가됩니다.

5. Create(생성)를 선택합니다.

를 사용하여 목록 만들기AWS SDK for Python (Boto3)

목록 이름을 지정하여 목록을 만듭니다. 목록을 만들 때 선택적으로 설명을 제공하거나 변수 유형을 연결하거나 목록에 항목을 추가할 수 있습니다. 또는 나중에 항목이나 설명을 추가하여 목록을 업데이트할 수 있습니다. 목록을 만들 때 변수 유형을 할당하지 않은 경우 나중에 목록에 변수 유형을 할당할 수 있습니다. 목록의 변수 유형은 할당된 후에는 변경할 수 없습니다.

> **Important**
>
> 목록에 IP 주소가 포함된 경우 IP_ADDRESS를 변수 유형으로 지정해야 합니다. 변수 유형을 할당하지 않는 경우 Amazon Fraud Detector는 목록을 FREE_FORM_TEXT 변수 유형으로 가정합니다.

다음 예제에서는 CreateListAPI 작업을 사용하여 설명, 변수 유형을 제공하고 네 개의 목록 항목을 추가하여 목록을 만듭니다.`allow_email_ids`

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.create_list (
     name = 'allow_email_ids',
     description = 'legitimate email_ids'
     variableType = 'EMAIL_ADDRESS',
     elements = ['emailId _1', 'emailId_2', 'emailId_3','emailId_4']
     )
```

목록에 항목 추가

목록을 만든 후 언제든지 목록에 항목을 추가하거나 추가할 수 있습니다. 목록에 항목을 추가하거나 추가할 경우, 목록에 연결된 규칙을 업데이트할 필요가 없습니다. 규칙은 새로 추가된 항목을 자동으로 통합합니다.

목록에는 최대 100,000개의 고유한 항목이 포함될 수 있으며 각 항목은 최대 320자까지 입력할 수 있습니다.

Amazon Fraud Detector 콘솔에서 API를 사용하거나 SDK를 사용하거나AWS SDK를 사용하여 항목을 추가할 수 있습니다.AWS CLI

Amazon Fraud Detector 콘솔을 사용하여 목록에 항목 추가

목록에 하나 이상의 항목을 추가하려면

1. AWS관리 콘솔을 열고 계정에 로그인합니다. Amazon Fraud Detector로 이동합니다.
2. 왼쪽 탐색 창에서 목록을 선택합니다.

3. 목록 페이지에서 항목을 추가할 목록을 선택합니다.
4. 목록 세부정보 페이지에서 목록 데이터 탭을 선택하고 데이터 추가를 선택합니다.
5. 목록 데이터 추가 상자에서 각 줄에 항목을 하나씩 추가하거나 스프레드시트에서 항목을 복사하여 붙여넣습니다. 항목을 구분할 때 쉼표를 사용하지 마십시오.
6. 추가(Add)를 선택합니다.

를 사용하여 목록에 항목 추가AWS SDK for Python (Boto3)

다음 예제에서는 UpdateListAPI 작업을 사용하여`allow_email_ids` 목록에 새 항목 두 개를 추가합니다. 추가하려는 항목이 목록에서 고유한지 확인하십시오.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.update_list (
     name = 'allow_email_ids',
     updateMode = 'APPEND'
     elements = ['emailId_11','emailId_12']
```

목록에 변수 유형 지정

규칙에 사용하는 모든 목록은 Amazon Fraud Detector의변수 유형 변수 유형과 연결되어야 합니다. 기본적으로 아마존 Fraud Detector 목록을 FREE_FORM_TEXT 변수 유형이라고 가정합니다. IP 주소로 구성된 목록은 IP_ADDRESS 변수 유형과 연결되어야 한다는 점에 유의해야 합니다.

목록을 만들 때 또는 나중에 언제든지 목록을 변수 유형과 연결할 수 있습니다. 목록을 이미 변수 유형과 연결했는데 나중에 변경하려는 경우 새 목록을 만들어야 합니다. 목록의 변수 유형은 변경할 수 없습니다.

Amazon Fraud Detector 콘솔에서 API를 사용하거나AWS SDK를 사용하여 변수 유형을 할당할 수 있습니다.AWS CLI

Amazon Fraud Detector 콘솔을 사용하여 목록에 변수 유형 할당

목록에 변수 유형을 지정하려면

1. AWS관리 콘솔을 열고 계정에 로그인합니다. Amazon Fraud Detector로 이동합니다.

2. 왼쪽 탐색 창에서 목록을 선택합니다.
3. 목록 페이지에서 변수 유형을 할당하려는 목록을 선택합니다.
4. 목록 세부정보 페이지에서 작업을 선택하고 목록 편집을 선택합니다.
5. 편집 목록 상자에서 목록의 변수 유형을 선택합니다.
6. 저장을 선택합니다.

를 사용하여 목록에 변수 유형 지정AWS SDK for Python (Boto3)

다음 예제에서는 UpdateListAPI 작업을 사용하여`allow_ip_address` 목록에 변수 유형을 할당합니다.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.update_list (
     name = 'allow_ip_address',
     variableType = 'IP_ADDRESS'
)
```

목록 삭제

어떤 규칙에도 사용되지 않는 목록을 삭제할 수 있습니다. 목록을 삭제하면 Amazon Fraud Detector는 해당 목록과 목록의 모든 항목을 영구적으로 삭제합니다.

Amazon Fraud Detector 콘솔에서 API를 사용하거나AWS CLI 또는AWS SDK를 사용하여 목록을 삭제할 수 있습니다.

Amazon Fraud Detector 콘솔을 사용하여 목록 삭제

목록을 삭제하려면

1. AWS관리 콘솔을 열고 계정에 로그인합니다. Amazon Fraud Detector로 이동합니다.
2. 왼쪽 탐색 창에서 목록을 선택합니다.
3. 목록 페이지에서 삭제하려는 목록을 선택합니다.
4. 목록 세부정보 페이지에서 작업을 선택하고 목록 삭제를 선택합니다.
5. 목록 삭제를 선택합니다.

를 사용하여 목록 삭제AWS SDK for Python (Boto3)

다음 예에서는 DeleteListAPI 작업을 사용하여allow_email_ids 삭제합니다.

```
import boto3
                        fraudDetector = boto3.client('frauddetector')
fraudDetector.delete_list(
   name = 'allow_email_ids'
)
```

목록에서 항목 삭제

목록에서 하나 이상의 항목을 언제든지 삭제할 수 있습니다. 목록에서 항목을 삭제하면 해당 목록과 관련된 규칙을 업데이트할 필요가 없습니다. 규칙에는 업데이트된 목록이 자동으로 통합됩니다.

Amazon Fraud Detector 콘솔에서 API를 사용하거나AWS CLI 또는AWS SDK를 사용하여 목록에서 항목을 삭제할 수 있습니다.

Amazon Fraud Detector 콘솔을 사용하여 목록에서 항목 삭제

목록에서 하나 이상의 항목을 삭제하려면

1. AWS관리 콘솔을 열고 계정에 로그인합니다. Amazon Fraud Detector로 이동합니다.
2. 왼쪽 탐색 창에서 목록을 선택합니다.
3. 목록 페이지에서 삭제하려는 항목이 있는 목록을 선택합니다.
4. 목록 세부정보 페이지에서 목록 데이터 탭을 선택하고 삭제하려는 항목을 선택합니다.
5. 삭제를 선택하고 삭제를 다시 선택하여 확인합니다.

를 사용하여 목록에서 항목을 삭제합니다.AWS SDK for Python (Boto3)

다음 예제에서 UpdateListAPI 작업은allow_email_ids 목록에서 항목을 삭제합니다.

```
import boto3
                        fraudDetector = boto3.client('frauddetector')
fraudDetector.update_list(
   name = 'allow_email_ids',
   updateMode = 'REMOVE',
   elements = ['emailId_4', 'emailId_12']
```

```
)
```

목록에서 모든 항목 삭제

목록이 규칙에서 사용되고 있지 않은 경우 목록의 모든 항목을 삭제할 수 있습니다. 목록에 있는 모든 항목을 삭제하고 나중에 동일한 목록에 항목을 추가할 수 있습니다.

Amazon Fraud Detector 콘솔에서 API를 사용하거나AWS CLI 또는AWS SDK를 사용하여 목록에서 항목을 삭제할 수 있습니다.

Amazon Fraud Detector 콘솔을 사용하여 목록에서 모든 항목을 삭제합니다.

목록에서 모든 항목을 삭제하려면

1. AWS관리 콘솔을 열고 계정에 로그인합니다. Amazon Fraud Detector로 이동합니다.
2. 왼쪽 탐색 창에서 목록을 선택합니다.
3. 목록 페이지에서 삭제하려는 항목이 있는 목록을 선택합니다.
4. 목록 세부정보 페이지에서 목록 데이터 탭을 선택하고 모두 삭제를 선택합니다.
5. 모두 삭제 상자에`delete all` 입력하여 확인한 다음 모든 목록 데이터 삭제를 선택합니다.

를 사용하여 목록에서 모든 항목을 삭제합니다.AWS SDK for Python (Boto3)

다음 예제에서 UpdateListAPI 작업은`allow_email_ids` 목록에서 모든 항목을 삭제합니다.

```
import boto3
                        fraudDetector = boto3.client('frauddetector')
fraudDetector.update_list(
   name = 'allow_email_ids',
   updateMode = 'REPLACE',
   elements = []
)
```

결과

결과는 사기 예측의 결과입니다. 가능한 각 사기 예측 결과에 대한 결과를 생성할 수 있습니다. 예를 들어 결과가 위험 수준 (고위험, 중간 위험, 낮은_위험) 또는 조치 (승인, 검토) 를 나타내기를 원할 수 있

습니다. 결과를 만든 후 하나 이상의 결과를 추가할 수 있습니다. GetEventPrediction응답의 일환으로 Amazon Fraud Detector는 일치하는 모든 규칙에 대해 정의된 결과를 반환합니다.

결과 만들기

Amazon Fraud Detector 콘솔에서 put-result 명령, PutOutcomeAPI 또는 를 사용하여 결과를 생성할 수 AWS SDK for Python (Boto3) 있습니다.

Amazon Fraud Detector 콘솔을 사용하여 결과 생성

하나 이상의 결과를 만든 후

1. AWS관리 콘솔을 열고 계정에 로그인합니다. Amazon Fraud Detector로 이동합니다.
2. 왼쪽 탐색 창에서 결과를 선택합니다.
3. 결과 페이지에서 생성을 선택합니다.
4. 새 결과 페이지에서 다음을 입력합니다.
    a. 결과 이름에 결과 이름을 입력합니다.
    b. 결과 설명에 설명을 입력합니다.
5. 결과 저장을 선택합니다.
6. 추가 결과를 생성하려면 2~5단계를 반복합니다.

를 사용하여 결과 생성AWS SDK for Python (Boto3)

다음 예제에서는PutOutcome API를 사용하여 세 가지 결과를 생성합니다. 그들 은verify_customerreview, 및approve. 결과를 생성한 후 규칙에 결과를 할당할 수 있습니다.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.put_outcome(
name = 'verify_customer',
description = 'this outcome initiates a verification workflow'
)

fraudDetector.put_outcome(
name = 'review',
description = 'this outcome sidelines event for review'
)
```

```
fraudDetector.put_outcome(
name = 'approve',
description = 'this outcome approves the event'
)
```

결과 삭제

규칙 버전에 사용되는 결과는 삭제할 수 없습니다.

결과를 삭제하면 Amazon Fraud Detector는 해당 결과를 영구적으로 삭제하며 데이터는 더 이상 Amazon Fraud Detector에 저장되지 않습니다.

Amazon Fraud Detector 콘솔에서 delete-result 명령을 사용하거나 DeleteOutcomeAPI를 사용하거나 다음을 사용하여 결과를 삭제할 수 있습니다.AWS SDK for Python (Boto3)

Amazon Fraud Detector 콘솔에서 결과 삭제

결과를 삭제하려면

1. 에 로그인한 후 https://console.aws.amazon.com/frauddetector 에서 Amazon Fraud Detector 콘솔을 엽니다.AWS Management Console
2. Amazon Fraud Detector 콘솔의 왼쪽 탐색 창에서 리소스를 선택한 다음 결과를 선택합니다.
3. 삭제하고 싶은 결과를 선택합니다.
4. [Actions]를 선택한 후 [Delete]를 선택합니다.
5. 결과 이름을 입력한 다음 결과 삭제를 선택합니다.

를 사용하여 결과 삭제AWS SDK for Python (Boto3)

다음 예제에서는 DeleteOutcomeAPI를 사용하여`verify_customer` 결과를 삭제합니다. 결과를 삭제한 후에는 더 이상 규칙에 결과를 할당할 수 없습니다.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.delete_outcome(
name = 'verify_customer'
)
```

엔터티

엔티티는 이벤트를 수행하는 사람 또는 사물을 나타냅니다. 엔터티 유형에 따라 엔터티가 분류됩니다. 분류의 예로는 고객, 판매자, 사용자 또는 계정이 있습니다. 이벤트 데이터세트의 일부로 엔티티 유형 (ENTITY_TYPE) 과 엔티티 식별자 (ENTITY_ID) 를 제공하여 이벤트를 수행한 특정 엔티티를 나타냅니다.

Amazon Fraud Detector는 이벤트에 대한 사기 예측을 생성할 때 엔티티 유형을 사용하여 누가 이벤트를 수행했는지 표시합니다. 사기 예측에 사용하려는 개체 유형은 먼저 Amazon Fraud Detector에서 생성한 다음 이벤트 유형을 생성할 때 이벤트에 추가해야 합니다.

엔터티

Amazon Fraud Detector 콘솔에서 put-entity-type명령을 사용하거나 PutEntityTypeAPI를 사용하거나를 사용하여 엔티티 유형을 생성할 수AWS SDK for Python (Boto3) 있습니다. 아래`customer` 예제에서는 Amazon SDK for Python (Boto3) 를 사용해 엔터티 사기 탐지 모델을 학습하기 위해 이벤트 유형과 연결할 개체 유형을 만들려면 사용 사례에 적합한 이벤트 데이터세트의 개체 유형을 사용하십시오.

Amazon Fraud Detector 콘솔을 사용하여 개체 유형을 생성합니다.

엔터티

1. AWS관리 콘솔을 열고 계정에 로그인합니다.
2. Amazon Fraud Detector로 이동하여 왼쪽 내비게이션에서 엔티티를 선택한 다음 생성을 선택합니다.
3. 엔티티 생성 페이지에서 고객을 엔티티 유형 이름으로 입력합니다. 엔터티 설명 (선택 사항)
4. 엔티티 생성을 선택합니다.

를 사용하여 엔티티 유형을 생성합니다.AWS SDK for Python (Boto3)

다음AWS SDK for Python (Boto3) 코드 예제는`PutEntityType` API를 사용하여 엔티티 유형을 만듭니다`customer`. 사기 탐지 모델을 학습하기 위해 이벤트 유형과 연결할 개체 유형을 만들려면 해당 사용 사례에 적합한 이벤트 데이터세트의 엔터티를 사용하십시오.

```
import boto3
fraudDetector = boto3.client('frauddetector')
```

```
fraudDetector.put_entity_type(
name = 'customer',
description = 'customer'
)
```

엔터티 유형 삭제

Amazon Fra티 Dethon티 (Fra티 티 티) 에서는 이벤트 유형에 포함된 엔터티 먼저 엔티티가 연결된 이벤트 유형을 삭제한 다음 엔티티 유형을 삭제해야 합니다.

엔티티 유형을 삭제하면 Amazon Fraud Detector는 해당 엔티티 유형을 영구적으로 삭제하며 데이터는 더 이상 Amazon Fraud Detector에 저장되지 않습니다.

개체 유형은 Amazon Fraud Detector 콘솔에서 delete-entity-type명령을 사용하거나 DeleteEntityTypeAPI를 사용하거나 다음을 사용하여 삭제할 수 있습니다.AWS SDK for Python (Boto3)

Amazon Fraud Detector 콘솔에서 개체 유형을 삭제합니다.

엔터티

1. 에 로그인한 후 https://console.aws.amazon.com/frauddetector 에서 Amazon Python용 탐지 콘솔을 엽니다.AWS Management Console
2. Amazon Fraud Detector 콘솔의 왼쪽 탐색 창에서 리소스를 선택한 다음 엔티티를 선택합니다.
3. 삭제티(티) 를 선택합니다.
4. [Actions]를 선택한 후 [Delete]를 선택합니다.
5. 엔티티 유형 이름을 입력한 다음 엔티티 유형 삭제를 선택합니다.

를 사용하여 엔티티 유형을 삭제합니다.AWS SDK for Python (Boto3)

다음AWS SDK for Python (Boto3) 예제 코드는 DeleteEntityTypeAPI를 사용하여 엔티티 유형 고객을 삭제합니다.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.delete_entity_type (
```

```
name = 'customer'

)
```

다음을 사용하여 Amazon Fraud Detector 리소스를 관리합니다.AWS CloudFormation

Amazon FraFraFraud DeDeDeDe에서 리소스 및 인프라를 생성하고 관리하는 데 소요되는 시간을 줄일 수 있도록 Amazon Fraud DeDeDe에서 이러한 서비스를 모델링하고 설정하는 데 도움이 되는 서비스인 Amazon Fraud DeDeDeDeDeDeDeDeDeDe에서 이러한 서비스를 제공하는 서비스인 Amazon FraDeDeDeDeDeDeDeDeDeDeDeAWS CloudFormation 필요한 모든 Amazon Fraud De에서 이러한 리소스를 프로비저닝하고 구성하는 템플릿을 생성하면 에서 이러한 리소스를AWS CloudFormation 프로비저닝하고 구성합니다. EntityType EventType 템플릿을 재사용하면 여러 AWS 계정 및 리전에서 리소스를 일관되게 반복적으로 프로비저닝하고 구성할 수 있습니다.

AWS 사용에 따르는 추가 요금은 없습니다 CloudFormation.

Amazon Fraud DeAmazon Fraud DeA

Amazon Fraud DeDeDe에 대한 리소스를 프로비저닝하고 구성하려면 템플릿 을 이해하려면 AWS CloudFormation템플릿 을 이해해야 합니다. 템플릿은 JSON 또는 YAML로 서식 지정된 텍스트 파일입니다. 이 템플릿은 AWS CloudFormation 스택에서 프로비저닝할 리소스에 대해 설명합니다. JSON 또는 YAML에 익숙하지 않은 경우 AWS CloudFormation Designer를 사용하면 AWS CloudFormation 템플릿을 시작하는 데 도움이 됩니다. 자세한 내용은 AWS CloudFormation 사용 설명서에서 AWS CloudFormation Designer이란 무엇입니까?를 참조하세요.

또한AWS CloudFormation 템플릿을 사용하여 Amazon Fraud DeDe에서 Fraud DeDe 리소스에 대한 JSON 및 YAML 템플릿의 예를 비롯한 자세한 내용은 AWS CloudFormation사용 설명서 에서 Amazon FraFraud De에서 Amazon Fraud De에서 Amazon Fraud DeDe에서 Amazon Fraud DeDe에서 Amazon Fraud DeDeDe에서 Amazon Fraud DeDe

이미 사용 CloudFormation 중인 경우 추가 IAM 정책이나 CloudTrail 로깅을 관리할 필요가 없습니다.

Amazon FraFraud De에서 Fraud De이

CloudFormation 콘솔 또는 AWS CLI를 통해 Amazon Fraud Detector 스택을 생성, 업데이트 및 삭제할 수 있습니다.

스택을 생성하려면 AWS에서 스택에 포함할 리소스를 설명하는 CloudFormation 템플릿이 있어야 합니다. 또한 이미 생성한 Amazon Fraud Detector 리소스를 새 스택 또는 기존 스택으로 가져와서 CloudFormation 관리용으로 가져올 수 있습니다.

스택 관리에 대한 자세한 지침은 AWS CloudFormation사용 설명서에서 스택 생성, 업데이트 및 삭제 방법을 참조하십시오.

Amazon FraFraud De에서 FraFraud De이

AWS CloudFormation스택을 정리하는 방법은 전적으로 귀하에게 달려 있습니다. 일반적으로 수명 주기와 소유권별로 스택을 구성하는 것이 가장 좋습니다. 즉, 리소스 변경 빈도 또는 리소스 업데이트를 담당하는 팀별로 리소스를 그룹화합니다.

각 검출기 및 탐지 로직 (예: 규칙, 변수 등) 에 대한 스택을 생성하여 스택을 구성하도록 선택할 수 있습니다. 다른 서비스를 사용하는 경우 Amazon Fraud Detector 리소스를 다른 서비스의 리소스와 함께 사용할지 여부를 고려해야 합니다. 예를 들어 데이터를 수집하는 데 도움이 되는 Kinesis 리소스와 데이터를 처리하는 Amazon Fraud Detector 리소스가 포함된 스택을 만들 수 있습니다. 이는 사기 팀의 모든 제품이 함께 작동하도록 하는 효과적인 방법이 될 수 있습니다.

Amazon Fraud De에서 Fraud Fraud CloudFormation Detector

Amazon Fraud Detector에는 모든 CloudFormation 템플릿에서 사용할 수 있는 표준 파라미터 외에도 배포 동작을 관리하는 데 도움이 되는 두 가지 추가 파라미터가 도입되었습니다. 이러한 매개 변수 중 하나 또는 둘 다를 포함하지 CloudFormation 않으면 아래 표시된 기본값이 사용됩니다.

파라미터	값	기본값
DetectorVersionStatus	ACTIVE: 새로운/업데이트된 감지기 버전을 활성 상태로 설정합니다. 초안: 새로운/업데이트된 감지기 버전을 초안 상태로 설정합니다.	초안
인라인	TRUE: 스택 생성/업데이트/삭제 시 리소스를 생성/업데이트/삭제할 수 CloudFormation 있습니다. FALSE: 객체가 CloudFormation 존재하는지 확인할 수 있지만 객체를 변경할 수는 없습니다.	TRUE

Amazon FraFraud De에 대한 샘플AWS CloudFormation 템플릿

다음은 감에 대한 샘플 YAML 템플릿과 같이 필요한 샘플AWS CloudFormation YAML 템플릿과 같이 필요한 샘플 YAML 템플릿과 같이 필요한 샘플 YAML 템플릿과 같이

```
# Simple Detector resource containing inline Rule, EventType, Variable, EntityType and
 Label resource definitions
Resources:
  TestDetectorLogicalId:
    Type: AWS::FraudDetector::Detector
    Properties:
      DetectorId: "sample_cfn_created_detector"
      DetectorVersionStatus: "DRAFT"
      Description: "A detector defined and created in a CloudFormation stack!"

      Rules:
        - RuleId: "over_threshold_investigate"
          Description: "Automatically sends transactions of $10000 or more to an
 investigation queue"
          DetectorId: "sample_cfn_created_detector"
          Expression: "$amount >= 10000"
          Language: "DETECTORPL"
          Outcomes:
            - Name: "investigate"
              Inline: true
        - RuleId: "under_threshold_approve"
          Description: "Automatically approves transactions of less than $10000"
          DetectorId: "sample_cfn_created_detector"
          Expression: "$amount <10000"
          Language: "DETECTORPL"
          Outcomes:
            - Name: "approve"
              Inline: true
      EventType:
        Inline: "true"
        Name: "online_transaction"
        EventVariables:
          - Name: "amount"
            DataSource: 'EVENT'
            DataType: 'FLOAT'
            DefaultValue: '0'
            VariableType: "PRICE"
            Inline: 'true'
```

```
        EntityTypes:
          - Name: "customer"
            Inline: 'true'
        Labels:
          - Name: "legitimate"
            Inline: 'true'
          - Name: "fraudulent"
            Inline: 'true'
```

AWS CloudFormation에 대해 자세히 알아보기

AWS CloudFormation에 대한 자세한 내용은 다음 리소스를 참조하세요.

- AWS CloudFormation
- AWS CloudFormation 사용 설명서
- AWS CloudFormation API 참조
- AWS CloudFormation 명령줄 인터페이스 사용 설명서

사기 예측

Amazon Fraud Detector를 사용하여 단일 이벤트에 대한 부정 행위를 실시간으로 예측하거나 일련의 이벤트에 대한 부정 행위를 오프라인으로 예측할 수 있습니다. 단일 이벤트 또는 일련의 이벤트에 대한 부정 행위 예측을 생성하려면 Amazon Fraud Detector에 다음 정보를 제공해야 합니다.

- 사기 예측 로직
- 이벤트 메타데이터

사기 탐지 로직

사기 예측 로직은 하나 이상의 규칙을 사용하여 이벤트와 관련된 데이터를 평가한 다음 결과 및 사기 예측 점수를 제공합니다. 다음 구성 요소를 사용하여 사기 예측 로직을 생성합니다.

- 이벤트 유형 - 이벤트의 구조를 정의합니다.
- 모델 - 사기 예측을 위한 알고리즘 및 데이터 요구 사항을 정의합니다.
- 변수 - 이벤트와 관련된 데이터 요소를 나타냅니다.
- 규칙 - 사기 예측 중 변수 값을 해석하는 방법을 Amazon Fraud Detector에 알려줌
- 결과 - 사기 예측을 통해 생성된 결과
- 탐지기 버전 - 특정 이벤트에 대한 사기 예측 로직 포함

사기 탐지 로직을 생성하는 데 사용되는 구성 요소에 대한 자세한 내용은 Amazon Fraud Detector 개념을 참조하십시오. 사기 예측 생성을 시작하기 전에 사기 예측 로직이 포함된 탐지기 버전을 만들어 게시했는지 확인하세요. Fraud Detector 탐지 콘솔 또는 API를 사용하여 탐지기 버전을 만들고 게시할 수 있습니다. 콘솔 사용에 대한 지침은 시작하기 (콘솔) 를 참조하십시오. API 사용에 대한 지침은 탐지기 버전 생성을 참조하십시오.

이벤트 메타데이터

이벤트 메타데이터는 평가 중인 이벤트의 세부 정보를 제공합니다. 평가하려는 각 이벤트에는 탐지기 버전과 관련된 이벤트 유형의 각 변수 값이 포함되어야 합니다. 또한 이벤트 메타데이터에는 다음이 포함되어야 합니다.

- EVENT_ID — 이벤트의 식별자입니다. 예를 들어, 이벤트가 온라인 거래인 경우 EVENT_ID는 고객에게 제공되는 거래 참조 번호일 수 있습니다.

  EVENT_ID에 대한 중요 참고 사항

  - 해당 이벤트에서만 사용할 수 있는 고유한 것이어야 합니다.
  - 비즈니스에 의미 있는 정보를 나타내야 합니다.
  - 정규 표현식 패턴을 충족해야 합니다.`^[0-9a-z_-]+$`.
  - 반드시 저장해야 합니다. EVENT_ID는 이벤트에 대한 참조이며 이벤트 삭제와 같은 이벤트 작업을 수행하는 데 사용됩니다.
  - EVENT_ID에 타임스탬프를 추가하는 것은 나중에 이벤트를 업데이트하려고 할 때 문제가 발생할 수 있으므로 권장하지 않습니다. 정확히 동일한 EVENT_ID를 제공해야 하기 때문입니다.
- ENTITY_TYPE — 판매자 또는 고객과 같이 이벤트를 수행하는 엔티티입니다.
- ENTITY_ID - 이벤트를 수행하는 개체의 식별자입니다. ENTITY_ID는 다음 정규 표현식 패턴을 충족해야`^[0-9a-z_-]+$` 합니다. 평가 시 ENTITY_ID를 사용할 수 없는 경우 unknown이라는 문자열을 전달하십시오.
- EVENT_타임스탬프 - 이벤트가 발생한 시간입니다. 타임스탬프는 UTC 기준 ISO 8601 표준이어야 합니다.

실시간 예측

`GetEventPrediction`API를 호출하여 사기 행위에 대한 온라인 활동을 실시간으로 평가할 수 있습니다. 각 요청에서 단일 이벤트에 대한 정보를 제공하고 지정된 탐지기와 관련된 사기 예측 로직을 기반으로 모델 점수 및 결과를 동기적으로 수신합니다.

실시간 사기 예측의 작동 방식

`GetEventPrediction`API는 지정된 감지기 버전을 사용하여 이벤트에 제공된 이벤트 메타데이터를 평가합니다. 평가 중에 Amazon Fraud Detector는 먼저 탐지기 버전에 추가된 모델에 대한 모델 점수를 생성한 다음 그 결과를 평가 규칙에 전달합니다. 규칙은 규칙 실행 모드에서 지정한 대로 실행됩니다 (탐지기 버전 만들기 참조). 응답의 일환으로 Amazon Fraud Detector는 모델 점수와 일치하는 규칙과 관련된 모든 결과를 제공합니다.

실시간 사기 예측하기

실시간 부정 행위를 예측하려면 사기 예측 모델 및 규칙 또는 단순한 규칙 세트가 포함된 탐지기를 만들어 게시했는지 확인하세요.

AWS명령줄 인터페이스 (AWSCLI) 또는 Amazon Fraud Detector SDK 중 하나를 사용하여 GetEventPredictionAPI 작업을 호출하여 이벤트에 대한 부정 행위를 실시간으로 예측할 수 있습니다.

API를 사용하려면 각 요청과 함께 단일 이벤트의 정보를 제공하십시오. 요청의 일부로 Amazon FrauddetectorId Detector가 이벤트를 평가하는 데 사용할 항목을 지정해야 합니다. 필요한 경우 a를 지정할 수 있습니다detectorVersionId. detectorVersionIda가 지정되지 않은 경우 Amazon Fraud Detector는 해당ACTIVE 버전의 탐지기를 사용합니다.

선택적으로 필드에 데이터를 전달하여 SageMaker 모델을 호출할 데이터를 보낼 수externalModelEndpointBlobs 있습니다.

다음을 사용하여 부정 행위를 예측하세요AWS SDK for Python (Boto3)

사기 예측을 생성하려면GetEventPrediction API를 호출하세요. 아래 예제는 완료했다고파트 B: 사기 예측 생성 가정합니다. 응답의 일부로 모델 점수와 일치하는 규칙 및 해당 결과를 받게 됩니다. aws-fraud-detector-samples GitHub 저장소에서GetEventPrediction 요청의 추가 예를 찾을 수 있습니다.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.get_event_prediction(
detectorId = 'sample_detector',
eventId = '802454d3-f7d8-482d-97e8-c4b6db9a0428',
eventTypeName = 'sample_registration',
eventTimestamp = '2020-07-13T23:18:21Z',
entities = [{'entityType':'sample_customer', 'entityId':'12345'}],
eventVariables = {
    'email_address' : 'johndoe@exampledomain.com',
    'ip_address' : '1.2.3.4'
}
)
```

Batch 예측

Amazon Fraud Detector의 배치 예측 작업을 사용하여 실시간 채점이 필요하지 않은 일련의 이벤트에 대한 예측을 얻을 수 있습니다. 예를 들어 배치 예측 작업을 만들어 오프라인으로 수행하거나 매시간 proof-of-concept, 매일 또는 매주 이벤트 위험을 소급하여 평가할 수 있습니다.

Amazon Fraud Detector 콘솔을 사용하거나 AWS 명령줄 인터페이스 (AWSCLI) 또는 Amazon Fraud Detector SDK 중 하나를 사용하여 CreateBatchPredictionJobAPI 작업을 호출하여 배치 예측 작업을 생성할 수 있습니다.

주제

- 배치 예측의 작동 방식
- 입력 및 출력 파일
- 배치 예측 가져오기
- IAM 역할에 대한 지침
- 다음을 사용하여 일괄 사기 예측을 얻을 수 있습니다. AWS SDK for Python (Boto3)

배치 예측의 작동 방식

`CreateBatchPredictionJob`API 작업은 지정된 탐지기 버전을 사용하여 Amazon S3 버킷에 있는 입력 CSV 파일에 제공된 데이터를 기반으로 예측합니다. 그러면 API가 결과 CSV 파일을 S3 버킷으로 반환합니다.

Batch 예측 작업은 작업과 동일한 방식으로 모델 점수와 예측 결과를 계산합니다. `GetEventPrediction` 마찬가지로 배치 예측 작업을 만들려면 먼저 이벤트 유형을 만들고 선택적으로 모델을 학습시킨 다음 배치 작업의 이벤트를 평가하는 감지기 버전을 만들어야 합니다. `GetEventPrediction`

배치 예측 작업에서 평가한 이벤트 위험 점수의 가격은 `GetEventPrediction` API에서 생성한 점수의 가격과 동일합니다. 자세한 내용은 Amazon Fraud Detector 요금을 참조하십시오.

배치 예측 작업은 한 번에 하나만 실행할 수 있습니다.

입력 및 출력 파일

입력 CSV 파일에는 선택한 감지기 버전과 관련된 이벤트 유형과 일치하는 헤더가 포함되어야 합니다. 입력 데이터 파일의 최대 크기는 1GB입니다. 이벤트 수는 이벤트 규모에 따라 달라집니다.

Amazon Fraud Detector는 출력 데이터를 저장할 별도의 위치를 지정하지 않는 한 입력 파일과 동일한 버킷에 출력 파일을 생성합니다. 출력 파일에는 입력 파일의 원본 데이터와 다음과 같은 추가 열이 포함됩니다.

- `MODEL_SCORES`— 선택한 감지기 버전과 관련된 각 모델의 이벤트에 대한 모델 점수를 자세히 설명합니다.
- `OUTCOMES`— 선택한 감지기 버전 및 규칙에 따라 평가된 이벤트 결과를 자세히 설명합니다.
- `STATUS`— 이벤트가 성공적으로 평가되었는지 여부를 나타냅니다. 이벤트가 성공적으로 평가되지 않은 경우 이 열에는 실패 원인 코드가 표시됩니다.
- `RULE_RESULTS`— 규칙 실행 모드를 기준으로 일치하는 모든 규칙의 목록입니다.

배치 예측 가져오기

다음 단계에서는 이미 이벤트 유형을 만들고, 해당 이벤트 유형 (선택 사항) 을 사용하여 모델을 학습시키고, 해당 이벤트 유형에 대한 탐지기 버전을 생성했다고 가정합니다.

배치 예측을 가져오려면

1. 에 로그인한 후 https://console.aws.amazon.com/frauddetector 에서 Amazon Fraud Detector 콘솔을 엽니다. AWS Management Console
2. Amazon Fraud Detector 콘솔의 왼쪽 탐색 창에서 Batch 예측을 선택한 다음 새 배치 예측을 선택합니다.
3. Job 이름에서 배치 예측 작업의 이름을 지정합니다. 이름을 지정하지 않으면 Amazon Fraud Detector가 임의로 작업 이름을 생성합니다.
4. 검출기에서 이 배치 예측에 사용할 검출기를 선택합니다.
5. 검출기 버전에서 이 배치 예측에 대한 검출기 버전을 선택합니다. 어떤 상태에서든 검출기 버전을 선택할 수 있습니다. 검출기에 `Active` 상태가 검출기 버전이 있는 경우 해당 버전이 자동으로 선택되지만 필요한 경우 이 선택을 변경할 수도 있습니다.
6. IAM 역할에서 입력 및 출력 Amazon S3 버킷에 대한 읽기 및 쓰기 액세스 권한이 있는 역할을 선택하거나 생성합니다. 자세한 정보는 IAM 역할에 대한 지침 섹션을 참조하세요.

   배치 예측을 가져오려면 `CreateBatchPredictionJob` 작업을 호출하는 IAM 역할에 입력 S3 버킷에 대한 읽기 권한과 출력 S3 버킷에 대한 쓰기 권한이 있어야 합니다. 버킷 권한에 대한 자세한 내용은 Amazon S3 사용 설명서의 사용자 정책 예를 참조하십시오.
7. 입력 데이터 위치에서 입력 데이터의 Amazon S3 위치를 지정합니다. 출력 파일을 다른 S3 버킷에 넣으려면 출력을 위한 별도의 데이터 위치를 선택하고 출력 데이터를 위한 Amazon S3 위치를 제공하십시오.
8. (선택 사항) 배치 예측 작업에 사용할 태그를 생성합니다.
9. 시작을 선택합니다.

   Amazon Fraud Detector는 배치 예측 작업을 생성하며 작업 상태는 입니다`In progress`. Batch 예측 작업 처리 시간은 이벤트 수와 감지기 버전 구성에 따라 달라집니다.

진행 중인 배치 예측 작업을 중지하려면 [배치 예측 작업 세부 정보] 페이지로 이동하여 [작업] 을 선택한 다음 [배치 예측 중지] 를 선택합니다. 배치 예측 작업을 중지하면 해당 작업에 대한 결과를 받을 수 없습니다.

배치 예측 작업의 상태가 로 Complete 변경되면 지정된 출력 Amazon S3 버킷에서 작업 출력을 검색할 수 있습니다. 출력 파일의 이름은 다음과 같은 형식입니다batch prediction job name_file creation timestamp_output.csv. 예를 들어, 라는 작업의 출력 파일은 mybatchjob 입니다mybatchjob_ 1611170650_output.csv.

배치 예측 작업으로 평가된 특정 이벤트를 검색하려면 Amazon Fraud Detector 콘솔의 왼쪽 탐색 창에서 과거 예측 검색을 선택합니다.

완료된 배치 예측 작업을 삭제하려면 배치 예측 작업 세부 정보 페이지로 이동하고 작업을 선택한 다음 배치 예측 삭제를 선택합니다.

IAM 역할에 대한 지침

배치 예측을 가져오려면 CreateBatchPredictionJob작업을 호출하는 IAM 역할에 입력 S3 버킷에 대한 읽기 권한과 출력 S3 버킷에 대한 쓰기 권한이 있어야 합니다. 버킷 권한에 대한 자세한 내용은 Amazon S3 사용 설명서에서 사용자 정책 예제를 참조하세요. Amazon Fraud Detector 콘솔에서는 Batch 예측을 위한 IAM 역할을 선택할 수 있는 세 가지 옵션이 있습니다.

1. 새 Batch 예측 작업을 생성할 때 역할을 생성하십시오.
2. Amazon Fraud Detector 콘솔에서 이전에 생성한 기존 IAM 역할을 선택합니다. 이 단계를 수행하기 전에 역할에 `S3:PutObject` 권한을 추가해야 합니다.
3. 이전에 생성한 IAM 역할에 대한 사용자 지정 ARN을 입력합니다.

IAM 역할과 관련된 오류가 발생하면 다음을 확인합니다.

1. Amazon S3 입력 및 출력 버킷은 감지기와 동일한 리전에 있습니다.
2. 사용 중인 IAM 역할에는 입력 S3 버킷에 대한 `s3:GetObject` 권한과 출력 S3 버킷에 대한 `s3:PutObject` 권한이 있습니다.
3. 사용 중인 IAM 역할에는 서비스 `frauddetector.amazonaws.com` 주체에 대한 신뢰 정책이 있습니다.

다음을 사용하여 일괄 사기 예측을 얻을 수 있습니다. AWS SDK for Python (Boto3)

다음 예제에서는 CreateBatchPredictionJobAPI에 대한 샘플 요청을 보여 줍니다. 배치 예측 작업에는 탐지기, 감지기 버전 및 이벤트 유형 이름과 같은 기존 리소스가 포함되어야 합니다. 다음 예제에서는

이벤트 유형sample_registration, 감지기 및 감지기 sample_detector 버전을 1 만들었다고 가정합니다.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.create_batch_prediction_job (
   jobId = 'sample_batch',
   inputPath = 's3://bucket_name/input_file_name.csv',
   outputPath = 's3://bucket_name/',
   eventTypeName = 'sample_registration',
   detectorName = 'sample_detector',
   detectorVersion = '1',
   iamRoleArn = 'arn:aws:iam::**:role/service-role/AmazonFraudDetector-DataAccessRole-
**'
)
```

예측에 대한 설명

예측 설명은 각 이벤트 변수가 모델의 사기 예측 점수에 어떤 영향을 미쳤는지에 대한 통찰력을 제공하며, 사기 예측의 일부로 자동 생성됩니다. 각 사기 예측의 위험 점수는 1에서 1000 사이입니다. 예측 설명은 규모 (0-5, 최고 5점) 및 방향 (주행 점수 상승 또는 하락) 측면에서 각 사건 변수가 위험 점수에 미치는 영향에 대한 세부 정보를 제공합니다. 다음 작업에 대한 예측 설명을 사용할 수도 있습니다.

- 이벤트가 검토를 위해 신고될 때 수동 조사 중에 주요 위험 지표를 식별하기 위함입니다.
- 잘못된 예측으로 이어지는 근본 원인 (예: 합법적인 사건의 경우 높은 위험 점수) 을 좁히기 위함입니다.
- 이벤트 데이터 전반의 사기 패턴을 분석하고 데이터세트에 편향이 있는 경우 이를 탐지하기 위함입니다.

> **Important**
>
> 예측 설명은 자동으로 생성되며 2021년 6월 30일 이후에 학습된 모델에 대해서만 사용할 수 있습니다. 2021년 6월 30일 이전에 훈련된 모델에 대한 예측 설명을 받으려면 해당 모델을 다시 학습시키십시오.

예측 설명은 모델 학습에 사용된 각 이벤트 변수에 대해 다음과 같은 값 세트를 제공합니다.

상대적 영향

변수의 크기를 기준으로 사기 예측 점수에 미치는 영향을 시각적으로 참조합니다. 상대적 영향 값은 사기 위험의 별점 (0-5, 5가 가장 높음) 및 방향 (증가/감소) 으로 구성됩니다.

- 사기 위험을 증가시킨 변수는 빨간색 별표로 표시됩니다. 빨간색 별의 수가 많을수록 변수가 더 많이 사기 점수를 높이고 사기 가능성을 높입니다.
- 사기 위험을 줄인 변수는 녹색 별표로 표시됩니다. 녹색으로 표시된 시작 횟수가 많을수록 변수로 인해 사기 위험 점수가 낮아지고 사기 가능성이 낮아집니다.
- 모든 변수에 별이 0개라는 것은 변수 자체만으로는 사기 위험을 크게 변화시키지 않았음을 나타냅니다.

원시 설명 값

사기의 로그 배당률로 표현되는 해석되지 않은 원시 값을 제공합니다. 이러한 값은 일반적으로 -10에서 +10 사이이지만 범위는 -무한대에서+무한대까지입니다.

- 양수 값은 변수로 인해 위험 점수가 상승했음을 나타냅니다.
- 음수 값은 변수로 인해 위험 점수가 낮아졌음을 나타냅니다.

Amazon Fraud Detector 콘솔에는 예측 설명 값이 다음과 같이 표시됩니다. 색상이 지정된 별 등급과 해당하는 원시 수치를 통해 변수 간의 상대적 영향을 쉽게 확인할 수 있습니다.

Prediction explanations - *preview*

This prediction is based on contribution from each variable to the overall likelihood of a fraudulent event. Prediction explanations give you better understanding of how an event's input variables influence fraud prediction scores. For details on calculations, refer to documentation

Show raw prediction explanation value

Variables that increased fraud risk

Name	Value	Relative impact	Raw explanation value
comp_255	whatsapp	★★★★★	0.49
req_255	0	★☆☆☆☆	0.29
sentiment_description	0.2	★☆☆☆☆	0.12
desc_255	this is the company description	☆☆☆☆☆	0.07
title	king	☆☆☆☆☆	0.07
required_experience	5	☆☆☆☆☆	0.04
required_education	masters	☆☆☆☆☆	0.03
has_questions	true	☆☆☆☆☆	0.01

Variables that decreased fraud risk

Name	Value	Relative impact	Raw explanation value
has_company_logo	true	★★★★★	-0.26
req_desc_similarity	0.3	★☆☆☆☆	-0.21
employment_type	temp	★☆☆☆☆	-0.21
job_location	california	☆☆☆☆☆	-0.11
job_function	engineer	☆☆☆☆☆	-0.06
industry	software	☆☆☆☆☆	-0.05
sentiment_requirements	0.5	☆☆☆☆☆	-0.01
telecommuting	yes	☆☆☆☆☆	-0.00
company_desc_similarity	0.0	☆☆☆☆☆	-0.00

예측 설명 보기

사기 예측을 생성한 후 Amazon Fraud Detector 콘솔에서 예측 설명을 볼 수 있습니다. AWSSDK의 API를 사용하여 예측 설명을 보려면 먼저 API를 호출하여 이벤트에 대한 예측 타임스탬프를 가져온 다음 `ListEventPrediction` API를 호출하여 예측 설명을 가져와야 합니다. `GetEventPredictionMetadata`

Amazon Fraud Detector 콘솔을 사용하여 예측 설명 보기

콘솔을 사용하여 예측 설명을 보려면

1. AWS콘솔을 열고 계정에 로그인합니다. Amazon Fraud Detector로 이동합니다.
2. 왼쪽 탐색 창에서 과거 예측 검색을 선택합니다.
3. 속성, 연산자, 값 필터를 사용하여 검토하려는 예측을 선택합니다.
4. 상단 필터 창에서 검토하려는 예측이 생성된 기간을 선택해야 합니다.

5. 결과 창에는 지정된 기간 동안 생성된 모든 예측 목록이 표시됩니다. 예측의 이벤트 ID를 클릭하면 예측 설명을 볼 수 있습니다.
6. 아래로 스크롤하여 예측 설명 창으로 이동합니다.
7. 모든 변수의 원시 예측 설명 값을 보려면 원시 예측 설명 값 표시 버튼을 설정합니다.

파이썬용 AWS SDK (Boto3) 를 사용한 예측 설명 보기

다음 예제는 SDK의 `ListEventPredictions` 및 `GetEventPredictionMetadata` API를 사용하여 예측 설명을 보기 위한 샘플 요청을 보여줍니다. AWS

예제 1: API를 사용하여 가장 최근 예측 목록 가져오기 **`ListEventPredictions`**

```
import boto3
fraudDetector = boto3.client('frauddetector')
fraudDetector.list_event_predictions(
  maxResults = 10,
  predictionTimeRange = {
     end_time: '2022-01-13T23:18:21Z',
     start_time: '2022-01-13T20:18:21Z'
    }
 )
```

예 2. API를 사용하여 이벤트 유형 “등록”에 대한 과거 예측 목록 가져오기 **`ListEventPredictions`**

```
import boto3
fraudDetector = boto3.client('frauddetector')
fraudDetector.list_event_predictions(
   eventType = {
      value = 'registration'
    }
   maxResults = 70,
   nextToken = "10",
   predictionTimeRange = {
     end_time: '2021-07-13T23:18:21Z',
     start_time: '2021-07-13T20:18:21Z'
    }
 )
```

예 3: **`GetEventPredictionMetadata`** API를 사용하여 지정된 기간 내에 생성된 특정 이벤트 ID, 이벤트 유형, 탐지기 ID 및 탐지기 버전 ID에 대한 과거 예측의 세부 정보를 가져옵니다.

이 요청에 predictionTimestamp 지정된 정보는 먼저 ListEventPredictions API를 호출하여 가져옵니다.

```
import boto3
fraudDetector = boto3.client('frauddetector')
fraudDetector.get_event_prediction_metadata (
   detectorId = 'sample_detector',
   detectorVersionId = '1',
   eventId = '802454d3-f7d8-482d-97e8-c4b6db9a0428',
   eventTypeName = 'sample_registration',
   predictionTimestamp = '2021-07-13T21:18:21Z'
 )
```

예측 설명이 계산되는 방식에 대한 이해

Amazon Fraud Detectory는 SHAP (Shapeley 추가 설명) 를 사용하여 모델 학습에 사용되는 각 이벤트 변수의 원시 설명 값을 계산하여 개별 이벤트 예측을 설명합니다. 원시 설명 값은 예측 생성 시 분류 알고리즘의 일부로 모델에 의해 계산됩니다. 이러한 원시 설명 값은 사기 확률의 로그에 대한 각 입력의 기여도를 나타냅니다. 원시 설명 값 (-무한대에서 +무한대까지) 은 매핑을 사용하여 상대적 영향 값 (-5에서 +5) 으로 변환됩니다. 원시 설명 값에서 도출된 상대 영향값은 사기 (양수) 또는 합법적 (음수) 확률이 증가하는 횟수를 나타내므로 예측 설명을 더 쉽게 이해할 수 있습니다.

Amazon Fraud Detector의 보안

클라우드 AWS 보안이 최우선 과제입니다. AWS 고객은 가장 보안에 민감한 조직의 요구 사항을 충족하도록 구축된 데이터 센터 및 네트워크 아키텍처를 활용할 수 있습니다.

보안은 기업과 기업 간의 공동 책임입니다. AWS 공동 책임 모델은 이 사항을 클라우드의 보안 및 클라우드 내 보안으로 설명합니다.

- 클라우드 보안 - AWS 클라우드에서 AWS 서비스를 실행하는 인프라를 보호하는 역할을 합니다. AWS AWS 또한 안전하게 사용할 수 있는 서비스를 제공합니다. Amazon Fraud Detector에 적용되는 규정 준수 프로그램에 대해 자세히 알아보려면 규정 준수 프로그램별 AWS 범위 내 서비스 규정 준수 프로그램 참조하십시오.
- 클라우드에서의 보안 — 귀하의 책임은 사용하는 AWS 서비스에 따라 결정됩니다. 또한 귀하는 귀사의 데이터의 민감도, 귀사의 요구 사항, 관련 법률 및 규정을 비롯한 기타 요소에 대해서도 책임이 있습니다.

이 설명서는 Amazon Fraud Detector를 사용할 때 공동 책임 모델을 적용하는 방법을 이해하는 데 도움이 됩니다. 다음 주제에서는 보안 및 규정 준수 목표를 충족하도록 Amazon Fraud Detector를 구성하는 방법을 보여줍니다. 또한 Amazon Fraud Detector 리소스를 모니터링하고 보호하는 데 도움이 되는 다른 AWS 서비스를 사용하는 방법도 알아봅니다.

주제

- Amazon Fraud Detector에서의 데이터 보호
- Amazon Fraud Detector의 자격 증명 및 액세스 관리
- Amazon Fraud Detector에서의 로깅 및 모니터링
- Amazon Fraud Detector에 대한 규정 준수 검증
- Amazon Fraud Detector의 복원력
- Amazon Fraud Detector의 인프라 보안

Amazon Fraud Detector에서의 데이터 보호

AWS 공동 책임 모델 공동 책임 모델 Amazon Fraud Detector의 데이터 보호에 적용됩니다. 이 모델에 설명된 대로 AWS 은 모든 모델을 실행하는 글로벌 인프라를 보호하는 역할을 AWS 클라우드합니다. 사용자는 인프라에서 호스팅되는 콘텐츠를 관리해야 합니다. 사용하는 AWS 서비스 의 보안 구성과 관

리 작업에 대한 책임도 사용자에게 있습니다. 데이터 프라이버시에 대한 자세한 내용은 데이터 프라이버시 FAQ를 참조하세요. 유럽의 데이터 보호에 대한 자세한 내용은AWS 보안 블로그에서 AWS 공동 책임 모델 및 GDPR 블로그 게시물을 참조하세요.

데이터 보호를 위해 AWS 계정 자격 증명을 보호하고 AWS IAM Identity Center OR AWS Identity and Access Management (IAM) 을 사용하여 개별 사용자를 설정하는 것이 좋습니다. 이렇게 하면 개별 사용자에게 자신의 직무를 충실히 이행하는 데 필요한 권한만 부여됩니다. 또한 다음과 같은 방법으로 데이터를 보호하는 것이 좋습니다.

- 각 계정에 멀티 팩터 인증 설정(MFA)을 사용하세요.
- SSL/TLS를 사용하여 리소스와 통신할 수 있습니다. AWS TLS 1.2는 필수이며 TLS 1.3를 권장합니다.
- 를 사용하여 API 및 사용자 활동 로깅을 설정합니다. AWS CloudTrail
- 포함된 모든 기본 보안 제어와 함께 AWS 암호화 솔루션을 사용하십시오 AWS 서비스.
- Amazon S3에 저장된 민감한 데이터를 검색하고 보호하는 데 도움이 되는 Amazon Macie와 같은 고급 관리형 보안 서비스를 사용하세요.
- 명령줄 인터페이스 또는 API를 AWS 통해 액세스할 때 FIPS 140-2로 검증된 암호화 모듈이 필요한 경우 FIPS 엔드포인트를 사용하십시오. 사용 가능한 FIPS 엔드포인트에 대한 자세한 내용은 FIPS(Federal Information Processing Standard) 140-2를 참조하세요.

고객의 이메일 주소와 같은 기밀 정보나 중요한 정보는 태그나 이름 필드와 같은 자유 양식 필드에 입력하지 않는 것이 좋습니다. 여기에는 Amazon Fraud Detector 또는 기타 콘솔 AWS CLI, API 또는 AWS SDK를 AWS 서비스 사용하여 작업하는 경우가 포함됩니다. 이름에 사용되는 태그 또는 자유 형식 텍스트 필드에 입력하는 모든 데이터는 청구 또는 진단 로그에 사용될 수 있습니다. 외부 서버에 URL을 제공할 때 해당 서버에 대한 요청을 검증하기 위해 보안 인증 정보를 URL에 포함시켜서는 안 됩니다.

저장 데이터 암호화

Amazon Fraud Detector는 사용자가 선택한 암호화 키를 사용하여 저장된 데이터를 암호화합니다. 다음 중 하나를 선택할 수 있습니다.

- AWS 소유한 KMS 키. 암호화 키를 지정하지 않으면 기본적으로 이 키로 데이터가 암호화됩니다.
- 고객이 관리하는 KMS 키. 키 정책을 사용하여 고객 관리형 KMS 키에 대한 액세스를 제어할 수 있습니다. 고객 관리형 KMS 키 생성 및 관리에 대한 자세한 내용은 을 참조하십시오. 키 관리

전송 중 데이터 암호화

Amazon Fraud Detector는 사용자 계정에서 데이터를 복사하여 내부 AWS 시스템에서 처리합니다. 기본적으로 Amazon Fraud Detector는 AWS 인증서와 함께 TLS 1.2를 사용하여 전송 데이터를 암호화합니다.

키 관리

Amazon Fraud Detector는 다음 두 가지 유형의 키 중 하나를 사용하여 데이터를 암호화합니다.

- AWS 소유한 KMS 키. 이 값이 기본값입니다.
- 고객이 관리하는 KMS 키.

고객 관리형 KMS 키 생성

KMS 콘솔 또는 API를 사용하여 고객 관리형 AWS KMS 키를 생성할 수 있습니다. CreateKey 키를 생성할 때는 다음 사항을 확인하십시오.

- 대칭 암호화 고객 관리형 KMS 키를 선택하십시오. Amazon Fraud Detector는 비대칭 KMS 키를 지원하지 않습니다. 자세한 내용은 키 관리 서비스 개발자 안내서의 비대칭 키를 참조하십시오 AWS KMS. AWS
- 단일 지역 KMS 키를 생성하십시오. Amazon Fraud Detector는 다중 지역 KMS 키를 지원하지 않습니다. 자세한 내용은 키 관리 서비스 개발자 안내서의 AWS KMS다중 지역 AWS 키를 참조하십시오.
- 다음 키 정책을 제공하여 Amazon Fraud Detector에 키 사용 권한을 부여하십시오.

```
    {
"Effect": "Allow",
"Principal": {
    "Service": "frauddetector.amazonaws.com"
},
"Action": [
    "kms:Encrypt",
    "kms:Decrypt",
    "kms:ReEncrypt*",
    "kms:GenerateDataKey*",
    "kms:DescribeKey",
    "kms:CreateGrant",
    "kms:RetireGrant"
```

```
    ],
    "Resource": "*"
}
```

주요 정책에 대한 자세한 내용은 키 관리 서비스 개발자 안내서의 AWS KMS에서의 AWS 키 정책 사용을 참조하십시오.

고객 관리형 KMS 키를 사용한 데이터 암호화

Amazon Fraud Detector의 PutKMS EncryptionKey API를 사용하면 고객 관리형 KMS 키를 사용하여 저장된 Amazon Fraud Detector 데이터를 암호화할 수 있습니다. API를 사용하여 언제든지 암호화 구성을 변경할 수 있습니다. `PutKMSEncryptionKey`

암호화된 데이터에 대한 중요 참고 사항

- 고객 관리형 KMS 키를 설정한 후 생성된 데이터는 암호화됩니다. 고객 관리형 KMS 키를 설정하기 전에 생성된 데이터는 암호화되지 않은 상태로 유지됩니다.
- 고객 관리형 KMS 키가 변경된 경우 이전 암호화 구성을 사용하여 암호화된 데이터는 다시 암호화되지 않습니다.

데이터 보기

고객 관리형 KMS 키를 사용하여 Amazon Fraud Detector 데이터를 암호화하는 경우 Amazon Fraud Detector 콘솔의 과거 예측 검색 영역에 있는 필터를 사용하여 이 방법을 사용하여 암호화된 데이터를 검색할 수 없습니다. 완전한 검색 결과를 얻으려면 다음 속성 중 하나 이상을 사용하여 결과를 필터링하십시오.

- 이벤트 ID
- 평가 타임스탬프
- 검출기 상태
- Detector 버전
- 모델 버전
- 모델 유형
- 규칙 평가 상태
- 규칙 실행 모드
- 룰 매칭 상태

- 규칙 버전
- 가변 데이터 소스

고객 관리형 KMS 키가 삭제되었거나 삭제가 예정된 경우 데이터를 사용하지 못할 수 있습니다. 자세한 내용은 KMS 키 삭제를 참조하십시오.

Amazon Fraud Detector 및 인터페이스 VPC 엔드포인트 ()AWS PrivateLink

인터페이스 VPC 엔드포인트를 생성하여 VPC와 Amazon Fraud Detector 사이에 프라이빗 연결을 설정할 수 있습니다. 인터페이스 엔드포인트는 인터넷 게이트웨이 AWS PrivateLink, NAT 디바이스, VPN 연결 또는 AWS Direct Connect 연결 없이 Amazon Fraud Detector API에 비공개로 액세스할 수 있는 기술인 에 의해 구동됩니다. VPC의 인스턴스는 Amazon Fraud Detector API와 통신하는 데 퍼블릭 IP 주소가 필요하지 않습니다. VPC와 Amazon Fraud Detector 간의 트래픽은 아마존 네트워크를 벗어나지 않습니다.

각 인터페이스 엔드포인트는 서브넷에서 하나 이상의 탄력적 네트워크 인터페이스로 표현됩니다.

자세한 내용은 Amazon VPC 사용 설명서의 인터페이스 VPC 엔드포인트 (AWS PrivateLink) 를 참조하십시오.

Amazon Fraud Detector VPC 엔드포인트에 대한 고려 사항

Amazon Fraud Detector의 인터페이스 VPC 엔드포인트를 설정하기 전에 Amazon VPC 사용 설명서의 인터페이스 엔드포인트 속성 및 제한을 검토해야 합니다.

Amazon Fraud Detector는 VPC에서 모든 API 작업을 호출할 수 있도록 지원합니다.

VPC 엔드포인트 정책은 Amazon Fraud Detector에 지원됩니다. 기본적으로 엔드포인트를 통해 Amazon Fraud Detector에 대한 전체 액세스가 허용됩니다. 자세한 정보는 VPC 사용 설명서의 VPC 엔드포인트를 통해 서비스에 대한 액세스 제어를 참조하세요.

Amazon Fraud Detector를 위한 인터페이스 VPC 엔드포인트 생성

Amazon VPC 콘솔 또는 () 를 사용하여 Amazon Fraud Detector 서비스를 위한 VPC 엔드포인트를 생성할 수 있습니다. AWS Command Line Interface AWS CLI자세한 내용은 VPC 사용 설명서의 인터페이스 엔드포인트 생성을 참조하세요.

다음 서비스 이름을 사용하여 Amazon Fraud Detector용 VPC 엔드포인트를 생성합니다.

- com.amazonaws.*region*.frauddetector

엔드포인트에 대한 프라이빗 DNS를 활성화하면 해당 지역의 기본 DNS 이름 (예:) 을 사용하여 Amazon Fraud Detector에 API 요청을 보낼 수 `frauddetector.us-east-1.amazonaws.com` 있습니다.

자세한 내용은 Amazon VPC 사용 설명서의 인터페이스 엔드포인트를 통해 서비스 액세스를 참조하세요.

Amazon Fraud Detector를 위한 VPC 엔드포인트 정책 생성

Amazon Fraud Detector의 인터페이스 VPC 엔드포인트에 대한 정책을 생성하여 다음을 지정할 수 있습니다.

- 작업을 수행할 수 있는 보안 주체.
- 수행할 수 있는 작업
- 작업을 수행할 수 있는 리소스

자세한 내용은 Amazon VPC 사용 설명서의 VPC 엔드포인트를 통해 서비스에 대한 액세스 제어를 참조하세요.

다음 예제 VPC 엔드포인트 정책은 VPC 인터페이스 엔드포인트에 액세스할 수 있는 모든 사용자가 이름이 지정된 Amazon Fraud Detector 탐지기에 액세스할 수 있도록 지정합니다. `my_detector`

```
{
  "Statement": [
      {
          "Action": "frauddetector:*Detector",
          "Effect": "Allow",
          "Resource": "arn:aws:frauddetector:us-east-1:123456789012:detector/
my_detector",
          "Principal": "*"
      }
  ]
}
```

이 예제에서 다음은 거부됩니다.

- 기타 아마존 Fraud Detector API 작업

- 아마존 Fraud Detector `GetEventPrediction` API 호출

> **Note**
>
> 이 예시에서도 사용자는 VPC 외부에서 다른 Amazon Fraud Detector API 작업을 수행할 수 있습니다. API 직접 호출을 VPC 내부의 API 직접 호출로 제한하는 방법에 대한 자세한 내용은 Amazon Fraud Detector 신원 정보 기반 정책을 참조하세요.

서비스 개선을 위한 데이터 사용 거부

모델을 학습시키고 예측을 생성하기 위해 제공하는 과거 이벤트 데이터는 서비스를 제공하고 유지하는 용도로만 사용됩니다. 이 데이터는 Amazon Fraud Detector의 품질을 개선하는 데도 사용될 수 있습니다. Amazon은 고객의 신뢰, 개인 정보 보호 및 콘텐츠 보안을 최우선으로 생각하며, Google이 사용자에 대한 약속을 준수하도록 보장합니다. 자세한 내용은 데이터 프라이버시 FAQ를 참조하십시오.

AWS Organizations User Guide의 AI 서비스 수신 거부 정책 페이지를 방문하여 여기에 설명된 절차에 따라 Amazon Fraud Detector의 개발 또는 품질 개선에 이벤트 데이터를 사용하지 않도록 선택할 수 있습니다.

> **Note**
>
> 옵트아웃 정책을 사용할 수 있으려면 AWS Organizations에서 AWS 계정을 중앙에서 관리해야 합니다. AWS 계정을 위한 조직을 아직 생성하지 않은 경우 조직 생성 및 관리 페이지를 방문하여 여기에 설명된 프로세스를 따르십시오.

Amazon Fraud Detector의 자격 증명 및 액세스 관리

AWS Identity and Access Management (IAM) 은 관리자가 리소스에 대한 액세스를 안전하게 제어할 수 AWS 서비스 있도록 AWS 도와줍니다. IAM 관리자는 Amazon Fraud Detector 리소스를 사용할 수 있는 인증 (로그인) 및 권한 부여 (권한 보유) 를 받을 수 있는 사용자를 제어합니다. IAM은 추가 AWS 서비스 비용 없이 사용할 수 있습니다.

주제

- 정책을 사용한 액세스 관리
- Amazon Fraud Detector가 IAM과 함께 작동하는 방식
- Amazon Fraud Detector ID 기반 정책 예제
- 혼동된 대리자 방지
- Amazon Fraud Detector ID 및 액세스 문제 해결

고객

Amazon Fraud Detector에서 수행하는 작업에 따라 AWS Identity and Access Management (IAM) 사용 방법이 다릅니다.

서비스 사용자 — Amazon Fraud Detector 서비스를 사용하여 업무를 수행하는 경우 관리자가 필요한 자격 증명과 권한을 제공합니다. 더 많은 Amazon Fraud Detector 기능을 사용하여 업무를 수행함에 따라 추가 권한이 필요할 수 있습니다. 액세스 권한 관리 방식을 이해하면 적절한 권한을 관리자에게 요청할 수 있습니다. Amazon Fraud Detector의 기능에 액세스할 수 없는 경우 을 참조하십시오Amazon Fraud Detector ID 및 액세스 문제 해결.

서비스 관리자 — 회사에서 Amazon Fraud Detector 리소스를 담당하고 있다면 Amazon Fraud Detector에 대한 전체 액세스 권한을 가지고 있을 것입니다. 서비스 사용자가 액세스해야 하는 Amazon Fraud Detector 기능 및 리소스를 결정하는 것은 여러분의 몫입니다. 그런 다음, IAM 관리자에게 요청을 제출하여 서비스 사용자의 권한을 변경해야 합니다. 이 페이지의 정보를 검토하여 IAM의 기본 개념을 이해합니다. 회사에서 Amazon Fraud Detector와 함께 IAM을 사용하는 방법에 대한 자세한 내용은 을 참조하십시오Amazon Fraud Detector가 IAM과 함께 작동하는 방식.

IAM 관리자 — IAM 관리자인 경우 Amazon Fraud Detector에 대한 액세스를 관리하는 정책을 작성하는 방법에 대해 자세히 알아보는 것이 좋습니다. IAM에서 사용할 수 있는 Amazon Fraud Detector ID 기반 정책의 예를 보려면 을 참조하십시오. Amazon Fraud Detector ID 기반 정책 예제

자격 증명을 통한 인증

인증은 자격 증명을 AWS 사용하여 로그인하는 방법입니다. IAM 사용자로 인증 (로그인 AWS) 하거나 IAM 역할을 맡아 인증 (로그인) 해야 합니다. AWS 계정 루트 사용자

ID 소스를 통해 제공된 자격 증명을 사용하여 페더레이션 ID로 로그인할 수 있습니다. AWS AWS IAM Identity Center (IAM ID 센터) 사용자, 회사의 싱글 사인온 인증, Google 또는 Facebook 자격 증명이 페더레이션 ID의 예입니다. 페더레이션형 ID로 로그인할 때 관리자가 이전에 IAM 역할을 사용하여 ID 페더레이션을 설정했습니다. 페더레이션을 사용하여 액세스하는 경우 AWS 간접적으로 역할을 맡게 됩니다.

사용자 유형에 따라 AWS Management Console 또는 AWS 액세스 포털에 로그인할 수 있습니다. 로그인에 대한 자세한 내용은 AWS 로그인 사용 설명서의 내 로그인 방법을 참조하십시오. AWS AWS 계정

AWS 프로그래밍 방식으로 액세스하는 경우 자격 증명을 사용하여 요청에 암호화 방식으로 서명할 수 있는 소프트웨어 개발 키트 (SDK) 와 명령줄 인터페이스 (CLI) 를 AWS 제공합니다. AWS 도구를 사용하지 않는 경우 요청에 직접 서명해야 합니다. 권장 방법을 사용하여 직접 요청에 서명하는 방법에 대한 자세한 내용은 IAM 사용 설명서의 AWS API 요청 서명을 참조하십시오.

사용하는 인증 방법에 상관없이 추가 보안 정보를 제공해야 할 수도 있습니다. 예를 들어, AWS 계정의 보안을 강화하기 위해 다단계 인증 (MFA) 을 사용할 것을 권장합니다. 자세한 내용은 AWS IAM Identity Center 사용 설명서의 다중 인증 및 IAM 사용자 설명서의 AWS에서 다중 인증(MFA) 사용을 참조합니다.

AWS 계정 루트 사용자

계정을 AWS 계정만들 때는 먼저 계정의 모든 AWS 서비스 리소스에 대한 완전한 액세스 권한을 가진 하나의 로그인 ID로 시작합니다. 이 ID를 AWS 계정 루트 사용자라고 하며, 계정을 만들 때 사용한 이메일 주소와 비밀번호로 로그인하여 액세스할 수 있습니다. 일상적인 작업에 루트 사용자를 사용하지 않을 것을 강력히 권장합니다. 루트 사용자 보안 인증 정보를 보호하고 루트 사용자만 수행할 수 있는 태스크를 수행하는 데 사용하세요. 루트 사용자로 로그인해야 하는 전체 작업 목록은 IAM 사용 설명서의 Tasks that require root user credentials를 참조하세요.

사용자 및 그룹

IAM 사용자는 단일 사용자 또는 애플리케이션에 대한 특정 권한을 가진 사용자 내의 자격 증명입니다. AWS 계정 가능하면 암호 및 액세스 키와 같은 장기 자격 증명이 있는 IAM 사용자를 생성하는 대신 임시 자격 증명을 사용하는 것이 좋습니다. 하지만 IAM 사용자의 장기 자격 증명이 필요한 특정 사용 사례가 있는 경우 액세스 키를 교체하는 것이 좋습니다. 자세한 내용은 IAM 사용 설명서의 장기 보안 인증이 필요한 사용 사례의 경우 정기적으로 액세스 키 교체를 참조하세요.

IAM 그룹은 IAM 사용자 컬렉션을 지정하는 자격 증명입니다. 사용자는 그룹으로 로그인할 수 없습니다. 그룹을 사용하여 여러 사용자의 권한을 한 번에 지정할 수 있습니다. 그룹을 사용하면 대규모 사용자 집합의 권한을 더 쉽게 관리할 수 있습니다. 예를 들어, IAMAdmins라는 그룹이 있고 이 그룹에 IAM 리소스를 관리할 권한을 부여할 수 있습니다.

사용자는 역할과 다릅니다. 사용자는 한 사람 또는 애플리케이션과 고유하게 연결되지만, 역할은 해당 역할이 필요한 사람이라면 누구나 수임할 수 있습니다. 사용자는 영구적인 장기 자격 증명을 가지고 있지만, 역할은 임시 보안 인증만 제공합니다. 자세한 정보는 IAM 사용 설명서의 IAM 사용자를 만들어야 하는 경우(역할이 아님)를 참조하세요.

IAM 역할

IAM 역할은 특정 권한을 가진 사용자 AWS 계정 내의 자격 증명입니다. IAM 사용자와 유사하지만, 특정 개인과 연결되지 않습니다. 역할을 AWS Management Console 전환하여 에서 일시적으로 IAM 역할을 맡을 수 있습니다. AWS CLI 또는 AWS API 작업을 호출하거나 사용자 지정 URL을 사용하여 역할을 수임할 수 있습니다. 역할 사용 방법에 대한 자세한 정보는 IAM 사용 설명서의 IAM 역할 사용을 참조하세요.

임시 보안 인증 정보가 있는 IAM 역할은 다음과 같은 상황에서 유용합니다.

- 페더레이션 사용자 액세스 - 페더레이션 아이덴티티에 권한을 부여하려면 역할을 생성하고 해당 역할의 권한을 정의합니다. 연동 자격 증명이 인증되면 역할이 연결되고 역할에 정의된 권한이 부여됩니다. 페더레이션 역할에 대한 자세한 내용은 IAM 사용 설명서의 타사 자격 증명 공급자의 역할 만들기를 참조하세요. IAM Identity Center를 사용하는 경우 권한 세트를 구성합니다. 인증 후 자격 증명이 액세스할 수 있는 항목을 제어하기 위해 IAM Identity Center는 권한 세트를 IAM의 역할과 연관짓습니다. 권한 세트에 대한 자세한 내용은AWS IAM Identity Center 사용 설명서의 권한 세트를 참조하세요.
- 임시 IAM 사용자 권한 - IAM 사용자 또는 역할은 IAM 역할을 수임하여 특정 태스크에 대한 다양한 권한을 임시로 받을 수 있습니다.
- 크로스 계정 액세스 - IAM 역할을 사용하여 다른 계정의 사용자(신뢰할 수 있는 보안 주체)가 내 계정의 리소스에 액세스하도록 허용할 수 있습니다. 역할은 계정 간 액세스를 부여하는 기본적인 방법입니다. 그러나 일부 AWS 서비스경우에는 역할을 프록시로 사용하는 대신 정책을 리소스에 직접 연결할 수 있습니다. 크로스 계정 액세스를 위한 역할과 리소스 기반 정책의 차이점을 알아보려면 IAM 사용 설명서의 IAM 역할과 리소스 기반 정책의 차이를 참조하세요.
- 서비스 간 액세스 — 일부는 다른 AWS 서비스서비스의 기능을 AWS 서비스 사용합니다. 예컨대, 어떤 서비스에서 호출을 수행하면 일반적으로 해당 서비스는 Amazon EC2에서 애플리케이션을 실행하거나 Amazon S3에 객체를 저장합니다. 서비스는 호출하는 보안 주체의 권한을 사용하거나, 서비스 역할을 사용하거나, 또는 서비스 연결 역할을 사용하여 이 작업을 수행할 수 있습니다.
  - 순방향 액세스 세션 (FAS) — IAM 사용자 또는 역할을 사용하여 작업을 수행하는 경우 보안 AWS 주체로 간주됩니다. 일부 서비스를 사용하는 경우 다른 서비스에서 다른 작업을 시작하는 작업을 수행할 수 있습니다. FAS는 전화를 거는 주체의 권한을 다운스트림 AWS 서비스서비스에 AWS 서비스 요청하기 위한 요청과 결합하여 사용합니다. FAS 요청은 다른 서비스 AWS 서비스 또는 리소스와의 상호 작용이 필요한 요청을 서비스가 수신한 경우에만 이루어집니다. 이 경우 두 작업을 모두 수행할 수 있는 권한이 있어야 합니다. FAS 요청 시 정책 세부 정보는 전달 액세스 세션을 참조하세요.

  - 서비스 역할 - 서비스 역할은 서비스가 사용자를 대신하여 태스크를 수행하기 위해 맡는 IAM 역할입니다. IAM 관리자는 IAM 내에서 서비스 역할을 생성, 수정 및 삭제할 수 있습니다. 자세한 정보는 IAM 사용자 설명서의 AWS 서비스에 대한 권한을 위임할 역할 생성을 참조합니다.
  - 서비스 연결 역할 — 서비스 연결 역할은 에 연결된 서비스 역할의 한 유형입니다. AWS 서비스서비스는 사용자를 대신하여 작업을 수행하기 위해 역할을 수임할 수 있습니다. 서비스 연결 역할은 사용자에게 AWS 계정 표시되며 해당 서비스가 소유합니다. IAM 관리자는 서비스 링크 역할의 권한을 볼 수 있지만 편집은 할 수 없습니다.
- Amazon EC2에서 실행되는 애플리케이션 — IAM 역할을 사용하여 EC2 인스턴스에서 실행되고 API 요청을 AWS CLI 하는 애플리케이션의 임시 자격 증명을 관리할 수 있습니다. AWS 이는 EC2 인스턴스 내에 액세스 키를 저장할 때 권장되는 방법입니다. EC2 인스턴스에 AWS 역할을 할당하고 모든 애플리케이션에서 사용할 수 있게 하려면 인스턴스에 연결된 인스턴스 프로필을 생성합니다. 인스턴스 프로파일에는 역할이 포함되어 있으며 EC2 인스턴스에서 실행되는 프로그램이 임시 보안 인증을 얻을 수 있습니다. 자세한 정보는 IAM 사용 설명서의 IAM 역할을 사용하여 Amazon EC2 인스턴스에서 실행되는 애플리케이션에 권한 부여를 참조하세요.

IAM 역할을 사용할지 또는 IAM 사용자를 사용할지를 알아보려면 IAM 사용자 설명서의 IAM 역할(사용자 대신)을 생성하는 경우를 참조합니다.

정책을 사용한 액세스 관리

정책을 생성하고 이를 AWS ID 또는 리소스에 AWS 연결하여 액세스를 제어할 수 있습니다. 정책은 ID 또는 리소스와 연결될 때 AWS 해당 권한을 정의하는 객체입니다. AWS 주도자 (사용자, 루트 사용자 또는 역할 세션) 가 요청할 때 이러한 정책을 평가합니다. 정책에서 권한은 요청이 허용되거나 거부되는 지를 결정합니다. 대부분의 정책은 JSON 문서로 AWS 저장됩니다. JSON 정책 문서의 구조와 콘텐츠에 대한 자세한 정보는 IAM 사용 설명서의 JSON 정책 개요를 참조하세요.

관리자는 AWS JSON 정책을 사용하여 누가 무엇에 액세스할 수 있는지 지정할 수 있습니다. 즉, 어떤 보안 주체가 어떤 리소스와 어떤 조건에서 작업을 수행할 수 있는지를 지정할 수 있습니다.

기본적으로, 사용자와 역할에는 어떠한 권한도 없습니다. 사용자에게 사용자가 필요한 리소스에서 작업을 수행할 권한을 부여하려면 IAM 관리자가 IAM 정책을 생성하면 됩니다. 그런 다음 관리자가 IAM 정책을 역할에 추가하고, 사용자가 역할을 수임할 수 있습니다.

IAM 정책은 작업을 수행하기 위해 사용하는 방법과 상관없이 작업에 대한 권한을 정의합니다. 예를 들어, `iam:GetRole`태스크를 허용하는 정책이 있다고 가정합니다. 해당 정책을 사용하는 사용자는 AWS Management Console, AWS CLI, 또는 AWS API에서 역할 정보를 가져올 수 있습니다.

ID 기반 정책

ID 기반 정책은 IAM 사용자, 사용자 그룹 또는 역할과 같은 자격 증명에 연결할 수 있는 JSON 권한 정책 문서입니다. 이러한 정책은 사용자와 역할이 어떤 리소스와 어떤 조건에서 어떤 작업을 수행할 수 있는 지를 제어합니다. 자격 증명 기반 정책을 생성하는 방법을 알아보려면 IAM 사용 설명서의 IAM 정책 생성을 참조하세요.

자격 증명 기반 정책은 인라인 정책 또는 관리형 정책으로 한층 더 분류할 수 있습니다. 인라인 정책은 단일 사용자, 그룹 또는 역할에 직접 포함됩니다. 관리형 정책은 내 여러 사용자, 그룹 및 역할에 연결할 수 있는 독립형 정책입니다. AWS 계정관리형 정책에는 AWS 관리형 정책과 고객 관리형 정책이 포함됩니다. 관리형 정책 또는 인라인 정책을 선택하는 방법을 알아보려면 IAM 사용 설명서의 관리형 정책과 인라인 정책의 선택을 참조하세요.

리소스 기반 정책

리소스 기반 정책은 리소스에 연결하는 JSON 정책 설명서입니다. 리소스 기반 정책의 예는 IAM 역할 신뢰 정책과 Amazon S3 버킷 정책입니다. 리소스 기반 정책을 지원하는 서비스에서 서비스 관리자는 이러한 정책을 사용하여 특정 리소스에 대한 액세스를 통제할 수 있습니다. 정책이 연결된 리소스의 경우 정책은 지정된 보안 주체가 해당 리소스와 어떤 조건에서 어떤 작업을 수행할 수 있는 지를 정의합니다. 리소스 기반 정책에서 보안 주체를 지정해야 합니다. 보안 주체에는 계정, 사용자, 역할, 연동 사용자 등이 포함될 수 있습니다. AWS 서비스

리소스 기반 정책은 해당 서비스에 있는 인라인 정책입니다. IAM의 AWS 관리형 정책은 리소스 기반 정책에 사용할 수 없습니다.

액세스 제어 목록(ACL)

액세스 제어 목록(ACLs)은 어떤 보안 주체(계정 멤버, 사용자 또는 역할)가 리소스에 액세스할 수 있는 권한을 가지고 있는 지를 제어합니다. ACLs는 JSON 정책 문서 형식을 사용하지 않지만 리소스 기반 정책과 유사합니다.

ACL을 지원하는 서비스의 예로는 아마존 S3와 아마존 VPC가 있습니다. AWS WAF ACL에 대해 자세히 알아보려면 Amazon Simple Storage Service 개발자 안내서의 액세스 제어 목록(ACL) 개요를 참조하세요.

기타 정책 타입

AWS 일반적이지 않은 추가 정책 유형을 지원합니다. 이러한 정책 타입은 더 일반적인 정책 타입에 따라 사용자에게 부여되는 최대 권한을 설정할 수 있습니다.

- 권한 경계 – 권한 경계는 보안 인증 기반 정책에 따라 IAM 엔터티(IAM 사용자 또는 역할)에 부여할 수 있는 최대 권한을 설정하는 고급 기능입니다. 개체에 대한 권한 경계를 설정할 수 있습니다. 그 결과로 얻는 권한은 엔터티의 자격 증명 기반 정책과 그 권한 경계의 교집합입니다. `Principal` 필드에서 사용자나 역할을 보안 주체로 지정하는 리소스 기반 정책은 권한 경계를 통해 제한되지 않습니다. 이러한 정책 중 하나에 포함된 명시적 거부는 허용을 재정의합니다. 권한 경계에 대한 자세한 정보는 IAM 사용자 설명서의 IAM 엔터티에 대한 권한 경계를 참조합니다.
- 서비스 제어 정책 (SCP) - SCP는 조직 또는 조직 단위 (OU) 에 대한 최대 권한을 지정하는 JSON 정책입니다. AWS Organizations AWS Organizations 사업체가 소유한 여러 AWS 계정 개를 그룹화하고 중앙에서 관리하는 서비스입니다. 조직에서 모든 기능을 활성화할 경우 서비스 제어 정책(SCP)을 임의의 또는 모든 계정에 적용할 수 있습니다. SCP는 각 항목을 포함하여 구성원 계정의 엔티티에 대한 권한을 제한합니다. AWS 계정 루트 사용자조직 및 SCP에 대한 자세한 정보는AWS Organizations 사용 설명서의 SCP 작동 방식을 참조하세요.
- 세션 정책 – 세션 정책은 역할 또는 연합된 사용자에 대해 임시 세션을 프로그래밍 방식으로 생성할 때 파라미터로 전달하는 고급 정책입니다. 결과적으로 얻는 세션의 권한은 사용자 또는 역할 자격 증명 기반 정책의 교차 및 세션 정책입니다. 또한 권한을 리소스 기반 정책에서 가져올 수도 있습니다. 이러한 정책 중 하나에 포함된 명시적 거부는 허용을 재정의합니다. 자세한 정보는 IAM 사용자 설명서의 세션 정책을 참조합니다.

여러 정책 타입

여러 정책 타입이 요청에 적용되는 경우 결과 권한은 이해하기가 더 복잡합니다. 여러 정책 유형이 관련되어 있을 때 요청을 허용할지 여부를 AWS 결정하는 방법을 알아보려면 IAM 사용 설명서의 정책 평가 로직을 참조하십시오.

Amazon Fraud Detector가 IAM과 함께 작동하는 방식

IAM을 사용하여 Amazon Fraud Detector에 대한 액세스를 관리하려면 먼저 Amazon Fraud Detector와 함께 사용할 수 있는 IAM 기능이 무엇인지 이해해야 합니다. Amazon Fraud Detector 및 기타 AWS 서비스가 IAM과 어떻게 연동되는지 자세히 알아보려면 IAM 사용 설명서의 IAM과 연동되는AWS 서비스를 참조하십시오.

주제

- Amazon Fraud Detector 신원 정보 기반 정책
- Amazon Fraud Detector 리소스 기반 정책
- Amazon Fraud Detector 태그를 기반으로 한 인증
- Amazon Fraud Detector IAM 역할

Amazon Fraud Detector 신원 정보 기반 정책

IAM 자격 증명 기반 정책을 사용하면 허용되거나 거부되는 작업와 리소스뿐 아니라 작업이 허용되거나 거부되는 조건을 지정할 수 있습니다. Amazon Fraud Detector는 특정 작업, 리소스 및 조건 키를 지원합니다. JSON 정책에서 사용하는 모든 요소에 대해 알고 싶다면 IAM 사용 설명서의 IAM JSON 정책 요소 참조를 참조하세요.

Amazon Fraud Detector를 시작하려면 Amazon Fraud Detector 작업에 대한 액세스가 제한되고 필요한 권한을 가진 사용자를 생성하는 것이 좋습니다. 필요하다면 그 밖의 권한을 추가할 수 있습니다. 다음 정책은 Amazon Fraud Detector를 사용하는 데 필요한 권한을 `AmazonFraudDetectorFullAccessPolicy` 제공합니다`AmazonS3FullAccess`. 이러한 정책을 사용하여 Amazon Fraud Detector를 설정하는 방법에 대한 자세한 내용은 을 참조하십시오Amazon Fraud Detector 설정.

작업

관리자는 AWS JSON 정책을 사용하여 누가 무엇에 액세스할 수 있는지 지정할 수 있습니다. 즉, 어떤 보안 주체가 어떤 리소스와 어떤 조건에서 작업을 수행할 수 있는지를 지정할 수 있습니다.

JSON 정책의 `Action`요소는 정책에서 액세스를 허용하거나 거부하는 데 사용할 수 있는 태스크를 설명합니다. 정책 작업은 일반적으로 관련 AWS API 작업과 이름이 같습니다. 일치하는 API 작업이 없는 권한 전용 작업 같은 몇 가지 예외도 있습니다. 정책에서 여러 작업이 필요한 몇 가지 작업도 있습니다. 이러한 추가 작업을 일컬어 종속 작업이라고 합니다.

연결된 작업을 수행할 수 있는 권한을 부여하기 위한 정책에 작업을 포함하십시오.

Amazon Fraud Detector의 정책 조치는 조치 앞에 다음 접두사를 사용합니다. `frauddetector:` 예를 들어 Amazon Fraud Detector `CreateRule` API 작업을 사용하여 규칙을 생성하려면 정책에 `frauddetector:CreateRule` 작업을 포함해야 합니다. 정책 문에는 `Action` 또는 `NotAction` 요소가 포함되어야 합니다. Amazon Fraud Detector는 이 서비스로 수행할 수 있는 작업을 설명하는 자체 작업 세트를 정의합니다.

명령문 하나에 여러 태스크를 지정하려면 다음과 같이 쉼표로 구분합니다.

```
"Action": [
      "frauddetector:action1",
      "frauddetector:action2"
```

와일드카드(*)를 사용하여 여러 작업을 지정할 수 있습니다. 예를 들어, `Describe`라는 단어로 시작하는 모든 태스크를 지정하려면 다음 태스크를 포함합니다.

```
"Action": "frauddetector:Describe*"
```

Amazon Fraud Detector 작업 목록을 보려면 IAM 사용 설명서의 Amazon Fraud Detector에서 정의한 작업을 참조하십시오.

리소스

관리자는 AWS JSON 정책을 사용하여 누가 무엇에 액세스할 수 있는지 지정할 수 있습니다. 즉, 어떤 보안 주체가 어떤 리소스와 어떤 조건에서 작업을 수행할 수 있는지 지정할 수 있습니다.

`Resource` JSON 정책 요소는 작업이 적용되는 하나 이상의 개체를 지정합니다. 문장에는 `Resource`또는 `NotResource`요소가 반드시 추가되어야 합니다. 모범 사례에 따라 Amazon 리소스 이름(ARN)을 사용하여 리소스를 지정합니다. 리소스 수준 권한이라고 하는 특정 리소스 타입을 지원하는 작업에 대해 이 작업을 수행할 수 있습니다.

작업 나열과 같이 리소스 수준 권한을 지원하지 않는 작업의 경우, 와일드카드(*)를 사용하여 해당 문이 모든 리소스에 적용됨을 나타냅니다.

```
"Resource": "*"
```

Amazon Fraud Detector에서 정의한 리소스 유형에는 모든 Amazon Fraud Detector 리소스 ARN이 나열되어 있습니다.

예를 들어 명령문에 `my_detector` 탐지기를 지정하려면 다음 ARN을 사용하십시오.

```
"Resource": "arn:aws:frauddetector:us-east-1:123456789012:detector/my_detector"
```

ARN 형식에 대한 자세한 내용은 Amazon 리소스 이름 (ARN) 및 AWS 서비스 네임스페이스를 참조하십시오.

특정 계정에 속하는 모든 탐지기를 지정하려면 와일드카드 (*) 를 사용합니다.

```
"Resource": "arn:aws:frauddetector:us-east-1:123456789012:detector/*"
```

리소스 생성 작업과 같은 일부 Amazon Fraud Detector 작업은 특정 리소스에서 수행할 수 없습니다. 이러한 경우, 와일드카드(*)를 사용해야 합니다.

```
"Resource": "*"
```

Amazon Fraud Detector 리소스 유형 및 해당 ARN 목록을 보려면 IAM 사용 설명서의 Amazon Fraud Detector에서 정의한 리소스를 참조하십시오. 각 리소스의 ARN을 지정할 수 있는 작업에 대해 알아보려면 Amazon Fraud Detector에서 정의한 작업을 참조하십시오.

조건 키

관리자는 AWS JSON 정책을 사용하여 누가 무엇에 액세스할 수 있는지 지정할 수 있습니다. 즉, 어떤 보안 주체가 어떤 리소스와 어떤 조건에서 작업을 수행할 수 있는지 지정할 수 있습니다.

`Condition` 요소(또는 `Condition` 블록)를 사용하면 정책이 발효되는 조건을 지정할 수 있습니다. `Condition` 요소는 옵션입니다. 같거나 작음과 같은 조건 연산자를 사용하여 정책의 조건을 요청의 값과 일치시키는 조건식을 생성할 수 있습니다.

한 문에서 여러 `Condition`요소를 지정하거나 단일 `Condition`요소에서 여러 키를 지정하는 경우 AWS 는 논리적 `AND`태스크를 사용하여 평가합니다. 단일 조건 키에 여러 값을 지정하는 경우 는 논리적 `OR` 연산을 사용하여 조건을 AWS 평가합니다. 명문의 권한을 부여하기 전에 모든 조건을 충족해야 합니다.

조건을 지정할 때 자리 표시자 변수를 사용할 수도 있습니다. 예를 들어, IAM 사용자에게 IAM 사용자 이름으로 태그가 지정된 경우에만 리소스에 액세스할 수 있는 권한을 부여할 수 있습니다. 자세한 정보는 IAM 사용 설명서의 IAM 정책 요소: 변수 및 태그를 참조하세요.

AWS 글로벌 조건 키 및 서비스별 조건 키를 지원합니다. 모든 AWS 글로벌 조건 키를 보려면 IAM 사용 AWS 설명서의 글로벌 조건 컨텍스트 키를 참조하십시오.

Amazon Fraud Detector는 자체 조건 키 세트를 정의하며 일부 글로벌 조건 키 사용도 지원합니다. 모든 AWS 글로벌 조건 키를 보려면 IAM 사용 설명서의AWS 글로벌 조건 컨텍스트 키를 참조하십시오.

Amazon Fraud Detector 조건 키 목록을 보려면 IAM 사용 설명서의 Amazon Fraud Detector의 조건 키를 참조하십시오. 조건 키를 사용할 수 있는 작업 및 리소스를 알아보려면 Amazon Fraud Detector에서 정의한 작업을 참조하십시오.

예제

Amazon Fraud Detector ID 기반 정책의 예를 보려면 을 참조하십시오. Amazon Fraud Detector ID 기반 정책 예제

Amazon Fraud Detector 리소스 기반 정책

Amazon Fraud Detector는 리소스 기반 정책을 지원하지 않습니다.

Amazon Fraud Detector 태그를 기반으로 한 인증

Amazon Fraud Detector 리소스에 태그를 첨부하거나 Amazon Fraud Detector에 요청을 통해 태그를 전달할 수 있습니다. 태그를 기반으로 액세스를 제어하려면 `aws:ResourceTag/key-name`, `aws:RequestTag/key-name` 또는 `aws:TagKeys` 조건 키를 사용하여 정책의 조건 요소에 태그 정보를 제공합니다.

Amazon Fraud Detector IAM 역할

IAM 역할은 AWS 계정 내에서 특정 권한을 가진 엔티티입니다.

Amazon Fraud Detector에서의 임시 자격 증명 사용

임시 보안 인증을 사용하여 페더레이션을 통해 로그인하거나, IAM 역할을 맡거나, 교차 계정 역할을 맡을 수 있습니다. AssumeRole또는 와 같은 AWS STS API 작업을 호출하여 임시 보안 자격 증명을 얻습니다 GetFederationToken.

Amazon Fraud Detector는 임시 자격 증명 사용을 지원합니다.

서비스 연결 역할

서비스 연결 역할을 사용하면 AWS 서비스가 다른 서비스의 리소스에 액세스하여 사용자를 대신하여 작업을 완료할 수 있습니다. 서비스 연결 역할은 IAM 계정에 나타나고 서비스가 소유합니다. IAM 관리자는 서비스 연결 역할의 권한을 볼 수 있지만 편집할 수 없습니다.

Amazon Fraud Detector는 서비스 연결 역할을 지원하지 않습니다.

서비스 역할

이 기능을 사용하면 서비스가 사용자를 대신하여 서비스 역할을 수임할 수 있습니다. 이 역할을 사용하면 서비스가 다른 서비스의 리소스에 액세스해 사용자를 대신해 작업을 완료할 수 있습니다. 서비스 역할은 계정에 나타나고, 해당 계정이 소유합니다. 즉, 관리자가 이 역할에 대한 권한을 변경할 수 있습니다. 그러나 권한을 변경하면 서비스의 기능이 손상될 수 있습니다.

Amazon Fraud Detector는 서비스 역할을 지원합니다.

Amazon Fraud Detector ID 기반 정책 예제

기본적으로 사용자 및 IAM 역할에는 Amazon Fraud Detector 리소스를 생성하거나 수정할 권한이 없습니다. 또한 AWS Management Console AWS CLI, 또는 AWS API를 사용하여 작업을 수행할 수 없습니다. 관리자는 지정된 리소스에서 특정 API 태스크를 수행할 수 있는 권한을 사용자와 역할에게 부여

하는 IAM 정책을 생성해야 합니다. 그런 다음 관리자는 해당 권한이 필요한 사용자 또는 그룹에 이러한 정책을 연결해야 합니다.

이러한 예제 JSON 정책 문서를 사용하여 IAM 자격 증명 기반 정책을 생성하는 방법을 알아보려면 IAM 사용 설명서의 JSON 탭에서 정책 생성을 참조하십시오.

주제

- 정책 모범 사례
- Amazon Fraud Detector에 대한 AWS 관리형 (사전 정의) 정책
- 사용자가 자신의 고유한 권한을 볼 수 있도록 허용
- Amazon Fraud Detector 리소스에 대한 전체 액세스 허용
- Amazon Fraud Detector 리소스에 대한 읽기 전용 액세스 허용
- 특정 리소스에 대한 액세스 허용
- 이중 모드 API를 사용할 때 특정 리소스에 대한 액세스를 허용합니다.
- 태그에 따른 액세스 제한

정책 모범 사례

ID 기반 정책은 누군가가 사용자 계정에서 Amazon Fraud Detector 리소스를 생성, 액세스 또는 삭제할 수 있는지 여부를 결정합니다. 이 작업으로 인해 AWS 계정에 비용이 발생할 수 있습니다. ID 기반 정책을 생성하거나 편집할 때는 다음 지침과 권장 사항을 따르세요.

- AWS 관리형 정책을 시작하고 최소 권한 권한으로 이동 — 사용자 및 워크로드에 권한 부여를 시작하려면 여러 일반적인 사용 사례에 권한을 부여하는 AWS 관리형 정책을 사용하십시오. 해당 내용은 에서 사용할 수 있습니다. AWS 계정사용 사례에 맞는 AWS 고객 관리형 정책을 정의하여 권한을 더 줄이는 것이 좋습니다. 자세한 정보는 IAM 사용 설명서의 AWS managed policies(관리형 정책) 또는 AWS managed policies for job functions(직무에 대한 관리형 정책)를 참조하세요.
- 최소 권한 적용 – IAM 정책을 사용하여 권한을 설정하는 경우 태스크를 수행하는 데 필요한 권한만 부여합니다. 이렇게 하려면 최소 권한으로 알려진 특정 조건에서 특정 리소스에 대해 수행할 수 있는 작업을 정의합니다. IAM을 사용하여 권한을 적용하는 방법에 대한 자세한 정보는 IAM 사용 설명서에 있는 Policies and permissions in IAM(IAM의 정책 및 권한)을 참조하세요.
- IAM 정책의 조건을 사용하여 액세스 추가 제한 – 정책에 조건을 추가하여 작업 및 리소스에 대한 액세스를 제한할 수 있습니다. 예를 들어 SSL을 사용하여 모든 요청을 전송해야 한다고 지정하는 정책 조건을 작성할 수 있습니다. 예를 AWS 서비스들어 특정 작업을 통해 서비스 작업을 사용하는 경우 조건을 사용하여 서비스 작업에 대한 액세스 권한을 부여할 수도 AWS CloudFormation있습니다. 자세한 정보는 IAM 사용 설명서의 IAM JSON 정책 요소: 조건을 참조하세요.

- IAM Access Analyzer를 통해 IAM 정책을 검증하여 안전하고 기능적인 권한 보장 – IAM Access Analyzer에서는 IAM 정책 언어(JSON)와 모범 사례가 정책에서 준수되도록 신규 및 기존 정책을 검증합니다. IAM Access Analyzer는 100개 이상의 정책 확인 항목과 실행 가능한 추천을 제공하여 안전하고 기능적인 정책을 작성하도록 돕습니다. 자세한 정보는 IAM 사용 설명서의 IAM Access Analyzer 정책 검증을 참조하tpdy.
- 멀티 팩터 인증 (MFA) 필요 - IAM 사용자 또는 루트 사용자가 필요한 시나리오가 있는 경우 추가 보안을 위해 AWS 계정 MFA를 활성화하십시오. API 작업을 직접 호출할 때 MFA가 필요하면 정책에 MFA 조건을 추가합니다. 자세한 정보는 IAM 사용 설명서의 Configuring MFA-protected API access(MFA 보호 API 액세스 구성)를 참조하세요.

IAM의 모범 사례에 대한 자세한 내용은 IAM 사용 설명서의 IAM의 보안 모범 사례를 참조하세요.

Amazon Fraud Detector에 대한 AWS 관리형 (사전 정의) 정책

AWS 에서 생성하고 관리하는 독립형 IAM 정책을 제공하여 많은 일반적인 사용 사례를 해결합니다. AWS이러한 AWS 관리형 정책은 일반적인 사용 사례에 필요한 권한을 부여하므로 필요한 권한을 조사하지 않아도 됩니다. 자세한 내용은 AWS Identity and Access Management 관리 사용 설명서의 AWS 관리형 정책을 참조하십시오.

계정의 사용자에게 연결할 수 있는 다음과 같은 AWS 관리형 정책은 Amazon Fraud Detector에만 적용됩니다.

`AmazonFraudDetectorFullAccess`: Amazon Fraud Detector 리소스, 작업 및 다음을 포함한 지원되는 작업에 대한 전체 액세스 권한을 부여합니다.

- Amazon의 모든 모델 엔드포인트를 나열하고 설명하십시오. SageMaker
- 계정의 모든 IAM 역할을 나열하십시오.
- 모든 Amazon S3 버킷을 나열합니다.
- IAM 전달 역할이 Amazon Fraud Detector에 역할을 전달하도록 허용

이 정책은 무제한 S3 액세스를 제공하지 않습니다. 모델 교육 데이터 세트를 S3에 업로드해야 하는 경우 `AmazonS3FullAccess` 관리형 정책 (또는 범위가 축소된 사용자 지정 Amazon S3 액세스 정책) 도 필요합니다.

IAM 콘솔에 로그인하고 정책 이름으로 검색하여 정책의 권한을 검토할 수 있습니다. 또한 필요에 따라 Amazon Fraud Detector 작업 및 리소스에 대한 권한을 허용하도록 사용자 지정 IAM 정책을 생성할 수 있습니다. 정책이 필요한 사용자 또는 그룹에 이러한 사용자 지정 정책을 연결할 수 있습니다.

사용자가 자신의 고유한 권한을 볼 수 있도록 허용

이 예시는 IAM 사용자가 자신의 사용자 자격 증명에 연결된 인라인 및 관리형 정책을 볼 수 있도록 허용하는 정책을 생성하는 방법을 보여줍니다. 이 정책에는 콘솔에서 또는 API를 사용하여 프로그래밍 방식으로 이 작업을 완료할 수 있는 AWS CLI 권한이 포함됩니다. AWS

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

Amazon Fraud Detector 리소스에 대한 전체 액세스 허용

다음 예제는 사용자에게 모든 Amazon Fraud Detector 리소스 및 작업에 대한 AWS 계정 전체 액세스 권한을 제공합니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "frauddetector:*"
            ],
            "Resource": "*"
        }
    ]
}
```

Amazon Fraud Detector 리소스에 대한 읽기 전용 액세스 허용

이 예시에서는 Amazon Fraud Detector 리소스에 대한 AWS 계정 읽기 전용 액세스 권한을 사용자에게 부여합니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "frauddetector:GetEventTypes",
                "frauddetector:BatchGetVariable",
                "frauddetector:DescribeDetector",
                "frauddetector:GetModelVersion",
                "frauddetector:GetEventPrediction",
                "frauddetector:GetExternalModels",
                "frauddetector:GetLabels",
                "frauddetector:GetVariables",
                "frauddetector:GetDetectors",
                "frauddetector:GetRules",
                "frauddetector:ListTagsForResource",
                "frauddetector:GetKMSEncryptionKey",
                "frauddetector:DescribeModelVersions",
                "frauddetector:GetDetectorVersion",
                "frauddetector:GetPrediction",
                "frauddetector:GetOutcomes",
                "frauddetector:GetEntityTypes",
                "frauddetector:GetModels"
            ],
```

```
            "Resource": "*"
        }
    ]
}
```

특정 리소스에 대한 액세스 허용

이 리소스 수준 정책 예시에서는 특정 Detector 리소스 하나를 제외한 모든 작업 및 리소스에 대한 AWS 계정 액세스 권한을 사용자에게 부여합니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "frauddetector:*"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Deny",
            "Action": [
                "frauddetector:*Detector"
            ],
            "Resource": "arn:${Partition}:frauddetector:${Region}:${Account}:detector/
${detector-name}"
        }
    ]
}
```

이중 모드 API를 사용할 때 특정 리소스에 대한 액세스를 허용합니다.

Amazon Fraud Detector는 목록 작성 및 설명 작업 모두로 작동하는 이중 모드 가져오기 API를 제공합니다. 매개 변수 없이 이중 모드 API를 호출하면 해당 AWS 계정 API와 관련된 지정된 리소스 목록이 반환됩니다. 파라미터와 함께 이중 모드 API를 호출하면 지정된 리소스의 세부 정보가 반환됩니다. 리소스는 모델, 변수, 이벤트 유형 또는 엔티티 유형일 수 있습니다.

이중 모드 API는 IAM 정책의 리소스 수준 권한을 지원합니다. 하지만 리소스 수준 권한은 요청의 일부로 하나 이상의 파라미터가 제공된 경우에만 적용됩니다. 예를 들어, 사용자가 GetVariablesAPI를 호출하여 변수 이름을 제공하고 변수 리소스 또는 변수 이름에 IAM Deny 정책이 연결되어 있는 경우

사용자에게 오류가 발생합니다. AccessDeniedException 사용자가 변수 이름을 지정하지 않고 GetVariables API를 호출하면 모든 변수가 반환되어 정보 유출이 발생할 수 있습니다.

사용자가 특정 리소스의 세부 정보만 볼 수 있게 하려면 IAM 거부 NotResource 정책에서 IAM 정책 요소를 사용하십시오. 이 정책 요소를 IAM 거부 정책에 추가하면 사용자는 블록에 지정된 리소스의 세부 정보만 볼 수 있습니다. NotResource 자세한 내용은 IAM 사용 설명서의 IAM JSON 정책 NotResource 요소:를 참조하십시오.

다음 예제 정책은 사용자가 Amazon Fraud Detector의 모든 리소스에 액세스할 수 있도록 허용합니다. 하지만 NotResource 정책 요소는 GetVariablesAPI 호출을 접두사 user*job_*, 및 var* 가 있는 변수 이름으로만 제한하는 데 사용됩니다.

```
{
 "Version": "2012-10-17",
 "Statement": [
  {
    "Effect": "Allow",
    "Action": "frauddetector:*",
    "Resource": "*"
  },
 {
    "Effect": "Deny",
    "Action": "frauddetector:GetVariables",
    "NotResource": [
       "arn:aws:frauddetector:*:*:variable/user*",
       "arn:aws:frauddetector:*:*:variable/job_*",
       "arn:aws:frauddetector:*:*:variable/var*"
    ]
  }
 ]
}
```

응답

이 예제 정책의 경우 응답은 다음과 같은 동작을 나타냅니다.

- 변수 이름을 포함하지 않는 GetVariables 호출은 요청이 Deny 문에 매핑되기 때문에 AccessDeniedException 오류가 발생합니다.
- 허용되지 않는 변수 이름을 포함하는 GetVariables 호출은 변수 이름이 NotResource 블록의 변수 이름에 매핑되지 않기 때문에 AccessDeniedException 오류가 발생합니다. 예를 들어, 변수 이름

을 email_address 사용하여 GetVariables 호출하면 AccessDeniedException 오류가 발생합니다.

- NotResource블록의 변수 이름과 일치하는 변수 이름을 포함하는 GetVariables 호출은 예상대로 반환됩니다. 예를 들어, 변수 이름이 포함된 GetVariables 호출은 job_cpa 변수의 세부 정보를 job_cpa 반환합니다.

태그에 따른 액세스 제한

이 예제 정책은 리소스 태그를 기반으로 Amazon Fraud Detector에 대한 액세스를 제한하는 방법을 보여줍니다. 이 예시에서는 다음을 가정합니다.

- 에서 Team1과 Team2라는 두 개의 다른 그룹을 정의했습니다. AWS 계정
- 네 개의 탐지기를 만들었습니다.
- Team1의 구성원이 2개의 탐지기에서 API 호출을 할 수 있도록 허용하고 싶습니다.
- Team2의 구성원이 다른 두 탐지기에서 API 호출을 할 수 있도록 허용하고 싶습니다.

API 직접 호출에 대한 액세스를 제어하려면(예)

1. Team1에서 사용하는 A 탐지기에 Project 키와 값이 포함된 태그를 추가하세요.
2. Team2에서 사용하는 B 탐지기에 Project 키와 값이 포함된 태그를 추가합니다.
3. 키와 값이 포함된 태그가 있는 탐지기에 대한 액세스를 거부하는 ResourceTag 조건으로 IAM 정책을 생성하고 해당 정책을 Project B Team1에 연결합니다.
4. 키와 Project 값이 A 포함된 태그가 있는 탐지기에 대한 액세스를 거부하는 ResourceTag 조건을 포함하는 IAM 정책을 생성하고 해당 정책을 Team2에 연결합니다.

다음은 키와 값이 1인 태그가 있는 모든 Amazon Fraud Detector 리소스에서 특정 작업을 거부하는 정책의 예입니다. Project B

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "frauddetector:*",
      "Resource": "*"
    },
```

```
    {
      "Effect": "Deny",

      "Action": [

        "frauddetector:CreateModel",
        "frauddetector:CancelBatchPredictionJob",
        "frauddetector:CreateBatchPredictionJob",
        "frauddetector:DeleteBatchPredictionJob",
        "frauddetector:DeleteDetector"
      ],

      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceTag/Project": "B"
        }
      }
    }
  ]
}
```

혼동된 대리자 방지

대리인을 혼동하는 문제는 작업을 수행할 권한이 없는 주체가 더 많은 권한을 가진 주체에게 해당 작업을 수행하도록 강요할 때 발생합니다. AWS 계정의 리소스에 대한 타사 액세스 권한 (교차 계정이라고 함) 또는 기타 AWS 서비스 (크로스 서비스라고 함) 에 제공하는 경우 계정을 보호하는 데 도움이 되는 도구를 제공합니다.

서비스 간 혼동 대리인 문제는 한 서비스 (통화 서비스) 가 다른 서비스 (호출 서비스) 에 전화를 걸 때 발생할 수 있습니다. 직접 호출하는 서비스는 다른 고객의 리소스에 대해 액세스 권한이 없는 방식으로 작동하게 권한을 사용하도록 조작될 수 있습니다. 이를 방지하기 위해 서비스 리소스에 대한 액세스 권한이 부여된 서비스 주체를 사용하는 모든 서비스의 데이터를 보호하는 데 도움이 되는 정책을 만들 수 있습니다.

Amazon Fraud Detector는 권한 정책에서 서비스 역할을 사용하여 서비스가 사용자를 대신하여 다른 서비스의 리소스에 액세스할 수 있도록 지원합니다. 역할에는 두 가지 정책이 필요합니다. 즉, 역할을 수임할 수 있는 보안 주체를 지정하는 역할 신뢰 정책과 역할로 수행할 수 있는 작업을 지정하는 권한 정책이 필요합니다. 서비스가 사용자를 대신하여 역할을 맡을 경우 서비스 보안 주체는 역할 신뢰 정책의 `sts:AssumeRole` 작업을 수행하도록 허용되어야 합니다. 서비스가 `sts:AssumeRole` 호출되면

서비스 주체가 역할의 권한 정책에서 허용하는 리소스에 액세스하는 데 사용하는 임시 보안 자격 증명 세트를 AWS STS 반환합니다.

서비스 간에 혼동되는 대리인 문제를 방지하기 위해 Amazon Fraud Detector에서는 역할 신뢰 정책의 aws:SourceArn및 aws:SourceAccount글로벌 조건 컨텍스트 키를 사용하여 예상 리소스에서 생성된 요청으로만 역할에 대한 액세스를 제한할 것을 권장합니다.

는 계정 ID를 aws:SourceAccount 지정하고 교차 서비스 액세스와 관련된 리소스의 ARN을 aws:SourceArn 지정합니다. 는 ARN 형식을 사용하여 aws:SourceArn 지정해야 합니다. 동일한 정책 설명에서 둘 다 aws:SourceAccount 동일한 계정 ID를 사용하는 경우 둘 다 동일한 계정 ID를 사용하는지 확인하십시오. aws:SourceArn

혼동된 대리인 문제로부터 보호하는 가장 효과적인 방법은 리소스의 전체 ARN이 포함된 aws:SourceArn 글로벌 조건 컨텍스트 키를 사용하는 것입니다. 리소스의 전체 ARN을 모르거나 여러 리소스를 지정하는 경우 ARN의 알 수 없는 부분에 대해 와일드카드 (*) 와 함께 aws:SourceArn 글로벌 컨텍스트 조건 키를 사용하십시오. 예를 들어 arn:aws:*servicename*:*:*123456789012*:*입니다. Amazon Fraud Detector 리소스 및 권한 정책에서 사용할 수 있는 작업에 대한 자세한 내용은 Amazon Fraud Detector의 작업, 리소스 및 조건 키를 참조하십시오.

다음 역할 신뢰 정책 예제는 aws:SourceArn 조건 키에 와일드카드 (*) 를 사용하여 Amazon Fraud Detector가 계정 ID와 연결된 여러 리소스에 액세스할 수 있도록 허용합니다.

```
{
        "Version": "2012-10-17",
        "Statement": [
             {
               "Effect": "Allow",
               "Principal": {
               "Service": [
                   "frauddetector.amazonaws.com"
                   ]
               },
               "Action": "sts:AssumeRole",
               "Condition": {
               "StringEquals": {
                   "aws:SourceAccount": "123456789012"
               },
                 "StringLike": {
                    "aws:SourceArn": "arn:aws:frauddetector:us-west-2:123456789012:*"
        }
```

```
      }
    }
  ]
}
```

다음 역할 신뢰 정책은 Amazon Fraud Detectory의 `external-model` 리소스에만 액세스할 수 있도록 허용합니다. 조건 `aws:SourceArn` 블록의 매개변수를 확인하십시오. 리소스 한정자는 `PutExternalModel` API 호출을 위해 제공된 모델 엔드포인트를 사용하여 빌드됩니다.

```
    {
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": [
          "frauddetector.amazonaws.com"
        ]
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "123456789012"
        },
        "StringLike": {
          "aws:SourceArn": "arn:aws:frauddetector:us-west-2:123456789012:external-
model/MyExternalModeldoNotDelete-ReadOnly"
        }
      }
    }
  ]
}
```

Amazon Fraud Detector ID 및 액세스 문제 해결

다음 정보를 사용하면 Amazon Fraud Detector 및 IAM을 사용할 때 발생할 수 있는 일반적인 문제를 진단하고 해결하는 데 도움이 됩니다.

주제

- Amazon Fraud Detector에서 조치를 취할 권한이 없습니다.

- 저는 IAM을 수행할 권한이 없습니다. PassRole
- 내 AWS 계정 외부의 사용자가 내 Amazon Fraud Detector 리소스에 액세스할 수 있도록 허용하고 싶습니다.
- Amazon Fraud Detector는 주어진 역할을 맡을 수 없었습니다.

Amazon Fraud Detector에서 조치를 취할 권한이 없습니다.

조치를 취할 권한이 없다는 AWS Management Console 메시지가 표시되면 관리자에게 도움을 요청해야 합니다. 관리자는 로그인 보안 인증 정보를 제공한 사람입니다.

다음 예제 오류는 `mateojackson` 사용자가 콘솔을 사용하여 *####* 대한 세부 정보를 보려고 하지만 `frauddetector:`*`GetDetectors`* 권한이 없는 경우 발생합니다.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 frauddetector:GetDetectors on resource: my-example-detector
```

이 경우 Mateo는 *`my-example-detector`* 작업을 사용하여 `frauddetector:`*`GetDetectors`* 리소스에 액세스하도록 허용하는 정책을 업데이트하라고 관리자에게 요청합니다.

저는 IAM을 수행할 권한이 없습니다. PassRole

작업을 수행할 권한이 없다는 오류가 발생하는 경우 Amazon Fraud Detector에 역할을 전달할 수 있도록 정책을 업데이트해야 합니다. `iam:PassRole`

일부 AWS 서비스 서비스에서는 새 서비스 역할 또는 서비스 연결 역할을 생성하는 대신 기존 역할을 해당 서비스에 전달할 수 있습니다. 이렇게 하려면 사용자가 서비스에 역할을 전달할 수 있는 권한을 가지고 있어야 합니다.

다음 예제 오류는 이라는 IAM 사용자가 Amazon Fraud Detector에서 콘솔을 사용하여 작업을 `marymajor` 수행하려고 할 때 발생합니다. 하지만 작업을 수행하려면 서비스 역할이 부여한 권한이 서비스에 있어야 합니다. Mary는 서비스에 역할을 전달할 수 있는 권한을 가지고 있지 않습니다.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

이 경우 Mary가 `iam:PassRole`작업을 수행할 수 있도록 Mary의 정책을 업데이트해야 합니다.

도움이 필요한 경우 AWS 관리자에게 문의하십시오. 관리자는 로그인 자격 증명을 제공한 사람입니다.

내 AWS 계정 외부의 사용자가 내 Amazon Fraud Detector 리소스에 액세스할 수 있도록 허용하고 싶습니다.

다른 계정의 사용자 또는 조직 외부의 사람이 리소스에 액세스할 때 사용할 수 있는 역할을 생성할 수 있습니다. 역할을 수임할 신뢰할 수 있는 사람을 지정할 수 있습니다. 리소스 기반 정책 또는 액세스 제어 목록(ACL)을 지원하는 서비스의 경우 이러한 정책을 사용하여 다른 사람에게 리소스에 대한 액세스 권한을 부여할 수 있습니다.

자세히 알아보려면 다음을 참조하세요.

- Amazon Fraud Detector가 이러한 기능을 지원하는지 알아보려면 을 참조하십시오Amazon Fraud Detector가 IAM과 함께 작동하는 방식.
- 소유하고 AWS 계정 있는 모든 리소스에 대한 액세스를 제공하는 방법을 알아보려면 IAM 사용 설명서의 다른 AWS 계정 IAM 사용자에게 액세스 권한 제공을 참조하십시오.
- 제3자에게 리소스에 대한 액세스 권한을 제공하는 방법을 알아보려면 IAM 사용 설명서의 타사 AWS 계정AWS 계정 소유에 대한 액세스 제공을 참조하십시오.
- ID 페더레이션을 통해 액세스 권한을 제공하는 방법을 알아보려면 IAM 사용 설명서의 외부에서 인증된 사용자에게 액세스 권한 제공(자격 증명 연동)을 참조하세요.
- 크로스 계정 액세스를 위한 역할과 리소스 기반 정책 사용의 차이점을 알아보려면 IAM 사용 설명서의 IAM 역할과 리소스 기반 정책의 차이를 참조하세요.

Amazon Fraud Detector는 주어진 역할을 맡을 수 없었습니다.

Amazon Fraud Detector가 지정된 역할을 맡을 수 없다는 오류가 발생하는 경우 지정된 역할에 대한 신뢰 관계를 업데이트해야 합니다. Amazon Fraud Detector를 신뢰할 수 있는 개체로 지정하면 서비스가 역할을 맡을 수 있습니다. Amazon Fraud Detector를 사용하여 역할을 생성하면 이 신뢰 관계가 자동으로 설정됩니다. Amazon Fraud Detector에서 생성하지 않은 IAM 역할에 대해서만 이 신뢰 관계를 설정하면 됩니다.

Amazon Fraud Detector와 기존 역할에 대한 신뢰 관계를 구축하려면

1. https://console.aws.amazon.com/iam/에서 IAM 콘솔을 엽니다.
2. 탐색 창에서 역할을 선택합니다.
3. 수정하려는 역할의 이름을 선택하고 신뢰 관계 탭을 선택합니다.
4. 신뢰 관계 편집을 선택합니다.
5. [Policy Document] 아래에 다음을 붙어 넣고 [Update Trust Policy]를 선택합니다.

```
{
    "Version": "2012-10-17",
    "Statement": [ {
        "Effect": "Allow",
        "Principal": {
            "Service": "frauddetector.amazonaws.com"
        },
        "Action": "sts:AssumeRole"
    } ]
}
```

Amazon Fraud Detector에서의 로깅 및 모니터링

AWS는 Amazon Fraud Detector를 감시하고, 문제 발생 시 보고하고, 적절한 경우 자동 조치를 취할 수 있도록 다음과 같은 모니터링 도구를 제공합니다.

- Amazon은 실행 중인 AWS 리소스와 애플리케이션을 AWS 실시간으로 CloudWatch 모니터링합니다. 에 대한 CloudWatch 자세한 내용은 Amazon CloudWatch 사용 설명서를 참조하십시오.
- AWS CloudTrail계정에서 또는 AWS 계정을 대신하여 이루어진 API 호출 및 관련 이벤트를 캡처하고 지정한 Amazon S3 버킷으로 로그 파일을 전송합니다. 자세한 CloudTrail 내용은 AWS CloudTrail 사용 설명서를 참조하십시오.

Amazon Fraud Detector 모니터링에 대한 자세한 내용은 을 참조하십시오Amazon Fraud Detector 모니터링.

Amazon Fraud Detector에 대한 규정 준수 검증

제3자 감사자는 SOC, PCI, FedRAMP, HIPAA와 같은 여러 AWS 규정 준수 프로그램의 일환으로 AWS 서비스의 보안 및 규정 준수를 평가합니다.

특정 규정 준수 프로그램의 범위 내에 AWS 서비스 있는지 알아보려면AWS 서비스 규정 준수 프로그램의 범위별 범위 내 규정 준수 프로그램별 규정을 선택하십시오. 일반 정보는 AWS 규정 준수 프로그램AWS 보증 프로그램 규정AWS 참조하십시오.

를 사용하여 AWS Artifact타사 감사 보고서를 다운로드할 수 있습니다. 자세한 내용은 의 보고서 https://docs.aws.amazon.com/artifact/latest/ug/downloading-documents.html 참조하십시오 AWS Artifact.

사용 시 규정 준수 AWS 서비스 책임은 데이터의 민감도, 회사의 규정 준수 목표, 관련 법률 및 규정에 따라 결정됩니다. AWS 규정 준수에 도움이 되는 다음 리소스를 제공합니다.

- 보안 및 규정 준수 퀵 스타트 가이드 - 이 배포 가이드에서는 아키텍처 고려 사항을 설명하고 보안 및 규정 준수에 AWS 중점을 둔 기본 환경을 배포하기 위한 단계를 제공합니다.
- Amazon Web Services의 HIPAA 보안 및 규정 준수를 위한 설계 — 이 백서에서는 기업이 HIPAA 적격 애플리케이션을 만드는 AWS 데 사용할 수 있는 방법을 설명합니다.

> **Note**
>
> 모든 AWS 서비스 사람이 HIPAA 자격을 갖춘 것은 아닙니다. 자세한 내용은 HIPAA 적격 서비스 참조를 참조하십시오.

- AWS 규정 준수 리소스AWS — 이 워크북 및 가이드 모음은 해당 산업 및 지역에 적용될 수 있습니다.
- AWS 고객 규정 준수 가이드 — 규정 준수의 관점에서 공동 책임 모델을 이해하십시오. 이 가이드에서는 보안을 유지하기 위한 모범 사례를 AWS 서비스 요약하고 여러 프레임워크 (미국 표준 기술 연구소 (NIST), 결제 카드 산업 보안 표준 위원회 (PCI), 국제 표준화기구 (ISO) 등) 에서 보안 제어에 대한 지침을 매핑합니다.
- AWS Config 개발자 안내서의 규칙을 사용하여 리소스 평가 — 이 AWS Config 서비스는 리소스 구성이 내부 관행, 업계 지침 및 규정을 얼마나 잘 준수하는지 평가합니다.
- AWS Security Hub— 이를 AWS 서비스 통해 내부 AWS보안 상태를 포괄적으로 파악할 수 있습니다. Security Hub는 보안 제어를 사용하여 AWS 리소스를 평가하고 보안 업계 표준 및 모범 사례에 대한 규정 준수를 확인합니다. 지원되는 서비스 및 제어 목록은 Security Hub 제어 참조를 참조하십시오.
- Amazon GuardDuty — 환경에 의심스럽고 악의적인 활동이 있는지 AWS 계정모니터링하여 워크로드, 컨테이너 및 데이터에 대한 잠재적 위협을 AWS 서비스 탐지합니다. GuardDuty 특정 규정 준수 프레임워크에서 요구하는 침입 탐지 요구 사항을 충족하여 PCI DSS와 같은 다양한 규정 준수 요구 사항을 해결하는 데 도움이 될 수 있습니다.
- AWS Audit Manager— 이를 AWS 서비스 통해 AWS 사용량을 지속적으로 감사하여 위험을 관리하고 규정 및 업계 표준을 준수하는 방법을 단순화할 수 있습니다.

Amazon Fraud Detector의 복원력

AWS 글로벌 인프라는 AWS 리전 및 가용 영역을 중심으로 구축됩니다. AWS 리전에서는 물리적으로 분리되고 격리된 다수의 가용 영역을 제공하며 이러한 가용 영역은 짧은 지연 시간, 높은 처리량 및 높은 중복성을 갖춘 네트워크에 연결되어 있습니다. 가용 영역을 사용하면 중단 없이 영역 간에 자동으로 장애 극복 조치가 이루어지는 애플리케이션 및 데이터베이스를 설계하고 운영할 수 있습니다. 가용 영역은 기존의 단일 또는 다중 데이터 센터 인프라보다 가용성, 내결함성, 확장성이 뛰어납니다.

AWS 리전 및 가용 영역에 대한 자세한 내용은 AWS 글로벌 인프라를 참조하십시오.

Amazon Fraud Detector의 인프라 보안

관리형 서비스인 Amazon Fraud Detector는 AWS 글로벌 네트워크 보안의 보호를 받습니다. AWS 보안 서비스 및 인프라 AWS 보호 방법에 대한 자세한 내용은 AWS 클라우드 보안을 참조하십시오. 인프라 보안 모범 사례를 사용하여 AWS 환경을 설계하려면 Security Pillar AWS Well-Architected Framework의 인프라 보호를 참조하십시오.

AWS 게시된 API 호출을 사용하여 네트워크를 통해 Amazon Fraud Detector에 액세스할 수 있습니다. 고객은 다음을 지원해야 합니다.

- 전송 계층 보안(TLS) TLS 1.2는 필수이며 TLS 1.3을 권장합니다.
- DHE(Ephemeral Diffie-Hellman) 또는 ECDHE(Elliptic Curve Ephemeral Diffie-Hellman)와 같은 완전 전송 보안(PFS)이 포함된 암호 제품군 Java 7 이상의 최신 시스템은 대부분 이러한 모드를 지원합니다.

또한 요청은 액세스 키 ID 및 IAM 주체와 관련된 비밀 액세스 키를 사용하여 서명해야 합니다. 또는 AWS Security Token Service(AWS STS)를 사용하여 임시 보안 인증을 생성하여 요청에 서명할 수 있습니다.

Amazon Fraud Detector 모니터링

모니터링은 Amazon Fraud Detector 및 기타 AWS 솔루션의 안정성, 가용성 및 성능을 유지하는 데 있어 중요한 부분입니다. AWS는 Amazon Fraud Detector를 감시하고, 문제 발생 시 보고하고, 적절한 경우 자동 조치를 취할 수 있도록 다음과 같은 모니터링 도구를 제공합니다.

- Amazon은 실행 중인 AWS 리소스와 애플리케이션을 AWS 실시간으로 CloudWatch 모니터링합니다. 지표를 수집 및 추적하고, 맞춤 대시보드를 생성할 수 있으며, 지정된 지표가 지정한 임계값에 도달하면 사용자에게 알리거나 조치를 취하도록 경보를 설정할 수 있습니다. 자세한 내용은 Amazon CloudWatch 사용 설명서를 참조하십시오.
- AWS CloudTrail계정에서 또는 AWS 계정을 대신하여 이루어진 API 호출 및 관련 이벤트를 캡처하고 지정한 Amazon S3 버킷으로 로그 파일을 전송합니다. 어떤 사용자 및 계정이 AWS를 호출했는지 어떤 소스 IP 주소에 호출이 이루어졌는지 언제 호출이 발생했는지 확인할 수 있습니다. 자세한 내용은 AWS CloudTrail 사용 설명서를 참조하십시오.

주제

아마존을 통한 Amazon Fraud Detector 모니터링 CloudWatch

원시 데이터를 수집하여 읽기 가능한 거의 실시간 지표로 처리하는 Amazon Fraud Detector를 사용하여 CloudWatch 모니터링할 수 있습니다. 이러한 통계는 15개월간 보관되므로 기록 정보에 액세스하고 웹 애플리케이션 또는 서비스가 어떻게 실행되고 있는지 전체적으로 더 잘 파악할 수 있습니다. 특정 임계값을 주시하다가 해당 임계값이 충족될 때 알림을 전송하거나 조치를 취하도록 경보를 설정할 수도 있습니다. 자세한 내용은 Amazon CloudWatch 사용 설명서를 참조하십시오.

주제

Amazon Fraud Detector용 CloudWatch 메트릭스 사용.

측정치를 사용하려면 다음 정보를 지정해야 합니다.

- 메트릭 네임스페이스. 네임스페이스는 CloudWatch Amazon Fraud Detector가 지표를 게시하는 데 사용하는 컨테이너입니다. CloudWatch ListMetricsAPI 또는 list-metrics 명령을 사용하여 Amazon Fraud Detector의 지표를 보는 경우 네임스페이스를 지정하십시오`AWS/FraudDetector`.
- 지표 측정기준. 차원은 측정치를 고유하게 식별하는 데 도움이 되는 이름-값 쌍입니다. 예를 들어 차원 이름일 수 있습니다. `DetectorId` 지표 측정기준 지정은 선택사항입니다.
- `GetEventPrediction`와 같은 지표 이름.

AWS Management Console AWS CLI, 또는 CloudWatch API를 사용하여 Amazon Fraud Detector에 대한 모니터링 데이터를 가져올 수 있습니다. Amazon AWS 소프트웨어 개발 키트 (SDK) 또는 CloudWatch API 도구 중 하나를 통해 CloudWatch API를 사용할 수도 있습니다. 콘솔에는 CloudWatch API의 원시 데이터를 기반으로 한 일련의 그래프가 표시됩니다. 필요에 따라 콘솔에 표시되거나 API에서 가져온 그래프를 사용하는 것이 더 나을 수 있습니다.

다음 목록은 몇 가지 일반적인 지표 사용 사례를 보여 줍니다. 모든 사용 사례를 망라한 것은 아니지만 시작하는 데 참고가 될 것입니다.

방법	관련 지표
수행된 예측 수를 추적하려면 어떻게 해야 하나요?	`GetEventPrediction` 지표를 모니터링합니다.
`GetEventPrediction` 오류를 모니터링하려면 어떻게 해야 하나요?	`GetEventPrediction5xxError` 및 `GetEventPrediction4xxError` 지표를 사용하세요.
`GetEventPrediction` 호출의 지연 시간은 어떻게 모니터링할 수 있습니까?	`GetEventPredictionLatency` 측정치를 사용합니다.

Amazon Fraud Detector를 모니터링하려면 적절한 CloudWatch 권한이 있어야 CloudWatch 합니다. 자세한 내용은 Amazon의 인증 및 액세스 제어를 참조하십시오 CloudWatch.

Amazon Fraud Detector 메트릭에 액세스

다음 단계는 CloudWatch 콘솔을 사용하여 Amazon Fraud Detector 지표에 액세스하는 방법을 보여줍니다.

지표를 보려면(콘솔)

1. https://console.aws.amazon.com/cloudwatch/ 에서 CloudWatch 콘솔을 엽니다.
2. 지표를 선택하고 모든 지표 탭을 선택한 다음 Fraud Detector를 선택합니다.
3. 지표 차원을 선택합니다.
4. 목록에서 원하는 지표를 선택하고 그래프의 기간을 선택합니다.

경보 만들기

CloudWatch 알람 상태가 변경될 때 Amazon Simple Service (Amazon SNS) 메시지를 보내는 경보를 생성할 수 있습니다. 경보는 지정한 기간 동안 단일 지표를 감시합니다. 기간 수에 대한 주어진 임계값과 지표 값을 비교하여 하나 이상의 작업을 수행합니다. 이 작업은 Amazon SNS 주제 또는 Auto Scaling 정책에 전송되는 알림입니다.

경보는 지속적인 상태 변경에 대한 조치만 호출합니다. CloudWatch 경보는 단순히 특정 상태에 있다는 이유만으로 조치를 호출하지 않습니다. 상태가 변경되어 지정된 기간 동안 유지되어야 합니다.

경보를 설정하려면(콘솔)

1. 에 AWS Management Console 로그인하고 https://console.aws.amazon.com/cloudwatch/ 에서 CloudWatch 콘솔을 엽니다.
2. 탐색 창에서 [Alarms] 를 선택하고 [알람 만들기] 를 선택합니다. 그러면 알람 생성 마법사가 열립니다.
3. 지표 선택을 선택하세요.
4. 모든 지표 탭에서 Fraud Detector를 선택합니다.
5. [Detector ID별] 을 선택한 다음 GetEventPrediction지표를 선택합니다.
6. 그래프로 표시된 지표 탭을 선택합니다.
7. Statistic(통계)에서 Sum(합계)를 선택합니다.
8. 지표 선택을 선택하세요.
9. 조건의 경우 임계값 유형에서 [정적] 을 선택하고 [Whenever...] 에서 [더 크게] 를 선택한 다음 원하는 최대값을 입력합니다. 다음을 선택합니다.
10. 기존 SNS 주제에 경보를 전송하려면 다음 주소로 알림 전송:에서 기존 SNS 주제를 선택합니다. 새 이메일 구독 목록의 이름과 이메일 주소를 설정하려면 새 목록을 선택합니다. CloudWatch 목록을 저장하고 필드에 표시하므로 이 목록을 사용하여 향후 경보를 설정할 수 있습니다.

> **Note**
>
> 새 목록을 사용하여 새 SNS 주제를 만드는 경우, 의도한 수신자가 알림을 받기 전에 이메일 주소를 확인해야만 합니다. SNS는 경보가 경보 상태에 진입할 때만 이메일을 전송합니다. 이메일 주소가 확인되기 전에 이러한 경보 상태 변경이 발생하면 의도한 수신자는 알림을 받지 못합니다.

11. 다음을 선택합니다. 알람의 이름과 선택적 설명을 추가하세요. 다음을 선택합니다.
12. 경보 생성을 선택합니다.

Amazon Fraud Detector 메트릭스

Amazon Fraud Detector는 다음 측정치를 에 CloudWatch 전송합니다. 모든 지표는 다음 통계를 지원합니다. `Average`,`Minimum`,`Maximum`,`Sum`.

지표	설명
`GetEventPrediction`	GetEventPrediction API 요청 수. 유효한 차원: `DetectorID`
`GetEventPredictionLatency`	요청의 클라이언트 요청에 응답하는 데 걸린 시간 간격. GetEventPrediction 유효한 차원: `DetectorID` 단위: 밀리초
`GetEventPrediction4XXError`	Amazon Fraud Detector에서 4xx HTTP 응답 코드를 반환한 GetEventPrediction 요청 수입니다. 각 4xx 응답에 대해 1개가 전송됩니다. 유효한 차원: `DetectorID`
`GetEventPrediction5XXError`	Amazon Fraud Detector에서 5xx HTTP 응답 코드를 반환한 GetEventPrediction 요청 수입니다. 각 5xx 응답에 대해 1개가 전송됩니다.

지표	설명
	유효한 차원: DetectorID
Prediction	예측 수입니다. 성공하면 1이 전송됩니다. 유효한 차원: DetectorID , DetectorVersionID
PredictionLatency	예측 작업에 소요된 시간 간격. 유효한 차원: DetectorID , DetectorVersionID 단위: 밀리초
PredictionError	Amazon Fraud Detector에서 오류가 발생한 예측 수입니다. 오류가 발생하면 1이 전송됩니다. 유효한 차원: DetectorID , DetectorVersionID
VariableUsed	평가 과정에서 변수가 사용된 GetEventPrediction 요청의 수입니다. 유효 치수:DetectorID ,DetectorVersionID , VariableName
VariableDefaultReturned	변수가 이벤트 속성의 일부로 존재하지 않아 평가 중에 변수의 기본값이 사용된 GetEventPrediction 요청 수입니다. 유효 치수:DetectorID ,DetectorVersionID , VariableName
RuleNotEvaluated	이전 규칙이 일치하여 규칙이 평가되지 않은 GetEventPrediction 요청 수입니다. 유효 치수:DetectorID ,DetectorVersionID , RuleID

지표	설명
RuleEvaluateTrue	규칙이 True로 트리거되고 규칙 결과가 반환된 GetEventPrediction 요청 수입니다. 유효 치수:DetectorID ,DetectorVersionID , RuleID
RuleEvaluateFalse	규칙이 False로 평가된 GetEventPrediction 요청 수입니다. 유효 치수:DetectorID ,DetectorVersionID , RuleID
RuleEvaluateError	규칙이 잘못 평가된 GetEventPrediction 요청 수 유효 치수:DetectorID ,, DetectorVersionID RuleID
OutcomeReturned	지정된 결과가 반환된 GetEventPrediction 통화 수입니다. 유효 치수:DetectorID ,DetectorVersionID , OutcomeName
ModelInvocation (Amazon SageMaker model endpoint)	평가의 일환으로 SageMaker 모델 엔드포인트가 호출된 GetEventPrediction 요청 수. 유효 치수:DetectorID ,, DetectorVersionID ModelEndpoint
ModelInvocationError (Amazon SageMaker model endpoint)	호출된 SageMaker 모델 엔드포인트가 평가 중에 오류를 반환한 GetEventPrediction 요청 수입니다. 유효 치수:DetectorID ,, DetectorVersionID ModelEndpoint

지표	설명
`ModelInvocationLatency (Amazon SageMaker model endpoint)`	Amazon Fraud Detector에서 확인한 대로 가져온 모델이 응답하는 데 걸리는 시간 간격입니다. 이 간격에는 모델 호출만 포함됩니다. 유효 치수:`DetectorID` ,, `DetectorVersionID` `ModelEndpoint` 단위: 밀리초
`ModelInvocation`	평가 과정에서 모델이 호출된 GetEventPrediction 요청 수. 유효 치수:`DetectorID` ,`DetectorVersionID` , `ModelType` `ModelID`
`ModelInvocationError`	Amazon Fraud Detector 모델이 평가 중에 오류를 반환한 GetEventPrediction 요청 수입니다. 유효 치수:`DetectorID` ,`DetectorVersionID` ,`ModelType` , `ModelID`
`ModelInvocationLatency`	Amazon Fraud Detector에서 본 Amazon 사기 탐지기 모델이 응답하는 데 걸리는 시간 간격입니다. 이 간격에는 모델 호출만 포함됩니다. 유효 치수:`DetectorID` ,`DetectorVersionID` , `ModelType` `ModelID` 단위: 밀리초

를 사용하여 Amazon Fraud Detector API 호출 로깅 AWS CloudTrail

Amazon Fraud Detector는 Amazon Fraud Detector에서 사용자, 역할 또는 AWS 서비스가 수행한 작업의 기록을 제공하는 서비스와 통합되어 있습니다. AWS CloudTrail CloudTrail Amazon Fraud Detector 콘솔에서의 호출, 코드에서 Amazon Fraud Detector API로의 호출 등 Amazon Fraud Detector에 대한 모든 API 호출을 이벤트로 캡처합니다.

추적을 생성하면 Amazon Fraud Detector에 대한 CloudTrail 이벤트를 포함하여 Amazon S3 버킷으로 이벤트를 지속적으로 전송할 수 있습니다. 트레일을 구성하지 않아도 CloudTrail 콘솔의 이벤트 기록에서 가장 최근 이벤트를 계속 볼 수 있습니다. 에서 수집한 CloudTrail 정보를 사용하여 Amazon Fraud Detector에 이루어진 요청, 요청한 IP 주소, 요청한 사람, 요청 시기 및 추가 세부 정보를 확인할 수 있습니다.

자세한 CloudTrail 내용은 AWS CloudTrail 사용 설명서를 참조하십시오.

Amazon Fraud Detector 정보: CloudTrail

CloudTrail 계정을 만들면 AWS 계정에서 활성화됩니다. Amazon Fraud Detector에서 활동이 발생하면 해당 활동이 CloudTrail 이벤트 기록의 다른 AWS 서비스 이벤트와 함께 이벤트에 기록됩니다. AWS 계정에서 최근 이벤트를 보고, 검색하고, 다운로드할 수 있습니다. 자세한 내용은 이벤트 기록으로 CloudTrail 이벤트 보기를 참조하십시오.

Amazon Fraud Detector의 이벤트를 포함하여 AWS 계정에서 진행 중인 이벤트 기록을 보려면 트레일을 생성하십시오. 트레일을 사용하면 CloudTrail Amazon S3 버킷으로 로그 파일을 전송할 수 있습니다. 콘솔에서 추적을 생성하면 기본적으로 모든 AWS Regions에 추적이 적용됩니다. 트레일은 AWS 파티션에 있는 모든 지역의 이벤트를 기록하고 지정한 Amazon S3 버킷으로 로그 파일을 전송합니다. 또한 CloudTrail 로그에서 수집된 이벤트 데이터를 추가로 분석하고 이에 따라 조치를 취하도록 다른 AWS 서비스를 구성할 수 있습니다. 자세한 내용은 다음을 참조하십시오.

- 추적 생성 개요
- CloudTrail 지원되는 서비스 및 통합
- 에 대한 Amazon SNS 알림 구성 CloudTrail
- 여러 지역에서 CloudTrail 로그 파일 수신 및 여러 계정으로부터 CloudTrail 로그 파일 수신

Amazon Fraud Detector는 모든 작업 (API 작업) 을 CloudTrail 로그 파일에 이벤트로 기록할 수 있도록 지원합니다. 자세한 내용은 작업을 참조하십시오.

모든 이벤트 및 로그 항목에는 요청을 생성한 사용자에 대한 정보가 들어 있습니다. 보안 인증 정보를 이용하면 다음을 쉽게 판단할 수 있습니다.

- 요청을 루트로 했는지 아니면 사용자 자격 증명으로 했는지 여부
- 역할 또는 페더레이션 사용자에 대한 임시 보안 인증을 사용하여 요청이 생성되었는지 여부.
- 요청이 다른 AWS 서비스에 의해 이루어졌는지 여부.

자세한 내용은 CloudTrail 사용자 ID 요소를 참조하십시오.

Amazon Fraud Detector 로그 파일 항목의 이해

트레일은 지정한 Amazon S3 버킷에 이벤트를 로그 파일로 전송할 수 있는 구성입니다. CloudTrail 로그 파일은 하나 이상의 로그 항목을 포함합니다. 이벤트는 모든 소스의 단일 요청을 나타내며 요청된 작업, 작업 날짜 및 시간, 요청 매개 변수 등에 대한 정보를 포함합니다. CloudTrail 로그 파일은 공개 API 호출의 정렬된 스택 트레이스가 아니므로 특정 순서로 표시되지 않습니다.

다음 예제는 `GetDetectors` 작업을 보여주는 CloudTrail 로그 항목을 보여줍니다.


```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "principal-id",
        "arn": "arn:aws:iam::user-arn",
        "accountId": "account-id",
        "accessKeyId": "access-key",
        "userName": "user-name"
    },
    "eventTime": "2019-11-22T02:18:03Z",
    "eventSource": "frauddetector.amazonaws.com",
    "eventName": "GetDetectors",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "source-ip-address",
    "userAgent": "aws-cli/1.11.16 Python/2.7.11 Darwin/15.6.0 botocore/1.4.73",
    "requestParameters": null,
    "responseElements": null,
    "requestID": "request-id",
    "eventID": "event-id",
    "eventType": "AwsApiCall",
    "recipientAccountId": "recipient-account-id"
}
```

문제 해결

다음 섹션은 Amazon Fraud Detector를 사용할 때 발생할 수 있는 문제를 해결하는 데 도움이 됩니다.

교육 데이터 문제 해결

이 섹션의 정보를 사용하면 모델을 학습할 때 Amazon Fraud Detector 콘솔의 모델 교육 진단 창에 표시될 수 있는 문제를 진단하고 해결하는 데 도움이 됩니다.

모델 교육 진단 창에 표시되는 문제는 다음과 같이 분류됩니다. 문제를 해결하기 위한 요구 사항은 문제 범주에 따라 다릅니다.

- 오류 - 모델 학습이 실패합니다. 모델을 성공적으로 학습시키려면 이러한 문제를 해결해야 합니다.
- 경고 - 모델 학습이 계속되지만 일부 변수가 학습 프로세스에서 제외될 수 있습니다. 데이터셋의 품질을 개선하려면 이 섹션의 관련 지침을 확인하세요.
- 정보 (Info) - 모델 학습에는 영향을 미치지 않으며 모든 변수가 학습에 사용됩니다. 데이터셋의 품질과 모델 성능을 더욱 개선하려면 이 섹션의 관련 지침을 확인하는 것이 좋습니다.

주제

해당 데이터셋의 사기 발생률이 불안정합니다.

문제 유형: 오류

설명

주어진 데이터의 사기율이 시간이 지나면서 너무 불안정합니다. 시간이 지나면서 사기 행위와 합법적인 사건이 일관되게 샘플링되는지 확인하십시오.

원인

이 오류는 데이터세트의 사기 및 합법적인 이벤트가 고르지 않게 분산되어 있고 다른 시간대에서 가져온 경우 발생합니다. Amazon Fraud Detector 모델 교육은 EVENT_TIMESTAMP를 기반으로 데이터 세트를 샘플링하고 파티셔닝합니다. 예를 들어, 데이터 세트가 지난 6개월 동안 가져온 사기 이벤트로 구성되어 있지만 마지막 달의 합법적인 이벤트만 포함된 경우 데이터 세트는 불안정한 것으로 간주됩니다. 데이터세트가 불안정하면 모델 성능 평가에 편향이 생길 수 있습니다.

솔루션

부정 행위와 합법적 이벤트 데이터를 같은 시간대에 제공해야 합니다. 그러면 시간이 지나도 사기 발생률이 크게 변하지 않습니다.

데이터 부족

1. 문제 유형: 오류

   설명

   사기 이벤트로 분류되는 행은 50개 미만입니다. 사기 이벤트와 합법적 이벤트 모두 최소 개수인 50개를 초과하는지 확인하고 모델을 다시 학습시키십시오.

   원인

   이 오류는 데이터세트에 모델 학습에 필요한 이벤트 수보다 부정 행위로 분류된 이벤트 수가 적은 경우 발생합니다. Amazon Fraud Detector에서는 모델을 학습시키기 위해 최소 50개의 사기 이벤트가 필요합니다.

   솔루션

   데이터 세트에 최소 50개의 사기 이벤트가 포함되어 있는지 확인하십시오. 필요한 경우 더 긴 기간을 적용하여 이를 보장할 수 있습니다.

2. 문제 유형: 오류

   설명

   50개 미만의 행이 정상 이벤트로 분류됩니다. 사기 이벤트와 합법적 이벤트 모두 최소 개수인 $threshold 를 초과하는지 확인하고 모델을 재학습하십시오.

   원인

   이 오류는 데이터세트에 합법적인 것으로 표시된 이벤트 수가 모델 학습에 필요한 것보다 적은 경우 발생합니다. Amazon Fraud Detector에서는 모델을 학습시키기 위해 최소 50개의 합법적인 이벤트가 필요합니다.

   솔루션

   데이터세트에 최소 50개의 합법적인 이벤트가 포함되어 있는지 확인하십시오. 필요한 경우 더 긴 기간을 적용하여 이를 보장할 수 있습니다.
3. 문제 유형: 오류

   설명

   사기와 관련된 고유 개체 수가 100개 미만입니다. 성과를 개선하려면 사기 조직의 예를 더 많이 포함하는 것을 고려해 보세요.

   원인

   이 오류는 데이터세트에 모델 학습에 필요한 개체 수보다 부정 이벤트가 발생한 개체 수가 적은 경우 발생합니다. 거래 사기 인사이트 (TFI) 모델에서는 사기 영역을 최대한 커버하기 위해 사기 사건이 발생한 개체가 100개 이상 있어야 합니다. 모든 사기 이벤트가 소규모 집단에 의해 수행되는 경우 모델이 제대로 일반화되지 않을 수 있습니다.

   솔루션

   데이터세트에 사기 이벤트가 있는 개체가 100개 이상 포함되어 있는지 확인하세요. 필요한 경우 더 긴 기간을 포함하도록 할 수 있습니다.
4. 문제 유형: 오류

   설명

   합법적인 개체와 관련된 고유 개체의 수가 100개 미만입니다. 성과를 높이려면 합법적인 단체의 예를 더 많이 포함하는 것을 고려해 보세요.

원인

이 오류는 데이터세트에 모델 학습에 필요한 항목 수보다 적으면 정상 이벤트가 발생한 개체 수가 적은 경우에 발생합니다. 거래 사기 인사이트 (TFI) 모델에서는 사기 영역을 최대한 커버하기 위해 합법적인 이벤트가 있는 주체가 100개 이상 있어야 합니다. 모든 합법적인 이벤트가 소규모 집단에 의해 수행되면 모델이 제대로 일반화되지 않을 수 있습니다.

솔루션

데이터셋에 합법적인 이벤트가 있는 개체가 100개 이상 포함되어 있는지 확인하세요. 필요한 경우 더 긴 기간을 포함하도록 할 수 있습니다.

5. 문제 유형: 오류

   설명

   데이터세트의 행이 100개 미만입니다. 전체 데이터세트에 100개 이상의 행이 있고 50개 이상의 행이 허위로 분류되어 있는지 확인하세요.

   원인

   이 오류는 데이터세트에 포함된 레코드가 100개 미만인 경우 발생합니다. Amazon Fraud Detector에서는 모델 교육을 위해 데이터 세트에 있는 최소 100개의 이벤트 (레코드) 데이터를 필요로 합니다.

   솔루션

   데이터세트에 100개가 넘는 이벤트의 데이터가 있는지 확인하십시오.

누락되었거나 다른 EVENT_LABEL 값

1. 문제 유형: 오류

   설명

   EVENT_LABEL 열의 1% 이상이 null이거나 모델 구성에 정의된 값 이외의 값입니다. **$label_values** EVENT_LABEL 열의 누락된 값이 1% 미만이고 해당 값이 모델 구성에 정의된 값인지 확인하십시오. **$label_values**

   원인

이 오류는 다음 원인 중 하나로 인해 발생합니다.

- 훈련 데이터가 들어 있는 CSV 파일의 레코드 중 EVENT_LABEL 열에 누락된 값이 있는 레코드가 1% 를 넘습니다.
- 훈련 데이터가 들어 있는 CSV 파일의 레코드 중 EVENT_LABEL 열에 이벤트 유형과 관련된 값과 다른 값이 있는 레코드가 1% 를 넘습니다.

OFI (온라인 사기 인사이트) 모델에서는 각 레코드의 EVENT_LABEL 열을 이벤트 유형과 관련된 (또는 매핑된) 레이블 중 하나로 채워야 합니다. `CreateModelVersion`

솔루션

누락된 EVENT_LABEL 값으로 인해 이 오류가 발생한 경우 해당 레코드에 적절한 레이블을 할당하거나 데이터세트에서 해당 레코드를 삭제해 보세요. 일부 레코드의 레이블이 포함되지 않아 이 오류가 발생하는 경우**`label_values`**, EVENT_LABEL 열의 모든 값을 이벤트 유형의 레이블에 추가하고 모델 생성 시 사기 또는 합법적 (사기, 합법적) 에 매핑해야 합니다.

2. 문제 유형: 정보

설명

EVENT_LABEL 열에는 모델 구성에 정의된 값 이외의 널 값 또는 레이블 값이 포함되어 있습니다. **`$label_values`** 이러한 일관성 없는 값은 학습 전에 '사기 아님'으로 변환되었습니다.

원인

다음과 같은 이유 중 하나로 인해 이 정보를 얻을 수 있습니다.

- 훈련 데이터가 들어 있는 CSV 파일의 레코드 중 EVENT_LABEL 열에 누락된 값이 있는 레코드는 1% 미만입니다.
- 훈련 데이터를 포함하는 CSV 파일의 레코드 중 EVENT_LABEL 열에 이벤트 유형과 관련된 값과 다른 값이 있는 레코드는 1% 미만입니다.

두 경우 모두 모델 훈련이 성공합니다. 하지만 레이블 값이 누락되거나 매핑되지 않은 이벤트의 레이블 값은 올바른 것으로 변환됩니다. 이것이 문제라고 생각되면 아래 제공된 해결 방법을 따르십시오.

솔루션

데이터세트에 누락된 EVENT_LABEL 값이 있는 경우 데이터세트에서 해당 레코드를 삭제해 보세요. EVENT_LABELS에 제공된 값이 매핑되지 않은 경우 각 이벤트의 모든 값이 사기 또는 합법적(사기, 합법적) 에 매핑되었는지 확인하세요.

누락되거나 잘못된 EVENT_TIMESTAMP 값

1. 문제 유형: 오류

   설명

   훈련 데이터 세트에는 허용된 형식을 준수하지 않는 타임스탬프가 포함된 EVENT_TIMESTAMP가 포함되어 있습니다. 형식이 허용되는 날짜/타임스탬프 형식 중 하나인지 확인하십시오.

   원인

   이 오류는 EVENT_TIMESTAMP 열에 Amazon Fraud Detector에서 지원하는 타임스탬프 형식을 준수하지 않는 값이 포함된 경우 발생합니다.

   솔루션

   EVENT_TIMESTAMP 열에 제공된 값이 지원되는 타임스탬프 형식과 호환되는지 확인하십시오. EVENT_TIMESTAMP 열에 누락된 값이 있는 경우 지원되는 타임스탬프 형식을 사용하여 값을 채우거나, 또는 같은 문자열을 입력하는 대신 이벤트를 완전히 삭제하는 방법을 고려할 수 있습니다. `none` `null` `missing`
2. 문제 유형: 오류

   훈련 데이터 세트에 누락된 값이 있는 EVENT_TIMESTAMP가 포함되어 있습니다. 누락된 값이 없는지 확인하세요.

   원인

   이 오류는 데이터셋의 EVENT_TIMESTAMP 열에 누락된 값이 있는 경우 발생합니다. Amazon Fraud Detector에서는 데이터세트의 EVENT_TIMESTAMP 열에 값이 있어야 합니다.

   솔루션

   데이터세트의 EVENT_TIMESTAMP 열에 값이 있고 해당 값이 지원되는 타임스탬프 형식을 준수하는지 확인하십시오. EVENT_TIMESTAMP 열에 누락된 값이 있는 경우 지원되는 타임스탬프 형식을

사용하여 값을 채우거나, 또는 같은 문자열을 입력하는 대신 이벤트를 완전히 삭제하는 방법을 고려할 수 있습니다. none null missing

데이터가 인제스트되지 않았습니다.

문제 유형: 오류

설명

교육에 필요한 인제스트된 이벤트를 찾을 수 없습니다. 교육 구성을 확인하세요.

원인

이 오류는 Amazon Fraud Detector에 저장된 이벤트 데이터로 모델을 생성하고 있지만 모델 교육을 시작하기 전에 Amazon Fraud Detector로 데이터 세트를 가져오지 않은 경우 발생합니다.

솔루션

Amazon Fraud Detector 콘솔의 CreateBatchImportJob API 작업, API 작업 또는 일괄 가져오기 기능을 사용하여 먼저 이벤트 데이터를 가져온 다음 모델을 학습시킵니다. SendEvent 자세한 내용은 저장된 이벤트 데이터세트를 참조하십시오.

> **Note**
>
> 데이터 가져오기를 마친 후 모델 학습에 사용하기 전에 10분 정도 기다린 후 데이터를 사용하는 것이 좋습니다.

Amazon Fraud Detector 콘솔을 사용하여 각 이벤트 유형에 대해 이미 저장된 이벤트 수를 확인할 수 있습니다. 자세한 내용은 저장된 이벤트의 지표 보기를 참조하십시오.

변수가 충분하지 않음

문제 유형: 오류

설명

데이터셋에는 학습에 적합한 변수가 2개 이상 포함되어야 합니다.

원인

이 오류는 데이터셋에 모델 학습에 적합한 변수가 2개 미만인 경우 발생합니다. Amazon Fraud Detector는 모든 검증을 통과한 경우에만 모델 학습에 적합한 변수를 고려합니다. 변수가 검증에 실패하면 해당 변수는 모델 교육에서 제외되며 모델 교육 진단에서 메시지를 볼 수 있습니다.

솔루션

데이터셋에 값이 채워지고 모든 데이터 검증을 통과한 변수가 두 개 이상 있는지 확인하세요. 단, 열 헤더 (EVENT_TIMESTAMP, EVENT_ID, ENTITY_ID, EVENT_LABEL 등) 를 제공한 이벤트 메타데이터 행은 변수로 간주되지 않습니다.

변수 유형이 누락되었거나 올바르지 않습니다.

문제 유형: 경고

설명

의 예상 데이터 유형은 **$variable_name** 숫자입니다. 데이터세트를 검토 및 **$variable_name** 업데이트하고 모델을 재학습하세요.

원인

변수가 NUMERIC 변수로 정의되어 있지만 데이터세트에 NUMERIC으로 변환할 수 없는 값이 있는 경우 이 경고가 표시됩니다. 따라서 해당 변수는 모델 학습에서 제외됩니다.

솔루션

이 변수를 NUMERIC 변수로 유지하려면 제공하는 값을 부동 숫자로 변환할 수 있어야 합니다. 변수에 누락된 값이 있는 경우, 또는 같은 nonene 문자열로 채우지 마십시오. `null` `missing` 변수에 숫자가 아닌 값이 포함되어 있는 경우 변수를 CATEGORICAL 또는 FREE_FORM_TEXT 변수 유형으로 다시 생성하십시오.

변수 값이 누락되었습니다.

문제 유형: 경고

설명

의 **$threshold** **$variable_name** 값보다 큰 값이 훈련 데이터세트에서 누락되었습니다. 데이터세트를 수정하고 **$variable_name** 성능을 개선하려면 다시 훈련해 보세요.

원인

누락된 값이 너무 많아 지정된 변수가 삭제되는 경우 이 경고가 표시됩니다. Amazon Fraud Detector에서는 변수에 누락된 값을 입력할 수 있습니다. 하지만 한 변수에 결측값이 너무 많으면 모델에 크게 영향을 주지 않으므로 모델 교육 시 해당 변수가 삭제됩니다.

솔루션

먼저, 누락된 값이 데이터 수집 및 준비의 실수로 인한 것이 아닌지 확인하십시오. 실수라면 모델 학습에서 제외하는 방안을 고려해 볼 수 있습니다. 하지만 누락된 값이 가치가 있다고 생각하면서도 해당 변수를 유지하고 싶다면 모델 학습과 실시간 추론 모두에서 누락된 값을 상수로 수동으로 채울 수 있습니다.

고유 변수 값이 충분하지 않습니다.

문제 유형: 경고

설명

의 고유 값 수가 100개 **$variable_name** 미만입니다. 데이터세트를 검토 및 **$variable_name** 업데이트하고 모델을 재학습하세요.

원인

지정된 변수의 고유 값 수가 100개 미만인 경우 이 경고가 표시됩니다. 임계값은 변수 유형에 따라 다릅니다. 고유한 값이 거의 없는 경우 데이터셋이 해당 변수의 특징 공간을 포함할 만큼 충분히 일반적이지 않을 위험이 있습니다. 따라서 모델이 실시간 예측에 대해 잘 일반화되지 않을 수 있습니다.

솔루션

먼저 변수 분포가 실제 비즈니스 트래픽을 나타내는지 확인하십시오. 그런 다음 `full_customer_name` 대신 `last_name` 별도로 사용하는 것과 같이 카디널리티가 더 높은 미세 훈련된 변수를 채택하거나 변수 유형을 CATEGORICAL로 변경하여 카디널리티를 낮출 수 있습니다. `first_name`

잘못된 변수 표현식

1. 문제 유형: 정보

   설명

   50% 이상의 **$email_variable_name** 값이 예상 정규 표현식 http://emailregex.com 과 일치하지 않습니다. 성능 개선을 위해 데이터세트를 **$email_variable_name** 수정하고 다시 학습시키는 것을 고려해 보십시오.

원인

이 정보는 데이터세트의 50% 이상의 레코드에 일반 이메일 표현식을 준수하지 않는 이메일 값이 있어 검증에 실패한 경우 표시됩니다.

솔루션

정규 표현식에 맞게 이메일 변수 값의 형식을 지정합니다. 누락된 이메일 값이 있는 경우 `nonenull`, 또는 같은 문자열로 채우지 말고 비워 두는 것이 좋습니다`missing`.

2. 문제 유형: 정보

설명

50% 이상의 **$IP_variable_name** 값이 IPv4 또는 IPv6 주소의 정규 표현식과 일치하지 않습니다 https://digitalfortress.tech/tricks/top-15 - /. commonly-used-regex 성능 개선을 위해 데이터세트를 수정하고 다시 **$IP_variable_name** 학습시키는 것을 고려해 보세요.

원인

이 정보는 데이터세트의 50% 이상의 레코드가 IP 값이 정규 IP 표현식을 준수하지 않아 검증에 실패한 경우 표시됩니다.

솔루션

정규 표현식을 준수하도록 IP 값의 형식을 지정합니다. 누락된 IP 값이 있는 경우 `nonenull`, 또는 같은 문자열로 채우지 말고 비워 두는 것이 좋습니다`missing`.

3. 문제 유형: 정보

설명

50% 가 넘는 **$phone_variable_name** 값이 기본 전화 정규 표현식 /$pattern/과 일치하지 않습니다. 성능 개선을 위해 데이터세트를 수정하고 **$phone_variable_name** 다시 학습시키는 것을 고려해 보세요.

원인

이 정보는 데이터세트에 있는 레코드의 50% 가 넘는 레코드의 전화번호가 일반 전화번호 표현식을 준수하지 않아 검증에 실패한 경우에 표시됩니다.

솔루션

전화번호의 형식을 정규 표현식에 맞게 지정합니다. 전화번호가 누락된 경우 nonenull, 또는 등의 문자열로 채우지 말고 비워 두는 것이 좋습니다missing.

고유 엔티티가 충분하지 않음

문제 유형: 정보

설명

고유 개체 수가 1500개 미만입니다. 성능을 개선하려면 더 많은 데이터를 포함하는 것을 고려해 보세요.

원인

이 정보는 데이터셋의 고유 항목 수가 권장 수보다 적은 경우 표시됩니다. 거래 사기 인사이트 (TFI) 모델은 시계열 집계와 일반 거래 기능을 모두 사용하여 최상의 성능을 제공합니다. 데이터세트에 포함된 고유 개체가 너무 적은 경우 IP_ADDRESS, EMAIL_ADDRESS와 같은 대부분의 일반 데이터에 고유한 값이 없을 수 있습니다. 그러면 이 데이터셋이 해당 변수의 특징 공간을 포괄할 만큼 충분히 일반적이지 않을 위험도 있습니다. 따라서 새로운 엔티티에서 발생하는 트랜잭션에 대해서는 모델이 제대로 일반화되지 않을 수 있습니다.

솔루션

더 많은 엔티티를 포함하세요. 필요한 경우 훈련 데이터 시간 범위를 확장하세요.

할당량

각 Amazon WebAWS 계정 Service에는 이전에 한도라고 했던 기본 할당량이 있습니다. 다르게 표시되지 않는 한, 리전별로 각 할당량이 적용됩니다. 아래 표에 나와 있는 모든 조정 가능한 할당량에 대해 할당량 증가 요청을 할 수 있습니다. 자세한 내용은 할당량 증가 요청을 참조하세요.

다음 표에는 구성 요소별 Amazon Fraud Detector 할당량이 요약되어 있습니다.

Amazon Fraud Fraud Detector

할당량 이름	기본 할당량	조정 가능
교육 데이터 크기	5GB	아니요
계정당 모델	50	아니요
모델당	200	아니요
계정당 배포된 모델 버전	5	아니요
계정당 동시 교육 작업	3	아니요
모델별 동시 교육 작업	1	아니요

Amazon Fraud Detector 탐지기/변수/결과/규칙

할당량 이름	기본 할당량	조정 가능
계정당	5000	아니요
계정당 규칙	5000	아니요
규칙당	3	아니요
계정당	5000	아니요
계정당	100	아니요

할당량 이름	기본 할당량	조정 가능
검출기별 목록	30	아니요
검출기별 초안 버전	100	아니요
검출기 버전별 모델	10	아니요
계정당	100	아니요
계정당	100	아니요
계정당	100	아니요

Amazon Fraud Fraud Detector

할당량 이름	기본 할당량	조정 가능
GetEventPrediction 초당	200TPS	예
GetEventPrediction API 호출당 페이로드 크기	256KB	아니요
GetEventPrediction API 호출당 입력 수	5000	아니요

문서 기록

다음 표에서는 Amazon 사기 탐지기 사용 설명서의 중요한 변경 사항을 설명합니다. 또한 보내주시는 피드백을 반영하기 위해 Amazon Fraud Detector 사용 설명서를 자주 업데이트합니다.

변경 사항	설명	날짜
새 변수 및 데이터 유형	Amazon Fraud Detector에는 유용한 정보를 추출하는 데 사용할 수 있는 새로운 변수 유형과 데이터 유형이 도입되었습니다.	2023년 6월 5일
이벤트 오케스트레이션	이벤트 오케스트레이션을 사용하면 Amazon을 사용하여 다운스트림 처리를 AWS 서비스 위해 이벤트를 쉽게 전송할 수 있습니다. EventBridge	2023년 5월 30일
리스트	Lists 리소스를 사용하면 규칙의 일부로 IP 주소 또는 이메일 주소와 같은 값 집합을 참조할 수 있습니다. 규칙의 목록을 사용하여 액세스 또는 트랜잭션을 허용하거나 거부할 수 있습니다.	2023년 2월 14일
데이터 모델 탐색기	데이터 모델 탐색기는 Amazon Fraud Detector에서 사기 탐지 모델을 생성하는 데 필요한 데이터 요소에 대한 통찰력을 제공합니다. 이벤트 데이터세트를 준비하기 전에 데이터 모델 탐색기를 사용하세요.	2022년 12월 15일
계정 인계 인사이트 모델	ATI (계정 탈취 인사이트) 모델을 사용하여 악의적인 탈취, 피	2022년 7월 21일

	싱 또는 자격 증명 도난으로 인해 손상된 계정을 탐지할 수 있습니다.	
챕터 업데이트	Amazon Fraud Detector에 대한 추가 정보로 소개 장을 업데이트했습니다.	2022년 4월 11일
변수 강화	제공하는 일부 원시 데이터를 보강하여 이러한 데이터 요소를 사용하고 2022년 2월 8일 이전에 학습된 모델의 성능을 높일 수 있습니다.	2022년 2월 8일
옵트아웃 정책	옵트아웃 정책을 사용하여 이벤트 데이터가 Amazon Fraud Detector의 개발 또는 품질 향상에 사용되는 것을 거부하십시오.	2022년 1월 6일
혼란스러운 대리인 예방	타사 또는 크로스 서비스 엔티티가 자신을 대신하여 행동할 권한이 있는 엔티티를 조작하여 계정의 리소스에 액세스하지 못하도록 하는 정책을 만드세요.	2021년 12월 6일
이벤트 데이터세트 생성	이벤트 데이터세트 만들기에 제공된 지침을 사용하여 모델 학습을 위한 데이터를 준비하고 수집하세요.	2021년 11월 22일
예측에 대한 설명	예측 설명을 사용하면 각 이벤트 변수가 모델의 부정 행위 예측 점수에 어떤 영향을 미쳤는지 파악할 수 있습니다.	2021년 11월 10일

문제 해결	학습 데이터 문제 해결의 정보를 사용하면 모델을 학습할 때 Amazon Fraud Detector 콘솔에 나타날 수 있는 문제를 진단하고 해결하는 데 도움이 됩니다.	2021년 10월 11일
거래 사기 인사이트 모델	거래 사기 인사이트 (TFI) 모델을 사용하여 온라인 또는 card-not-present 거래 사기를 탐지할 수 있습니다.	2021년 10월 11일
저장된 이벤트	이벤트 데이터를 Amazon Fraud Detector에 저장하고 저장된 데이터를 사용하여 나중에 모델을 학습시킬 수 있습니다. Amazon Fraud Detector에 이벤트 데이터를 저장하면 자동 계산된 변수를 사용하는 모델을 학습시켜 성능을 개선하고, 모델 재교육을 간소화하고, 사기 레이블을 업데이트하여 기계 학습 피드백 루프를 종료할 수 있습니다.	2021년 10월 11일
모델 변수 중요도	모델 변수 중요도를 사용하여 모델의 성능을 높이거나 낮추는 요인과 가장 큰 영향을 미치는 모델 변수를 파악할 수 있습니다. 그런 다음 모델을 조정하여 전반적인 성능을 개선하세요.	2021년 7월 9일
AWS CloudFormation와 통합	Amazon 사기 탐지기 리소스를 관리하는 AWS CloudFormation 데 사용합니다.	2021년 5월 10일

배치 예측	배치 예측을 사용하면 실시간 채점이 필요하지 않은 일련의 이벤트에 대한 예측을 얻을 수 있습니다.	2021년 3월 31일
챕터 재작업	시작하기 및 기타 섹션 재작업	2020년 7월 17일
최초 릴리스	최초 릴리스	2019년 12월 2일

기계 번역으로 제공되는 번역입니다. 제공된 번역과 원본 영어의 내용이 상충하는 경우에는 영어 버전이 우선합니다.