aws

사용자 가이드

Amazon Fraud Detector

버전 latest

Amazon Fraud Detector: 사용자 가이드

Table of Contents

Amazon Fraud Detector란 무엇입니까?

Amazon Fraud Detector는 잠재적 사기 활동의 탐지를 온라인에서 자동화하는 완전관리형 사기 탐지 서비스입니다. 이러한 활동에는 무단 트랜잭션과 가짜 계정 생성이 포함됩니다. Amazon Fraud Detector는 기계 학습을 사용하여 데이터를 분석하는 방식으로 작동합니다. 이는 Amazon에서 20년 이상 사기 탐지에 대한 노련한 전문 지식을 기반으로 하는 방식으로 수행됩니다.

Amazon Fraud Detector를 사용하여 사용자 지정 사기 감지 모델을 구축하고, 모델의 사기 평가를 해석하는 결정 로직을 추가하고, 가능한 각 사기 평가에 대한 검토를 위해 통과 또는 전송과 같은 결과를 할당할 수 있습니다. Amazon Fraud Detector를 사용하면 사기 활동을 탐지하는 데 기계 학습 전문 지식이 필요하지 않습니다.

시작하려면 조직에서 수집한 사기 데이터를 수집하고 준비합니다. 그런 다음 Amazon Fraud Detector는이 데이터를 사용하여 사용자를 대신하여 사용자 지정 사기 탐지 모델을 훈련, 테스트 및 배포합니다. 이 프로세스의 일환으로 Amazon Fraud Detector는에서 사기 패턴을 학습한 기계 학습 모델 AWS과 Amazon의 자체 사기 전문 지식을 사용하여 사기 데이터를 평가하고 모델 점수 및 모델 성능 데이터를 생성합니다. 모델의 점수를 해석하고 각 사기 평가를 처리하는 방법에 대한 결과를 할당하도록 결정 로직을 구성합니다.

이점

Amazon Fraud Detector는 다음과 같은 이점을 제공합니다. 이러한 이점을 통해 사기 관리 시스템을 구축하고 유지 관리하는 데 전통적으로 필요한 시간과 리소스를 투자할 필요 없이 사기를 빠르게 감지할 수 있습니다.

자동 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 새 계정 생성, 온라인 결제, 게스트 체크아웃과 같은 온라인 트랜잭션에서 잠재적 사기를 식별할 수 있습니다.

사기 모델은 자동화된 프로세스를 통해 생성되므로 모델 생성 및 훈련과 관련된 많은 단계를 잊어버릴 수 있습니다. 이러한 단계에는 데이터 검증 및 보강, 기능 엔지니어링, 알고리즘 선택, 하이퍼파라미터 튜닝 및 모델 배포가 포함됩니다.

Amazon Fraud Detector를 사용하여 사기 탐지 모델을 생성하려면 회사의 과거 사기 데이터 세트만 업로드하고 모델 유형을 선택합니다. 그런 다음 Amazon Fraud Detector는 사용 사례에 가장 적합한 사기

탐지 알고리즘을 자동으로 찾아 모델을 생성합니다. 사기 탐지 모델을 생성하기 위해 코딩을 알고 있거나 기계 학습 전문 지식이 있어야 할 필요는 없습니다.

진화하고 학습하는 사기 모델

사기 탐지 모델은 변화하는 사기 상황을 따라잡기 위해 지속적으로 진화해야 합니다. Amazon Fraud Detector는 계정 연령, 마지막 활동 이후 경과 시간 및 활동 수를 포함한 정보를 계산하여 이를 자동으로 수행합니다. 그 결과 모델은 거래를 자주 하는 신뢰할 수 있는 고객과 사기범의 일반적인 지속적인 시도 간의 차이를 학습합니다. 이렇게 하면 재훈련 세션 간에 모델의 성능을 더 오래 유지할 수 있습니다.

사기 모델 성능 시각화

제공한 데이터를 사용하여 모델을 훈련한 후 Amazon Fraud Detector는 모델 성능을 검증합니다. 또한 성능을 평가할 수 있는 시각적 도구도 제공합니다. 훈련하는 각 모델에 대해 모델 성능 점수, 점수 분포 그래프, 혼동 행렬, 임계값 테이블 및 모델 성능에 미치는 영향에 따라 순위를 매긴 모든 입력을 볼 수 있습니다. 이러한 성능 도구를 사용하여 모델이 어떻게 수행되고 있는지, 어떤 입력이 모델 성능을 좌우하는지 알아볼 수 있습니다. 필요한 경우 모델을 조정하여 전체 성능을 개선할 수 있습니다.

사기 예측

Amazon Fraud Detector는 조직의 비즈니스 활동에 대한 사기 예측을 생성합니다. 사기 예측은 사기 위험에 대한 비즈니스 활동의 평가입니다. Amazon Fraud Detector는 활동과 연결된 데이터와 함께 예측 로직을 사용하여 예측을 생성합니다. 사기 탐지 모델을 생성할 때이 데이터를 제공했습니다. 단일 활동에 대한 사기 예측을 실시간으로 가져오거나 일련의 활동에 대한 사기 예측을 오프라인으로 가져올 수 있습니다.

사기 예측 설명 시각화

Amazon Fraud Detector는 사기 예측 프로세스의 일부로 예측 설명을 생성합니다. 예측 설명은 모델을 훈련하는 데 사용되는 각 데이터 요소가 모델의 사기 예측 점수에 어떤 영향을 미쳤는지에 대한 통찰력을 제공합니다. 예측 설명은 테이블 및 그래프와 같은 시각적 도구를 사용하여 제공됩니다. 이러한 도구를 사용하여 각 데이터 요소가 예측 점수에 미치는 영향을 시각적으로 식별할 수 있습니다. 그런 다음이 정보를 사용하여 데이터 세트 전반의 사기 패턴을 분석하고 편향이 있는 경우 이를 감지할 수 있습니다. 마지막으로 예측 설명을 사용하여 수동 사기 조사 프로세스 중에 주요 위험 지표를 식별할 수도 있습니다. 이렇게 하면 거짓 긍정 예측으로 이어지는 근본 원인을 좁힐 수 있습니다.

규칙 기반 작업

사기 탐지 모델을 훈련한 후 규칙을 추가하여 데이터 수락, 검토를 위한 데이터 전송 또는 더 많은 데이터 수집과 같이 평가된 데이터에 대한 조치를 취할 수 있습니다. 규칙은 사기 예측 중에 데이터를 해석

하는 방법을 Amazon Fraud Detector에 알려주는 조건입니다. 예를 들어, 검토할 의심스러운 고객 계정에 플래그를 지정하는 규칙을 생성할 수 있습니다. 감지된 모델 점수가 미리 결정된 임계값보다 크고 계정 결제의 권한 부여 코드(AUTH_CODE)가 유효하지 않은 경우이 규칙을 시작하도록 설정할 수 있습니다.

핵심 개념 및 용어

다음은 Amazon Fraud Detector에 사용되는 핵심 개념 및 용어 목록입니다.

Event

이벤트는 사기 위험에 대해 평가되는 조직의 비즈니스 활동입니다. Amazon Fraud Detector는 이벤트에 대한 사기 예측을 생성합니다.

레이블

레이블은 단일 이벤트를 사기 또는 합법적으로 분류합니다. 레이블은 Amazon Fraud Detector에서 기계 학습 모델을 훈련하는 데 사용됩니다.

개체

엔터티는 이벤트를 수행 중인 사용자를 나타냅니다. 이벤트를 수행한 특정 엔터티를 나타내기 위해 회사 사기 데이터의 일부로 엔터티 ID를 제공합니다.

이벤트 유형

이벤트 유형은 Amazon Fraud Detector로 전송되는 이벤트의 구조를 정의합니다. 여기에는 이벤트의 일부로 전송된 데이터, 이벤트를 수행하는 개체(예: 고객), 이벤트를 분류하는 레이블이 포함됩니다. 예제 이벤트 유형에는 온라인 결제 트랜잭션, 계정 등록 및 인증이 포함됩니다.

엔터티 유형

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

이벤트 데이터 세트

이벤트 데이터 세트는 특정 비즈니스 활동 또는 이벤트에 대한 회사의 기록 데이터입니다. 예를 들어 회사의 이벤트는 온라인 계정 등록일 수 있습니다. 단일 이벤트(등록)의 데이터에는 연결된 IP 주소, 이메일 주소, 결제 주소 및 이벤트 타임스탬프가 포함될 수 있습니다. Amazon Fraud Detector에 이벤트 데이터 세트를 제공하여 사기 탐지 모델을 생성하고 훈련합니다.

모델

모델은 기계 학습 알고리즘의 출력입니다. 이러한 알고리즘은 코드로 구현되며 사용자가 제공하는 이벤트 데이터에 대해 실행됩니다.

모델 유형

모델 유형은 모델 훈련 중에 사용되는 알고리즘, 보강 및 기능 변환을 정의합니다. 또한 모델 훈련을 위한 데이터 요구 사항도 정의합니다. 이러한 정의는 특정 유형의 사기에 맞게 모델을 최적화하는 기능을 합니다. 모델을 생성할 때 사용할 모델 유형을 지정합니다.

모델 훈련

모델 훈련은 제공된 이벤트 데이터 세트를 사용하여 사기 이벤트를 예측할 수 있는 모델을 생성하는 프로세스입니다. 모델 훈련 프로세스의 모든 단계는 완전히 자동화됩니다. 이러한 단계에는 데이터 검증, 데이터 변환, 특성 엔지니어링, 알고리즘 선택 및 모델 최적화가 포함됩니다.

모델 점수

모델 점수는 회사의 과거 사기 데이터의 평가 결과입니다. 모델 훈련 프로세스 중에 Amazon Fraud Detector는 데이터 세트의 사기 활동을 평가하고 0~1000 사이의 점수를 생성합니다. 이 점수의 경우 0은 사기 위험이 낮음을 나타내고 1000은 사기 위험이 가장 높음을 나타냅니다. 점수 자체는 거짓 긍정 비율(FPR)과 직접 관련이 있습니다.

모델 버전

모델 버전은 모델 훈련의 출력입니다.

모델 배포

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

Amazon SageMaker AI 모델 엔드포인트

Amazon Fraud Detector를 사용하여 모델을 빌드하는 것 외에도 선택적으로 Amazon Fraud Detector 평가에서 SageMaker AI 호스팅 모델 엔드포인트를 사용할 수 있습니다.

SageMaker AI에서 모델을 빌드하는 방법에 대한 자세한 내용은 를 사용하여 모델 훈련을 Amazon SageMaker AI 참조하세요.

감지기

감지기에는 사기에 대해 평가하려는 특정 이벤트에 대한 모델 및 규칙과 같은 감지 로직이 포함되어 있습니다. 모델 버전을 사용하여 감지기를 생성합니다.

Detector 버전

감지기에는 여러 버전이 있을 수 있으며 각 버전은 `Draft`, `Active`또는의 상태를 갖습니다`Inactive`. 한 번에 하나의 감지기 버전만 `Active` 상태가 될 수 있습니다.

변수

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

규칙

규칙은 사기 예측 중에 변수 값을 해석하는 방법을 Amazon Fraud Detector에 알려주는 조건입니다. 규칙은 하나 이상의 변수, 로직 표현식 및 하나 이상의 결과로 구성됩니다. 규칙에 사용되는 변수는 감지기가 평가하는 이벤트 데이터 세트의 일부여야 합니다. 또한 각 감지기에는 하나 이상의 규칙이 연결되어 있어야 합니다.

결과

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

사기 예측

사기 예측은 단일 이벤트 또는 이벤트 세트에 대한 사기 평가입니다. Amazon Fraud Detector는 규칙에 따라 모델 점수와 결과를 동기식으로 제공하여 단일 온라인 이벤트에 대한 사기 예측을 실시간으로 생성합니다. Amazon Fraud Detector는 오프라인에서 이벤트 세트에 대한 사기 예측을 생성합니다. 예측을 사용하여 오프라인 proof-of-concept 수행하거나 시간별, 일별 또는 주별로 사기 위험을 소급 평가할 수 있습니다.

사기 예측 설명

사기 예측 설명은 각 변수가 모델의 사기 예측 점수에 어떤 영향을 미쳤는지에 대한 통찰력을 제공합니다. 각 변수가 위험 점수의 규모(0~5 범위, 5는 가장 높음) 및 방향(점수를 더 높거나 낮게 주행) 측면에서 위험 점수에 미치는 영향에 대한 정보를 제공합니다.

Amazon Fraud Detector 작동 방식

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

자동화된 모델 훈련 프로세스에는 특정 비즈니스 사용 사례에 대한 사기를 감지하는 기계 학습 알고리즘을 선택하고, 제공한 데이터를 검증하고, 모델 성능을 개선하기 위해 데이터 조작을 수행하는 작업이

포함됩니다. 모델을 훈련한 후 Amazon Fraud Detector는 모델 점수 및 기타 모델 성능 지표를 생성합니다. 점수와 성능 지표를 사용하여 모델 성능을 평가할 수 있습니다. 필요한 경우 훈련을 위해 제공한 데이터 세트에서 데이터 요소를 추가하거나 제거하고 모델을 재훈련하여 모델 점수를 개선할 수 있습니다.

모델을 생성, 훈련 및 활성화한 후에는 규칙이라고도 하는 결정 로직을 구성하여 비즈니스에서 생성된 데이터를 해석하는 방법을 모델에 알리고 각 활동의 해석을 처리하는 방법에 대한 결과를 할당해야 합니다. 결과는 활동을 승인 또는 검토하는 등의 작업을 나타내거나, 고위험, 중간 위험 및 저위험과 같은 활동의 위험 수준을 나타낼 수 있습니다.

감지기는 모델과 관련 규칙을 보관하는 컨테이너입니다. 감지기를 생성, 테스트하고 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 생성한 규칙과 사용하여 결과를 분석하고 결과를 할당합니다. Amazon Fraud Detector를 사용하면 단일 비즈니스 활동의 데이터를 실시간으로 평가하거나 오프라인으로 여러 비즈니스 활동의 데이터를 평가할 수 있습니다.

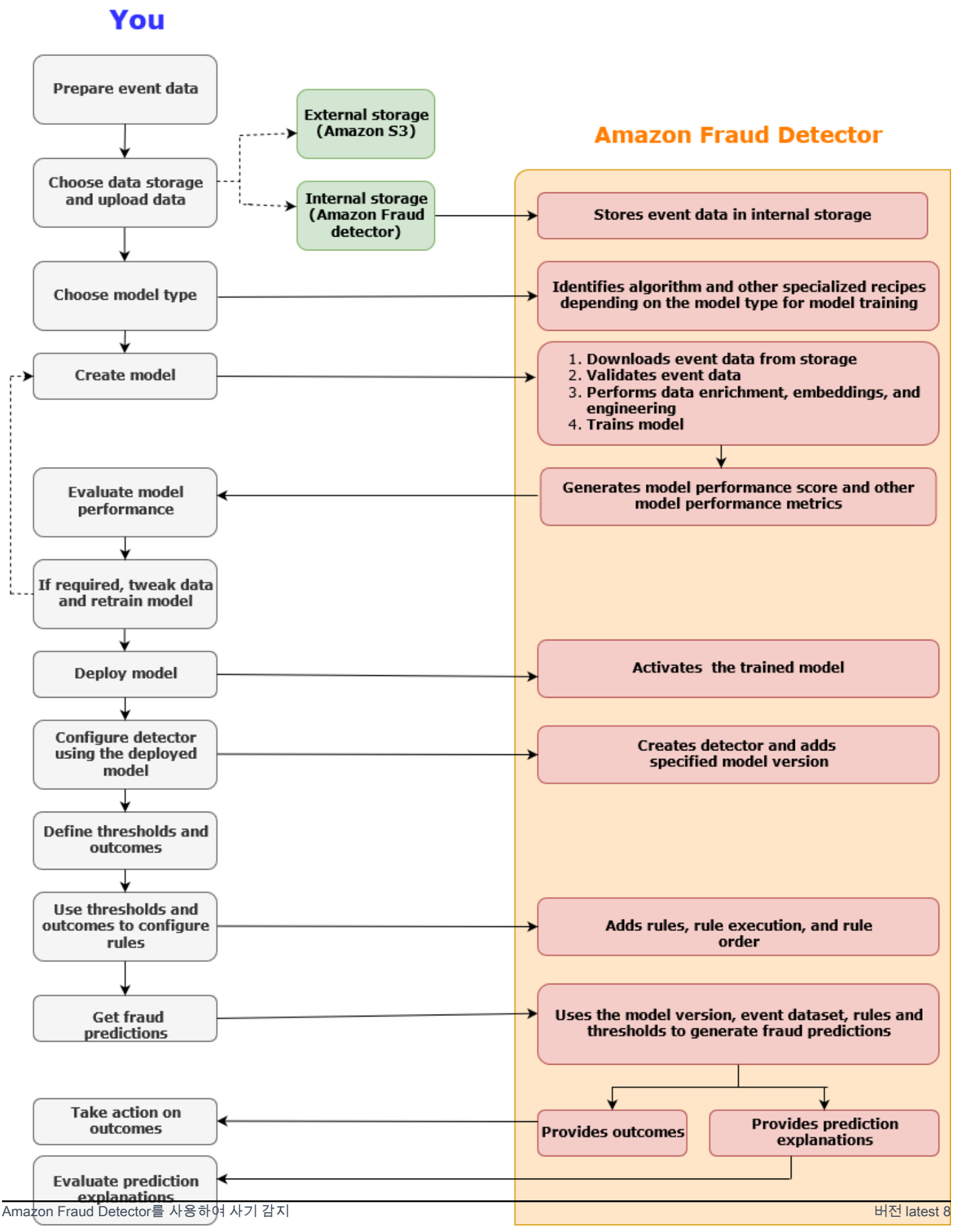
활동 중 하나로 온라인 자금 이체를 하는 사업체가 있다고 가정해 보겠습니다. Amazon Fraud Detector를 사용하여 실시간으로 자금 이체에 대한 사기 요청을 탐지하려고 합니다. 시작하려면 먼저 Amazon Fraud Detector에 과거 자금 이체 요청의 데이터를 제공해야 합니다. Amazon Fraud Detector는이 데이터를 사용하여 자금 이체에 대한 사기 요청을 감지하도록 사용자 지정된 모델을 생성하고 훈련합니다. 그런 다음 모델을 추가하고 모델을 위한 규칙을 구성하여 데이터를 해석하여 감지기를 생성합니다. 온라인 자금 이체 활동에 대한 규칙의 예로는 xyz@example.com 이메일 주소에서 자금 이체 요청이 오는 경우 검토를 위해 요청을 보내는 것이 있습니다. 비즈니스의 프로덕션 환경에서 자금 이체 요청이 수신되면 모델은 요청과 함께 제공된 데이터를 분석하고 규칙을 사용하여 결과를 할당합니다. 그런 다음 할당된 결과에 따라 요청에 대한 작업을 수행할 수 있습니다.

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

Amazon Fraud Detector를 사용하여 사기를 감지하는 데 사용할 워크플로에 대한 자세한 내용은 섹션을 참조하세요. Amazon Fraud Detector를 사용하여 사기 감지

Amazon Fraud Detector를 사용하여 사기 감지

이 섹션에서는 Amazon Fraud Detector를 사용하여 사기를 탐지하기 위한 일반적인 워크플로를 설명합니다. 또한 이러한 작업을 수행하는 방법을 요약합니다. 다음 다이어그램은 Amazon Fraud Detector를 사용하여 사기를 탐지하기 위한 워크플로의 개략적인 보기를 제공합니다.


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 후 예측 설명을 기반으로 성능 점수와 지표를 평가해야 합니다. 이렇게 하면 주요 위험 지표를 식별하고, 오탐지로 이어지는 근본 원인을 좁히고, 데이터 세트 전반의 사기 패턴을 분석하고, 편향이 있는 경우 이를 감지할 수 있습니다. 예측의 정확도를 높이기 위해 데이터세트를 조정하여 새 데이터 또는 수정된 데이터를 포함할 수 있습니다. 그런 다음 업데이트된 데이터 세트로 모델을 재학습할 수 있습니다. 더 많은 데이터를 사용할 수 있게 되면 정확도를 높이기 위해 모델을 계속 재학습합니다.

Amazon Fraud Detector에 액세스

Amazon Fraud Detector는 여러에서 사용할 수 AWS 리전 있으며 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, 훈련, 배포, 테스트, 실행 및 관리할 수 있습니다.

AWS Management Console - Amazon Fraud Detector는 웹 기반 사용자 인터페이스인 Amazon Fraud Detector 콘솔을 제공합니다. 에 가입한 경우 Amazon Fraud Detector 콘솔에 액세스할 AWS 계정수 있습니다. 자세한 내용은 Amazon Fraud Detector 설정을 참조하세요.

AWS Command Line Interface (AWS CLI) - 명령줄 셸의 명령을 사용하여 Amazon Fraud Detector를 AWS 서비스비롯한 광범위한와 상호 작용하는 데 사용할 수 있는 인터페이스를 제공합니다. Amazon Fraud Detector에 대한 AWS CLI 명령은 Amazon Fraud Detector 콘솔에서 제공하는 기능과 동일한 기능을 구현합니다.

AWS SDK - 언어별 APIs 제공하고 서명 계산, 요청 재시도 처리 및 오류 처리와 같은 많은 연결 세부 정보를 관리합니다. 자세한 내용은 빌드할 도구 AWS 페이지로 이동하여 SDK 섹션으로 스크롤한 다음 더하기(+) 기호를 선택하여 섹션을 확장합니다.

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 참조하세요.

요금

Amazon Fraud Detector를 사용하면 사용한 만큼만 비용을 지불하면 됩니다. 최소 요금이나 사전 약정은 없습니다. 모델을 훈련하고 호스팅하는 데 사용되는 컴퓨팅 시간, 사용하는 스토리지 양 및 사기 예측량을 기준으로 요금이 부과됩니다. 자세한 내용은 Amazon Fraud Detector 요금을 참조하세요.

Amazon Fraud Detector 설정

Amazon Fraud Detector를 사용하려면 먼저 Amazon Web Services(AWS) 계정이 필요하며 모든 인터페이스에 대한 AWS 계정 액세스 권한을 부여하는 권한을 설정해야 합니다. 나중에 Amazon Fraud Detector 리소스를 생성하기 시작할 때 Amazon Fraud Detector가 계정에 액세스하여 사용자를 대신하여 작업을 수행하고 소유한 리소스에 액세스할 수 있는 권한을 부여해야 합니다.

Amazon Fraud Detector 사용을 설정하려면이 섹션의 다음 작업을 완료하세요.

- 가입합니다 AWS.
- 가 Amazon Fraud Detector 인터페이스 AWS 계정 에 액세스할 수 있도록 권한을 설정합니다.
- Amazon Fraud Detector에 액세스하는 데 사용할 인터페이스를 설정합니다.

이 단계를 완료한 후 Amazon Fraud Detector를 계속 시작하려면 Amazon Fraud Detector 시작하기 섹션을 참조하세요.

에 가입 AWS

Amazon Web Services(AWS)에 가입하면 AWS 계정 가 Amazon Fraud Detector를 AWS포함한의 모든 서비스에 자동으로 등록됩니다. 사용한 서비스에 대해서만 청구됩니다. 이미이 있는 경우 다음 작업으로 AWS 계정건너뜁니다.

에 가입 AWS 계정

이 없는 경우 다음 단계를 AWS 계정완료하여 생성합니다.

에 가입하려면 AWS 계정

1. https://portal.aws.amazon.com/billing/signup을 엽니다.
2. 온라인 지시 사항을 따릅니다.

   가입 절차의 일부로는 전화 또는 문자 메시지를 수신하고 전화 키패드에 확인 코드를 입력하는 것이 포함됩니다.

   에 가입하면 AWS 계정AWS 계정 루트 사용자이 생성됩니다. 루트 사용자에게는 계정의 모든 AWS 서비스 및 리소스에 액세스할 권한이 있습니다. 보안 모범 사례는 사용자에게 관리 액세스

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

AWS 는 가입 프로세스가 완료된 후 확인 이메일을 보냅니다. 언제든지 https://aws.amazon.com/으로 이동하고 내 계정을 선택하여 현재 계정 활동을 보고 계정을 관리할 수 있습니다.

관리자 액세스 권한이 있는 사용자 생성

에 가입한 후 일상적인 작업에 루트 사용자를 사용하지 않도록 관리 사용자를 AWS 계정보호 AWS IAM Identity Center, AWS 계정 루트 사용자활성화 및 생성합니다.

보안 AWS 계정 루트 사용자

1. 루트 사용자를 선택하고 AWS 계정 이메일 주소를 입력하여 계정 소유자AWS Management Console로에 로그인합니다. 다음 페이지에서 비밀번호를 입력합니다.

   루트 사용자를 사용하여 로그인하는 데 도움이 필요하면 AWS 로그인 User Guide의 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 참조하세요.

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 자격 증명 소스 IAM Identity Center 디렉터리 로 사용하는 방법에 대한 자습서는 사용 AWS IAM Identity Center 설명서의 기본값으로 사용자 액세스 구성을 IAM Identity Center 디렉터리 참조하세요.

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 AWS 로그인 사용 설명서의 AWS 액세스 포털에 로그인을 참조하세요.

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.

   지침은 AWS IAM Identity Center 사용 설명서의 Add groups를 참조하세요.

Amazon Fraud Detector 인터페이스에 액세스할 수 있는 권한 설정

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

보안 모범 사례에 따라 Amazon Fraud Detector 작업으로 액세스가 제한되고 필요한 권한이 있는 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
  - SageMaker AI의 모든 모델 엔드포인트 나열 및 설명
  - 계정의 모든 IAM 역할 나열
  - 모든 Amazon S3 버킷 나열
  - IAM 역할 전달이 Amazon Fraud Detector에 역할을 전달하도록 허용
- `AmazonS3FullAccess`

  에 대한 전체 액세스를 허용합니다 Amazon Simple Storage Service. 이는 훈련 데이터 세트를 Amazon S3에 업로드해야 하는 경우에 필요합니다.

다음은 IAM 사용자를 생성하고 필요한 권한을 할당하는 방법을 설명합니다.

사용자를 생성하고 필요한 권한을 할당하려면

1. 에 로그인 AWS Management Console 하고 https://console.aws.amazon.com/iam/ 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 Management Console 액세스 확인란을 선택한 다음 사용자 암호를 구성합니다.
5. (선택 사항) 기본적으로 AWS 에서는 새 사용자가 처음 로그인할 때 새 암호를 생성해야 합니다. 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: 권한을 선택합니다.
7. 그룹 생성을 선택합니다.
8. 그룹 이름에를 입력합니다**AmazonFraudDetectorGroup**.
9. 정책 목록에서 AmazonFraudDetectorFullAccessPolicy 및 AmazonS3FullAccess의 확인란을 선택합니다. 그룹 생성을 선택합니다.
10. 그룹 목록에서 새로운 그룹의 확인란을 선택합니다. 목록에 그룹이 표시되지 않으면 새로 고침을 선택합니다.
11. 다음: 태그를 선택합니다.
12. (선택 사항) 태그를 키 값 페어로 연결하여 메타데이터를 사용자에게 추가합니다. IAM에서 태그를 사용하는 방법에 대한 지침은 IAM 사용자 및 역할 태그 지정을 참조하세요.
13. 다음: 검토를 선택하여 새 사용자의 사용자 세부 정보 및 권한 요약을 확인합니다. 계속 진행할 준비가 되었으면 사용자 생성을 선택합니다.

를 사용하여 Amazon Fraud Detector에 액세스하도록 인터페이스 설정

Amazon Fraud Detector 콘솔 AWS CLI또는 AWS SDK를 사용하여 Amazon Fraud Detector에 액세스할 수 있습니다. 이를 사용하려면 먼저 AWS CLI 및 AWS SDK를 설정합니다.

Amazon Fraud Detector 콘솔에 액세스

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

Amazon Fraud Detector 콘솔에 액세스하려면

1. 로 이동하여에 https://console.aws.amazon.com/ 로그인합니다 AWS 계정.
2. Amazon Fraud Detector로 이동합니다.

Amazon Fraud Detector 콘솔을 사용하면 모델과 감지기, 변수, 이벤트, 개체, 레이블 및 결과와 같은 사기 탐지 리소스를 생성하고 관리할 수 있습니다. 예측을 생성하고 모델의 성능과 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 AWS CLI 를 사용하여 터미널의 명령 프롬프트에서 Amazon Fraud Detector 콘솔에서 제공하는 것과 유사한 기능에 대한 명령을 실행할 수 있습니다.

를 설정하려면 AWS CLI

AWS CLI를 다운로드하고 구성합니다. 지침은 AWS Command Line Interface 사용 설명서의 다음 주제를 참조하세요.

- AWS 명령줄 인터페이스 설정
- AWS 명령줄 인터페이스 구성

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

AWS SDK 설정

AWS SDKs를 사용하여 사기 탐지 리소스를 생성 및 관리하고 사기 예측을 가져오는 코드를 작성할 수 있습니다. AWS SDKs는 JavaScript 및 Python(Boto3)에서 Amazon Fraud Detector를 지원합니다.

설정하려면 AWS SDK for Python (Boto3)

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

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

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

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

Java용 AWS SDKs를 설정하려면

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

Amazon Fraud Detector 시작하기

시작하기 전에의 단계를 Amazon Fraud Detector를 사용하여 사기 감지 읽고 완료했는지 확인하세요Amazon Fraud Detector 설정.

이 섹션의 실습 자습서를 사용하여 Amazon Fraud Detector를 사용하여 사기 탐지 모델을 구축, 훈련 및 배포하는 방법을 알아봅니다. 이 자습서에서는 기계 학습 모델을 사용하여 새 계정 등록이 사기인지 예측하는 사기 분석가의 역할을 맡습니다. 모델은 계정 등록의 데이터를 사용하여 훈련해야 합니다. Amazon Fraud Detector는이 자습서의 계정 등록 데이터 세트 예제를 제공합니다. 자습서를 시작하기 전에 예제 데이터 세트를 업로드해야 합니다.

다음 인터페이스 중 하나를 사용하여 Amazon Fraud Detector를 시작할 수 있습니다. 자습서를 시작하기 전에에 대한 지침을 따라야 합니다. 예제 데이터 세트 가져오기 및 업로드

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

예제 데이터 세트 가져오기 및 업로드

이 자습서에서 사용하는 예제 데이터 세트는 온라인 계정 등록에 대한 세부 정보를 제공합니다. 데이터 세트는 UTF-8 형식의 쉼표로 구분된 값(CSV)을 사용하는 텍스트 파일에 있습니다. CSV 데이터 세트 파일의 첫 번째 행에는 헤더가 포함되어 있습니다. 헤더 행 뒤에 여러 데이터 행이 옵니다. 이러한 각 행은 단일 계정 등록의 데이터 요소로 구성됩니다. 편의를 위해 데이터에 레이블이 지정됩니다. 데이터 세트의 열은 계정 등록이 사기인지 여부를 식별합니다.

예제 데이터 세트를 가져오고 업로드하려면

1. 샘플로 이동합니다.

   온라인 계정 등록 데이터가 있는 데이터 파일은 registration_data_20K_minimum.csv와 registration_data_20K_full.csv 두 개입니다. 파일에는 ip_address와 email_address라는 두 가지 변수만 `registration_data_20K_minimum` 포함되어 있습니다. 파일에는 다른 변수가 `registration_data_20K_full` 포함되어 있습니다. 이러한 변수는 각 이벤트에 대한 것이며 billing_address, phone_number 및 user_agent를 포함합니다. 두 데이터 파일에는 두 개의 필수 필드도 포함되어 있습니다.

   - EVENT_TIMESTAMP - 이벤트가 발생한 시기를 정의합니다.
   - EVENT_LABEL - 이벤트를 사기 또는 합법적으로 분류합니다.

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

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

   이 단계에서는 데이터 세트를 저장할 외부 스토리지를 생성합니다. 이 외부 스토리지는 Amazon S3 버킷입니다. Amazon S3에 대한 자세한 내용은 Amazon S3란 무엇인가요?를 참조하세요.

   a. 에 로그인 AWS Management Console 하고 https://console.aws.amazon.com/s3/ 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 제공해야 합니다. 버킷의 목적을 설명하는 이름을 사용하는 것이 좋습니다.
   d. 에서 버킷을 생성할 AWS 리전 를 AWS 리전선택합니다. 선택한 리전은 Amazon Fraud Detector를 지원해야 합니다. 지연 시간을 줄이려면 지리적 위치에 가장 가까운 AWS 리전 를 선택합니다. Amazon Fraud Detector를 지원하는 리전 목록은 글로벌 인프라 안내서의 리전 테이블을 참조하세요.
   e. 이 자습서에서는 객체 소유권에 대한 기본 설정, 퍼블릭 액세스 차단에 대한 버킷 설정, 버킷 버전 관리 및 태그를 그대로 둡니다.
   f. 기본 암호화의 경우이 자습서에서 비활성화를 선택합니다.
   g. 버킷 구성을 검토한 다음 버킷 생성을 선택합니다.

3. Amazon S3 버킷에 예제 데이터 파일을 업로드합니다.

   이제 버킷이 있으므로 이전에 다운로드한 예제 파일 중 하나를 방금 생성한 Amazon S3 버킷에 업로드합니다.

   a. 버킷에는 버킷 이름이 나열됩니다. 버킷을 선택합니다.
   b. 업로드를 선택합니다.
   c. 파일 및 폴더에서 파일 추가를 선택합니다.
   d. 컴퓨터에서 다운로드한 예제 데이터 파일 중 하나를 선택한 다음 열기를 선택합니다.
   e. 대상, 권한 및 속성에 대한 기본 설정을 그대로 둡니다.
   f. 구성을 검토한 다음 업로드를 선택합니다.
   g. 예제 데이터 파일은 Amazon S3 버킷에 업로드됩니다. 버킷 위치를 기록해 둡니다. 객체에서 방금 업로드한 예제 데이터 파일을 선택합니다.

h. 객체 개요에서 S3 URI 아래에 위치를 복사합니다. 이 위치는 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 자습서는 두 부분으로 구성됩니다. 첫 번째 부분에서는 사기 탐지 모델을 빌드, 훈련 및 배포하는 방법을 설명합니다. 두 번째 부분에서는 모델을 사용하여 사기 예측을 실시간으로 생성하는 방법을 다룹니다. 모델은 S3 버킷에 업로드하는 예제 데이터 파일을 사용하여 훈련됩니다. 이 자습서를 마치면 다음 작업을 완료합니다.

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

> **Important**
>
> 계속하기 전에에 대한 지침을 따랐는지 확인합니다. 예제 데이터 세트 가져오기 및 업로드

파트 A: Amazon Fraud Detector 모델 구축, 훈련 및 배포

파트 A에서는 비즈니스 사용 사례를 정의하고, 이벤트를 정의하고, 모델을 빌드하고, 모델을 훈련하고, 모델의 성능을 평가하고, 모델을 배포합니다.

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 생성하고 훈련하는 데 사용할 모델 유형을 권장합니다. 또한 데이터 모델 탐색기는 데이터 세트에 포함해야 하는 필수, 권장 및 선택적 데이터 요소에 대한 인사이트를 제공합니다. 데이터 세트는 사기 탐지 모델을 생성하고 훈련하는 데 사용됩니다.

  이 자습서에서 비즈니스 사용 사례는 새 계정 등록입니다. 비즈니스 사용 사례를 지정하면 데이터 모델 탐색기는 사기 탐지 모델을 생성하기 위한 모델 유형을 권장하고 데이터 세트를 생성하는 데 필요한 데이터 요소 목록도 제공합니다. 새 계정 등록의 데이터가 포함된 샘플 데이터 세트를 이미 업로드했으므로 새 데이터 세트를 생성할 필요가 없습니다.

a. AWS Management Console을 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 훈련하는 데 사용할 알고리즘, 보강 및 변환을 정의합니다.

   권장 모델 유형을 기록해 둡니다. 나중에 모델을 생성할 때이 정보가 필요합니다.

e. 데이터 모델 인사이트 창은 사기 탐지 모델을 생성하고 훈련하는 데 필요한 필수 및 권장 데이터 요소에 대한 인사이트를 제공합니다.

   다운로드한 샘플 데이터 세트를 살펴보고 테이블에 나열된 모든 필수 및 권장 데이터 요소가 있는지 확인합니다.

   나중에 특정 비즈니스 사용 사례에 맞는 모델을 생성할 때 제공된 인사이트를 사용하여 데이터 세트를 생성합니다.

2단계: 이벤트 유형 생성

- 이 단계에서는 사기를 평가할 비즈니스 활동(이벤트)을 정의합니다. 이벤트를 정의하려면 데이터 세트에 있는 변수, 이벤트를 시작하는 개체 및 이벤트를 분류하는 레이블을 설정해야 합니다. 이 자습서에서는 계정 등록 이벤트를 정의합니다.

  a. AWS Management Console을 열고 계정에 로그인합니다. Amazon Fraud Detector로 이동합니다.

  b. 왼쪽 탐색 창에서 이벤트를 선택합니다.

  c. 이벤트 유형 페이지에서 생성을 선택합니다.

  d. 이벤트 유형 세부 정보에서 이벤트 유형 이름으로 `sample_registration`를 입력하고 선택적으로 이벤트에 대한 설명을 입력합니다.

  e. 개체에서 개체 생성을 선택합니다.

  f. 개체 생성 페이지에서 개체 유형 이름으로 `sample_customer`를 입력합니다. 선택적으로 개체 유형에 대한 설명을 입력합니다.

  g. 개체 생성을 선택합니다.

h. 이벤트 변수의 이 이벤트의 변수를 정의하는 방법 선택에서 훈련 데이터 세트에서 변수 선택을 선택합니다.

i. IAM 역할에서 IAM 역할 생성을 선택합니다.

j. IAM 역할 생성 페이지에서 예제 데이터를 업로드한 S3 버킷의 이름을 입력하고 역할 생성을 선택합니다.

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

l. 업로드를 선택합니다.

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

m. 레이블 - 선택 사항에서 레이블에 대해 새 레이블 생성을 선택합니다.

n. 레이블 생성 페이지에서 이름으로 `fraud`를 입력합니다. 이 레이블은 예제 데이터 세트의 사기 계정 등록을 나타내는 값에 해당합니다.

o. 레이블 생성을 선택합니다.

p. 두 번째 레이블을 생성한 다음 이름으로 `legit`를 입력합니다. 이 레이블은 예제 데이터 세트의 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 정의에 모델 이름으로 `sample_fraud_detection_model`를 입력합니다. 선택적으로 모델에 대한 설명을 추가합니다.
3. 모델 유형에서 온라인 사기 인사이트 모델을 선택합니다.
4. 이벤트 유형에서 sample_registration을 선택합니다. 1단계에서 생성한 이벤트 유형입니다.
5. 과거 이벤트 데이터에서

   a. 이벤트 데이터 소스에서 S3에 저장된 이벤트 데이터를 선택합니다.

   b. IAM 역할에서 1단계에서 생성한 역할을 선택합니다.

   c. 훈련 데이터 위치에 예제 데이터 파일의 S3 URI 경로를 입력합니다.

6. Next(다음)를 선택합니다.

4단계: 모델 훈련

1. 모델 입력에서 모든 확인란을 선택한 상태로 둡니다. 기본적으로 Amazon Fraud Detector는 과거 이벤트 데이터 세트의 모든 변수를 모델 입력으로 사용합니다.
2. 레이블 분류에서 사기 레이블은 예제 데이터 세트의 사기 이벤트를 나타내는 값에 해당하므로 사기 레이블에서 사기를 선택합니다. 합법적인 레이블의 경우이 레이블이 예제 데이터 세트의 합법적인 이벤트를 나타내는 값에 해당하므로 적법성을 선택합니다.
3. 레이블이 지정되지 않은 이벤트 처리의 경우이 예제 데이터 세트에 대한 레이블이 지정되지 않은 이벤트 무시 기본 선택을 유지합니다.
4. Next(다음)를 선택합니다.
5. 검토 후 모델 생성 및 훈련을 선택합니다. Amazon Fraud Detector는 모델을 생성하고 모델의 새 버전을 훈련하기 시작합니다.

   모델 버전에서 상태 열은 모델 훈련의 상태를 나타냅니다. 예제 데이터 세트를 사용하는 모델 훈련을 완료하는 데 약 45분이 걸립니다. 모델 훈련이 완료된 후 상태가 배포 준비 완료로 변경됩니다.

5단계: 모델 성능 검토

Amazon Fraud Detector를 사용하는 중요한 단계는 모델 점수와 성능 지표를 사용하여 모델의 정확도를 평가하는 것입니다. 모델 훈련이 완료되면 Amazon Fraud Detector는 모델 훈련에 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 모델을 선택합니다.

   b. 모델 페이지에서 방금 훈련한 모델(sample_fraud_detection_model)을 선택한 다음 1.0을 선택합니다. 이 버전은 모델의 Amazon Fraud Detector가 생성한 버전입니다.
2. 모델 성능 전체 점수와 Amazon Fraud Detector가이 모델에 대해 생성한 기타 모든 지표를 살펴봅니다.

   이 페이지의 모델 성능 점수 및 성능 지표에 대한 자세한 내용은 모델 점수 및 섹션을 참조하세요모델 성능 지표.

   훈련된 모든 Amazon Fraud Detector 모델에는이 자습서에서 모델에 대해 표시되는 성능 지표와 유사한 실제 사기 탐지 성능 지표가 있을 것으로 예상할 수 있습니다.

6단계: 모델 배포

훈련된 모델의 성능 지표를 검토하고 이를 사용할 준비가 되면 사기 예측을 생성하여 모델을 배포할 수 있습니다.

1. Amazon Fraud Detector 콘솔의 왼쪽 탐색 창에서 모델을 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 상태는 배포 상태를 표시합니다. 배포가 완료되면 상태가 활성으로 변경됩니다. 이는 모델 버전이 활성화되어 사기 예측을 생성하는 데 사용할 수 있음을 나타냅니다. 를 계속 진행파트 B: 사기 예측 생성하여 사기 예측을 생성하는 단계를 완료합니다.

파트 B: 사기 예측 생성

사기 예측은 비즈니스 활동(이벤트)에 대한 사기 평가입니다. Amazon Fraud Detector는 탐지기를 사용하여 사기 예측을 생성합니다. 감지기에는 사기에 대해 평가하려는 특정 이벤트에 대한 모델 및 규칙과 같은 감지 로직이 포함되어 있습니다. 탐지 로직은 규칙을 사용하여 모델과 연결된 데이터를 해석하는 방법을 Amazon Fraud Detector에 알립니다. 이 자습서에서는 이전에 업로드한 계정 등록 예제 데이터 세트를 사용하여 계정 등록 이벤트를 평가합니다.

파트 A에서는 모델을 생성, 훈련 및 배포했습니다. 파트 B에서는 `sample_registration` 이벤트 유형에 대한 감지기를 빌드하고, 배포된 모델을 추가하고, 규칙 및 규칙 실행 순서를 생성한 다음, 사기 예측을 생성하는 데 사용하는 감지기 버전을 생성하고 활성화합니다.

1단계: 감지기 빌드

감지기를 생성하려면

1. Amazon Fraud Detector 콘솔의 왼쪽 탐색 창에서 감지기를 선택합니다.
2. 감지기 생성을 선택합니다.
3. 감지기 세부 정보 정의 페이지에서 감지기 이름`sample_detector`에를 입력합니다. 선택적으로와 같은 감지기에 대한 설명을 입력합니다`my sample fraud detector`.
4. 이벤트 유형에서 sample_registration을 선택합니다. 이 이벤트는이 자습서의 파트 A에서 생성한 이벤트입니다.
5. Next(다음)를 선택합니다.

2단계: 모델 추가

이 자습서의 파트 A를 완료했다면 감지기에 추가할 수 있는 Amazon Fraud Detector 모델이 이미 있을 것입니다. 아직 모델을 생성하지 않은 경우 파트 A로 이동하여 모델을 생성, 훈련 및 배포하는 단계를 완료한 다음 파트 B를 계속 진행합니다.

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 사기 예측을 평가할 때 모델 성능 점수를 해석하는 방법을 Amazon Fraud Detector에 알려주는 조건입니다. 이 자습서에서는 , 및 `high_fraud_risk`의 세 가지 규칙을 생성합니다`medium_fraud_risk`.`low_fraud_risk`.

1. 규칙 추가 페이지의 규칙 정의에서 규칙 이름에를 입력하고 설명 - 선택 사항에서 규칙에 대한 설명**`This rule captures events with a high ML model score`**으로를 입력합니다. `high_fraud_risk`
2. 표현식에서 Amazon Fraud Detector 간소화된 규칙 표현식을 사용하여 다음 규칙 표현식을 입력합니다.

   ```
   $sample_fraud_detection_model_insightscore > 900
   ```
3. 결과에서 새 결과 생성을 선택합니다. 결과는 사기 예측의 결과이며 평가 중에 규칙이 일치하면 반환됩니다.
4. 새 결과 생성에서 결과 이름으로 `verify_customer`를 입력합니다. 필요한 경우 설명을 입력합니다.
5. 결과 저장을 선택합니다.
6. 규칙 추가를 선택하여 규칙 검증 검사기를 실행하고 규칙을 저장합니다. 생성 후 Amazon Fraud Detector는 감지기에서 규칙을 사용할 수 있도록 합니다.
7. 다른 규칙 추가를 선택한 다음 규칙 생성 탭을 선택합니다.
8. 다음 `low_fraud_risk` 규칙 세부 정보를 사용하여이 프로세스를 두 번 더 반복하여 `medium_fraud_risk` 및 규칙을 생성합니다.

- medium_fraud_risk

  규칙 이름: `medium_fraud_risk`

  결과: `review`

  표현식:

  ```
  $sample_fraud_detection_model_insightscore <= 900 and
  ```

  ```
  $sample_fraud_detection_model_insightscore > 700
  ```

- low_fraud_risk

  규칙 이름: `low_fraud_risk`

  결과: `approve`

  표현식:

  ```
  $sample_fraud_detection_model_insightscore <= 700
  ```

  이러한 값은이 자습서에 사용되는 예제입니다. 자체 감지기에 대한 규칙을 생성할 때 모델 및 사용 사례에 적합한 값을 사용합니다.

9. 세 규칙을 모두 생성한 후 다음을 선택합니다.

   규칙 생성 및 작성에 대한 자세한 내용은 규칙 및 섹션을 참조하세요규칙 언어 참조.

4단계: 규칙 실행 및 규칙 순서 구성

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

기본 규칙 실행 모드는 입니다`FIRST_MATCHED`.

첫 번째 일치

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

에서 규칙을 실행하는 순서는 결과적으로 발생하는 사기 예측 결과에 영향을 미칠 수 있습니다. 규칙을 생성한 후 다음 단계에 따라 규칙을 원하는 순서로 실행하도록 다시 정렬합니다.

high_fraud_risk 규칙이 아직 규칙 목록 상단에 없는 경우 순서를 선택한 다음 1을 선택합니다. 그러면 high_fraud_risk 첫 번째 위치로 이동합니다.

규칙이 두 번째 위치에 있고 medium_fraud_risk 규칙low_fraud_risk이 세 번째 위치에 있도록이 프로세스를 반복합니다.

모두 일치

일치하는 모든 규칙 실행 모드는 규칙 순서에 관계없이 일치하는 모든 규칙에 대한 결과를 반환합니다. ALL_MATCHED를 지정하면 Amazon Fraud Detector는 모든 규칙을 평가하고 일치하는 모든 규칙에 대한 결과를 반환합니다.

이 자습서FIRST_MATCHED에서를 선택한 후 다음을 선택합니다.

5단계: 감지기 버전 검토 및 생성

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

1. 검토 및 생성 페이지에서 구성한 감지기 세부 정보, 모델 및 규칙을 검토합니다. 변경해야 하는 경우 해당 섹션 옆의 편집을 선택합니다.
2. 감지기 생성을 선택합니다. 생성되면 감지기의 첫 번째 버전이 감지기 버전 테이블에 Draft 상태와 함께 나타납니다.

   초안 버전을 사용하여 감지기를 테스트합니다.

6단계: 감지기 버전 테스트 및 활성화

Amazon Fraud Detector 콘솔에서 테스트 실행 기능을 사용하여 모의 데이터를 사용하여 탐지기의 로직을 테스트할 수 있습니다. 이 자습서에서는 예제 데이터 세트의 계정 등록 데이터를 사용할 수 있습니다.

1. 감지기 버전 세부 정보 페이지 하단의 테스트 실행으로 스크롤합니다.
2. 이벤트 메타데이터에 이벤트가 발생한 시점의 타임스탬프를 입력하고 이벤트를 수행하는 개체의 고유 식별자를 입력합니다. 이 자습서에서는 타임스탬프의 날짜 선택기에서 날짜를 선택하고 개체 ID에 “1234”를 입력합니다.
3. 이벤트 변수에 테스트할 변수 값을 입력합니다. 이 자습서에서는 ip_address 및 email_address 필드만 필요합니다. 이는 Amazon Fraud Detector 모델을 훈련하는 데 사용되는

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

- ip_address: `205.251.233.178`
- email_address: `johndoe@exampledomain.com`

4. 테스트 실행을 선택합니다.
5. Amazon Fraud Detector는 규칙 실행 모드를 기반으로 사기 예측 결과를 반환합니다. 규칙 실행 모드가 인 경우 반환`FIRST_MATCHED`된 결과는 일치하는 첫 번째 규칙에 해당합니다. 첫 번째 규칙은 우선 순위가 가장 높은 규칙입니다. true로 평가되면 일치됩니다. 규칙 실행 모드가 인 경우 반환`ALL_MATCHED`된 결과는 일치하는 모든 규칙에 해당합니다. 즉, 모두 true로 평가됩니다. 또한 Amazon Fraud Detector는 감지기에 추가된 모든 모델의 모델 점수를 반환합니다.

   입력을 변경하고 몇 가지 테스트를 실행하여 다양한 결과를 볼 수 있습니다. 테스트에 예제 데이터 세트의 ip_address 및 email_address 값을 사용하고 결과가 예상대로인지 확인할 수 있습니다.
6. 감지기가 작동하는 방식에 만족하면에서 `Draft`로 승격합니다`Active`. 이렇게 하면 탐지기를 실시간 사기 탐지에 사용할 수 있습니다.

   감지기 버전 세부 정보 페이지에서 작업, 게시, 버전 게시를 선택합니다. 그러면 감지기의 상태가 초안에서 활성으로 변경됩니다.

   이때 모델과 관련 탐지기 로직은 Amazon Fraud Detector `GetEventPrediction` API를 사용하여 실시간으로 사기에 대한 온라인 활동을 평가할 준비가 됩니다. CSV 입력 파일과 `CreateBatchPredictionJob` API를 사용하여 이벤트를 오프라인으로 평가할 수도 있습니다. 사기 예측에 대한 자세한 내용은 섹션을 참조하세요. 사기 예측

이 자습서를 완료하여 다음을 수행했습니다.

- 이벤트 데이터 세트 예제를 Amazon S3에 업로드했습니다.
- 예제 데이터 세트를 사용하여 Amazon Fraud Detector 사기 탐지 모델을 생성하고 교육했습니다.
- Amazon Fraud Detector가 생성한 모델 성능 점수 및 기타 성능 지표를 확인했습니다.
- 사기 탐지 모델을 배포했습니다.
- 감지기를 생성하고 배포된 모델을 추가했습니다.
- 감지기에 규칙, 규칙 실행 순서 및 결과가 추가되었습니다.
- 다양한 입력을 제공하고 규칙 및 규칙 실행 순서가 예상대로 작동하는지 확인하여 감지기를 테스트했습니다.
- 감지기를 게시하여 활성화했습니다.

자습서: 사용 시작하기 AWS SDK for Python (Boto3)

이 자습서에서는 Amazon Fraud Detector 모델을 빌드 및 훈련한 다음이 모델을 사용하여를 사용하여 실시간 사기 예측을 생성하는 방법을 설명합니다 AWS SDK for Python (Boto3). 모델은 Amazon S3 버킷에 업로드하는 계정 등록 예제 데이터 파일을 사용하여 훈련됩니다.

이 자습서를 마치면 다음 작업을 완료합니다.

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

사전 조건

다음은이 자습서의 사전 조건 단계입니다.

- 완료됨Amazon Fraud Detector 설정.

  이미가 있는 경우 Boto3 SDK 버전 1.14.29 이상을 사용하고 있는지 AWS SDK 설정확인합니다.
- 이 자습서에 필요한 예제 데이터 세트 가져오기 및 업로드 파일을 제출하는 지침을 따랐습니다.

시작

1단계: Python 환경 설정 및 확인

Boto는 Python용 Amazon Web Services(AWS) SDK입니다. 이를 사용하여 생성, 구성 및 관리할 수 있습니다 AWS 서비스. Boto3를 설치하는 방법에 대한 지침은 AWS SDK for Python(Boto3)을 참조하세요.

설치 후 다음 Python 예제 명령을 AWS SDK for Python (Boto3)실행하여 환경이 올바르게 구성되었는지 확인합니다. 환경이 올바르게 구성된 경우 응답에는 감지기 목록이 포함됩니다. 감지기가 생성되지 않은 경우 목록이 비어 있습니다.

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

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

2단계: 변수, 개체 유형 및 레이블 생성

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

변수 생성

변수는 이벤트 유형, 모델 및 규칙을 생성하는 데 사용할 데이터 세트의 데이터 요소입니다.

다음 예제에서는 CreateVariable API를 사용하여 두 개의 변수를 생성합니다. 변수는 `email_address` 및 입니다`ip_address`. 해당하는 변수 유형인 `EMAIL_ADDRESS` 및에 할당합니다`IP_ADDRESS`. 이러한 변수는 업로드한 예제 데이터 세트의 일부입니다. 변수 유형을 지정하면 Amazon Fraud Detector는 모델 훈련 중 및 예측을 가져올 때 변수를 해석합니다. 연결된 변수 유형이 있는 변수만 모델 훈련에 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>'
)
```

개체 유형 생성

개체는 이벤트를 수행하는 사용자를 나타내며 개체 유형은 개체를 분류합니다. 분류 예에는 고객, 판매자 또는 계정이 포함됩니다.

다음 예제에서는 PutEntityType API를 사용하여 `sample_customer`엔터티 유형을 생성합니다.

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

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

레이블 생성

레이블은 이벤트를 사기 또는 합법적으로 분류하며 사기 탐지 모델을 훈련하는 데 사용됩니다. 모델은 이러한 레이블 값을 사용하여 이벤트를 분류하는 방법을 학습합니다.

다음 예제에서는 Putlabel API를 사용하여 `fraud` 및 레이블 2개를 생성합니다`legit`.

```
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 예제에서는 PutEventType API를 사용하여 이벤트 유형을 생성합니다`sample_registration`. 이전 단계에서 생성한 변수(`email_address`,`ip_address`), 엔터티 유형(`sample_customer`), 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, 훈련 및 배포

Amazon Fraud Detector는 모델을 교육하여 특정 이벤트 유형에 대한 사기를 탐지하는 방법을 알아봅니다. 이전 단계에서 이벤트 유형을 생성했습니다. 이 단계에서는 이벤트 유형에 대한 모델을 생성하고 훈련합니다. 모델은 모델 버전의 컨테이너 역할을 합니다. 모델을 훈련할 때마다 새 버전이 생성됩니다.

다음 예제 코드를 사용하여 온라인 사기 인사이트 모델을 생성하고 교육합니다. 이 모델을 라고 합니다`sample_fraud_detection_model`. Amazon S3에 업로드한 계정 등록 예제 데이터 세트를 `sample_registration` 사용하는 이벤트 유형에 대한 것입니다.

Amazon Fraud Detector가 지원하는 다양한 모델 유형에 대한 자세한 내용은 섹션을 참조하세요모델 유형 선택.

모델 생성

다음 예제에서는 CreateModel API를 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 훈련

다음 예제에서는 CreateModelVersion API를 사용하여 모델을 훈련합니다. `trainingDataSource` 및 예제 데이터 세트를 저장한 Amazon S3 위치와 `'EXTERNAL_EVENTS'`용 Amazon S3 버킷의 RoleArn에를 지정합니다`externalEventsDetail`. `trainingDataSchema` 파라미터의 경우

Amazon Fraud Detector가 예제 데이터를 해석하는 방법을 지정합니다. 보다 구체적으로, 포함할 변수와 이벤트 레이블을 분류하는 방법을 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']
        }
    }
},
         externalEventsDetail = {
              'dataLocation' : 's3://amzn-s3-demo-bucket/your-example-data-
filename.csv',
              'dataAccessRoleArn' : 'role_arn'
}
)
```

모델을 여러 번 훈련할 수 있습니다. 모델을 훈련할 때마다 새 버전이 생성됩니다. 모델 훈련이 완료되면 모델 버전 상태가 로 업데이트됩니다TRAINING_COMPLETE. 모델 성능 점수 및 기타 모델 성능 지표를 검토할 수 있습니다.

모델 성능 검토

Amazon Fraud Detector를 사용하는 중요한 단계는 모델 점수와 성능 지표를 사용하여 모델의 정확도를 평가하는 것입니다. 모델 훈련이 완료되면 Amazon Fraud Detector는 모델 훈련에 사용되지 않은 데이터의 15%를 사용하여 모델 성능을 검증합니다. 모델 성능 점수 및 기타 성능 지표를 생성합니다.

DescribeModelVersions API를 사용하여 모델 성능을 검토합니다. 이 모델의 모델 성능 전체 점수와 Amazon Fraud Detector에서 생성한 기타 모든 지표를 살펴봅니다.

모델 성능 점수 및 성능 지표에 대한 자세한 내용은 모델 점수 및 섹션을 참조하세요모델 성능 지표.

훈련된 모든 Amazon Fraud Detector 모델에이 자습서의 지표와 유사한 실제 사기 탐지 성능 지표가 있을 것으로 예상할 수 있습니다.

모델 배포

훈련된 모델의 성능 지표를 검토한 후 모델을 배포하고 Amazon Fraud Detector에서 사기 예측을 생성할 수 있도록 합니다. 훈련된 모델을 배포하려면 UpdateModelVersionStatus API를 사용합니다. 다음 예제에서는 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: 감지기, 결과, 규칙 및 감지기 버전 생성

감지기에는 모델 및 규칙과 같은 감지 로직이 포함되어 있습니다. 이 로직은 사기에 대해 평가하려는 특정 이벤트에 대한 것입니다. 규칙은 예측 중에 변수 값을 해석하는 방법을 Amazon Fraud Detector에 알리기 위해 지정하는 조건입니다. 결과는 사기 예측의 결과입니다. 감지기에는 DRAFT, ACTIVE 또는 INACTIVE 상태의 각 버전이 있는 여러 버전이 있을 수 있습니다. 감지기 버전에는 이와 연결된 규칙이 하나 이상 있어야 합니다.

다음 예제 코드를 사용하여 감지기, 규칙, 결과를 생성하고 감지기를 게시합니다.

감지기 생성

다음 예제에서는 PutDetector API를 사용하여 `sample_registration` 이벤트 유형에 대한 `sample_detector` 감지기를 생성합니다.

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

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

결과 생성

가능한 각 사기 예측 결과에 대한 결과가 생성됩니다. 다음 예제에서는 PutOutcome API를 사용하여 , 및 verify_customerreview의 세 가지 결과를 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 데이터 세트의 하나 이상의 변수, 로직 표현식 및 하나 이상의 결과로 구성됩니다.

다음 예제에서는 CreateRule API를 사용하여 , 및 high_riskmedium_risk라는 세 가지 규칙을 생성합니다low_risk. 규칙 표현식을 생성하여 모델 성능 점수 sample_fraud_detection_model_insightscore 값을 다양한 임계값과 비교합니다. 이는 이벤트의 위험 수준을 결정하고 이전 단계에서 정의된 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']
     )
```

감지기 버전 생성

감지기 버전은 사기 예측을 가져오는 데 사용되는 모델과 규칙을 정의합니다.

다음 예제에서는 CreateDetectorVersion API를 사용하여 감지기 버전을 생성합니다. 모델 버전 세부 정보, 규칙 및 규칙 실행 모드 FIRST_MATCHED를 제공하여 이를 수행합니다. 규칙 실행 모드는 규칙을 평가하기 위한 시퀀스를 지정합니다. 규칙 실행 모드 FIRST_MATCHED는 규칙이 처음 일치할 때 중지되는 순서대로 먼저 또는 마지막으로 평가되도록 지정합니다.

```
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 단계에서 `sample_detector` 생성된 탐지기를 사용하여 `sample_registration` 이벤트 유형에 대한 사기 예측을 실시간으로 생성합니다. 감지기는 Amazon S3에 업로드된 예제 데이터를 평가합니다. 응답에는 모델 성능 점수와 일치하는 규칙과 관련된 모든 결과가 포함됩니다.

다음 예제에서는 GetEventPrediction API를 사용하여 각 요청에 대해 단일 계정 등록의 데이터를 제공합니다. 이 자습서에서는 계정 등록 예제 데이터 파일에서 데이터(email_address 및 ip_address)를 가져옵니다. 상단 헤더 라인 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 다음을 수행했습니다.

- 이벤트 데이터 세트 예제를 Amazon S3에 업로드했습니다.
- 모델을 생성하고 훈련하는 데 사용되는 변수, 엔터티 및 레이블을 생성했습니다.
- 예제 데이터 세트를 사용하여 모델을 생성하고 훈련했습니다.
- Amazon Fraud Detector가 생성한 모델 성능 점수 및 기타 성능 지표를 확인했습니다.
- 사기 탐지 모델을 배포했습니다.
- 감지기를 생성하고 배포된 모델을 추가했습니다.
- 감지기에 규칙, 규칙 실행 순서 및 결과가 추가되었습니다.
- 감지기 버전을 생성했습니다.
- 다양한 입력을 제공하고 규칙 및 규칙 실행 순서가 예상대로 작동하는지 확인하여 감지기를 테스트했습니다.

(선택 사항) Jupyter(iPython) 노트북을 사용하여 Amazon Fraud Detector APIs 탐색

Amazon Fraud Detector APIs를 사용하는 방법에 대한 자세한 예는 aws-fraud-detector-samples GitHub 리포지토리를 참조하세요. 노트북에서 다루는 주제에는 Amazon Fraud Detector APIs를 사용하여 모델과 탐지기를 빌드하고 `GetEventPrediction` API를 사용하여 배치 사기 예측 요청을 하는 것이 모두 포함됩니다.

다음 단계

이제 모델과 탐지기를 생성했으므로 더 자세히 살펴보고 모델과 탐지기를 생성하고 사기 예측을 생성할 수 있습니다.

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

- 모델 훈련을 위한 이벤트 데이터 세트를 준비하고 생성합니다.
- 이벤트 유형 생성
- 모델 생성
- 감지기 생성

- 사기 예측 가져오기
- Amazon Fraud Detector 리소스(특히 변수, 개체, 결과 및 레이블) 관리
- 보안 및 규정 준수 목표에 맞게 Amazon Fraud Detector 구성
- Amazon Fraud Detector 모니터링 및 Amazon Fraud Detector API 호출 로깅
- Amazon Fraud Detector 관련 문제 해결

이벤트 데이터 세트

이벤트 데이터 세트는 회사의 과거 사기 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 훈련됩니다. 모델 유형은 모델 훈련에 사용되는 알고리즘과 변환을 지정합니다. 모델 훈련은 제공한 데이터 세트를 사용하여 사기 이벤트를 예측할 수 있는 모델을 생성하는 프로세스입니다. 자세한 내용은 Amazon Fraud Detector 작동 방식을 참조하세요.

사기 탐지 모델을 생성하는 데 사용되는 데이터 세트는 이벤트의 세부 정보를 제공합니다. 이벤트는 사기 위험에 대한 평가가 이루어지는 비즈니스 활동입니다. 예를 들어 계정 등록은 이벤트일 수 있습니다. 계정 등록 이벤트와 연결된 데이터는 이벤트 데이터 세트일 수 있습니다. Amazon Fraud Detector는이 데이터 세트를 사용하여 계정 등록 사기를 평가합니다.

모델 생성을 위해 Amazon Fraud Detector에 데이터 세트를 제공하기 전에 모델 생성 목표를 정의해야 합니다. 또한 모델을 사용할 방법을 결정하고 모델이 특정 요구 사항에 따라 성능을 발휘하는지 여부를 평가하기 위한 지표를 정의해야 합니다.

예를 들어 계정 등록 사기를 평가하는 사기 탐지 모델을 생성하는 목표는 다음과 같을 수 있습니다.

- 합법적인 등록을 자동 승인하려면
- 이후 조사를 위해 사기 등록을 캡처합니다.

목표를 결정한 후 다음 단계는 모델을 사용할 방법을 결정하는 것입니다. 사기 탐지 모델을 사용하여 등록 사기를 평가하는 몇 가지 예는 다음과 같습니다.

- 각 계정 등록에 대한 실시간 사기 탐지.
- 매시간 모든 계정 등록의 오프라인 평가.

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

- 프로덕션의 현재 기준보다 일관되게 더 나은 성능을 발휘합니다.
- Y% 오탐률로 X% 사기 등록을 캡처합니다.
- 사기인 자동 승인 등록의 최대 5%를 허용합니다.

이벤트 데이터 세트 구조

Amazon Fraud Detector를 사용하려면 UTF-8 형식의 쉼표로 구분된 값(CSV)을 사용하여 이벤트 데이터 세트를 텍스트 파일로 제공해야 합니다. CSV 데이터 세트 파일의 첫 번째 줄에는 파일 헤더가 포함되어야 합니다. 파일 헤더는 이벤트 메타데이터와 이벤트와 연결된 각 데이터 요소를 설명하는 이벤트 변수로 구성됩니다. 헤더 뒤에 이벤트 데이터가 옵니다. 각 줄은 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 생성하고 훈련하는 데 사용되는 모델 유형에 따라 Amazon Fraud Detector는 특정 이벤트 메타데이터를 제공해야 합니다. CSV 파일 헤더에서 이벤트 메타데이터를 지정할 때는 Amazon Fraud Detector에서 지정한 것과 동일한 이벤트 메타데이터 이름을 사용하고 대문자만 사용합니다.
- 이벤트 변수 - 사기 탐지 모델을 생성하고 훈련하는 데 사용하려는 이벤트와 관련된 데이터 요소를 나타냅니다. 비즈니스 사용 사례 및 사기 탐지 모델을 생성하고 훈련하는 데 사용되는 모델 유형에 따라 Amazon Fraud Detector는 특정 이벤트 변수를 제공하도록 요구하거나 권장할 수 있습니다. 또한 모델 훈련에 포함하려는 이벤트의 다른 이벤트 변수를 선택적으로 제공할 수 있습니다. 온라인 등록 이벤트에 대한 이벤트 변수의 몇 가지 예는 이메일 주소, IP 주소 및 전화번호일 수 있습니다. CSV 파일 헤더에서 이벤트 변수 이름을 지정할 때는 원하는 변수 이름을 사용하고 소문자만 사용합니다.
- 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 파일에서 각 행에는 단일 등록의 데이터가 포함됩니다. 행의 각 데이터 요소는 해당 이벤트 메타데이터 또는 이벤트 변수와 일치해야 합니다.

다음은 계정 등록 이벤트의 데이터가 포함된 CSV 파일의 예입니다. 헤더 행에는 이벤트 메타데이터가 대문자로, 이벤트 변수가 소문자로, 이벤트 데이터가 뒤에 옵니다. 데이터 세트의 각 행에는 헤더에 해당하는 각 데이터 요소와 함께 단일 계정 등록과 연결된 데이터 요소가 포함되어 있습니다.

데이터 모델 탐색기를 사용하여 이벤트 데이터 세트 요구 사항 가져오기

모델을 생성하도록 선택한 모델 유형은 데이터 세트의 요구 사항을 정의합니다. Amazon Fraud Detector는 사용자가 제공한 데이터 세트를 사용하여 사기 탐지 모델을 생성하고 교육합니다. Amazon Fraud Detector가 모델을 생성하기 전에 데이터세트가 크기, 형식 및 기타 요구 사항을 충족하는지 확인합니다. 데이터 세트가 요구 사항을 충족하지 않으면 모델 생성 및 훈련이 실패합니다. 데이터 모델 탐색기를 사용하여 비즈니스 사용 사례에 사용할 모델 유형을 식별하고 식별된 모델 유형의 데이터 세트 요구 사항에 대한 인사이트를 얻을 수 있습니다.

데이터 모델 탐색기

데이터 모델 탐색기는 Amazon Fraud Detector 콘솔의 도구로, 비즈니스 사용 사례를 Amazon Fraud Detector에서 지원하는 모델 유형에 맞게 조정합니다. 또한 데이터 모델 탐색기는 Amazon Fraud Detector가 사기 탐지 모델을 생성하는 데 필요한 데이터 요소에 대한 인사이트를 제공합니다. 이벤트 데이터 세트를 준비하기 전에 데이터 모델 탐색기를 사용하여 Amazon Fraud Detector가 비즈니스용으로 권장하는 모델 유형을 파악하고 데이터 세트를 생성하는 데 필요한 필수, 권장 및 선택적 데이터 요소의 목록을 확인합니다.

데이터 모델 탐색기를 사용하려면

1. AWS Management Console을 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 평가할 비즈니스 사용 사례를 선택합니다.
4. Amazon Fraud Detector는 비즈니스 사용 사례와 일치하는 권장 모델 유형을 표시합니다. 모델 유형은 Amazon Fraud Detector가 사기 탐지 모델을 훈련하는 데 사용할 알고리즘, 보강 및 변환을 정의합니다.

   권장 모델 유형을 기록해 둡니다. 나중에 모델을 생성할 때이 정보가 필요합니다.

> **Note**
>
> 비즈니스 사용 사례를 찾을 수 없는 경우 설명의 연락 링크를 사용하여 비즈니스 사용 사례에 대한 세부 정보를 제공하세요. 비즈니스 사용 사례에 대한 사기 탐지 모델을 생성하는 데 사용할 모델 유형을 권장합니다.

5. 데이터 모델 인사이트 창은 비즈니스 사용 사례에 대한 사기 탐지 모델을 생성하고 교육하는 데 필요한 필수, 권장 및 선택적 데이터 요소에 대한 인사이트를 제공합니다. 인사이트 창의 정보를 사용하여 이벤트 데이터를 수집하고 데이터 세트를 생성합니다.

이벤트 데이터 수집

이벤트 데이터를 수집하는 것은 모델을 생성하는 데 중요한 단계입니다. 이는 사기를 예측하는 모델의 성능이 데이터 세트의 품질에 따라 달라지기 때문입니다. 이벤트 데이터를 수집하기 시작할 때 데이터 모델 탐색기가 데이터 세트를 생성하기 위해 제공한 데이터 요소 목록을 염두에 두세요. 모든 필수(이벤트 메타데이터) 데이터를 수집하고 모델 생성 목표를 기반으로 포함할 권장 및 선택적 데이터 요소(이벤트 변수)를 결정해야 합니다. 또한 포함하려는 각 이벤트 변수의 형식과 데이터 세트의 총 크기를 결정하는 것이 중요합니다.

이벤트 데이터 세트 품질

모델의 고품질 데이터 세트를 수집하려면 다음을 수행하는 것이 좋습니다.

- 성숙한 데이터 수집 - 최신 데이터를 사용하면 최신 사기 패턴을 식별하는 데 도움이 됩니다. 그러나 사기 사용 사례를 감지하려면 데이터가 성숙되도록 합니다. 성숙 기간은 비즈니스에 따라 달라지며 2주에서 3개월까지 걸릴 수 있습니다. 예를 들어 이벤트에 신용 카드 트랜잭션이 포함된 경우 데이터의 성숙도는 신용 카드의 결제 기간 또는 조사자가 결정을 내리는 데 걸린 시간에 따라 결정될 수 있습니다.

  모델을 훈련하는 데 사용된 데이터 세트가 비즈니스에 따라 성숙할 충분한 시간을 가졌는지 확인합니다.
- 데이터 배포가 크게 드리프트되지 않도록 - Amazon Fraud Detector 모델 훈련은 EVENT_TIMESTAMP를 기반으로 데이터 세트를 샘플링하고 분할합니다. 예를 들어 데이터 세트가 지난 6개월에서 가져온 사기 이벤트로 구성되지만 합법적인 이벤트의 마지막 달만 포함된 경우 데이터 배포는 드리프팅되고 불안정한 것으로 간주됩니다. 불안정한 데이터 세트는 모델 성능 평가에 편향을 초래할 수 있습니다. 데이터 배포가 크게 드리프트되는 경우 현재 데이터 배포와 유사한 데이터를 수집하여 데이터 세트의 균형을 맞추는 것이 좋습니다.
- 데이터세트가 모델이 구현/테스트되는 사용 사례를 나타내는지 확인합니다. 그렇지 않으면 예상 성능이 편향될 수 있습니다. 모델을 사용하여 모든 실내 신청자를 자동으로 거부하지만 모델은 이전에 승인된 기록 데이터/레이블이 있는 데이터 세트로 훈련된다고 가정해 보겠습니다. 그러면 평가가 거부된 신청자의 표현이 없는 데이터 세트를 기반으로 하므로 모델의 평가가 부정확할 수 있습니다.

이벤트 데이터 형식

Amazon Fraud Detector는 모델 훈련 프로세스의 일환으로 대부분의 데이터를 필요한 형식으로 변환합니다. 그러나 Amazon Fraud Detector가 데이터 세트를 검증할 때 나중에 문제를 방지하는 데 도움이 될 수 있는 데이터를 제공하는 데 쉽게 사용할 수 있는 몇 가지 표준 형식이 있습니다. 다음 표에서는 권장 이벤트 메타데이터를 제공하기 위한 형식에 대한 지침을 제공합니다.

> **Note**
>
> CSV 파일을 생성할 때 아래 나열된 대로 이벤트 메타데이터 이름을 대문자로 입력해야 합니다.

메타데이터 이름	형식	필수
EVENT_ID	제공된 경우 다음 요구 사항을 충족해야 합니다. • 해당 이벤트에 고유합니다. • 비즈니스에 의미 있는 정보를 나타냅니다. • 정규식 패턴(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 하면 정확히 동일한 EVENT_ID를 제공해야 하기 때문입니다.	모델 유형에 따라 다름
EVENT_TIMESTAMP	• 다음 형식 중 하나로 지정해야 합니다. • %yyyy-%mm-%ddT%hh:%mm:%ssZ(밀리초 없이 UTC 전용 ISO 8601 표준)	예

메타데이터 이름	형식	필수
	예: 2019-11-30T13:01:01Z • %yyyy/%mm/%dd %hh:%mm:%ss(AM/PM) 예: 2019/11/30 1:01:01 PM 또는 2019/11/30 13:01:01 • %mm/%dd/%yyyy %hh:%mm:%ss 예: 11/30/2019 1:01:01 PM, 11/30/2019 13:01:01 • %mm/%dd/%yy %hh:%mm:%ss 예: 11/30/19 1:01:01 PM, 11/30/19 13:01:01 • Amazon Fraud Detector는 이벤트 타임스탬프에 대한 날짜/타임스탬프 형식을 구문 분석할 때 다음과 같은 가정을 합니다. • ISO 8601 표준을 사용하는 경우 이전 사양과 정확히 일치해야 합니다. • 다른 형식 중 하나를 사용하는 경우 추가 유연성이 있습니다. • 월과 일에는 단일 또는 두 자릿수를 입력할 수 있습니다. 예를 들어 1/12/2019은 유효한 날짜입니다.	

메타데이터 이름	형식	필수
	• hh:mm:ss가 없는 경우 hh:mm:ss를 포함할 필요가 없습니다(즉, 단순히 날짜를 입력할 수 있음). 시간 및 분(예: hh:mm)의 하위 집합만 제공할 수도 있습니다. 시간 제공만 지원되지 않습니다. 밀리초도 지원되지 않습니다. • AM/PM 레이블을 입력하면 12시간 클럭이 가정됩니다. AM/PM 정보가 없는 경우 24시간 시계가 가정됩니다. • 날짜 요소의 구분 기호로 “/” 또는 “-”를 사용할 수 있습니다. 타임스탬프 요소에는 “:”가 가정됩니다.	
ENTITY_ID	• 정규식 패턴인를 따라야 합니다`^[0-9A-Za-z_.@+-]+$` . • 평가 시 개체 ID를 사용할 수 없는 경우 개체 ID를 알 수 없음으로 지정합니다.	모델 유형에 따라 다름
ENTITY_TYPE	모든 문자열을 사용할 수 있습니다.	모델 유형에 따라 다름
EVENT_LABEL	'사기', '법률', '1' 또는 '0'과 같은 모든 레이블을 사용할 수 있습니다.	LABEL_TIMESTAMP가 포함된 경우 필수

메타데이터 이름	형식	필수
LABEL_TIMESTAMP	타임스탬프 형식을 따라야 합니다.	EVENT_LABEL이 포함된 경우 필수

이벤트 변수에 대한 자세한 내용은 변수를 참조하세요.

> **Important**
>
> Account Takeover Insights(ATI) 모델을 생성하는 경우 데이터 준비 및 선택에 대한 자세한 내용은 데이터 준비 섹션을 참조하세요.

Null 또는 누락된 값

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

최소 변수

모델을 생성할 때 데이터 세트에는 필요한 이벤트 메타데이터 외에도 최소 2개의 이벤트 변수가 포함되어야 합니다. 두 이벤트 변수는 검증 검사를 통과해야 합니다.

이벤트 데이터 세트 크기

필수

성공적인 모델 훈련을 위해서는 데이터 세트가 다음과 같은 기본 요구 사항을 충족해야 합니다.

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

권장

성공적인 모델 훈련과 우수한 모델 성능을 위해 데이터 세트에 다음을 포함하는 것이 좋습니다.

- 최소 3주 동안 기록 데이터를 포함하되 최대 6개월 동안 데이터를 포함시킵니다.

- 최소 10K개의 총 이벤트 데이터를 포함합니다.
- 사기로 분류된 최소 400개의 이벤트(행)와 합법적으로 분류된 400개의 이벤트(행)를 포함합니다.
- 모델 유형에 ENTITY_ID가 필요한 경우 100개 이상의 고유 엔터티를 포함합니다.

데이터 세트 검증

Amazon Fraud Detector는 모델 생성을 시작하기 전에 모델 훈련을 위한 데이터 세트에 포함된 변수가 크기, 형식 및 기타 요구 사항을 충족하는지 확인합니다. 데이터 세트가 검증을 통과하지 못하면 모델이 생성되지 않습니다. 모델을 생성하기 전에 먼저 검증을 통과하지 못한 변수를 수정해야 합니다. Amazon Fraud Detector는 모델 훈련을 시작하기 전에 데이터 세트의 문제를 식별하고 해결하는 데 사용할 수 있는 데이터 프로파일러를 제공합니다.

데이터 프로파일러

Amazon Fraud Detector는 모델 훈련을 위해 데이터를 프로파일링하고 준비하기 위한 오픈 소스 도구를 제공합니다. 이 자동화된 데이터 프로파일러는 일반적인 데이터 준비 오류를 방지하고 모델 성능에 부정적인 영향을 미칠 수 있는 잘못 매핑된 변수 유형과 같은 잠재적 문제를 식별하는 데 도움이 됩니다. 프로파일러는 변수 통계, 레이블 분포, 범주형 및 숫자 분석, 변수 및 레이블 상관관계를 포함하여 데이터 세트에 대한 직관적이고 포괄적인 보고서를 생성합니다. 변수 유형에 대한 지침과 데이터세트를 Amazon Fraud Detector에 필요한 형식으로 변환하는 옵션을 제공합니다.

데이터 프로파일러 사용

자동화된 데이터 프로파일러는 몇 번의 클릭으로 쉽게 시작할 수 있는 AWS CloudFormation 스택으로 빌드됩니다. 모든 코드는 Github에서 사용할 수 있습니다. 데이터 프로파일러를 사용하는 방법에 대한 자세한 내용은 Amazon Fraud Detector용 자동 데이터 프로파일러를 사용하여 더 빠르게 모델 학습 블로그의 지침을 따르세요.

일반적인 이벤트 데이터 세트 오류

다음은 이벤트 데이터 세트를 검증할 때 Amazon Fraud Detector에서 발생하는 몇 가지 일반적인 문제입니다. 데이터 프로파일러를 실행한 후 모델을 생성하기 전에이 목록을 사용하여 데이터 세트에 오류가 있는지 확인합니다.

- CSV 파일은 UTF-8 형식이 아닙니다.
- 데이터 세트의 이벤트 수가 100개 미만입니다.
- 사기 또는 합법적으로 식별된 이벤트 수가 50개 미만입니다.
- 사기 이벤트와 연결된 고유 엔터티의 수가 100개 미만입니다.

- EVENT_TIMESTAMP에서 값의 0.1% 이상이 지원되는 날짜/타임스탬프 형식 이외의 null 또는 값을 포함합니다.
- EVENT_LABEL에서 값의 1% 이상은 이벤트 유형에 정의된 값 이외의 null 또는 값을 포함합니다.
- 모델 훈련에 사용할 수 있는 변수는 2개 미만입니다.

데이터 세트 스토리지

데이터 세트를 수집한 후 Amazon Fraud Detector를 사용하여 내부적으로 또는 Amazon Simple Storage Service(Amazon S3)를 사용하여 외부적으로 데이터 세트를 저장합니다. 사기 예측을 생성하는 데 사용하는 모델을 기반으로 데이터 세트를 저장할 위치를 선택하는 것이 좋습니다. 모델 유형에 대한 자세한 내용은 모델 유형 선택을 참조하세요. 데이터 세트 저장에 대한 자세한 내용은 섹션을 참조하세요이벤트 데이터 스토리지.

이벤트 유형

Amazon Fraud Detector를 사용하면 이벤트에 대한 사기 예측을 생성할 수 있습니다. 이벤트 유형은 Amazon Fraud Detector로 전송되는 개별 이벤트의 구조를 정의합니다. 정의되면 특정 이벤트 유형에 대한 위험을 평가하는 모델과 감지기를 빌드할 수 있습니다.

이벤트의 구조에는 다음이 포함됩니다.

- 개체 유형: 이벤트를 수행하는 사용자를 분류합니다. 예측 중에 이벤트를 수행한 사용자를 정의할 개체 유형과 개체 ID를 지정합니다.
- 변수: 이벤트의 일부로 전송할 수 있는 변수를 정의합니다. 변수는 모델 및 규칙에서 사기 위험을 평가하는 데 사용됩니다. 일단 추가되면 변수는 이벤트 유형에서 제거할 수 없습니다.
- 레이블: 이벤트를 사기 또는 합법적으로 분류합니다. 모델 훈련 중에 사용됩니다. 일단 추가되면 이벤트 유형에서 레이블을 제거할 수 없습니다.

이벤트 유형 생성

사기 탐지 모델을 생성하기 전에 먼저 이벤트 유형을 생성해야 합니다. 이벤트 유형을 생성하려면 사기를 평가할 비즈니스 활동(이벤트)을 정의해야 합니다. 이벤트를 정의하려면 사기 평가를 위해 포함할 데이터 세트의 이벤트 변수를 식별하고, 이벤트를 시작하는 개체와 이벤트를 분류하는 레이블을 지정해야 합니다.

이벤트 유형을 생성하기 위한 사전 조건

이벤트 유형을 생성하기 전에 다음을 완료했는지 확인합니다.

- 데이터 모델 탐색기 도구를 사용하여 Amazon Fraud Detector에서 사기 탐지 모델을 생성하는 데 필요한 데이터 요소에 대한 인사이트를 얻었습니다.
- Data Models Explorer에서 얻은 인사이트를 사용하여 이벤트 데이터 세트를 생성하고 Amazon S3 버킷에 데이터 세트를 업로드했습니다.
- Amazon Fraud Detector가이 이벤트에 대한 사기 탐지 모델을 생성하는 데 사용할 변수개체, 및를 생성레이블했습니다. 생성한 변수, 개체 유형 및 레이블이 이벤트 데이터 세트에 포함되어 있는지 확인합니다.

Amazon Fraud Detector 콘솔에서 API, AWS CLI또는 AWS SDK를 사용하여 이벤트 유형을 생성할 수 있습니다.

Amazon Fraud Detector 콘솔에서 이벤트 유형 생성

이벤트 유형을 생성하려면

1. AWS Management Console을 열고 계정에 로그인합니다. Amazon Fraud Detector로 이동합니다.
2. 왼쪽 탐색 창에서 이벤트를 선택합니다.
3. 이벤트 유형 페이지에서 생성을 선택합니다.
4. 이벤트 유형 세부 정보에서,
   a. 이름에 이벤트 이름을 입력합니다.
   b. 선택적으로 설명에 설명을 입력합니다.
   c. 개체에서 이벤트에 대해 생성한 개체 유형을 선택합니다.
5. 이벤트 변수에서
   - 이 이벤트의 변수를 정의하는 방법 선택에서
     - 이 이벤트에 대한 이벤트 변수를 이미 생성한 경우 변수 목록에서 변수 선택을 선택하고 변수에서이 이벤트에 대해 생성한 변수를 선택합니다.
     - 이 이벤트에 대한 변수를 생성하지 않은 경우 훈련 데이터 세트에서 변수 선택을 선택합니다.
       - IAM 역할에서 Amazon Fraud Detector가 데이터 세트가 포함된 Amazon S3 버킷에 액세스하는 데 사용할 IAM 역할을 선택합니다.
       - 데이터 위치에 데이터 세트 위치의 경로를 입력합니다. 다음과 유사한 `S3 URI` 경로를 사용합니다`S3://your-bucket-name/example dataset filename.csv`.
       - 업로드를 선택합니다.
       - 변수에는 Amazon Fraud Detector가 데이터 세트 파일에서 추출한 모든 이벤트 변수 이름이 표시됩니다.

         사기 감지를 위해 변수를 포함하려면 변수 유형에서 변수 유형을 선택합니다. 제거를 선택하여 사기 탐지에 포함되는 변수에서 변수를 제거합니다. 목록의 각 변수에 대해이 단계를 반복합니다.
6. 레이블(선택 사항)의 레이블에서이 이벤트에 대해 생성한 레이블을 선택합니다. 사기 및 합법적인 이벤트에 대해 각각 하나의 레이블을 선택해야 합니다.
7. 이 이벤트에 대한 자동 다운스트림 처리를 설정하려면 Amazon EventBridge를 사용한 이벤트 오케스트레이션 - 선택 사항에서 Amazon EventBridge를 사용한 이벤트 오케스트레이션 활성화를 켭니다. 이벤트 오케스트레이션에 대한 자세한 내용은 섹션을 참조하세요이벤트 오케스트레이션.

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

8. 이벤트 유형 생성을 선택합니다.

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

다음 예제에서는 `PutEventType` API에 대한 샘플 요청을 보여줍니다. 이 예제에서는 변수 `ip_address` 및 `email_address`, 레이블 `legit` 및 `fraud`, 개체 유형를 생성했다고 가정합니다`sample_customer`. 이러한 리소스를 생성하는 방법에 대한 자세한 내용은 섹션을 참조하세요리소스.

> **Note**
>
> 변수, 엔터티 유형 및 레이블을 이벤트 유형에 추가하기 전에 먼저 생성해야 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 삭제하고 이벤트와 연결된 데이터는 더 이상 Amazon Fraud Detector에 저장되지 않습니다.

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

1. 에 로그인 AWS Management Console 하고 https://console.aws.amazon.com/frauddetector Amazon Fraud Detector 콘솔을 엽니다.
2. 콘솔의 왼쪽 탐색 창에서 과거 예측 검색을 선택합니다.

3. 삭제할 이벤트를 선택합니다.
4. 작업을 선택한 다음 이벤트 삭제를 선택합니다.
5. **delete**를 입력한 다음 이벤트 삭제를 선택합니다.

> **Note**
>
> 그러면 작업으로 전송된 이벤트 데이터 및 SendEvent 작업을 통해 생성된 예측 데이터를 포함하여 해당 이벤트 ID와 연결된 모든 레코드가 삭제됩니다GetEventPrediction.

Amazon Fraud Detector에 저장되었지만 평가되지 않은(즉, SendEvent 작업을 통해 저장된) 이벤트를 삭제하려면 DeleteEvent 요청을 하고 이벤트 ID 및 이벤트 유형 ID를 지정해야 합니다. 이벤트와 이벤트와 관련된 예측 기록을 모두 삭제하려면 deleteAuditHistory 파라미터 값을 “true”로 설정합니다. deleteAuditHistory 파라미터가 “true”로 설정된 경우 삭제 작업이 완료된 후 최대 30초 동안 검색을 통해 이벤트 데이터를 사용할 수 있습니다.

이벤트 유형과 연결된 모든 이벤트를 삭제하려면

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

이벤트 유형에 대해 저장된 이벤트 수에 따라 저장된 모든 이벤트를 삭제하는 데 시간이 걸릴 수 있습니다. 예를 들어 1GB 데이터 세트(평균 고객의 경우 약 1~2백만 개의 이벤트)를 삭제하는 데 약 2시간이 걸립니다. 이 기간 동안이 이벤트 유형의 Amazon Fraud Detector로 보내는 새 이벤트는 저장되지 않지만 GetEventPrediction 작업을 통해 사기 예측을 계속 생성할 수 있습니다.

이벤트 유형을 삭제하려면

감지기 또는 모델에 사용되거나 연결된 저장된 이벤트가 있는 이벤트 유형은 삭제할 수 없습니다. 이벤트 유형을 삭제하기 전에 해당 이벤트 유형과 연결된 모든 이벤트를 삭제해야 합니다.

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

1. Amazon Fraud Detector 콘솔의 왼쪽 탐색 창에서 리소스를 선택한 다음 이벤트를 선택합니다.
2. 삭제할 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 Fraud Detector를 사용하여 내부적으로 또는 Amazon Simple Storage Service(Amazon S3)를 사용하여 외부적으로 데이터 세트를 저장합니다. 사기 예측을 생성하는 데 사용하는 모델을 기반으로 데이터 세트를 저장할 위치를 선택하는 것이 좋습니다. 다음은이 두 스토리지 옵션에 대한 세부 분석입니다.

- 내부 스토리지 - 데이터 세트는 Amazon Fraud Detector에 저장됩니다. 이벤트와 연결된 모든 이벤트 데이터는 함께 저장됩니다. 언제든지 Amazon Fraud Detector에 저장된 이벤트 데이터 세트를 업로드할 수 있습니다. Amazon Fraud Detector API로 이벤트를 한 번에 하나씩 스트리밍하거나 배치 가져오기 기능을 사용하여 대규모 데이터 세트(최대 1GB)를 가져올 수 있습니다. Amazon Fraud Detector에 저장된 데이터 세트를 사용하여 모델을 훈련할 때 데이터 세트의 크기를 제한하는 시간 범위를 지정할 수 있습니다.
- 외부 스토리지 - 데이터 세트는 Amazon Fraud Detector 이외의 외부 데이터 소스에 저장됩니다. 현재 Amazon Fraud Detector는 이러한 목적으로 Amazon Simple Storage Service(Amazon S3) 사용을 지원합니다. 모델이 Amazon S3에 업로드된 파일에 있는 경우 해당 파일은 압축되지 않은 데이터 5GB를 초과할 수 없습니다. 이보다 크면 데이터 세트의 시간 범위를 줄여야 합니다.

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

모델 유형	호환되는 훈련 데이터 소스
온라인 사기 인사이트	외부 스토리지, 내부 스토리지
트랜잭션 사기 인사이트	내부 스토리지
계정 탈취 인사이트	내부 스토리지

Amazon Simple Storage Service를 사용하여 데이터 세트를 외부에 저장하는 방법에 대한 자세한 내용은 섹션을 참조하세요Amazon S3를 사용하여 이벤트 데이터를 외부에 저장 . Amazon Fraud Detector를 사용하여 데이터세트를 내부적으로 저장하는 방법에 대한 자세한 내용은 섹션을 참조하세요Amazon Fraud Detector를 사용하여 이벤트 데이터를 내부적으로 저장.

Amazon S3를 사용하여 이벤트 데이터를 외부에 저장

온라인 사기 인사이트 모델을 훈련하는 경우 Amazon S3를 사용하여 이벤트 데이터를 외부에 저장하도록 선택할 수 있습니다. Amazon S3에 이벤트 데이터를 저장하려면 먼저 CSV 형식으로 텍스트 파일을 생성하고 이벤트 데이터를 추가한 다음 Amazon S3 버킷에 CSV 파일을 업로드해야 합니다.

> **Note**
>
> Transaction Fraud Insights 및 Account Takeover Insights 모델 유형은 Amazon S3로 외부에 저장된 데이터 세트를 지원하지 않습니다.

CSV 파일 생성

Amazon Fraud Detector를 사용하려면 CSV 파일의 첫 번째 행에 열 헤더가 포함되어야 합니다. CSV 파일의 열 헤더는 이벤트 유형에 정의된 변수에 매핑되어야 합니다. 예제 데이터 세트는 섹션을 참조하세요. 예제 데이터 세트 가져오기 및 업로드

Online Fraud Insights 모델에는 최소 2개의 변수와 최대 100개의 변수가 있는 훈련 데이터 세트가 필요합니다. 이벤트 변수 외에도 훈련 데이터 세트에는 다음 헤더가 포함되어야 합니다.

- EVENT_TIMESTAMP - 이벤트가 발생한 시기를 정의합니다.
- EVENT_LABEL - 이벤트를 사기성 또는 합법적인 것으로 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 해당하고의 값은 레이블 목록의 값에 EVENT_LABEL 해당합니다.

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

이벤트 타임스탬프 형식

이벤트 타임스탬프가 필수 형식인지 확인합니다. 모델 빌드 프로세스의 일환으로 Online Fraud Insights 모델 유형은 이벤트 타임스탬프를 기반으로 데이터를 주문하고 훈련 및 테스트 목적으로 데이터를 분할합니다. 성능을 공정하게 추정하기 위해 모델은 먼저 훈련 데이터 세트를 훈련한 다음 테스트 데이터 세트에서이 모델을 테스트합니다.

Amazon Fraud Detector는 모델 훈련 EVENT_TIMESTAMP 중에의 값에 대해 다음과 같은 날짜/타임스탬프 형식을 지원합니다.

- %yyyy-%mm-%ddT%hh:%mm:%ssZ(밀리초 없이 UTC 전용 ISO 8601 표준)

  예: 2019-11-30T13:01:01Z
- %yyyy/%mm/%dd %hh:%mm:%ss(AM/PM)

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

  예: 11/30/2019 1:01:01 PM, 11/30/2019 13:01:01
- %mm/%dd/%yy %hh:%mm:%ss

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

Amazon Fraud Detector는 이벤트 타임스탬프에 대한 날짜/타임스탬프 형식을 구문 분석할 때 다음과 같이 가정합니다.

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

- 월과 일에는 한 자릿수 또는 두 자릿수를 입력할 수 있습니다. 예를 들어 1/12/2019은 유효한 날짜입니다.
- hh:mm:ss가 없는 경우 hh:mm:ss를 포함할 필요가 없습니다(예: 날짜를 제공하면 됩니다). 시간 및 분(예: hh:mm)의 하위 집합만 제공할 수도 있습니다. 시간만 제공하면 지원되지 않습니다. 밀리초도 지원되지 않습니다.
- AM/PM 레이블을 입력하면 12시간 클럭이 가정됩니다. AM/PM 정보가 없는 경우 24시간 시계를 가정합니다.
- 날짜 요소의 구분 기호로 “/” 또는 “-”를 사용할 수 있습니다. 타임스탬프 요소에는 “:”가 사용됩니다.

시간 경과에 따른 데이터 세트 샘플링

동일한 시간 범위의 사기 및 합법적인 샘플의 예를 제공하는 것이 좋습니다. 예를 들어 지난 6개월 동안의 사기 이벤트를 제공하는 경우 동일한 기간에 균등하게 적용되는 합법적인 이벤트도 제공해야 합니다. 데이터 세트에 매우 고르지 않은 사기 및 합법적인 이벤트 분포가 포함되어 있는 경우 "시간별 사기 분포가 허용할 수 없을 정도로 변동합니다. 데이터 세트를 올바르게 분할할 수 없습니다.” 일반적으로 이 오류의 가장 쉬운 해결 방법은 사기 이벤트와 합법적인 이벤트가 동일한 기간에 균등하게 샘플링되도록 하는 것입니다. 또한 짧은 기간 내에 사기가 크게 급증한 경우 데이터를 제거해야 할 수도 있습니다.

균등하게 분산된 데이터 세트를 생성하기에 충분한 데이터를 생성할 수 없는 경우 한 가지 접근 방식은 이벤트의 EVENT_TIMESTAMP를 균등하게 분산되도록 무작위화하는 것입니다. 그러나 Amazon Fraud Detector는 EVENT_TIMESTAMP를 사용하여 데이터 세트의 적절한 이벤트 하위 집합에 대한 모델을 평가하기 때문에 성능 지표가 비현실적으로 나타나는 경우가 많습니다.

Null 및 누락된 값

Amazon Fraud Detector는 null 및 누락 값을 처리합니다. 그러나 변수에 대한 null의 백분율은 제한되어야 합니다. EVENT_TIMESTAMP 및 EVENT_LABEL 열에는 누락된 값이 포함되어서는 안 됩니다.

파일 검증

다음 조건 중 하나라도 트리거되면 Amazon Fraud Detector가 모델 훈련에 실패합니다.

- CSV를 구문 분석할 수 없는 경우
- 열의 데이터 유형이 잘못된 경우

Amazon S3 버킷에 이벤트 데이터 업로드

이벤트 데이터로 CSV 파일을 생성한 후 Amazon S3 버킷에 파일을 업로드합니다.

Amazon S3 버킷에 업로드하려면

1. 에 로그인 AWS Management Console 하고 https://console.aws.amazon.com/s3/ 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 Simple Storage Service 사용 설명서의 버킷 이름 지정 규칙을 참조하세요.

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

   모든 설정을 활성화된 상태로 두는 것이 좋습니다. 퍼블릭 액세스 차단에 대한 자세한 내용은 Amazon Simple Storage Service 사용 설명서의 Amazon S3 스토리지에 대한 퍼블릭 액세스 차단을 참조하세요.

6. 버킷 생성을 선택합니다.
7. 훈련 데이터 파일을 Amazon S3 버킷에 업로드합니다. 훈련 파일의 Amazon S3 위치 경로(예: s3://bucketname/object.csv)를 기록해 둡니다.

Amazon Fraud Detector를 사용하여 이벤트 데이터를 내부적으로 저장

이벤트 데이터를 Amazon Fraud Detector에 저장하고 나중에 저장된 데이터를 사용하여 모델을 훈련하도록 선택할 수 있습니다. Amazon Fraud Detector에 이벤트 데이터를 저장하면 자동 계산 변수를 사용하는 모델을 훈련하여 성능을 개선하고, 모델 재학습을 간소화하고, 사기 레이블을 업데이트하여 기계 학습 피드백 루프를 닫을 수 있습니다. 이벤트는 이벤트 유형 리소스 수준에 저장되므로 동일한 이벤트 유형의 모든 이벤트는 단일 이벤트 유형 데이터 세트에 함께 저장됩니다. 이벤트 유형을 정의하는 과정에서 Amazon Fraud Detector 콘솔에서 이벤트 수집 설정을 전환하여 해당 이벤트 유형에 대한 이벤트를 저장할지 여부를 선택적으로 지정할 수 있습니다.

Amazon Fraud Detector에서 단일 이벤트를 저장하거나 많은 수의 이벤트 데이터 세트를 가져올 수 있습니다. GetEventPrediction API 또는 SendEvent API를 사용하여 단일 이벤트를 스트리밍할 수 있습니다. Amazon Fraud Detector 콘솔의 배치 가져오기 기능을 사용하거나 CreateBatchImportJob API를 사용하여 대규모 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.

스토리지를 위한 이벤트 데이터 준비

Amazon Fraud Detector를 사용하여 내부적으로 저장된 이벤트 데이터는 `Event Type` 리소스 수준에서 저장됩니다. 따라서 동일한 이벤트의 모든 이벤트 데이터는 단일에 저장됩니다`Event Type`. 저장된 이벤트는 나중에 새 모델을 훈련하거나 기존 모델을 재학습하는 데 사용할 수 있습니다. 저장된 이벤트 데이터를 사용하여 모델을 훈련할 때 선택적으로 이벤트의 시간 범위를 지정하여 훈련 데이터 세트의 크기를 제한할 수 있습니다.

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

Amazon Fraud Detector를 사용하여 내부적으로 데이터를 저장하기 위한 사전 조건

- 이벤트 데이터가 검증을 통과하고 데이터 세트가 성공적으로 저장되도록 하려면 데이터 모델 탐색기에서 제공하는 인사이트를 사용하여 데이터 세트를 준비해야 합니다.
- Amazon Fraud Detector에 저장하려는 이벤트 데이터에 대한 이벤트 유형을 생성했습니다. 그렇지 않은 경우 지침에 따라 이벤트 유형 생성을 수행합니다.

스마트 데이터 검증

배치 가져오기를 위해 Amazon Fraud Detector 콘솔에 데이터 세트를 업로드하면 Amazon Fraud Detector는 데이터를 가져오기 전에 스마트 데이터 검증(SDV)을 사용하여 데이터 세트를 검증합니다. SDV는 업로드된 데이터 파일을 스캔하고 누락된 데이터, 잘못된 형식 또는 데이터 유형과 같은 문제를 식별합니다. 데이터 세트를 검증하는 것 외에도 SDV는 식별된 모든 문제를 나열하고 가장 영향을 미치는 문제를 해결하기 위한 조치를 제안하는 검증 보고서도 제공합니다. SDV로 식별되는 일부 문제는 중요할 수 있으며 Amazon Fraud Detector가 데이터 세트를 성공적으로 가져오려면 먼저 해결해야 합니다. 자세한 내용은 스마트 데이터 검증 보고서 단원을 참조하십시오.

SDV는 파일 수준과 데이터(행) 수준에서 데이터 세트를 검증합니다. 파일 수준에서 SDV는 데이터 파일을 스캔하고 파일에 액세스할 수 있는 부적절한 권한, 잘못된 파일 크기, 파일 형식 및 헤더(이벤트 메타데이터 및 이벤트 변수)와 같은 문제를 식별합니다. 데이터 수준에서 SDV는 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 기본적으로 검증이 켜져 있습니다. 데이터 세트를 가져오기 전에 Amazon Fraud Detector가 스마트 데이터 검증을 사용하지 않도록 하려면 데이터 세트를 업로드할 때 Amazon Fraud Detector 콘솔에서 검증을 끄세요.

APIs 또는 AWS SDK를 사용할 때 저장된 데이터 검증

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

- 해당 이벤트 유형에 대한 EventIngestion 설정은 ENABLED입니다.
- 이벤트 타임스탬프는 업데이트할 수 없습니다. 이벤트 ID가 반복되고 EVENT_TIMESTAMP가 다른 이벤트는 오류로 처리됩니다.
- 변수 이름과 값은 예상 형식과 일치합니다. 자세한 내용은 변수 생성 섹션을 참조하세요.
- 필수 변수는 값으로 채워집니다.
- 모든 이벤트 타임스탬프는 18개월 이하이며 미래가 아닙니다.

배치 가져오기를 사용하여 이벤트 데이터 저장

배치 가져오기 기능을 사용하면 콘솔, API 또는 AWS SDK를 사용하여 Amazon Fraud Detector에서 대규모 과거 이벤트 데이터 세트를 빠르고 쉽게 업로드할 수 있습니다. 배치 가져오기를 사용하려면 모든 이벤트 데이터가 포함된 CSV 형식의 입력 파일을 생성하고, Amazon S3 버킷에 CSV 파일을 업로드하고, 가져오기 작업을 시작합니다. Amazon Fraud Detector는 먼저 이벤트 유형에 따라 데이터를 검증한 다음 전체 데이터 세트를 자동으로 가져옵니다. 데이터를 가져온 후에는 새 모델을 훈련하거나 기존 모델을 재학습하는 데 사용할 준비가 된 것입니다.

입력 및 출력 파일

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

CSV 파일 생성

Amazon Fraud Detector는 쉼표로 구분된 값(CSV) 형식의 파일에서만 데이터를 가져옵니다. CSV 파일의 첫 번째 행에는 연결된 이벤트 유형에 정의된 변수와 정확히 일치하는 열 헤더와 EVENT_ID, EVENT_TIMESTAMP, ENTITY_ID, ENTITY_TYPE이라는 네 가지 필수 변수가 포함되어야 합니다. EVENT_LABEL 및 LABEL_TIMESTAMP를 선택적으로 포함할 수도 있습니다(EVENT_LABEL이 포함된 경우 LABEL_TIMESTAMP 필요).

필수 변수 정의

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

명칭	설명	요구 사항
EVENT_ID	이벤트의 식별자입니다. 예를 들어 이벤트가 온라인 트랜잭션인 경우 EVENT_ID는 고객에게 제공된 트랜잭션 참조 번호일 수 있습니다.	• EVENT_ID는 배치 가져오기 작업에 필요합니다. • 해당 이벤트에 대해 고유해야 합니다.

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

명칭	설명	요구 사항

EVENT_TIMESTAMP	이벤트가 발생한 시간의 타임스탬프입니다. 타임스탬프는 UTC의 ISO 8601 표준이어야 합니다.	• EVENT_TIMESTAMP는 배치 가져오기 작업에 필요합니다.
		• 다음 형식 중 하나로 지정해야 합니다.
		  • %yyyy-%mm-%ddT%hh:%mm:%ssZ(밀리초 없이 UTC 전용 ISO 8601 표준)
		  예: 2019-11-30T13:01:01Z
		  • %yyyy/%mm/%dd %hh:%mm:%ss(AM/PM)
		  예: 2019/11/30 1:01:01 PM 또는 2019/11/30 13:01:01
		  • %mm/%dd/%yyyy %hh:%mm:%ss
		  예: 11/30/2019 1:01:01 PM, 11/30/2019 13:01:01
		  • %mm/%dd/%yy %hh:%mm:%ss
		  예: 11/30/19 1:01:01 PM, 11/30/19 13:01:01
		• Amazon Fraud Detector는 이벤트 타임스탬프에 대한 날짜/타임스탬프 형식을 구문 분석할 때 다음과 같이 가정합니다.

명칭	설명	요구 사항

		• ISO 8601 표준을 사용하는 경우 이전 사양과 정확히 일치해야 합니다.
		• 다른 형식 중 하나를 사용하는 경우 추가 유연성이 있습니다.
		  • 월과 일에는 한 자릿수 또는 두 자릿수를 입력할 수 있습니다. 예를 들어 1/12/2019은 유효한 날짜입니다.
		  • hh:mm:ss가 없는 경우 hh:mm:ss를 포함할 필요가 없습니다(즉, 단순히 날짜를 제공할 수 있음). 시간 및 분(예: hh:mm)의 하위 집합만 제공할 수도 있습니다. 시간만 제공하면 지원되지 않습니다. 밀리초도 지원되지 않습니다.
		  • AM/PM 레이블을 입력하면 12시간 클럭이 가정됩니다. AM/PM 정보가 없는 경우 24시간 시계를 가정합니다.
		  • 날짜 요소의 구분 기호로 “/” 또는 “-”를 사용할 수 있습니다. 타임스탬프 요소에는 “:”가 사용됩니다.

명칭	설명	요구 사항
ENTITY_ID	이벤트를 수행하는 개체의 식별자입니다.	• 배치 가져오기 작업에는 ENTITY_ID가 필요합니다. • 정규식 패턴을 따라야 합니다^[0-9A-Za-z_.@+-]+$. • 평가 시 엔터티 ID를 사용할 수 없는 경우 엔터티 ID를 알 수 없음으로 지정합니다.
ENTITY_TYPE	판매자 또는 고객과 같이 이벤트를 수행하는 엔터티	배치 가져오기 작업에는 ENTITY_TYPE이 필요합니다.
EVENT_LABEL	이벤트를 `fraudulent` 또는 `legitimate`로 분류합니다.	LABEL_TIMESTAMP가 포함된 경우 EVENT_LABEL이 필요합니다.
LABEL_TIMESTAMP	이벤트 레이블이 마지막으로 채워지거나 업데이트된 타임스탬프	• EVENT_LABEL이 포함된 경우 LABEL_TIMESTAMP가 필요합니다. • 타임스탬프 형식을 따라야 합니다.

배치 가져오기를 위해 Amazon S3에 CSV 파일 업로드

데이터로 CSV 파일을 생성한 후 Amazon Simple Storage Service(Amazon S3) 버킷에 파일을 업로드합니다.

Amazon S3 버킷에 이벤트 데이터를 업로드하려면

1. 에 로그인 AWS Management Console 하고 https://console.aws.amazon.com/s3/ 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 Simple Storage Service 사용 설명서의 버킷 이름 지정 규칙을 참조하세요.

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

4. 리전에서 버킷이 상주할 AWS 리전을 선택합니다. Amazon Fraud Detector를 사용하는 동일한 리전, 즉 미국 동부(버지니아 북부), 미국 동부(오하이오), 미국 서부(오레곤), 유럽(아일랜드), 아시아 태평양(싱가포르) 또는 아시아 태평양(시드니)을 선택해야 합니다.
5. Bucket settings for Block Public Access(퍼블릭 액세스 차단을 위한 버킷 설정)에서 버킷에 적용할 퍼블릭 액세스 차단 설정을 선택합니다.

   모든 설정을 활성화된 상태로 두는 것이 좋습니다. 퍼블릭 액세스 차단에 대한 자세한 내용은 Amazon Simple Storage Service 사용 설명서의 Amazon S3 스토리지에 대한 퍼블릭 액세스 차단을 참조하세요.
6. 버킷 생성을 선택합니다.
7. 훈련 데이터 파일을 Amazon S3 버킷에 업로드합니다. 훈련 파일의 Amazon S3 위치 경로(예: s3://bucketname/object.csv)를 기록해 둡니다.

Amazon Fraud Detector 콘솔에서 이벤트 데이터 일괄 가져오기

`CreateBatchImportJob` API 또는 AWS SDK를 사용하여 Amazon Fraud Detector 콘솔에서 많은 수의 이벤트 데이터 세트를 쉽게 가져올 수 있습니다. 계속하기 전에 지침에 따라 데이터 세트를 CSV 파일로 준비했는지 확인합니다. Amazon S3 버킷에도 CSV 파일을 업로드했는지 확인합니다.

Amazon Fraud Detector 콘솔 사용

콘솔에서 이벤트 데이터 일괄 가져오기

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 ON인지 확인합니다.
6. 이벤트 데이터 가져오기 창에서 새 가져오기를 선택합니다.
7. 새 이벤트 가져오기 페이지에서 다음 정보를 제공합니다.

   - [권장] 이 데이터 세트에 대해 스마트 데이터 검증 활성화 - 새를 기본 설정으로 설정한 상태로 둡니다.
   - 데이터에 대한 IAM 역할에서 가져오려는 CSV 파일이 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 결과 버튼을 위해 데이터 위치 분리를 클릭하고 유효한 Amazon S3 버킷 위치를 제공합니다.

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

8. 시작을 선택합니다.
9. 이벤트 데이터 가져오기 창의 상태 열에는 검증 및 가져오기 작업의 상태가 표시됩니다. 상단의 배너는 데이터 세트가 먼저 검증을 거친 다음 가져오기를 수행할 때 상태에 대한 높은 수준의 설명을 제공합니다.
10. 에 제공된 지침을 따릅니다데이터 세트 검증 및 가져오기 작업 진행 상황 모니터링.

데이터 세트 검증 및 가져오기 작업 진행 상황 모니터링

Amazon Fraud Detector 콘솔을 사용하여 배치 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 상황과 상태를 모니터링할 수 있습니다. 페이지 상단의 배너는 검증 결과와 가져오기 작업의 상태에 대한 간략한 설명을 제공합니다. 검증 결과 및 가

져오기 작업 상태에 따라 데이터 세트의 성공적인 검증 및 가져오기를 보장하기 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 표시기	의미	어떻게 해야 하나요?
데이터 검증이 시작되었습니다.	검증 진행 중	SDV가 데이터 세트 검증을 시작했습니다.	상태가 변경될 때까지 기다립니다.
데이터 세트의 오류로 인해 데이터 검증을 진행할 수 없습니다. 데이터 파일의 오류를 수정하고 새 가져오기 작업을 시작합니다. 자세한 내용은 검증 보고서를 참조하세요.	검증 실패	SDV가 데이터 파일에서 문제를 식별했습니다. 데이터 세트를 성공적으로 가져오려면 이러한 문제를 해결해야 합니다.	이벤트 데이터 가져오기 창에서 작업 ID를 선택하고 검증 보고서를 확인합니다. 보고서의 권장 사항에 따라 나열된 모든 오류를 해결합니다. 자세한 내용은 검증 보고서 사용 단원을 참조하십시오.
데이터 가져오기가 시작되었습니다. 검증이 성공적으로 완료되었습니다.	가져오기 진행 중	데이터 세트가 검증을 통과했습니다. AFD가 데이터 세트 가져오기를 시작했습니다.	상태가 변경될 때까지 기다립니다.
경고와 함께 검증이 완료되었습니다. 데이터 가져오기가 시작되었습니다.	가져오기 진행 중	데이터 세트의 일부 데이터가 검증에 실패했습니다. 그러나 검증을 통과	배너의 메시지를 모니터링하고 상태가 변경될 때까지 기다립니다.

배너 메시지	상태 표시기	의미	어떻게 해야 하나요?
		한 데이터는 가져오기에 대한 최소 데이터 크기 요구 사항을 충족합니다.	
데이터를 부분적으로 가져왔습니다. 일부 데이터는 검증에 실패했으며 가져오지 못했습니다. 자세한 내용은 검증 보고서를 참조하세요.	가져옴. 상태에 경고 아이콘이 표시됩니다.	검증에 실패한 데이터 파일의 일부 데이터를 가져오지 못했습니다. 검증을 통과한 나머지 데이터를 가져왔습니다.	이벤트 데이터 가져오기 창에서 작업 ID를 선택하고 검증 보고서를 확인합니다. 데이터 수준 경고 표의 권장 사항에 따라 나열된 경고를 해결합니다. 모든 경고를 해결할 필요는 없습니다. 그러나 데이터 세트에 성공적인 가져오기에 대한 검증을 통과한 데이터의 50% 이상이 있는지 확인합니다. 경고를 해결한 후 새 가져오기 작업을 시작합니다. 자세한 내용은 검증 보고서 사용 단원을 참조하십시오.
처리 오류로 인해 데이터 가져오기에 실패했습니다. 새 데이터 가져오기 작업 시작	가져오기에 실패했습니다.	일시적인 런타임 오류로 인해 가져오기에 실패했습니다.	새 가져오기 작업 시작
데이터를 성공적으로 가져왔습니다.	가져옴	검증 및 가져오기가 모두 성공적으로 완료되었습니다.	가져오기 작업의 작업 ID를 선택하여 세부 정보를 확인한 다음 모델 교육을 진행합니다.

> **Note**
>
> 데이터 세트를 Amazon Fraud Detector로 성공적으로 가져온 후 시스템에서 완전히 수집되도록 10분을 기다리는 것이 좋습니다.

스마트 데이터 검증 보고서

스마트 데이터 검증은 검증이 완료된 후 검증 보고서를 생성합니다. 검증 보고서는 SDV가 데이터 세트에서 식별한 모든 문제에 대한 세부 정보와 가장 큰 영향을 미치는 문제를 해결하기 위한 권장 작업을 제공합니다. 검증 보고서를 사용하여 문제가 무엇인지, 데이터 세트에서 문제가 있는 위치, 문제의 심각도 및 해결 방법을 확인할 수 있습니다. 검증이 성공적으로 완료되면 검증 보고서가 생성됩니다. 이 경우 보고서를 보고 나열된 문제가 있는지 확인하고 문제가 있는 경우 수정할지 여부를 결정할 수 있습니다.

> **Note**
>
> 현재 버전의 SDV는 데이터 세트에서 배치 가져오기가 실패할 수 있는 문제를 스캔합니다. 검증 및 배치 가져오기에 성공하더라도 데이터 세트에 모델 훈련이 실패할 수 있는 문제가 있을 수 있습니다. 검증 및 가져오기가 성공했더라도 검증 보고서를 보고 성공적인 모델 훈련을 위해 보고서에 나열된 문제를 해결하는 것이 좋습니다. 문제를 해결한 후 새 배치 가져오기 작업을 생성합니다.

검증 보고서 액세스

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

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

검증 보고서 사용

가져오기 작업의 검증 보고서 페이지에는이 가져오기 작업의 세부 정보, 발견된 경우 중요한 오류 목록, 발견된 경우 데이터 세트의 특정 이벤트(행)에 대한 경고 목록, 유효하지 않은 값, 각 변수의 누락된 값과 같은 정보가 포함된 데이터 세트의 간략한 요약이 제공됩니다.

- 작업 세부 정보 가져오기

가져오기 작업에 대한 세부 정보를 제공합니다. 가져오기 작업이 실패했거나 데이터 세트를 부분적으로 가져온 경우 결과 파일로 이동을 선택하여 가져오지 못한 이벤트의 오류 로그를 확인합니다.

- 심각한 오류

  SDV로 식별되는 데이터 세트에서 가장 영향을 미치는 문제에 대한 세부 정보를 제공합니다. 이 창에 나열된 모든 문제는 중요하며 가져오기를 진행하기 전에 해결해야 합니다. 중요한 문제를 해결하지 않고 데이터 세트를 가져오려고 하면 가져오기 작업이 실패할 수 있습니다.

  중요한 문제를 해결하려면 각 경고에 제공된 권장 사항을 따르세요. 심각한 오류 창에 나열된 모든 문제를 해결한 후 새 배치 가져오기 작업을 생성합니다.

- 데이터 수준 경고

  데이터 세트의 특정 이벤트(행)에 대한 경고 요약을 제공합니다. 데이터 수준 경고 창이 채워지면 데이터 세트의 일부 이벤트가 검증에 실패하여 가져오지 못한 것입니다.

  각 경고에 대해 설명 열에는 문제가 있는 이벤트 수가 표시됩니다. 또한 샘플 이벤트 IDs는 문제가 있는 나머지 이벤트를 찾기 위한 시작점으로 사용할 수 있는 샘플 이벤트 IDs의 일부 목록을 제공합니다. 경고에 제공된 권장 사항을 사용하여 문제를 해결합니다. 또한 문제에 대한 추가 정보를 보려면 출력 파일의 오류 로그를 사용합니다. 오류 로그는 배치 가져오기에 실패한 모든 이벤트에 대해 생성됩니다. 오류 로그에 액세스하려면 가져오기 작업 세부 정보 창에서 결과 파일로 이동을 선택합니다.

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

- 데이터 세트 요약

  데이터 세트의 검증 보고서 요약을 제공합니다. 경고 수 열에 0개 이상의 경고가 표시되면 해당 경고를 수정해야 하는지 여부를 결정합니다. 경고 수 열에 0이 표시되면 모델 훈련을 계속합니다.

Python용 AWS SDK(Boto3)를 사용하여 이벤트 데이터 일괄 가져오기

다음 예제에서는 CreateBatchImportJob API에 대한 샘플 요청을 보여줍니다. 배치 가져오기 작업에는 jobID, inputPath, outputPath, eventTypeName 및 iamRoleArn이 포함되어야 합니다. 작업이 CREATE_FAILED 상태로 존재하지 않는 한 jobID에는 이전 작업의 동일한 ID가 포함될 수 없습니다. inputPath 및 outputPath는 유효한 S3 경로여야 합니다. outputPath에서 파일 이름 지정을 옵트아웃할 수 있지만 유효한 S3 버킷 위치를 제공해야 합니다. eventTypeName 및 iamRoleArn이 있어야 합니다.

IAM 역할은 Amazon S3 버킷을 입력할 수 있는 읽기 권한과 Amazon S3 버킷을 출력할 수 있는 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-*************'
)
```

배치 가져오기 작업 취소

Amazon Fraud Detector 콘솔에서 `CancelBatchImportJob` 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 데이터 가져오기 창에서 취소하려는 진행 중인 가져오기 작업의 작업 ID를 선택합니다.
6. 이벤트 작업 페이지에서 작업을 클릭하고 이벤트 가져오기 취소를 선택합니다.
7. 이벤트 가져오기 중지를 선택하여 배치 가져오기 작업을 취소합니다.

Python용 AWS SDK(Boto3)를 사용하여 배치 가져오기 작업 취소

다음 예제에서는 `CancelBatchImportJob` API에 대한 샘플 요청을 보여줍니다. 가져오기 작업 취소에는 진행 중인 배치 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 위해 `GetEventPrediction` API로 전송되는 모든 이벤트는 Amazon Fraud Detector에 저장됩니다. 즉, Amazon Fraud Detector는 예측을 생성할 때 이벤트 데이터를 자동으로 저장하고 해당 데이터를 사용하여 계산된 변수를 거의 실시간으로 업데이트합니다. Amazon Fraud Detector 콘솔에서 이벤트 유형으로 이동하여 이벤트 수집을 OFF로 설정하거나 `PutEventType` API 작업을 사용하여 EventIngestion 값을 DISABLED로 업데이트하여 데이터 스토리지를 비활성화할 수 있습니다. `GetEventPrediction` API 작업에 대한 자세한 내용은 섹션을 참조하세요사기 예측.

> **Important**
>
> 이벤트 유형에 대해 이벤트 수집을 활성화한 후에는 활성화 상태를 유지하는 것이 좋습니다. 동일한 이벤트 유형에 대해 이벤트 수집을 비활성화한 다음 예측을 생성하면 동작이 일관되지 않을 수 있습니다.

SendEvent API 작업을 사용하여 이벤트 데이터 저장

`SendEvent` API 작업을 사용하면 해당 이벤트에 대한 사기 예측을 생성하지 않고도 Amazon Fraud Detector에 이벤트를 저장할 수 있습니다. 예를 들어 `SendEvent` 작업을 사용하여 기록 데이터 세트를 업로드할 수 있으며, 나중에 모델을 훈련하는 데 사용할 수 있습니다.

SendEvent API의 이벤트 타임스탬프 형식

`SendEvent` API를 사용하여 이벤트 데이터를 저장할 때는 이벤트 타임스탬프가 필수 형식인지 확인해야 합니다. Amazon Fraud Detector는 다음 날짜/타임스탬프 형식을 지원합니다.

- %yyyy-%mm-%ddT%hh:%mm:%ssZ(밀리초 없이 UTC 전용 ISO 8601 표준)

  예: 2019-11-30T13:01:01Z
- %yyyy/%mm/%dd %hh:%mm:%ss(AM/PM)

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

예: 11/30/2019 1:01:01 PM, 11/30/2019 13:01:01

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

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

Amazon Fraud Detector는 이벤트 타임스탬프에 대한 날짜/타임스탬프 형식을 구문 분석할 때 다음과 같이 가정합니다.

- ISO 8601 표준을 사용하는 경우 이전 사양과 정확히 일치해야 합니다.
- 다른 형식 중 하나를 사용하는 경우 추가 유연성이 있습니다.
  - 월과 일에는 한 자릿수 또는 두 자릿수를 입력할 수 있습니다. 예를 들어 1/12/2019은 유효한 날짜입니다.
  - hh:mm:ss가 없는 경우 hh:mm:ss를 포함할 필요가 없습니다(즉, 단순히 날짜를 제공할 수 있음). 시간 및 분(예: hh:mm)의 하위 집합만 제공할 수도 있습니다. 시간 제공만 지원되지 않습니다. 밀리초도 지원되지 않습니다.
  - AM/PM 레이블을 입력하면 12시간 클럭이 가정됩니다. AM/PM 정보가 없는 경우 24시간 시계를 가정합니다.
  - 날짜 요소의 구분 기호로 “/” 또는 “-”를 사용할 수 있습니다. 타임스탬프 요소에는 “:”가 사용됩니다.

다음은 SendEvent API 직접 호출의 예입니다.

```
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 GetEvent API를 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 데이터 세트의 지표 보기

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

이벤트 유형의 저장된 이벤트 지표를 보려면

1. AWS 콘솔을 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 창에 지표가 표시됩니다. 이러한 지표는 하루에 한 번 자동으로 업데이트됩니다.
6. 선택적으로 이벤트 지표 새로 고침을 클릭하여 지표를 수동으로 업데이트합니다.

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

이벤트 오케스트레이션

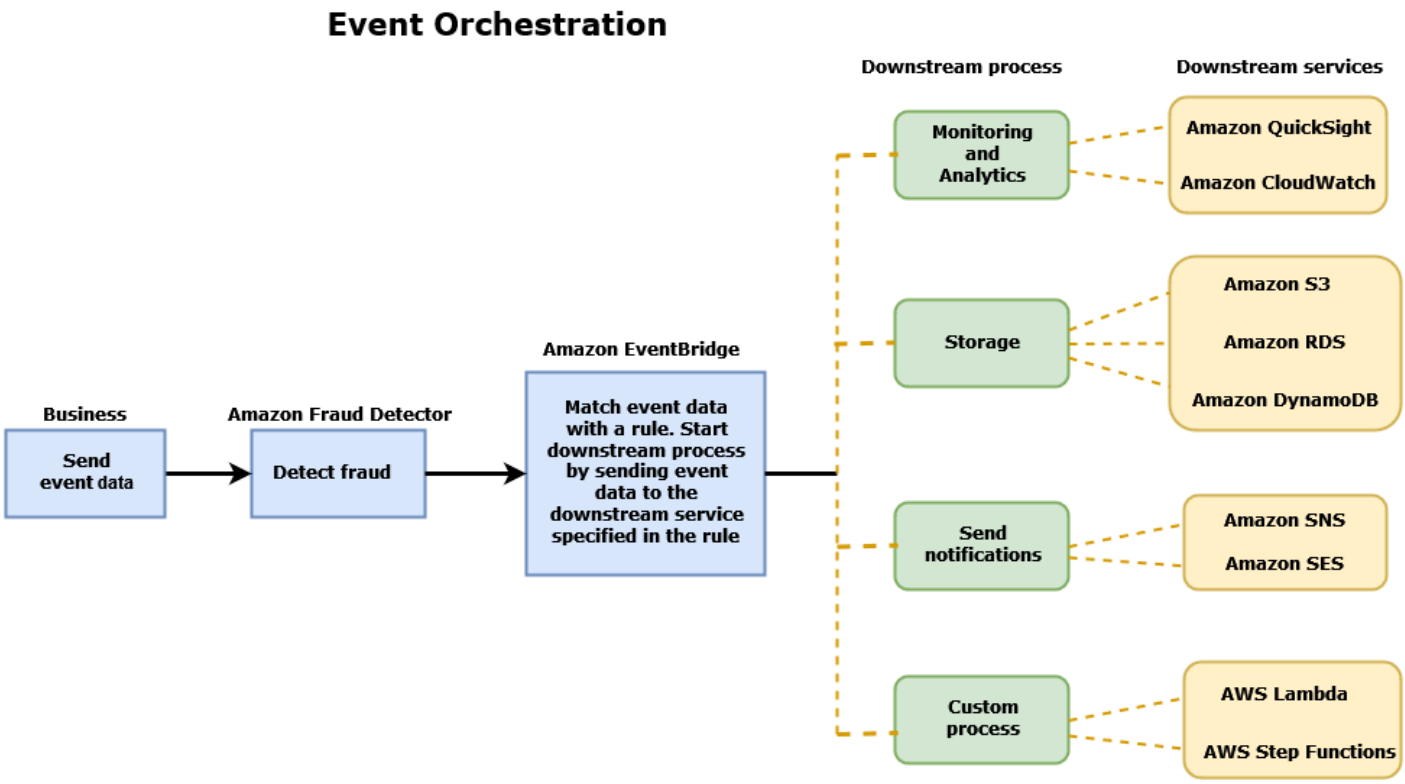
이벤트 오케스트레이션을 사용하면 Amazon EventBridge를 사용하여 다운스트림 처리를 AWS 서비스 위해 이벤트를에 쉽게 보낼 수 있습니다. Amazon Fraud Detector는 사기 탐지 후 이벤트 처리를 자동화하는 데 사용할 수 있는 간단한 규칙을 제공합니다. 이벤트 오케스트레이션을 사용하면 이벤트 데이터를 통해 인사이트를 얻기 위해 대시보드로 이벤트를 전송하고, 사기 탐지 결과를 기반으로 알림을 생성하고, 사기 탐지에서 학습한 결과를 기반으로 레이블을 사용하여 이벤트를 업데이트하는 등의 다운스트림 이벤트 프로세스를 자동화할 수 있습니다.

이벤트 오케스트레이션을 사용하면 Amazon EventBridge를 통해 AWS 환경의 서비스에 쉽게 액세스할 수 있습니다. API 대상을 사용하여 이벤트를 직접 AWS 서비스 또는 간접적으로 보내도록 Amazon EventBridge를 구성할 수 있습니다. 다운스트림 프로세스를 오케스트레이션 AWS 서비스 하는 데 사용하는를 대상이라고도 합니다. 다운스트림 처리를 오케스트레이션하는 데 사용할 수 있는 대상은 다음과 같습니다.

- 모니터링 및 분석용 - Amazon QuickSight, Amazon CloudWatch
- 스토리지용 - Amazon S3, Amazon RDS, Amazon DynamoDB
- 알림 전송용 - Amazon SNS, Amazon SES
- 사용자 지정 처리의 경우 - AWS Lambda, AWS Step Functions

Amazon EventBridge에서 지원하는 오케스트레이션 대상에 대한 자세한 내용은 Amazon EventBridge 대상을 참조하세요.

다음 다이어그램은 이벤트 오케스트레이션의 작동 방식을 전체적으로 보여줍니다.



이벤트 오케스트레이션 설정

이벤트에 대한 이벤트 오케스트레이션을 설정하려면 대상 서비스에서 프로세스를 설정하고, 이벤트 데이터를 수신 및 전송하도록 Amazon EventBridge를 구성하고, 다운스트림 프로세스를 시작하기 위한 조건을 지정하는 규칙을 Amazon EventBridge에서 생성해야 합니다. 다음 단계를 완료하여 이벤트 오케스트레이션을 설정합니다.

이벤트 오케스트레이션을 설정하려면

1. Amazon EventBridge 사용 설명서로 이동하여 Amazon EventBridge를 사용하는 방법을 알아봅니다. 사용 사례에 맞게 Amazon EventBridge에서 규칙을 생성하는 방법을 알아봅니다.
2. 에 대한 지침을 따릅니다Amazon Fraud Detector에서 이벤트 오케스트레이션 활성화.

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

3. 이벤트 데이터를 수신하고 처리하도록 대상 서비스를 설정합니다. 예를 들어 다운스트림 프로세스에 알림 전송이 포함되어 있고 Amazon SNS를 사용하려는 경우 Amazon SNS 콘솔로 이동하여 SNS 주제를 생성한 다음 엔드포인트를 주제에 구독합니다.

4. 지침에 따라 Amazon EventBridge 규칙을 생성합니다.

> **Important**
>
> Amazon EventBridge에서 이벤트 패턴을 빌드할 때는 소스 필드와 세부 정보 유형 필드에 `aws.frauddetector` `Event Prediction Result Returned`를 제공해야 합니다.

Amazon Fraud Detector에서 이벤트 오케스트레이션 활성화

이벤트 유형을 생성할 때 또는 이벤트 유형을 생성한 후 이벤트에 대해 이벤트 오케스트레이션을 활성화할 수 있습니다. Amazon Fraud Detector 콘솔에서 `put-event-type` 명령을 사용하거나 API를 사용하거나를 사용하여 이벤트 오케스트레이션을 활성화할 수 `PutEventType` 있습니다 AWS SDK for Python (Boto3).

Amazon Fraud Detector 콘솔에서 이벤트 오케스트레이션 활성화

이 예제에서는 이미 생성된 이벤트 유형에 대해 이벤트 오케스트레이션을 활성화합니다. 새 이벤트 유형을 생성하고 오케스트레이션을 활성화하려면의 지침을 따르세요이벤트 유형 생성.

이벤트 오케스트레이션을 활성화하려면

1. AWS Management Console을 열고 계정에 로그인합니다. Amazon Fraud Detector로 이동합니다.
2. 왼쪽 탐색 창에서 이벤트를 선택합니다.
3. 이벤트 유형 페이지에서 이벤트 유형을 선택합니다.
4. Amazon EventBridge로 이벤트 오케스트레이션 활성화를 켭니다.
5. 에 대한 3단계 지침을 계속 진행합니다이벤트 오케스트레이션 설정.

를 사용하여 이벤트 오케스트레이션 활성화 AWS SDK for Python (Boto3)

다음 예제에서는 이벤트 오케스트레이션을 활성화하도록 이벤트 유형을 업데이트`sample_registration`하기 위한 샘플 요청을 보여줍니다. 이 예제에서는 `PutEventType` API를 사용하고 변수 및 `email_address`, 레이블 `ip_address` 및 `legit` `fraud`, 엔터티 유형을 생성했다고 가정합니다`sample_customer`. 이러한 리소스를 생성하는 방법에 대한 자세한 내용은 리소스를 참조하세요.

```
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 오케스트레이션 비활성화

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

Amazon Fraud Detector 콘솔에서 이벤트 오케스트레이션 비활성화

이벤트 오케스트레이션을 비활성화하려면

1. AWS Management Console을 열고 계정에 로그인합니다. Amazon Fraud Detector로 이동합니다.
2. 왼쪽 탐색 창에서 이벤트를 선택합니다.
3. 이벤트 유형 페이지에서 이벤트 유형을 선택합니다.
4. Amazon EventBridge로 이벤트 오케스트레이션 활성화를 끕니다.

를 사용하여 이벤트 오케스트레이션 비활성화 AWS SDK for Python (Boto3)

다음 예제에서는 `PutEventType` API를 사용하여 이벤트 오케스트레이션을 비활성화`sample_registration`하도록 이벤트 유형을 업데이트하기 위한 샘플 요청을 보여줍니다.

```
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 훈련됩니다. 모델 유형은 모델 훈련에 사용되는 알고리즘과 변환을 지정합니다. 모델 훈련은 제공한 데이터 세트를 사용하여 사기 이벤트를 예측할 수 있는 모델을 생성하는 프로세스입니다.

모델을 생성하려면 먼저 모델 유형을 선택한 다음 모델 훈련에 사용할 데이터를 준비하고 제공해야 합니다.

모델 유형 선택

Amazon Fraud Detector에서 사용할 수 있는 모델 유형은 다음과 같습니다. 사용 사례에 적합한 모델 유형을 선택합니다.

- 온라인 사기 인사이트

  Online Fraud Insights 모델 유형은 평가 중인 개체에 대한 과거 데이터가 거의 없는 경우 사기를 감지하도록 최적화되어 있습니다. 예를 들어, 새 계정에 온라인으로 등록하는 신규 고객입니다.
- 트랜잭션 사기 인사이트

  Transaction Fraud Insights 모델 유형은 평가 중인 개체에 모델이 예측 정확도를 개선하기 위해 분석할 수 있는 상호 작용 기록이 있을 수 있는 사기 사용 사례(예: 과거 구매 기록이 있는 기존 고객)를 감지하는 데 가장 적합합니다.
- 계정 탈취 인사이트

  Account Takeover Insights 모델 유형은 계정이 피싱 또는 다른 유형의 공격으로 인해 손상되었는지 여부를 감지합니다. 로그인 시 사용되는 브라우저 및 디바이스와 같이 손상된 계정의 로그인 데이터는 계정과 연결된 과거 로그인 데이터와 다릅니다.

온라인 사기 인사이트

온라인 Fraud Insights는 감독을 받는 기계 학습 모델로, 사기 및 합법적인 거래의 과거 예를 사용하여 모델을 교육합니다. 온라인 사기 인사이트 모델은 적은 기록 데이터를 기반으로 사기를 탐지할 수 있습니다. 모델의 입력은 유연하므로 가짜 리뷰, 프로모션 남용, 게스트 체크아웃 사기 등 다양한 사기 위험을 감지하도록 조정할 수 있습니다.

온라인 Fraud Insights 모델은 데이터 보강, 변환 및 사기 분류에 기계 학습 알고리즘 앙상블을 사용합니다. 모델 훈련 프로세스의 일환으로 Online Fraud Insights는 IP 주소 및 BIN 번호와 같은 원시 데이터

요소를 IP 주소의 지리적 위치 또는 신용 카드 발급 은행과 같은 타사 데이터로 보강합니다. 타사 데이터 외에도 Online Fraud Insights는 Amazon 및에서 관찰된 사기 패턴을 고려하는 딥 러닝 알고리즘을 사용합니다 AWS. 이러한 사기 패턴은 그라데이션 트리 부스팅 알고리즘을 사용하여 모델에 입력 기능이 됩니다.

성능을 높이기 위해 Online Fraud Insights는 베이지안 최적화 프로세스를 통해 그라데이션 트리 부스팅 알고리즘의 하이퍼 파라미터를 최적화합니다. 다양한 모델 파라미터(예: 나무 수, 나무 깊이, 잎당 샘플 수)를 사용하여 수십 개의 다양한 모델을 순차적으로 훈련합니다. 또한 소수 사기 집단을 업웨이트하는 등 다양한 최적화 전략을 사용하여 매우 낮은 사기율을 처리합니다.

데이터 소스 선택

온라인 Fraud Insights 모델을 훈련할 때 외부(Amazon Fraud Detector 외부)에 저장되거나 Amazon Fraud Detector 내에 저장된 이벤트 데이터에 대해 모델을 훈련하도록 선택할 수 있습니다. Amazon Fraud Detector가 현재 지원하는 외부 스토리지는 Amazon Simple Storage Service(Amazon S3) 입니다. 가 외부 스토리지를 사용하는 경우 이벤트 데이터 세트를 쉼표로 구분된 값(CSV) 형식으로 Amazon S3 버킷에 업로드해야 합니다. 이러한 데이터 스토리지 옵션은 모델 훈련 구성 내에서 EXTERNAL_EVENTS(외부 스토리지의 경우) 및 INGESTED_EVENTS(내부 스토리지의 경우)라고 합니다. 사용 가능한 데이터 소스와 해당 소스에 데이터를 저장하는 방법에 대한 자세한 내용은 섹션을 참조하세요이벤트 데이터 스토리지.

데이터 준비

이벤트 데이터(Amazon S3 또는 Amazon Fraud Detector)를 저장하기로 선택한 위치에 관계없이 온라인 Fraud Insights 모델 유형에 대한 요구 사항은 동일합니다.

데이터 세트에는 열 헤더 EVENT_LABEL이 포함되어야 합니다. 이 변수는 이벤트를 사기 또는 합법적으로 분류합니다. CSV 파일(외부 스토리지)을 사용할 때는 파일의 각 이벤트에 EVENT_LABEL을 포함해야 합니다. 내부 스토리지의 경우 EVENT_LABEL 필드는 선택 사항이지만 모든 이벤트에 레이블을 지정하여 훈련 데이터 세트에 포함해야 합니다. 모델 훈련을 구성할 때 레이블이 지정되지 않은 이벤트를 무시할지, 레이블이 지정되지 않은 이벤트에 대해 합법적인 레이블을 맡을지, 레이블이 지정되지 않은 모든 이벤트에 대해 사기 레이블을 맡을지 선택할 수 있습니다.

데이터 선택

온라인 사기 인사이트 모델 훈련을 위한 데이터 선택에 대한 자세한 내용은 이벤트 데이터 수집을 참조하세요.

온라인 사기 인사이트 훈련은 EVENT_TIMESTAMP를 기반으로 과거 데이터를 샘플링하고 분할합니다. 데이터를 수동으로 샘플링할 필요가 없으며 그렇게 하면 모델 결과에 부정적인 영향을 미칠 수 있습니다.

이벤트 변수

Online Fraud Insights 모델에는 필요한 이벤트 메타데이터를 제외하고 모델 훈련을 위한 데이터 검증을 통과하고 모델당 최대 100개의 변수를 허용하는 최소 2개의 변수가 필요합니다. 일반적으로 변수를 많이 제공할수록 모델이 사기와 합법적인 이벤트를 더 잘 구별할 수 있습니다. 온라인 사기 인사이트 모델은 사용자 지정 변수를 포함한 수십 개의 변수를 지원할 수 있지만, 일반적으로 이러한 변수가 평가 중인 개체를 식별하는 데 가장 효과적이므로 IP 주소와 이메일 주소를 포함하는 것이 좋습니다.

데이터 검증

교육 프로세스의 일환으로 Online Fraud Insights는 데이터 세트에서 모델 교육에 영향을 미칠 수 있는 데이터 품질 문제를 검증합니다. 데이터를 검증한 후 Amazon Fraud Detector는 가능한 최상의 모델을 구축하기 위해 적절한 조치를 취합니다. 여기에는 잠재적 데이터 품질 문제에 대한 경고 실행, 데이터 품질 문제가 있는 변수 자동 제거 또는 오류 실행 및 모델 훈련 프로세스 중지가 포함됩니다. 자세한 내용은 데이터 세트 검증을 참조하세요.

트랜잭션 사기 인사이트

Transaction Fraud Insights 모델 유형은 온라인 또는 card-not-present 거래 사기를 탐지하도록 설계되었습니다. Transaction Fraud Insights는 지도형 기계 학습 모델로, 사기 및 합법적인 거래의 과거 예를 사용하여 모델을 훈련합니다.

Transaction Fraud Insights 모델은 데이터 보강, 변환 및 사기 분류에 기계 학습 알고리즘 앙상블을 사용합니다. 특성 엔지니어링 엔진을 활용하여 개체 수준 및 이벤트 수준 집계를 생성합니다. 모델 훈련 프로세스의 일환으로 Transaction Fraud Insights는 IP 주소 및 BIN 번호와 같은 원시 데이터 요소를 IP 주소의 지리적 위치 또는 신용 카드 발급 은행과 같은 타사 데이터로 보강합니다. 서드 파티 데이터 외에도 Transaction Fraud Insights는 Amazon에서 관찰된 사기 패턴을 고려하는 딥 러닝 알고리즘을 사용하며 AWS , 이러한 사기 패턴은 그라데이션 트리 부스팅 알고리즘을 사용하여 모델에 입력 기능이 됩니다.

성능을 높이기 위해 Transaction Fraud Insights는 베이지안 최적화 프로세스를 통해 그라데이션 트리 부스팅 알고리즘의 하이퍼 파라미터를 최적화하고 다양한 모델 파라미터(예: 나무 수, 나무 깊이, 잎당 샘플 수)와 매우 낮은 사기율을 처리하기 위해 소수 사기 집단을 업웨이트하는 등 다양한 최적화 전략을 사용하여 수십 가지 모델을 순차적으로 훈련합니다.

모델 훈련 프로세스의 일환으로 트랜잭션 사기 모델의 특성 엔지니어링 엔진은 훈련 데이터 세트 내의 각 고유 개체에 대한 값을 계산하여 사기 예측을 개선합니다. 예를 들어 훈련 프로세스 중에 Amazon Fraud Detector는 개체가 마지막으로 구매한 시간을 계산 및 저장하고 `GetEventPrediction` 또는 `SendEvent` API를 호출할 때마다이 값을 동적으로 업데이트합니다. 사기 예측 중에 이벤트 변수는 다른 개체 및 이벤트 메타데이터와 결합하여 트랜잭션이 사기인지 여부를 예측합니다.

데이터 소스 선택

Transaction Fraud Insights 모델은 Amazon Fraud Detector(INGESTED_EVENTS)를 사용하여 내부적으로 저장된 데이터 세트에 대해서만 훈련됩니다. 이렇게 하면 Amazon Fraud Detector가 평가 중인 개체에 대해 계산된 값을 지속적으로 업데이트할 수 있습니다. 사용 가능한 데이터 소스에 대한 자세한 내용은 섹션을 참조하세요. 이벤트 데이터 스토리지

데이터 준비

Transaction Fraud Insights 모델을 훈련하기 전에 이벤트 데이터 세트 준비에 언급된 대로 데이터 파일에 모든 헤더가 포함되어 있는지 확인합니다. Transaction Fraud Insights 모델은 수신된 새 엔터티를 데이터 세트의 사기 및 합법적인 엔터티의 예와 비교하므로 각 엔터티에 대해 많은 예제를 제공하는 것이 좋습니다.

Amazon Fraud Detector는 저장된 이벤트 데이터 세트를 훈련을 위한 올바른 형식으로 자동 변환합니다. 모델이 훈련을 완료한 후 성능 지표를 검토하고 훈련 데이터 세트에 개체를 추가해야 하는지 여부를 결정할 수 있습니다.

데이터 선택

기본적으로 Transaction Fraud Insights는 선택한 이벤트 유형에 대해 저장된 전체 데이터 세트를 훈련합니다. 선택적으로 시간 범위를 설정하여 모델 훈련에 사용되는 이벤트를 줄일 수 있습니다. 시간 범위를 설정할 때 모델을 훈련하는 데 사용되는 레코드가 충분히 성숙할 시간을 가졌는지 확인합니다. 즉, 합법적 및 사기 레코드가 올바르게 식별되도록 충분한 시간이 지났습니다. 예를 들어 차지백 사기의 경우 사기 이벤트를 올바르게 식별하는 데 60일 이상이 걸리는 경우가 많습니다. 최상의 모델 성능을 얻으려면 훈련 데이터 세트의 모든 레코드가 성숙해야 합니다.

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

Amazon Fraud Detector는 모델을 성공적으로 훈련시키기에 이벤트가 충분하지 않은 시간 범위를 선택하면 모델 훈련 중에 검증 오류를 반환합니다. 저장된 데이터 세트의 경우 EVENT_LABEL 필드는 선택 사항이지만 훈련 데이터 세트에 포함하려면 이벤트에 레이블을 지정해야 합니다. 모델 훈련을 구성

할 때 레이블이 지정되지 않은 이벤트를 무시할지, 레이블이 지정되지 않은 이벤트에 대한 합법적인 레이블을 맡을지, 레이블이 지정되지 않은 이벤트에 대한 사기성 레이블을 맡을지 선택할 수 있습니다.

이벤트 변수

모델을 훈련하는 데 사용되는 이벤트 유형은 필수 이벤트 메타데이터를 제외하고 데이터 검증을 통과하고 최대 100개의 변수를 포함할 수 있는 최소 2개의 변수를 포함해야 합니다. 일반적으로 변수를 많이 제공할수록 모델이 사기와 합법적인 이벤트를 더 잘 구별할 수 있습니다. Transaction Fraud Insight 모델은 사용자 지정 변수를 포함한 수십 개의 변수를 지원할 수 있지만 IP 주소, 이메일 주소, 결제 수단 유형, 주문 가격 및 카드 BIN을 포함하는 것이 좋습니다.

데이터 검증

훈련 프로세스의 일환으로 Transaction Fraud Insights는 훈련 데이터 세트에서 모델 훈련에 영향을 미칠 수 있는 데이터 품질 문제를 검증합니다. 데이터를 검증한 후 Amazon Fraud Detector는 가능한 최상의 모델을 구축하기 위해 적절한 조치를 취합니다. 여기에는 잠재적 데이터 품질 문제에 대한 경고 실행, 데이터 품질 문제가 있는 변수 자동 제거 또는 오류 실행 및 모델 훈련 프로세스 중지가 포함됩니다. 자세한 내용은 데이터 세트 검증을 참조하세요.

Amazon Fraud Detector는 고유한 개체 수가 1,500개 미만인 경우 훈련 데이터의 품질에 영향을 미칠 수 있으므로 경고를 발행하지만 모델을 계속 훈련합니다. 경고가 표시되면 성능 지표를 검토합니다.

계정 탈취 인사이트

계정 탈취 인사이트(ATI) 모델 유형은 계정이 악의적인 탈취, 피싱 또는 도난당한 자격 증명을 통해 침해되었는지 감지하여 사기 온라인 활동을 식별합니다. Account Takeover Insights는 온라인 비즈니스의 로그인 이벤트를 사용하여 모델을 훈련하는 기계 학습 모델입니다.

실시간 로그인 흐름 내에 훈련된 Account Takeover Insights 모델을 포함시켜 계정이 손상되었는지 감지할 수 있습니다. 모델은 다양한 인증 및 로그인 유형을 평가합니다. 여기에는 웹 애플리케이션 로그인, API 기반 인증 및 single-sign-on)가 포함됩니다. Account Takeover Insights 모델을 사용하려면 유효한 로그인 자격 증명이 제공된 후 GetEventPrediction API를 호출합니다. API는 계정이 손상될 위험을 정량화하는 점수를 생성합니다. Amazon Fraud Detector는 사용자가 정의한 점수와 규칙을 사용하여 로그인 이벤트에 대한 하나 이상의 결과를 반환합니다. 결과는 구성한 결과입니다. 받은 결과에 따라 각 로그인에 대해 적절한 조치를 취할 수 있습니다. 즉, 로그인에 대해 제공된 자격 증명을 승인하거나 챌린지할 수 있습니다. 예를 들어 추가 확인으로 계정 PIN을 요청하여 자격 증명에 이의를 제기할 수 있습니다.

Account Takeover Insights 모델을 사용하여 계정 로그인을 비동기적으로 평가하고 고위험 계정에 대해 조치를 취할 수도 있습니다. 예를 들어, 고위험 계정을 조사 대기열에 추가하여 인적 검토자가 계정 일시 중지와 같은 추가 조치를 취해야 하는지 여부를 결정할 수 있습니다.

Account Takeover Insights 모델은 비즈니스의 과거 로그인 이벤트가 포함된 데이터 세트를 사용하여 훈련됩니다. 이 데이터를 제공합니다. 필요에 따라 계정에 합법적 또는 사기성 레이블을 지정할 수 있습니다. 그러나 모델을 훈련하는 데는 필요하지 않습니다. Account Takeover Insights 모델은 계정의 성공적인 로그인 기록을 기반으로 이상을 감지합니다. 또한 악의적인 계정 탈취 이벤트의 위험 증가를 암시하는 사용자 동작의 이상을 탐지하는 방법을 알아봅니다. 예를 들어 일반적으로 동일한 디바이스 및 IP 주소 집합에서 로그인하는 사용자입니다. 사기범은 일반적으로 다른 디바이스와 지리적 위치에서 로그인합니다. 이 기법은 활동 이상 위험 점수를 생성하며, 이는 일반적으로 악성 계정 탈취의 주요 특성입니다.

Account Takeover Insights 모델을 훈련하기 전에 Amazon Fraud Detector는 기계 학습 기법의 조합을 사용하여 데이터 보강, 데이터 집계 및 데이터 변환을 수행합니다. 그런 다음 훈련 프로세스 중에 Amazon Fraud Detector는 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 동작을 설명하는 집계된 변수를 지속적으로 계산합니다. 사용자 동작의 예로는 사용자가 특정 IP 주소에서 로그인한 횟수가 있습니다. Amazon Fraud Detector는 이러한 추가 보강 및 집계를 사용하여 로그인 이벤트의 작은 입력 세트에서 강력한 모델 성능을 생성할 수 있습니다.

Account Takeover Insights 모델은 악의적인 행위자가 인간이든 로봇이든 상관없이 악의적인 행위자가 합법적인 계정에 액세스하는 인스턴스를 감지합니다. 모델은 계정 손상의 상대적 위험을 나타내는 단일 점수를 생성합니다. 손상되었을 수 있는 계정은 고위험 계정으로 플래그가 지정됩니다. 두 가지 방법 중 하나로 고위험 계정을 처리할 수 있습니다. 어느 쪽이든 추가 자격 증명 확인을 적용할 수 있습니다. 또는 수동 조사를 위해 계정을 대기열로 보낼 수 있습니다.

데이터 소스 선택

Account Takeover Insights 모델은 Amazon Fraud Detector에 내부적으로 저장된 데이터 세트에 대해 훈련됩니다. Amazon Fraud Detector로 로그인 이벤트 데이터를 저장하려면 사용자의 로그인 이벤트가 포함된 CSV 파일을 생성합니다. 각 이벤트에 대해 이벤트 타임스탬프, 사용자 ID, IP 주소, 사용자 에이전트, 로그인 데이터의 유효 여부와 같은 로그인 데이터를 포함합니다. CSV 파일을 생성한 후 먼저 Amazon Fraud Detector에 파일을 업로드한 다음 가져오기 기능을 사용하여 데이터를 저장합니다. 그런 다음 저장된 데이터를 사용하여 모델을 훈련할 수 있습니다. Amazon Fraud Detector를 사용하여 이벤트 데이터 세트를 저장하는 방법에 대한 자세한 내용은 섹션을 참조하세요. Amazon Fraud Detector를 사용하여 이벤트 데이터를 내부적으로 저장

데이터 준비

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

Accounts Takeover Insights 모델의 경우 CSV 파일의 헤더 라인에 다음 이벤트 메타데이터 및 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 - 판매자 또는 고객과 같이 로그인 이벤트를 수행하는 개체입니다.
- ENTITY_ID - 로그인 이벤트를 수행하는 개체의 식별자입니다.
- EVENT_TIMESTAMP - 로그인 이벤트가 발생한 시점의 타임스탬프입니다. 타임스탬프는 UTC의 ISO 8601 표준이어야 합니다.
- EVENT_LABEL(권장) - 이벤트를 사기 또는 합법적으로 분류하는 레이블입니다. "사기", "법률", "1" 또는 "0"과 같은 모든 레이블을 사용할 수 있습니다.

> **Note**
>
> - 이벤트 메타데이터는 대문자여야 합니다. 대/소문자를 구분합니다.
> - 로그인 이벤트에는 레이블이 필요하지 않습니다. 그러나 EVENT_LABEL 메타데이터를 포함하고 로그인 이벤트에 대한 레이블을 제공하는 것이 좋습니다. 레이블이 불완전하거나 산발적인 경우에도 괜찮습니다. 레이블을 제공하면 Amazon Fraud Detector는 레이블을 사용하여 계정 탈취 검색 속도를 자동으로 계산하고 모델 성능 차트 및 테이블에 표시합니다.

이벤트 변수

Accounts Takeover Insights 모델의 경우 제공해야 하는 필수(필수) 변수와 선택적 변수가 모두 있습니다. 변수를 생성할 때 변수를 올바른 변수 유형에 할당해야 합니다. 모델 훈련 프로세스의 일환으로

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

> **Note**
>
> 이벤트 변수 이름은 소문자여야 합니다. 대/소문자를 구분합니다.

필수 변수

Accounts Takeover Insights 모델을 훈련하려면 다음 변수가 필요합니다.

범주	변수 유형	설명
IP 주소	IP_ADDRESS	로그인 이벤트에 사용되는 IP 주소
브라우저 및 디바이스	USERAGENT	로그인 이벤트에 사용되는 브라우저, 디바이스 및 OS
유효한 자격 증명	VALIDCRED	로그인에 사용된 자격 증명이 유효한지 여부를 나타냅니다.

선택적 변수

다음 변수는 Accounts Takeover Insights 모델을 훈련하는 데 선택 사항입니다.

범주	유형	설명
브라우저 및 디바이스	지문	브라우저 또는 디바이스 지문의 고유 식별자
세션 ID	SESSION_ID	인증 세션의 식별자
레이블	EVENT_LABEL	이벤트를 사기 또는 합법적으로 분류하는 레이블입니다. "사기", "법률", "1" 또는 "0"과 같은 모든 레이블을 사용할 수 있습니다.

범주	유형	설명
Timestamp	LABEL_TIMESTAMP	레이블이 마지막으로 업데이트된 시점의 타임스탬프입니다. EVENT_LABEL이 제공된 경우 필요합니다.

> **Note**
>
> - 두 필수 변수의 선택적 변수에 변수 이름을 제공할 수 있습니다. 각 필수 및 선택적 변수를 올바른 변수 유형에 할당하는 것이 중요합니다.
> - 추가 변수를 제공할 수 있습니다. 그러나 Amazon Fraud Detector에는 Accounts Takeover Insights 모델을 훈련하기 위한 이러한 변수가 포함되지 않습니다.

데이터 선택

데이터 수집은 Account Takeover Insights 모델을 생성하는 데 중요한 단계입니다. 로그인 데이터 수집을 시작할 때 다음 요구 사항 및 권장 사항을 고려하세요.

필수

- 각각 2개 이상의 관련 로그인 이벤트가 있는 사용자 계정 예제를 1,500개 이상 제공합니다.
- 데이터 세트는 최소 30일의 로그인 이벤트를 포함해야 합니다. 나중에 모델을 훈련하는 데 사용할 이벤트의 특정 시간 범위를 지정할 수 있습니다.

권장

- 데이터 세트에는 실패한 로그인 이벤트의 예가 포함되어 있습니다. 이러한 실패한 로그인에 대해 선택적으로 “사기” 또는 “합법적”으로 레이블을 지정할 수 있습니다.
- 로그인 이벤트가 6개월 이상 지속되고 100K개의 엔터티를 포함하는 기록 데이터를 준비합니다.

최소 요구 사항을 이미 충족하는 데이터 세트가 없는 경우 SendEvent API 작업을 호출하여 이벤트 데이터를 Amazon Fraud Detector로 스트리밍하는 것이 좋습니다.

데이터 검증

Account Takeover Insights 모델을 생성하기 전에 Amazon Fraud Detector는 모델 훈련을 위해 데이터 세트에 포함된 메타데이터 및 변수가 크기 및 형식 요구 사항을 충족하는지 확인합니다. 자세한 내용은 데이터 세트 검증 단원을 참조하십시오. 또한 다른 요구 사항을 확인합니다. 데이터 세트가 검증을 통과하지 못하면 모델이 생성되지 않습니다. 모델을 성공적으로 생성하려면 다시 훈련하기 전에 검증을 통과하지 못한 데이터를 수정해야 합니다.

일반적인 데이터 세트 오류

Account Takeover Insights 모델 훈련을 위해 데이터 세트를 검증할 때 Amazon Fraud Detector는 이러한 문제와 기타 문제를 스캔하고 하나 이상의 문제가 발생하면 오류를 발생시킵니다.

- CSV 파일은 UTF-8 형식이 아닙니다.
- CSV 파일 헤더에는 EVENT_ID, ENTITY_ID또는 메타데이터 중 하나 이상이 포함되어 있지 않습니다EVENT_TIMESTAMP.
- CSV 파일 헤더에는 IP_ADDRESS, USERAGENT또는 변수 유형 중 하나 이상의 변수가 포함되어 있지 않습니다VALIDCRED.
- 동일한 변수 유형과 연결된 변수가 두 개 이상 있습니다.
- 에서 값의 0.1% 이상이 지원되는 날짜 및 타임스탬프 형식 이외의 null 또는 값을 EVENT_TIMESTAMP 포함합니다.
- 첫 번째 이벤트와 마지막 이벤트 사이의 일수는 30일 미만입니다.
- IP_ADDRESS 변수 유형 변수의 10% 이상이 유효하지 않거나 null입니다.
- USERAGENT 변수 유형 변수의 50% 이상이 null을 포함합니다.
- VALIDCRED 변수 유형의 모든 변수는 로 설정됩니다false.

모델 빌드

Amazon Fraud Detector 모델은 특정 이벤트 유형에 대한 사기를 탐지하는 방법을 학습합니다. Amazon Fraud Detector에서는 먼저 모델 버전을 위한 컨테이너 역할을 하는 모델을 생성합니다. 모델을 훈련할 때마다 새 버전이 생성됩니다. AWS 콘솔을 사용하여 모델을 생성하고 훈련하는 방법에 대한 자세한 내용은 섹션을 참조하세요3단계: 모델 생성.

각 모델에는 해당 모델 점수 변수가 있습니다. Amazon Fraud Detector는 모델을 생성할 때 사용자를 대신하여이 변수를 생성합니다. 규칙 표현식에서이 변수를 사용하여 사기 평가 중에 모델 점수를 해석할 수 있습니다.

를 사용하여 모델 훈련 및 배포 AWS SDK for Python (Boto3)

모델 버전은 CreateModel 및 CreateModelVersion 작업을 호출하여 생성됩니다.는 모델 버전을 위한 컨테이너 역할을 하는 모델을 CreateModel 시작합니다.는 훈련 프로세스를 CreateModelVersion 시작하여 모델의 특정 버전을 생성합니다. CreateModelVersion을 호출할 때마다 새 솔루션 버전이 생성됩니다.

다음 예제에서는 CreateModel API에 대한 샘플 요청을 보여줍니다. 이 예제에서는 Online Fraud Insights 모델 유형을 생성하고 이벤트 유형을 생성했다고 가정합니다sample_registration. 이벤트 유형 생성에 대한 자세한 내용은 섹션을 참조하세요이벤트 유형 생성.

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

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

CreateModelVersion API를 사용하여 첫 번째 버전을 훈련합니다. TrainingDataSource 및에 대해 훈련 데이터 세트의 소스 및 Amazon S3 위치를 ExternalEventsDetail 지정합니다. 의 경우 Amazon Fraud Detector가 훈련 데이터를 해석하는 방법, 특히 포함할 이벤트 변수와 이벤트 레이블을 분류하는 방법을 TrainingDataSchema 지정합니다. 기본적으로 Amazon Fraud Detector는 레이블이 지정되지 않은 이벤트를 무시합니다. 이 예제 코드는 AUTO에 unlabeledEventsTreatment를 사용하여 Amazon Fraud Detector가 레이블이 지정되지 않은 이벤트를 사용하는 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 인 새 모델 버전이 생성됩니다TRAINING_IN_PROGRESS. 훈련 중 언제든지를 호출UpdateModelVersionStatus하고 상태를 로 업데이트하여 훈련을 취소할 수 있습니다TRAINING_CANCELLED. 훈련이 완료되면 모델 버전 상태가 로 업데이트됩니다TRAINING_COMPLETE. Amazon Fraud Detector 콘솔을 사용하거나를 호출하여 모델 성능을 검토할 수 있습니다DescribeModelVersions. 모델 점수 및 성능을 해석하는 방법에 대한 자세한 내용은 모델 점수 및 섹션을 참조하세요모델 성능 지표.

모델 성능을 검토한 후 모델을 활성화하여 감지기가 실시간 사기 예측에서 사용할 수 있도록 합니다. Amazon Fraud Detector는 오토 스케일링이 켜져 있는 상태에서 중복성을 위해 여러 가용 영역에 모델을 배포하여 모델이 실행 중인 사기 예측 수에 따라 확장되도록 합니다. 모델을 활성화하려면 UpdateModelVersionStatus API를 호출하고 상태를 로 업데이트합니다ACTIVE.

```
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 유형에 따라 모델 점수를 다르게 생성합니다.

계정 탈취 인사이트(ATI) 모델의 경우 Amazon Fraud Detector는 집계된 값(원시 변수 세트를 결합하여 계산한 값)만 사용하여 모델 점수를 생성합니다. 새 개체의 첫 번째 이벤트에 대해 -1점이 생성되어 알 수 없는 위험을 나타냅니다. 이는 새 개체의 경우 집계 계산에 사용되는 값이 0 또는 null이기 때문입니다. Account Takeeover Insights(ATI) 모델은 동일한 개체와 기존 개체에 대한 모든 후속 이벤트에 대해 0~1000 사이의 모델 점수를 생성합니다. 여기서 0은 사기 위험이 낮음을 나타내고 1000은 사기 위

험이 높음을 나타냅니다. ATI 모델의 경우 모델 점수는 챌린지 비율(CR)과 직접 관련이 있습니다. 예를 들어 500점은 예상 5% 챌린지 비율에 해당하는 반면 900점은 예상 0.1% 챌린지 비율에 해당합니다.

온라인 사기 인사이트(OFI) 및 트랜잭션 사기 인사이트(TFI) 모델의 경우 Amazon Fraud Detector는 집계된 값(원시 변수 세트를 결합하여 계산한 값)과 원시 값(변수에 제공된 값)을 모두 사용하여 모델 점수를 생성합니다. 모델 점수는 0에서 1000 사이일 수 있습니다. 여기서 0은 사기 위험이 낮음을 나타내고 1000은 사기 위험이 높음을 나타냅니다. OFI 및 TFI 모델의 경우 모델 점수는 거짓 긍정률(FPR)과 직접 관련이 있습니다. 예를 들어 600점은 10%의 거짓 긍정률 추정치에 해당하고 900점은 2%의 거짓 긍정률 추정치에 해당합니다. 다음 표에는 특정 모델 점수가 예상 거짓 긍정 비율과 어떻게 상관관계가 있는지에 대한 세부 정보가 나와 있습니다.

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

모델 성능 지표

모델 훈련이 완료되면 Amazon Fraud Detector는 모델 훈련에 사용되지 않은 데이터의 15%를 사용하여 모델 성능을 검증합니다. 훈련된 Amazon Fraud Detector 모델은 검증 성능 지표와 유사한 실제 사기 탐지 성능을 가질 것으로 예상할 수 있습니다.

기업으로서 더 많은 사기를 탐지하고 합법적인 고객에게 마찰을 더하는 것 사이의 균형을 맞춰야 합니다. Amazon Fraud Detector는 올바른 밸런스를 선택하는 데 도움이 되도록 모델 성능을 평가하는 다음과 같은 도구를 제공합니다.

- 점수 분포 차트 - 모델 점수 분포의 히스토그램은 100,000개의 이벤트로 구성된 예제 모집단을 가정합니다. 왼쪽 Y축은 합법적인 이벤트를 나타내고 오른쪽 Y축은 사기 이벤트를 나타냅니다. 차트 영

역을 클릭하여 특정 모델 임계값을 선택할 수 있습니다. 그러면 혼동 행렬 및 ROC 차트의 해당 뷰가 업데이트됩니다.

- 혼동 행렬 - 모델 예측과 실제 결과를 비교하여 지정된 점수 임계값에 대한 모델 정확도를 요약합니다. Amazon Fraud Detector는 100,000개의 이벤트로 구성된 예제 모집단을 가정합니다. 사기 및 합법적인 이벤트의 배포는 비즈니스의 사기율을 시뮬레이션합니다.
  - 참 긍정 - 모델은 사기를 예측하고 이벤트는 실제로 사기입니다.
  - 거짓 긍정 - 모델은 사기를 예측하지만 이벤트는 실제로 합법적입니다.
  - 참 부정 - 모델은 합법적이고 이벤트는 실제로 합법적임을 예측합니다.
  - 거짓 부정 - 모델은 합법적이라고 예측하지만 실제로는 이벤트가 사기입니다.
  - 참 긍정률(TPR) - 모델이 탐지한 총 사기의 비율입니다. 캡처 속도라고도 합니다.
  - FPR(False positive rate) - 사기로 잘못 예측된 총 합법적인 이벤트의 비율입니다.
- 수신기 연산자 곡선(ROC) - 가능한 모든 모델 점수 임계값에 대해 거짓 긍정 비율의 함수로 실제 긍정 비율을 표시합니다. 고급 지표를 선택하여이 차트를 봅니다.
- 곡선하 면적(AUC) - 가능한 모든 모델 점수 임계값에서 TPR 및 FPR을 요약합니다. 예측력이 없는 모델은 AUC가 0.5인 반면, 완벽한 모델은 점수가 1.0입니다.
- 불확실성 범위 - 모델에서 예상되는 AUC 범위를 보여줍니다. 범위가 클수록(AUC > 0.1의 상한 및 하한 차이) 모델 불확실성이 높아집니다. 불확실성 범위가 크면(>0.1) 레이블이 더 많은 이벤트를 제공하고 모델을 재학습하는 것이 좋습니다.

모델 성능 지표를 사용하려면

1. 점수 분포 차트부터 시작하여 사기 및 합법적인 이벤트에 대한 모델 점수 분포를 검토합니다. 이상적으로는 사기와 합법적인 이벤트가 명확하게 구분됩니다. 이는 모델이 사기성 이벤트와 합법적인 이벤트를 정확하게 식별할 수 있음을 나타냅니다. 차트 영역을 클릭하여 모델 임계값을 선택합니다. 모델 점수 임계값을 조정하면 실제 긍정 및 거짓 긍정 비율에 어떤 영향을 미치는지 확인할 수 있습니다.

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

2. 혼동 행렬을 검토합니다. 선택한 모델 점수 임계값에 따라 100,000개의 이벤트 샘플을 기반으로 시뮬레이션된 영향을 볼 수 있습니다. 사기 및 합법적인 이벤트의 배포는 비즈니스의 사기율을 시

뮬레이션합니다. 이 정보를 사용하여 참 긍정 비율과 거짓 긍정 비율 간의 적절한 균형을 찾을 수 있습니다.

3. 자세한 내용을 보려면 고급 지표를 선택합니다. ROC 차트를 사용하여 모델 점수 임계값에 대한 참 긍정 비율과 거짓 긍정 비율 간의 관계를 이해합니다. ROC 곡선은 참 긍정 비율과 거짓 긍정 비율 간의 균형을 미세 조정하는 데 도움이 될 수 있습니다.

> **Note**
>
> 테이블을 선택하여 테이블 형식의 지표를 검토할 수도 있습니다.
> 테이블 보기에는 지표 정밀도도 표시됩니다. 정밀도는 사기로 예측된 모든 이벤트와 비교하여 사기로 정확하게 예측된 사기 이벤트의 백분율입니다.

4. 성능 지표를 사용하여 목표 및 사기 탐지 사용 사례에 따라 비즈니스에 최적의 모델 임계값을 결정합니다. 예를 들어 모델을 사용하여 새 계정 등록을 고위험, 중간 또는 저위험으로 분류하려는 경우 다음과 같이 세 가지 규칙 조건의 초안을 작성할 수 있도록 두 개의 임계값 점수를 식별해야 합니다.

   - 점수 > X는 위험이 높습니다.
   - 점수 < X이지만 > Y는 중간 위험입니다.
   - 점수가 < Y이면 위험이 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 값이 제공됩니다. 값이 가장 높은 모델 변수는 해당 모델 버전의 데이터 세트에 있는 다른 모델 변수보다 모델에 더 중요하며 기본적으로 상단에 나열됩니다. 마찬가지로 값이 가장 낮은 모델 변수는 기본적으로 하단에 나열되며 다른 모델 변수에 비해 가장 중요하지 않습니다. 모델 변수 중요도 값을 사용하면 모델의 성능을 좌우하는 입력에 대한 인사이트를 얻을 수 있습니다.

Amazon Fraud Detector 콘솔에서 또는 DescribeModelVersion API를 사용하여 훈련된 모델 버전의 모델 변수 중요도 값을 볼 수 있습니다.

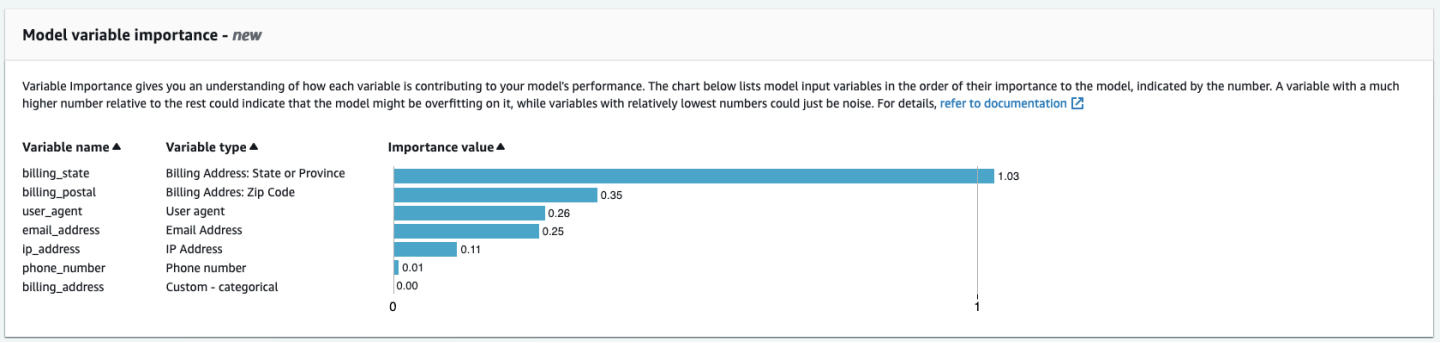
모델 변수 중요도는 모델 버전을 훈련하는 데 사용되는 각 변수에 대해 다음과 같은 값 집합을 제공합니다.

- 변수 유형: 변수 유형(예: IP 주소 또는 이메일). 자세한 내용은 변수 유형 단원을 참조하십시오. 계정 탈취 인사이트(ATI) 모델의 경우 Amazon Fraud Detector는 원시 변수 유형과 집계 변수 유형 모두에

대해 가변 중요도 값을 제공합니다. 원시 변수 유형은 사용자가 제공하는 변수에 할당됩니다. 집계 변수 유형은 Amazon Fraud Detector가 집계된 중요도 값을 계산하기 위해 결합한 원시 변수 집합에 할당됩니다.

- 변수 이름: 모델 버전을 훈련하는 데 사용된 이벤트 변수의 이름입니다(예: , `ip_addressemail_address`, `are_creadentials_valid`). 집계된 변수 유형의 경우 집계된 변수 중요도 값을 계산하는 데 사용된 모든 변수의 이름이 나열됩니다.
- 변수 중요도 값: 모델 성능에 대한 원시 또는 집계된 변수의 상대적 중요도를 나타내는 숫자입니다. 일반적인 범위: 0~10

Amazon Fraud Detector 콘솔에서 모델 변수 중요도 값은 온라인 사기 인사이트(OFI) 또는 트랜잭션 사기 인사이트(TFI) 모델에 대해 다음과 같이 표시됩니다. Account Takeover Insight(ATI) 모델은 원시 변수의 중요도 값 외에도 집계된 변수 중요도 값을 제공합니다. 시각적 차트를 사용하면 세로 점선으로 변수 간의 상대적 중요도를 쉽게 확인할 수 있어 가장 높은 순위의 변수의 중요도 값을 참조할 수 있습니다.



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

> **Important**
>
> 2021년 7월 9일 이전에 생성된 모델 버전에는 가변 중요도 값이 없습니다. 모델 변수 중요도 값을 생성하려면 모델의 새 버전을 훈련해야 합니다.

모델 변수 중요도 값 사용

모델 변수 중요도 값을 사용하여 모델의 성능을 높이거나 낮추는 요인과 가장 많이 기여하는 변수에 대한 인사이트를 얻을 수 있습니다. 그런 다음 모델을 조정하여 전반적인 성능을 개선합니다.

보다 구체적으로, 모델 성능을 개선하려면 도메인 지식을 기준으로 가변 중요도 값을 검사하고 훈련 데이터의 문제를 디버깅합니다. 예를 들어, 계정 ID가 모델의 입력으로 사용되었고 상단에 나열된 경우 가변 중요도 값을 살펴보세요. 변수 중요도 값이 나머지 값보다 훨씬 높으면 모델이 특정 사기 패턴에 과적합할 수 있습니다(예: 모든 사기 이벤트가 동일한 계정 ID에서 발생함). 그러나 변수가 사기 레이블에 의존하는 경우 레이블 유출이 발생할 수도 있습니다. 도메인 지식을 기반으로 한 분석 결과에 따라 변수를 제거하고 더 다양한 데이터 세트로 훈련하거나 모델을 그대로 유지할 수 있습니다.

마찬가지로 마지막으로 순위가 매겨진 변수를 살펴봅니다. 변수 중요도 값이 나머지 값보다 훨씬 낮은 경우이 모델 변수는 모델 훈련에 중요하지 않을 수 있습니다. 변수를 제거하여 더 간단한 모델 버전을 훈련하는 것을 고려할 수 있습니다. 모델에 두 개의 변수와 같은 변수가 거의 없는 경우 Amazon Fraud Detector는 여전히 변수 중요도 값을 제공하고 변수의 순위를 매깁니다. 그러나이 경우 인사이트는 제한됩니다.

> **Important**
>
> 1. 모델 변수 중요도 차트에 누락된 변수가 있는 경우 다음 이유 중 하나 때문일 수 있습니다. 데이터 세트의 변수를 수정하고 모델을 재학습하는 것이 좋습니다.
>   - 훈련 데이터 세트의 변수에 대한 고유 값 수가 100개 미만입니다.
>   - 변수 값이 0.9보다 크면 훈련 데이터 세트에서 누락됩니다.
> 2. 모델의 입력 변수를 조정하려면 매번 새 모델 버전을 훈련해야 합니다.

모델 변수 중요도 값 평가

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

- 가변 중요도 값은 항상 도메인 지식과 함께 평가해야 합니다.
- 모델 버전 내의 다른 변수의 변수 중요도 값과 비교하여 변수의 변수 중요도 값을 검사합니다. 단일 변수에 대해 변수 중요도 값을 독립적으로 고려하지 마십시오.
- 동일한 모델 버전 내에서 변수의 변수 중요도 값을 비교합니다. 모델 버전에서 변수의 변수 중요도 값이 다른 모델 버전에서 동일한 변수의 값과 다를 수 있으므로 모델 버전 간에 동일한 변수의 변수 중요도 값을 비교하지 마십시오. 동일한 변수와 데이터 세트를 사용하여 다른 모델 버전을 훈련하는 경우 동일한 변수 중요도 값을 반드시 생성하지는 않습니다.

모델 변수 중요도 순위 보기

모델 훈련이 완료되면 Amazon Fraud Detector 콘솔에서 또는 DescribeModelVersion API를 사용하여 훈련된 모델 버전의 모델 변수 중요도 순위를 볼 수 있습니다.

콘솔을 사용하여 모델 변수 중요도 순위를 보려면

1. AWS 콘솔을 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 확인합니다.
5. 아래로 스크롤하여 모델 변수 중요도 창을 봅니다.

모델 변수 중요도 값이 계산되는 방법 이해

각 모델 버전 훈련을 완료하면 Amazon Fraud Detector는 모델 변수 중요도 값과 모델의 성능 지표를 자동으로 생성합니다. 이를 위해 Amazon Fraud Detector는 SHapley Additive exPlanations(SHAP)를 사용합니다. SHAP는 기본적으로 모든 모델 변수의 가능한 모든 조합을 고려한 후 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. 예측에 각 모델 변수의 기여도를 할당하기 위해 SHAP는 가능한 모든 변수 조합 간의 모델 출력 차이를 고려합니다. 특정 변수 세트를 포함하거나 제거하여 모델 출력을 생성할 수 있는 모든 가능성을 포함함으로써 SHAP는 각 모델 변수의 중요도에 정확하게 액세스할 수 있습니다. 이는 모델 변수가 서로 높은 상관관계를 가질 때 특히 중요합니다.

ML 모델은 대부분의 경우 변수를 제거할 수 없습니다. 대신 모델에서 제거되거나 누락된 변수를 하나 이상의 기준(예: 사기가 아닌 이벤트)의 해당 변수 값으로 바꿀 수 있습니다. 적절한 기준 인스턴스를 선택하는 것은 어려울 수 있지만 Amazon Fraud Detector를 사용하면이 기준을 모집단 평균으로 설정하여 이를 쉽게 수행할 수 있습니다.

SageMaker AI 모델 가져오기

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

GetEventPrediction 요청의 일부로 전송된 이벤트 변수를 사용하도록 Amazon Fraud Detector를 구성할 수 있습니다. 이벤트 변수를 사용하도록 선택한 경우 입력 템플릿을 제공해야 합니다. Amazon Fraud Detector는이 템플릿을 사용하여 이벤트 변수를 SageMaker AI 엔드포인트를 호출하는 데 필요한 입력 페이로드로 변환합니다. 또는 GetEventPrediction 요청의 일부로 전송되는 byteBuffer를 사용하도록 SageMaker AI 모델을 구성할 수 있습니다.

Amazon Fraud Detector는 JSON 또는 CSV 입력 형식과 JSON 또는 CSV 출력 형식을 사용하는 SageMaker AI 알고리즘 가져오기를 지원합니다. 지원되는 SageMaker AI 알고리즘의 예로는 XGBoost, Linear Learner 및 Random Cut Forest가 있습니다.

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

SageMaker AI 모델을 가져오려면 PutExternalModel API를 사용합니다. 다음 예제에서는 SageMaker AI 엔드포인트sagemaker-transaction-model가 배포되었고, InService 상태이며, XGBoost 알고리즘을 사용한다고 가정합니다.

입력 구성은가 이벤트 변수를 사용하여 모델 입력을 구성하도록 지정합니다(useEventVariables는로 설정됨TRUE). XGBoost에 CSV 입력이 필요한 경우 입력 형식은 TEXT_CSV입니다. csvInputTemplate은 GetEventPrediction 요청의 일부로 전송된 변수에서 CSV 입력을 구성하는 방법을 지정합니다. 이 예제에서는 변수 order_amt, prev_amt hist_amt 및를 생성했다고 가정합니다payment_type.

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

> **Note**
>
> SageMaker AI 모델의 출력은 소스가 인 변수에 매핑되어야 합니다EXTERNAL_MODEL_SCORE. 변수를 사용하여 콘솔에서 이러한 변수를 생성할 수 없습니다. 모델 가져오기를 구성할 때 대신 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 또는 모델 버전 삭제

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

Amazon SageMaker AI 모델이 감지기 버전과 연결되어 있지 않은 경우에도 제거할 수 있습니다. SageMaker AI 모델을 제거하면 Amazon Fraud Detector에서 연결이 끊어지지만 SageMaker AI에서는 모델을 계속 사용할 수 있습니다.

모델 버전을 삭제하려면

`Ready to deploy` 상태인 모델 버전만 삭제할 수 있습니다. 모델 버전을에서 `ACTIVE` `Ready to deploy` 상태로 변경하려면 모델 버전을 배포 취소합니다.

1. 에 로그인 AWS Management Console 하고 https://console.aws.amazon.com/frauddetector Amazon Fraud Detector 콘솔을 엽니다.
2. Amazon Fraud Detector 콘솔의 왼쪽 탐색 창에서 모델을 선택합니다.
3. 삭제하려는 모델 버전이 포함된 모델을 선택합니다.
4. 삭제할 모델 버전을 선택합니다.
5. 작업을 선택한 후 삭제를 선택합니다.

6. 모델 버전 이름을 입력한 다음 모델 버전 삭제를 선택합니다.

모델 버전을 배포 취소하려면

감지기 버전(ACTIVE, ,)에서 사용 중인 모델 버전은 배포 취소할 수 없습니다INACTIVEDRAFT. 따라서 감지기 버전에서 사용 중인 모델 버전을 배포 해제하려면 먼저 감지기 버전에서 모델 버전을 제거합니다.

1. Amazon Fraud Detector 콘솔의 왼쪽 탐색 창에서 모델을 선택합니다.
2. 배포 해제하려는 모델 버전이 포함된 모델을 선택합니다.
3. 삭제할 모델 버전을 선택합니다.
4. 작업을 선택한 다음 모델 버전 배포 취소를 선택합니다.

모델을 삭제하려면

모델을 삭제하기 전에 먼저 모든 모델 버전을 삭제해야 하며 모델과 연결되어 있습니다.

1. Amazon Fraud Detector 콘솔의 왼쪽 탐색 창에서 모델을 선택합니다.
2. 삭제할 모델을 선택합니다.
3. 작업을 선택한 후 삭제를 선택합니다.
4. 모델 이름을 입력한 다음 모델 삭제를 선택합니다.

Amazon SageMaker AI 모델을 제거하려면

1. Amazon Fraud Detector 콘솔의 왼쪽 탐색 창에서 모델을 선택합니다.
2. 제거할 SageMaker AI 모델을 선택합니다.
3. 작업을 선택한 다음 모델 제거를 선택합니다.
4. 모델 이름을 입력한 다음 SageMaker AI 모델 제거를 선택합니다.

감지기

탐지기는 사기에 대해 평가하려는 특정 비즈니스 이벤트 하나에 대한 모델 및 규칙과 같은 사기 탐지 로직이 포함된 컨테이너입니다. 먼저 이미 정의한 이벤트를 지정하여 감지기를 생성하고 선택적으로 이벤트에 대해 Amazon Fraud Detector에서 이미 생성하고 훈련한 모델 버전을 추가합니다.

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

각 감지기 버전은 `DRAFT`, `ACTIVE`또는 상태여야 합니다`INACTIVE`. 한 번에 하나의 감지기 버전만 `ACTIVE` 상태가 될 수 있습니다. Amazon Fraud Detector는 `ACTIVE` 상태의 감지기 버전을 사용하여 사기 예측을 생성합니다.

감지기 생성

이미 정의한 이벤트 유형을 지정하여 감지기를 생성합니다. Amazon Fraud Detector에서 이미 훈련하고 배포한 모델을 선택적으로 추가할 수 있습니다. 모델을 추가하는 경우 규칙을 생성할 때 규칙 표현식에서 Amazon Fraud Detector에서 생성한 모델 점수를 사용할 수 있습니다(예: `$model score < 90`).

Amazon Fraud Detector 콘솔에서 PutDetector API를 사용하거나 put-detector 명령을 사용하거나 SDK를 사용하여 감지기를 생성할 수 있습니다. AWS API, 명령 또는 SDK를 사용하여 감지기를 생성하는 경우 감지기를 생성한 후에 대한 지침을 따릅니다감지기 버전 생성.

Amazon Fraud Detector 콘솔에서 감지기 생성

이 예제에서는 이벤트 유형을 생성했고 사기 예측에 사용할 모델 버전을 생성하고 배포했다고 가정합니다.

1단계: 감지기 빌드

1. Amazon Fraud Detector 콘솔의 왼쪽 탐색 창에서 감지기를 선택합니다.
2. 감지기 생성을 선택합니다.
3. 감지기 세부 정보 정의 페이지에서 감지기 이름`sample_detector`에를 입력합니다. 선택적으로 와 같은 감지기에 대한 설명을 입력합니다`my sample fraud detector`.

4. 이벤트 유형에서 사기 예측을 위해 생성한 이벤트 유형을 선택합니다.
5. Next(다음)를 선택합니다.

2단계: 배포된 모델 버전 추가

1. 이는 선택적 단계입니다. 감지기에 모델을 추가할 필요가 없습니다. 이 단계를 건너뛰려면 다음(Next)을 선택합니다.
2. 모델 추가 - 선택 사항에서 모델 추가를 선택합니다.
3. 모델 추가 페이지의 모델 선택에서 이전에 배포한 Amazon Fraud Detector 모델 이름을 선택합니다. 버전 선택에서 배포된 모델의 모델 버전을 선택합니다.
4. 모델 추가를 선택합니다.
5. Next(다음)를 선택합니다.

3단계: 규칙 추가

규칙은 사기 예측을 평가할 때 변수 값을 해석하는 방법을 Amazon Fraud Detector에 알려 주는 조건입니다. 이 예제에서는 모델 점수를 변수 값으로 사용하여 `high_fraud_risk`, 및 `medium_fraud_risk`라는 세 가지 규칙을 생성합니다`low_fraud_risk`. 자체 규칙, 규칙 표현식, 규칙 실행 순서 및 결과를 생성하려면 모델 및 사용 사례에 적합한 값을 사용합니다.

1. 규칙 추가 페이지의 규칙 정의에서 규칙 이름에를 입력하고 설명 - 선택 사항에서 규칙에 대한 설명**`This rule captures events with a high ML model score`**으로를 입력합니다. `high_fraud_risk`
2. 표현식에서 Amazon Fraud Detector 간소화된 규칙 표현식을 사용하여 다음 규칙 표현식을 입력합니다.

   ```
   $sample_fraud_detection_model_insightscore > 900
   ```
3. 결과에서 새 결과 생성을 선택합니다. 결과는 사기 예측의 결과이며 평가 중에 규칙이 일치하면 반환됩니다.
4. 새 결과 생성에서 결과 이름으로 `verify_customer`를 입력합니다. 필요한 경우 설명을 입력합니다.
5. 결과 저장을 선택합니다.
6. 규칙 추가를 선택하여 규칙 검증 검사기를 실행하고 규칙을 저장합니다. 생성 후 Amazon Fraud Detector는 감지기에서 규칙을 사용할 수 있도록 합니다.

7. 다른 규칙 추가를 선택한 다음 규칙 생성 탭을 선택합니다.
8. 다음 low_fraud_risk 규칙 세부 정보를 사용하여이 프로세스를 두 번 더 반복하여 medium_fraud_risk 및 규칙을 생성합니다.

   - medium_fraud_risk

     규칙 이름: medium_fraud_risk

     결과: review

     표현식:

     $sample_fraud_detection_model_insightscore <= 900 and

     $sample_fraud_detection_model_insightscore > 700

   - low_fraud_risk

     규칙 이름: low_fraud_risk

     결과: approve

     표현식:

     $sample_fraud_detection_model_insightscore <= 700
9. 사용 사례에 대한 모든 규칙을 생성한 후 다음을 선택합니다.

   규칙 생성 및 작성에 대한 자세한 내용은 규칙 및 섹션을 참조하세요규칙 언어 참조.

4단계: 규칙 실행 및 규칙 순서 구성

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

기본 규칙 실행 모드는 입니다FIRST_MATCHED.

첫 번째 일치

첫 번째 일치 규칙 실행 모드는 정의된 규칙 순서를 기반으로 첫 번째 일치 규칙의 결과를 반환합니다. FIRST_MATCHED를 지정하면 Amazon Fraud Detector는 처음부터 마지막까지 순차적으로 규칙을 평가하고 처음 일치하는 규칙에서 중지합니다. 그런 다음 Amazon Fraud Detector는 해당 단일 규칙에 대한 결과를 제공합니다.

에서 규칙을 실행하는 순서는 결과적으로 발생하는 사기 예측 결과에 영향을 미칠 수 있습니다. 규칙을 생성한 후 다음 단계에 따라 규칙을 원하는 순서로 실행하도록 다시 정렬합니다.

high_fraud_risk 규칙이 아직 규칙 목록 상단에 없는 경우 순서를 선택한 다음 1을 선택합니다. 그러면 high_fraud_risk 첫 번째 위치로 이동합니다.

규칙이 두 번째 위치에 있고 medium_fraud_risk 규칙low_fraud_risk이 세 번째 위치에 있도록이 프로세스를 반복합니다.

모두 일치

일치하는 모든 규칙 실행 모드는 규칙 순서에 관계없이 일치하는 모든 규칙에 대한 결과를 반환합니다. ALL_MATCHED를 지정하면 Amazon Fraud Detector는 모든 규칙을 평가하고 일치하는 모든 규칙에 대한 결과를 반환합니다.

이 자습서FIRST_MATCHED에서를 선택한 후 다음을 선택합니다.

5단계: 감지기 버전 검토 및 생성

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

1. 검토 및 생성 페이지에서 구성한 감지기 세부 정보, 모델 및 규칙을 검토합니다. 변경해야 하는 경우 해당 섹션 옆의 편집을 선택합니다.
2. 감지기 생성을 선택합니다. 생성되면 감지기의 첫 번째 버전이 감지기 버전 테이블에 Draft 상태와 함께 나타납니다.

   초안 버전을 사용하여 감지기를 테스트합니다.

를 사용하여 감지기 생성 AWS SDK for Python (Boto3)

다음 예제에서는 PutDetector API에 대한 샘플 요청을 보여줍니다. 감지기는 감지기 버전의 컨테이너 역할을 합니다. PutDetector API는 감지기가 평가할 이벤트 유형을 지정합니다. 다음 예제에서는 이벤트 유형을 생성했다고 가정합니다sample_registration.

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

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

```
)
```

감지기 버전 생성

감지기 버전은 사기 예측 생성 요청의 일부로 사용할 규칙, 규칙 실행 순서 및 선택적으로 모델 버전을 정의합니다. 감지기 내에 정의된 규칙을 감지기 버전에 추가할 수 있습니다. 평가된 이벤트 유형에 대해 훈련된 모델을 추가할 수도 있습니다.

각 감지기 버전의 상태는 DRAFT, ACTIVE또는 입니다INACTIVE. 한 번에 하나의 감지기 버전만 ACTIVE 상태가 될 수 있습니다. GetEventPrediction 요청 중에 Amazon Fraud Detector는 아무것도 DetectorVersion 지정하지 않으면 ACTIVE 감지기를 사용합니다.

규칙 실행 모드

Amazon Fraud Detector는 FIRST_MATCHED 및의 두 가지 규칙 실행 모드를 지원합니다ALL_MATCHED.

- 규칙 실행 모드가 인 경우 FIRST_MATCHEDAmazon Fraud Detector는 규칙을 먼저, 마지막으로 순차적으로 평가하여 일치하는 첫 번째 규칙에서 중지합니다. 그런 다음 Amazon Fraud Detector는 해당 단일 규칙에 대한 결과를 제공합니다. 규칙이 false(일치하지 않음)로 평가되면 목록의 다음 규칙이 평가됩니다.
- 규칙 실행 모드가 ALL_MATCHED인 경우 순서에 관계없이 평가의 모든 규칙이 병렬로 실행됩니다. Amazon Fraud Detector는 모든 규칙을 실행하고 일치하는 모든 규칙에 대해 정의된 결과를 반환합니다.

를 사용하여 감지기 버전 생성 AWS SDK for Python (Boto3)

다음 예제에서는 CreateDetectorVersion API에 대한 샘플 요청을 보여줍니다. 규칙 실행 모드가로 설정되어 FIRST_MATCHED있으므로 Amazon Fraud Detector는 규칙을 처음부터 마지막으로 순차적으로 평가하여 일치하는 첫 번째 규칙에서 중지합니다. 그런 다음 Amazon Fraud Detector는에서 해당 단일 규칙에 대한 결과를 제공합니다GetEventPrediction 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'
)
```

감지기 버전의 상태를 업데이트하려면 UpdateDetectorVersionStatus API를 사용합니다. 다음 예제에서는 감지기 버전 상태를에서 DRAFT로 업데이트합니다ACTIVE. GetEventPrediction 요청 중에 탐지기 ID가 지정되지 않은 경우 Amazon Fraud Detector는 탐지기 ACTIVE 버전을 사용합니다.

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

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

감지기, 감지기 버전 또는 규칙 버전 삭제

Amazon Fraud Detector에서 감지기를 삭제하기 전에 먼저 감지기와 연결된 모든 감지기 버전 및 규칙 버전을 삭제해야 합니다.

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

감지기 버전을 삭제하려면

DRAFT 또는 INACTIVE 상태의 감지기 버전만 삭제할 수 있습니다.

1. 에 로그인 AWS Management Console 하고 https://console.aws.amazon.com/frauddetector Amazon Fraud Detector 콘솔을 엽니다.
2. Amazon Fraud Detector 콘솔의 왼쪽 탐색 창에서 감지기를 선택합니다.
3. 삭제하려는 감지기 버전이 포함된 감지기를 선택합니다.
4. 삭제할 감지기 버전을 선택합니다.
5. 작업을 선택한 후 삭제를 선택합니다.
6. **delete**를 입력한 다음 감지기 삭제를 선택합니다.

규칙 버전을 삭제하려면

규칙 버전은 ACTIVE 또는 INACTIVE 감지기 버전에서 사용하지 않는 경우에만 삭제할 수 있습니다. 필요한 경우 규칙 버전을 삭제하기 전에 먼저 ACTIVE 감지기 버전을 로 이동한 INACTIVE다음 INACTIVE 감지기 버전을 삭제합니다.

1. Amazon Fraud Detector 콘솔의 왼쪽 탐색 창에서 감지기를 선택합니다.
2. 삭제하려는 규칙 버전이 포함된 감지기를 선택합니다.
3. 연결된 규칙 탭을 선택하고 삭제할 규칙을 선택합니다.
4. 삭제할 규칙 버전을 선택합니다.
5. 작업을 선택한 다음 규칙 버전 삭제를 선택합니다.
6. **delete**를 입력한 다음 버전 삭제를 선택합니다.

감지기를 삭제하려면

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

1. Amazon Fraud Detector 콘솔의 왼쪽 탐색 창에서 감지기를 선택합니다.
2. 삭제할 감지기를 선택합니다.
3. 작업을 선택한 다음 감지기 삭제를 선택합니다.
4. **delete**를 입력한 다음 감지기 삭제를 선택합니다.

리소스

모델, 규칙 및 탐지기는 변수, 결과, 레이블, 목록 및 엔터티와 같은 리소스를 사용하여 사기 위험의 이벤트를 평가합니다. 이 섹션에서는 리소스 생성 및 관리에 대한 정보를 제공합니다.

주제

변수

변수는 사기 예측에 사용하려는 데이터 요소를 나타냅니다. 이러한 변수는 모델 훈련을 위해 준비한 이벤트 데이터 세트, Amazon Fraud Detector 모델의 위험 점수 출력 또는 Amazon SageMaker AI 모델에서 가져올 수 있습니다. 이벤트 데이터 세트에서 가져온 변수에 대한 자세한 내용은 섹션을 참조하세요데이터 모델 탐색기를 사용하여 이벤트 데이터 세트 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 유형이 할당되어야 합니다. Amazon Fraud Detector는 IP 주소, 은행 식별 번호(BINs) 및 전화번호와 같이 사용자가 제공하는 일부 변수를 보강하여 이러한 변수를 사용하는 모델에 대한 추가 입력을 생성하고 성능을 높입니다.

데이터 타입

변수에는 변수가 나타내는 데이터 요소에 대한 데이터 형식이 있어야 하며, 선택적으로 미리 정의된 중 하나를 할당할 수 있습니다변수 유형. 변수 유형에 할당된 변수의 경우 데이터 유형이 미리 선택됩니다. 가능한 데이터 형식에는 다음 형식이 포함됩니다.

데이터 유형	설명	기본값	예제 값
String	문자, 정수 또는 둘 다의 조합	<비어 있음>	abc, 123, 1D3B

데이터 유형	설명	기본값	예제 값
Integer	양수 또는 음수 정수	0	1, -1
불	참 또는 거짓	False	True, False
DateTime	ISO 8601 표준 UTC 형식으로만 지정된 날짜 및 시간	<비어 있음>	2019-11-30T13:01:01Z
Float	소수점이 있는 숫자	0.0	4.01, 0.10

기본값

변수에는 기본값이 있어야 합니다. Amazon Fraud Detector가 사기 예측을 생성할 때 Amazon Fraud Detector가 변수 값을 수신하지 않는 경우이 기본값은 규칙 또는 모델을 실행하는 데 사용됩니다. 제공하는 기본값은 선택한 데이터 유형과 일치해야 합니다. AWS 콘솔에서 Amazon Fraud Detector는 `0` 정수의 경우 , 부울의 `false` 경우 , 부동 소수점의 `0.0` 경우 , 문자열의 경우 (비어 있음)의 기본값을 할당합니다. 이러한 데이터 유형에 대해 사용자 지정 기본값을 설정할 수 있습니다.

변수 유형

변수를 생성할 때 선택적으로 변수를 변수 유형에 할당할 수 있습니다. 변수 유형은 모델을 훈련하고 사기 예측을 생성하는 데 사용되는 일반적인 데이터 요소를 나타냅니다. 연결된 변수 유형이 있는 변수만 모델 훈련에 사용할 수 있습니다. 모델 훈련 프로세스의 일환으로 Amazon Fraud Detector는 변수와 연결된 변수 유형을 사용하여 변수 보강, 기능 엔지니어링 및 위험 점수를 수행합니다.

Amazon Fraud Detector에는 변수에 할당하는 데 사용할 수 있는 다음과 같은 변수 유형이 미리 정의되어 있습니다.

범주	변수 유형	설명	데이터 유형	예제
세션	IP_ADDRESS	이벤트 중에 수집되는 IP 주소	Strin	192.0.2.0

범주	변수 유형	설명	데이터 유형	예제
				참고: Amazon Fraud Detector는 이 데이터를 보강합니다. 자세한 내용은 지리적 위치 보강 단원

범주	변수 유형	설명	데이터 유형	예제
				을 참조하세요.
	USERAGEN	이벤트 중에 수집된 사용자 에이전트	Strin	Mozilla 5.0(Windows NT 10.0, Win64, x64,rv:68.0) Gecko 20100101
	지문	이벤트에 사용되는 디바이스의 고유 식별자입니다.	Strin	sadfow987u234
	SESSION_ID	이벤트의 활성 세션에 대한 세션 ID	Strin	sid123456789
	ARE_CREDENTIALS_VALID	이벤트 로그인에 사용되는 자격 증명이 유효한지 여부를 나타냅니다.	불	True
Us	EMAIL_ADDRESS	이벤트 중에 수집되는 이메일 주소	Strin	abc@domain.com

범주	변수 유형	설명	데이터 유형	예제
	PHONE_NUMBER	이벤트 중에 수집된 전화번호	String	+1 555-0100 참고: Amazon Fraud Detector는 이 데이터를 보강합니다. 자세한 내용은 전화번호 보강

범주	변수 유형	설명	데이터 유형	예제
				단원을 참조하세요.
결제	BILLING_NAME	결제 주소와 연결된 이름	Strin	John Doe

범주	변수 유형	설명	데이터 유형	예제
	BILLING_PHONE	결제 주소와 연결된 전화번호	Strin	+1 555-0100 참고: Amazon Fraud Detector는 이 데이터를 보강합니다. 자세한 내용은 전화번호 보강

범주	변수 유형	설명	데이터 유형	예제
				단원을 참조하세요.
	BILLING_ADDRESS_L	결제 주소의 첫 번째 줄	Strin	모든 거리
	BILLING_ADDRESS_L	결제 주소의 두 번째 줄	Strin	모든 단위 123
	청구_도시	결제 주소에 있는 도시	Strin	모든 도시
	BILLING_STATE	결제 주소에 있는 주 또는 도	Strin	모든 주 또는 도

범주	변수 유형	설명	데이터 유형	예제
	BILLING_COUNTRY	결제 주소에 있는 국가	Strin	모든 국가 참고: Amazon Fraud Detector는 이 데이터를 보강합니다. 자세한 내용은 지리적 위

범주	변수 유형	설명	데이터 유형	예제
				치보강 단원을 참조하세요.

범주	변수 유형	설명	데이터 유형	예제
	BILLING_ZIP	결제 주소에 있는 우편 번호	Strin	01234 참고: Amazon Fraud Detector는 이 데이터를 보강합니다. 자세한 내용은 지리적 위치 보강

범주	변수 유형	설명	데이터 유형	예제
				단원을 참조하세요.
배송	SHIPPING_NAME	배송 주소와 연결된 이름	Strin	John Doe

범주	변수 유형	설명	데이터 유형	예제
	SHIPPING_PHONE	배송 주소와 연결된 전화번호	Strin	+1 555-0100 참고: Amazon Fraud Detector는 이 데이터를 보강합니다. 자세한 내용은 전화번호 보강

범주	변수 유형	설명	데이터 유형	예제
				단원을 참조하세요.
	SHIPPING_ADDRESS_1	배송 주소의 첫 번째 줄	Strin	123 Any Street
	SHIPPING_ADDRESS_2	배송 주소의 두 번째 줄	Strin	유닛 123
	배송_도시	배송 주소에 있는 도시	Strin	모든 도시
	SHIPPING_STATE	배송 주소에 있는 주 또는 도	Strin	모든 상태

범주	변수 유형	설명	데이터 유형	예제
	SHIPPING_COUNTRY	배송 주소에 있는 국가	Strin	모든 국가 참고: Amazon Fraud Detector는 이 데이터를 보강합니다. 자세한 내용은 지리적 위

범주	변수 유형	설명	데이터 유형	예제
				치보강 단원을 참조하세요.

범주	변수 유형	설명	데이터 유형	예제
	SHIPPING_ZIP	배송 주소에 있는 우편 번호	Strin	01234 참고: Amazon Fraud Detector는 이 데이터를 보강합니다. 자세한 내용은 지리적 위치 보강

범주	변수 유형	설명	데이터 유형	예제
				단원을 참조하세요.
Payment	ORDER_ID	트랜잭션의 고유 식별자	Strin	LUX60
	요금	총 주문 가격	Strin	560.00
	CURRENCY_CODE	ISO 4217 통화 코드	Strin	USD
	결제_유형	이벤트 중에 결제에 사용되는 결제 방법	Strin	신용카드
	AUTH_COD	신용 카드 발급자 또는 발급 은행에서 보내는 영숫자 코드	Strin	0000
	AVS	카드 프로세서의 주소 확인 시스템(AVS) 응답 코드	Strin	Y
Pr	PRODUCT_ATEGORY	주문 항목의 제품 범주	Strin	주방

범주	변수 유형	설명	데이터 유형	예제
사용자 지정 (Custom)	NUMERIC	실제 숫자로 표현할 수 있는 변수	Floa	1.224
	CATEGORICAL	범주, 세그먼트 또는 그룹을 설명하는 변수	Strin	대형
	free_FORM_TEXT	이벤트의 일부로 캡처된 자유 형식 텍스트(예: 고객 리뷰 또는 의견)	Strin	자유 형식 텍스트 입력의 예

변수 유형에 변수 할당

모델 훈련에 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 변경하는 것이 매우 어려울 수 있습니다. 특히 여러 모델과 이벤트가 변수를 사용한 경우 더욱 그렇습니다.

미리 정의된 변수 유형 중 하나 또는 `FREE_FORM_TEXT`, `CATEGORICAL`또는 사용자 지정 변수 유형 중 하나를 변수에 할당할 수 있습니다`NUMERIC`.

변수를 올바른 변수 유형에 할당하기 위한 중요 참고 사항

1. 변수가 사전 정의된 변수 유형 중 하나와 일치하는 경우이 변수를 사용합니다. 변수 유형이 변수에 해당하는지 확인합니다. 예를 들어 ip_address 변수를 `EMAIL_ADDRESS` 변수 유형에 할당하면

ip_address 변수는 ASN, ISP, 지리적 위치 및 위험 점수와 같은 보강으로 보강되지 않습니다. 자세한 내용은 가변 보강 단원을 참조하십시오.

2. 변수가 사전 정의된 변수 유형과 일치하지 않는 경우 아래 나열된 권장 사항에 따라 사용자 지정 변수 유형 중 하나를 할당합니다.
3. 일반적으로 자연 순서가 없고 범주, 세그먼트 또는 그룹에 넣을 수 있는 변수에 CATEGORICAL 변수 유형을 할당합니다. 모델 훈련에 사용하는 데이터 세트에는 merchant_id, campaign_id 또는 policy_id와 같은 ID 변수가 있을 수 있습니다. 이러한 변수는 그룹을 나타냅니다(예: policy_id가 동일한 모든 고객은 그룹을 나타냄). 다음 데이터가 있는 변수에는 CATEGORICAL 변수 유형을 할당해야 합니다.
   - customer_ID, segment_ID, color_ID, department_code 또는 product_ID와 같은 데이터가 포함된 변수입니다.
   - true, false 또는 null 값이 있는 부울 데이터가 포함된 변수입니다.
   - 회사 이름, 제품 범주, 카드 유형 또는 추천 미디어와 같은 그룹 또는 범주에 넣을 수 있는 변수입니다.

   > **Note**
   >
   > ENTITY_ID는 Amazon Fraud Detector에서 ENTITY_ID 변수에 할당하는 데 사용되는 예약 변수 유형입니다. ENTITY_ID 변수는 평가하려는 작업을 시작하는 개체의 ID입니다. TFI(Transaction Fraud Insight) 모델 유형을 생성하는 경우 ENTITY_ID 변수를 제공해야 합니다. 데이터의 어떤 변수가 작업을 시작하는 엔터티를 고유하게 식별하고 이를 ENTITY_ID 변수로 전달해야 합니다. 데이터 세트에 있는 다른 모든 IDs에 CATEGORICAL 변수 유형이 있고 모델 훈련에 사용하는 경우 CATEGORICAL 변수 유형을 할당합니다. 데이터 세트의 개체가 아닌 다른 IDs의 예로는 merchant_ID, policy_ID 및 campaign_ID가 있습니다.

4. 텍스트 블록이 포함된 FREE_FORM_TEXT 변수에 변수 유형을 할당합니다. FREE_FORM_TEXT 변수 유형의 예로는 사용자 리뷰, 설명, 날짜 및 추천 코드가 있습니다. FREE_FORM_TEXT 데이터에는 구분 기호로 구분된 여러 토큰이 포함되어 있습니다. 구분 기호는 영숫자 및 밑줄 기호 이외의 모든 문자일 수 있습니다. 예를 들어 사용자 리뷰와 주석은 “공백” 구분 기호로 구분할 수 있으며, 날짜 및 추천 코드는 하이픈을 구분 기호로 사용하여 접두사, 접미사 및 중간 부분을 구분할 수 있습니다. Amazon Fraud Detector는 구분 기호를 사용하여 FREE_FORM_TEXT 변수에서 데이터를 추출합니다.
5. 실수이고 순서가 고유한 변수에 NUMERIC 변수 유형을 할당합니다. NUMERIC 변수의 예로는 day_of_the_week, incident_severity, customer_rating 등이 있습니다. 이러한 변수에 CATEGORICAL 변수 유형을 할당할 수 있지만 모든 실수 변수는 NUMERIC 변수 유형에 고유한 순서로 할당하는 것이 좋습니다.

가변 보강

Amazon Fraud Detector는 IP 주소, 은행 식별 번호(BINs) 및 전화번호와 같이 사용자가 제공하는 일부 원시 데이터 요소를 보강하여 이러한 데이터 요소를 사용하는 모델에 대한 추가 입력을 생성하고 성능을 향상시킵니다. 보강은 잠재적으로 의심스러운 상황을 식별하고 모델이 더 많은 사기를 포착하는 데 도움이 됩니다.

전화번호 보강

Amazon Fraud Detector는 전화번호 데이터를 지리적 위치, 원래 통신 사업자 및 전화번호의 유효성에 관련된 추가 정보로 보강합니다. 전화번호 보강은 2021년 12월 13일 이후에 훈련되고 국가 코드(+xxx)가 포함된 전화번호가 있는 모든 모델에 대해 자동으로 활성화됩니다. 모델에 전화번호 변수를 포함시키고 2021년 12월 13일 이전에 학습한 경우이 보강을 활용할 수 있도록 모델을 재학습하세요.

데이터가 성공적으로 보강되도록 전화번호 변수에 다음 형식을 사용하는 것이 좋습니다.

변수	형식	설명
PHONE_NUMBER	E.164 표준	전화번호와 함께 국가 코드(+xxx)를 포함해야 합니다.
BILLING_PHONE 및 SHIPPING_PHONE	E.164 표준	전화번호와 함께 국가 코드(+xxx)를 포함해야 합니다.

지리적 위치 보강

2022년 2월 8일부터 Amazon Fraud Detector는 이벤트에 제공하는 IP_ADDRESS, BILLING_ZIP 및 SHIPPING_ZIP 값 간의 물리적 거리를 계산합니다. 계산된 거리는 사기 탐지 모델의 입력으로 사용됩니다.

지리적 위치 보강을 활성화하려면 이벤트 데이터에 IP_ADDRESS, BILLING_ZIP 또는 SHIPPING_ZIP의 세 가지 변수 중 두 개 이상이 포함되어야 합니다. 또한 각 BILLING_ZIP 및 SHIPPING_ZIP 값에는 각각 유효한 BILLING_COUNTRY 코드와 SHIPPING_COUNTRY 코드가 있어야 합니다. 2022년 2월 8일 이전에 훈련된 모델이 있고 이러한 변수가 포함된 경우 지리 위치 보강을 활성화하기 위해 모델을 재훈련해야 합니다.

Amazon Fraud Detector가 데이터가 유효하지 않아 이벤트의 IP_ADDRESS, BILLING_ZIP 또는 SHIPPING_ZIP 값과 연결된 위치를 확인할 수 없는 경우, 대신 특수 자리 표시자 값이 사용됩니다. 예를 들어 이벤트에 유효한 IP_ADDRESS 및 BILLING_ZIP 값이 있지만 SHIPPING_ZIP 값이 유효하지 않다고 가정해 보겠습니다. 이 경우 보강은 IP_ADDRESS–> BILLING_ZIP에 대해서만 수행됩니다. IP_ADDRESS–>SHIPPING_ZIP 및 BILLING_ZIP–>SHIPPING_ZIP 에는 보강이 수행되지 않습니다. 대신 자리 표시자 값이 자리에 사용됩니다. 모델에 지리적 위치 보강이 활성화되어 있는지 여부에 관계없이 모델의 성능은 변경되지 않습니다.

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

지리적 위치 변수 형식

위치 데이터가 성공적으로 보강되도록 지리적 위치 변수에 다음 형식을 사용하는 것이 좋습니다.

변수	형식	설명
IP_ADDRESS	IPv4 주소	예: 1.1.1.1
BILLING_ZIP 및 SHIPPING_ZIP	지정된 국가의 ISO 3166-1 alpha-2 우편번호	자세한 내용은이 주제의 국가 및 지역 코드 섹션을 참조하세요.
BILLING_COUNTRY 및 SHIPPING_COUNTRY	ISO 3166-1 alpha-2 2자 표준 국가 코드	자세한 내용은이 주제의 국가 및 지역 코드 섹션을 참조하세요. Amazon Fraud Detector는 국가 이름의 모든 일반적인 변형을 ISO 3166-1 2자 표준 국가 코드와 일치시키려고 합니다. 그러나 올바르게 매칭될 것이라고 보장할 수는 없습니다.

국가 및 관할 구역 코드

다음 표에는 지리적 위치 보강을 위해 Amazon Fraud Detector에서 지원하는 국가 및 지역의 전체 목록이 나와 있습니다. 각 국가 및 지역에는 할당된 국가 코드(특히 ISO 3166-1 알파-2 2자 국가 코드)와 우편 번호가 있습니다.

우편 번호 형식

- 9 - 숫자
- a - 문자
- [X] - X는 선택 사항입니다. 예를 들어 Guersney "GY9[9] 9aa"는 "GY9 9aa"와 "GY99 9aa"가 모두 유효함을 의미합니다. 한 가지 형식을 사용합니다.
- [X/XX] - X 또는 XX를 사용할 수 있습니다. 예를 들어, Bermuda "aa[aa/99]"는 "aa"와 "aa 99"가 모두 유효함을 의미합니다. 이러한 형식 중 하나를 사용하지만 둘 다 사용하지는 않습니다.
- 일부 국가에는 고정 접두사가 있습니다. 예를 들어 Andorra의 우편 번호는 AD999입니다. 즉, 국가 코드는 문자 AD로 시작하고 세 개의 숫자로 시작해야 합니다.

코드	명칭	우편 번호
AD	안도라	AD999
AR	네덜란드령 안틸 제도	9999
AT	오스트리아	9999
AU	호주	9999
AZ	아제르바이잔	AZ 9999
BD	방글라데시	9999
BE	벨기에	9999
BG	불가리아	9999
BM	버뮤다	aa[aa/99]
BY	벨로루시	999999

코드	명칭	우편 번호
CA	캐나다	a9a 9a9
CH	스위스	9999
CL	칠레	9999999
CO	콜롬비아	999999
CR	코스타리카	99999
CY	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
FO	페로 제도	999
FR	프랑스	99999
GB	영국	a[a]9[a/9] 9aa
GG	건지	GY9[9] 9aa
GL	그린란드	9999

코드	명칭	우편 번호
GP	과들루프	99999
GT	과테말라	99999
GU	괌	99999
HR	크로아티아	99999
HU	헝가리	9999
IE	아일랜드	a99[a/9][a/9][a/9][a/9]
IM	맨 섬	IM9[9]9aa
IN	인도	999999
IS	아이슬란드	999
IT	이탈리아	99999
JE	저지	JE9[9]9aa
JP	일본	999-9999
KR	대한민국	99999
LI	리히텐슈타인	9999
LK	스리랑카	99999
LT	리투아니아	99999
LU	룩셈부르크	L-9999
LV	라트비아	LV-9999
MC	모나코	99999
MD	몰도바 공화국	9999

코드	명칭	우편 번호
MH	마셜 제도	99999
MK	북 마케도니아	9999
MP	북마리아나 제도	99999
MQ	마티니크	99999
MT	몰타	aaa 9999
MX	멕시코	99999
MY	말레이시아	99999
NL	네덜란드	9999 aa
NO	노르웨이	9999
NZ	뉴질랜드	9999
PH	필리핀	9999
PK	파키스탄	99999
PL	폴란드	99-999
PR	푸에르토리코	99999
PT	포르투갈	9999-999
PW	팔라우	99999
RE	레위니옹	99999
RO	루마니아	999999
RU	러시아 연방	999999
SE	스웨덴	999 99

코드	명칭	우편 번호
SG	싱가포르	999999
SI	슬로베니아	9999
SK	슬로바키아	999 99
SM	산마리노	99999
TH	태국	99999
TR	터키	99999
UA	우크라이나	99999
미국	미국	99999
UY	우루과이	99999
VI	미국령 버진 제도	99999
WF	월리스 푸투나	99999
YT	마요트	99999
ZA	남아프리카공화국	9999

Useragent 보강

Account Takeover Insights(ATI) 모델을 생성하는 경우 데이터 세트에 변수 유형의 `useragent` 변수를 제공해야 합니다. 이 변수에는 로그인 이벤트의 브라우저, 디바이스 및 OS 데이터가 포함됩니다. Amazon Fraud Detector는 사용자 에이전트 데이터를 `user_agent_family` `OS_family`, 및와 같은 추가 정보로 보강합니다`device_family`.

변수 생성

Amazon Fraud Detector 콘솔에서 create-variable 명령을 사용하거나 CreateVariable을 사용하거나를 사용하여 변수를 생성할 수 있습니다. AWS SDK for Python (Boto3)

Amazon Fraud Detector 콘솔을 사용하여 변수 생성

이 예제에서는 email_address 및 라는 두 개의 변수를 생성하고 해당 변수 유형(EMAIL_ADDRESS 및)에 ip_address할당합니다IP_ADDRESS. 이러한 변수는 예제로 사용됩니다. 모델 훈련에 사용할 변수를 생성하는 경우 사용 사례에 적합한 데이터 세트의 변수를 사용합니다. 변수를 생성하기 가변 보강 전에 변수 유형 및에 대해 읽어야 합니다.

변수를 생성하려면

1. AWS Management Console을 열고 계정에 로그인합니다.
2. Amazon Fraud Detector로 이동하여 왼쪽 탐색 창에서 변수를 선택한 다음 생성을 선택합니다.
3. 새 변수 페이지에서 변수 이름으로 email_address를 입력합니다. 선택적으로 변수에 대한 설명을 입력합니다.
4. 변수 유형에서 이메일 주소를 선택합니다.
5. Amazon Fraud Detector는이 변수 유형이 사전 정의되어 있으므로이 변수 유형에 대한 데이터 유형을 자동으로 선택합니다. 변수에 변수 유형이 자동으로 할당되지 않은 경우 목록에서 변수 유형을 선택합니다. 자세한 내용은 변수 유형 단원을 참조하십시오.
6. 변수의 기본값을 제공하려면 사용자 지정 기본값 정의를 선택하고 변수의 기본값을 입력합니다. 이 예제를 따르는 경우이 단계를 건너뜁니다.
7. 생성(Create)을 선택합니다.
8. email_address 개요 페이지에서 방금 생성한 변수의 세부 정보를 확인합니다.

   업데이트해야 하는 경우 편집을 선택하고 업데이트를 제공합니다. Save changes(변경 사항 저장)를 선택합니다.
9. 프로세스를 반복하여 다른 변수를 생성하고 변수 유형에 대해 IP 주소를 ip_address 선택합니다.
10. 변수 페이지에는 새로 생성된 변수가 표시됩니다.

> **Important**
>
> 데이터 세트에서 원하는 수만큼 변수를 생성하는 것이 좋습니다. 나중에 이벤트 유형을 생성할 때 모델을 훈련하여 사기를 감지하고 사기 감지를 생성하기 위해 포함할 변수를 결정할 수 있습니다.

를 사용하여 변수 생성 AWS SDK for Python (Boto3)

다음 예제에서는 CreateVariable API에 대한 요청을 보여줍니다. 이 예제에서는 `email_address` 및 라는 두 개의 변수를 생성하고 해당 변수 유형(`EMAIL_ADDRESS` 및)에 `ip_address`할당합니다`IP_ADDRESS`.

이러한 변수는 예제로 사용됩니다. 모델 훈련에 사용할 변수를 생성하는 경우 사용 사례에 적합한 데이터 세트의 변수를 사용합니다. 변수를 생성하기 가변 보강 전에 변수 유형 및에 대해 읽어야 합니다.

변수 소스를 지정해야 합니다. 변수 값이 파생되는 위치를 식별하는 데 도움이 됩니다. 변수 소스가 EVENT인 경우 변수 값은 GetEventPrediction 요청의 일부로 전송됩니다. 변수 값이 이면 Amazon Fraud Detector로 `MODEL_SCORE`채워집니다. `EXTERNAL_MODEL_SCORE`인 경우 변수 값은 가져온 SageMaker AI 모델로 채워집니다.

```
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 Detector가 해당 변수를 영구적으로 삭제하고 데이터는 더 이상 Amazon Fraud Detector에 저장되지 않습니다.

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

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

Amazon Fraud Detector 콘솔에서 변수를 삭제하거나, delete-variable CLI 명령을 사용하거나, DeleteVariable API를 사용하거나, AWS SDK for Python (Boto3)

콘솔을 사용하여 변수 삭제

변수를 삭제하려면

1. 에 로그인 AWS Management Console 하고 https://console.aws.amazon.com/frauddetector Amazon Fraud Detector 콘솔을 엽니다.
2. Amazon Fraud Detector 콘솔의 왼쪽 탐색 창에서 리소스를 선택한 다음 변수를 선택합니다.
3. 삭제할 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 DeleteVariable API를 사용하여 변수 customer_name을 삭제합니다.

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

fraudDetector.delete_variable (

name = 'customer_name'

)
```

레이블

레이블은 이벤트가 시기성인지 합법적인지를 분류합니다. 레이블은 이벤트 유형과 연결되며 Amazon Fraud Detector에서 기계 학습 모델을 훈련하는 데 사용됩니다. 온라인 사기 인사이트(OFI) 또는 거래 사기 인사이트(TFI) 모델을 훈련시키려는 경우 훈련 데이터 세트에서 최소 400개의 이벤트를 사기성 또는 합법적인 것으로 분류해야 합니다. 사기, 적법성, 1 또는 0과 같은 레이블을 사용하여 훈련 데이터

세트에서 이벤트를 분류할 수 있습니다. 훈련이 완료되면 훈련된 모델은 사기에 대한 이벤트를 평가하고 이러한 값을 사용하여 이벤트를 사기 또는 합법적으로 분류합니다.

먼저 훈련 데이터 세트에 사용된 값으로 레이블을 생성한 다음 사기 탐지 모델을 구축하고 훈련하는 데 사용되는 이벤트 유형과 레이블을 연결해야 합니다.

레이블 생성

Amazon Fraud Detector 콘솔에서 put-label 명령을 사용하거나 PutLabel API를 사용하거나를 사용하여 레이블을 생성할 수 있습니다 AWS SDK for Python (Boto3).

Amazon Fraud Detector 콘솔을 사용하여 레이블 생성

레이블을 생성하려면

1. AWS Management Console을 열고 계정에 로그인합니다.
2. Amazon Fraud Detector로 이동하여 왼쪽 탐색 창에서 레이블을 선택한 다음 생성을 선택합니다.
3. 레이블 생성 페이지에서 사기 이벤트의 레이블 이름을 레이블 이름으로 입력합니다. 레이블 이름은 훈련 데이터 세트의 사기 활동을 나타내는 레이블과 일치해야 합니다. 선택적으로 레이블에 대한 설명을 입력합니다.
4. 레이블 생성을 선택합니다.
5. 두 번째 레이블을 생성하고 합법적인 이벤트의 레이블 이름을 입력합니다. 레이블 이름이 훈련 데이터 세트의 합법적인 활동을 나타내는 값과 일치하는지 확인합니다.

를 사용하여 레이블 생성 AWS SDK for Python (Boto3)

다음 AWS SDK for Python (Boto3) 예제 코드는 PutLabel API를 사용하여 두 개의 레이블(사기, 적법)을 생성합니다. 레이블을 생성한 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'
)
```

레이블 업데이트

이벤트 데이터 세트가 Amazon Fraud Detector에 저장된 경우 이벤트에 대한 오프라인 사기 조사를 수행하고 기계 학습 피드백 루프를 닫으려는 경우와 같이 저장된 이벤트에 대한 레이블을 추가하거나 업데이트해야 할 수 있습니다.

updateupdate-event-label 명령, UpdateEventLabel API 또는를 사용하여 저장된 이벤트의 레이블을 추가하거나 업데이트할 수 있습니다. AWS SDK for Python (Boto3)

다음 AWS SDK for Python (Boto3) 예제 코드는 `UpdateEventLabel` API를 사용하여 이벤트 유형 등록과 연결된 레이블 사기를 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 레이블 업데이트

이벤트에 대한 오프라인 사기 조사를 수행하고 기계 학습 피드백 루프를 닫으려는 경우와 같이 Amazon Fraud Detector에 이미 저장된 이벤트에 대한 사기 레이블을 추가하거나 업데이트해야 할 수 있습니다. Amazon Fraud Detector에 이미 저장된 이벤트의 레이블을 업데이트하려면 `UpdateEventLabel` API 작업을 사용합니다. 다음은 UpdateEventLabel API 호출의 예입니다.

```
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'
)
```

레이블 삭제

레이블을 삭제하면 Amazon Fraud Detector가 해당 레이블을 영구적으로 삭제하고 데이터는 더 이상 Amazon Fraud Detector에 저장되지 않습니다.

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

Amazon Fraud Detector 콘솔에서, delete-label 명령을 사용하거나, DeleteLabel API를 사용하거나,를 사용하여 레이블을 삭제할 수 있습니다. AWS SDK for Python (Boto3)

콘솔을 사용하여 레이블 삭제

레이블을 삭제하려면

1. 에 로그인 AWS Management Console 하고 https://console.aws.amazon.com/frauddetector Amazon Fraud Detector 콘솔을 엽니다.
2. Amazon Fraud Detector 콘솔의 왼쪽 탐색 창에서 리소스를 선택한 다음 레이블을 선택합니다.
3. 삭제할 레이블을 선택합니다.
4. 작업을 선택한 후 삭제를 선택합니다.
5. 레이블 이름을 입력한 다음 레이블 삭제를 선택합니다.

를 사용하여 레이블 삭제 AWS SDK for Python (Boto3)

다음 AWS SDK for Python (Boto3) 예제 코드는 DeleteLabel API를 사용하여 레이블 적자를 삭제합니다.

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

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

규칙

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

- 변수 또는 목록 - 변수는 사기 예측에 사용할 이벤트 데이터 세트의 데이터 요소를 나타냅니다. 목록은 이벤트 데이터 세트의 변수에 대한 입력 데이터 요소 집합입니다. 규칙에 사용되는 변수는 평가된 이벤트 유형에서 사전 정의되어야 하며 규칙에 사용되는 목록은 변수 유형과 연결되어야 합니다. 자세한 내용은 변수 및 Lists 단원을 참조하세요.
- 표현식 - 규칙의 표현식은 비즈니스 로직을 캡처합니다. 규칙에서 변수를 사용하는 경우 변수, >, <, <=, >=와 같은 비교 연산자 및 값을 사용하여 간단한 규칙 표현식을 구성합니다. == . 목록을 사용하는 경우 규칙 표현식은 목록 항목, `in`및 목록 이름으로 구성됩니다. 자세한 내용은 규칙 언어 참조 단원을 참조하십시오. and 및를 사용하여 여러 표현식을 결합할 수 있습니다`or`. 모든 표현식은 부울 값(true 또는 false)으로 평가되어야 하며 길이가 4,000자 미만이어야 합니다. If-else 유형 조건은 지원되지 않습니다.
- 결과 - 규칙이 일치할 때 Amazon Fraud Detector에서 반환하는 응답입니다. 결과는 사기 예측의 결과를 나타냅니다. 가능한 각 사기 예측에 대한 결과를 생성하고 규칙에 추가할 수 있습니다. 자세한 내용은 결과 단원을 참조하십시오.

감지기에는 하나 이상의 관련 규칙이 있어야 합니다. 규칙은 최대 3개의 목록을 가질 수 있으며 감지기는 최대 30개의 목록을 가질 수 있습니다. 감지기 생성 프로세스의 일부로 규칙을 생성합니다. 새 규칙을 생성하고 기존 감지기와 연결할 수도 있습니다.

규칙 언어 참조

다음 섹션에서는 Amazon Fraud Detector의 표현식(즉, 규칙 작성) 기능을 간략하게 설명합니다.

변수 사용

평가된 이벤트 유형에 정의된 모든 변수를 표현식의 일부로 사용할 수 있습니다. 달러 기호를 사용하여 변수를 나타냅니다.

```
$example_variable < 100
```

목록 사용

변수 유형과 연결되고 규칙 표현식의 일부로 항목으로 채워진 모든 목록을 사용할 수 있습니다. 달러 기호를 사용하여 목록 항목 값을 표시합니다.

```
$example_list_variable in @list_name
```

비교, 멤버십 및 자격 증명 연산자

Amazon Fraud Detector에는 >, >=, <, <=,!=, ==, in, not in 등의 비교 연산자가 포함됩니다.

예를 들어, 다음과 같습니다.

예: <

```
$variable < 100
```

예: in, not in

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

예: !=

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

예: ==

```
$variable == 1000
```

연산자 테이블

연산자	Amazon 사기 탐지기 운영자
같음	==
같지 않음	!=
보다 큼	>

연산자	Amazon 사기 탐지기 운영자
보다 작음	<
크거나 같음	>=
작거나 같음	<=
있음	in
및	and
Or	or
아님	!

기본 수학

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

아래 규칙에서는 `$variable_1`를 사용하여 변수를 추가`$variable_2`하고 합계가 10 미만인지 확인합니다.

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

기본 수학 테이블 데이터

연산자	Amazon 사기 탐지기 운영자
더하기	+
마이너스	-
곱하기	*
나누기	/
모듈로	%

정규 표현식(regex)

정규식을 사용하여 표현식의 일부로 특정 패턴을 검색할 수 있습니다. 이는 변수 중 하나에 대한 특정 문자열 또는 숫자 값과 일치시키려는 경우에 특히 유용합니다. Amazon Fraud Detector는 정규식으로 작업할 때만 일치를 지원합니다(예: 제공된 문자열이 정규식과 일치하는지 여부에 따라 True/False를 반환함). Amazon Fraud Detector의 정규 표현식 지원은 java의 .matches()를 기반으로 합니다(RE2J 정규 표현식 라이브러리 사용). 인터넷에는 다양한 정규 표현식 패턴을 테스트하는 데 유용한 몇 가지 유용한 웹 사이트가 있습니다.

아래 첫 번째 예제에서는 먼저 변수를 소문자`email`로 변환합니다. 그런 다음 패턴이 `email` 변수`@gmail.com`에 있는지 확인합니다. 문자열을 명시적으로 확인할 수 있도록 두 번째 기간이 이스케이프됩니다`.com`.

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

두 번째 예에서는 변수에 국가 코드가 `phone_number` 포함되어 있는지 확인하여 전화번호가 미국인지 `+1` 확인합니다. 더하기 기호는 문자열을 명시적으로 확인할 수 있도록 이스케이프됩니다`+1`.

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

정규식 테이블

연산자	Amazon 사기 탐지기 예제
로 시작하는 문자열과 일치	regex_match("^mystring", $variable)
전체 문자열을 정확히 일치시킵니다.	regex_match("mystring", $variable)
새 줄을 제외한 모든 문자와 일치	regex_match(".", $variable)
'mystring' 이전의 새 줄을 제외한 모든 문자와 일치	regex_match(".*mystring", $variable)
이스케이프 특수 문자	\

누락된 값 확인

경우에 따라 값이 누락되었는지 확인하는 것이 좋습니다. Amazon Fraud Detector에서는 null로 표시됩니다. 다음 구문을 사용하여이 작업을 수행할 수 있습니다.

```
$variable != null
```

마찬가지로 값이 없는지 확인하려면 다음을 수행할 수 있습니다.

```
$variable == null
```

여러 조건

and 및를 사용하여 여러 표현식을 결합할 수 있습니다or. Amazon Fraud Detector는 단일 true 값이 발견되면 OR 표현식에서 중지되고, 단일 false 값이 발견되면 AND에서 중지됩니다.

아래 예제에서는 조건을 사용하여 두 and 가지 조건을 확인합니다. 첫 번째 문에서는 변수 1이 100 미만인지 확인합니다. 두 번째에서는 변수 2가 미국이 아닌지 확인합니다.

규칙이를 사용하므로 전체 조건이 TRUE로 평가되려면 and두 가지 모두 TRUE여야 합니다.

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

다음과 같이 괄호를 사용하여 부울 작업을 그룹화할 수 있습니다.

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

기타 표현식 유형

DateTime 함수

함수	설명	예제
getcurrentdatetime()	규칙 실행의 현재 시간을 ISO8601 UTC 형식으로 지정합니다. getepochmilliseconds(getcurrentdatetime())를 사용하여 추가 작업을 수행할 수 있습니다.	getcurrentdatetime() == "2023-03-28T18:34:02Z"
isbefore(DateTime1, DateTime2)	호출자 DateTime1이 DateTime2보다 이전인 경우 부울(True/False)을 반환합니다.	isbefore(getcurrentdatetime(), "2019-11-30T01:01:01Z") == "False"

함수	설명	예제
		isbefore(getcurrentdatetime(), "2050-11-30T01:05:01Z") == "True"
isafter(DateTime1, DateTime2)	호출자 DateTime1이 DateTime2 이후인 경우 부울(True/False)을 반환합니다.	isafter(getcurrentdatetime(), "2019-11-30T01:01:01Z") == "True" isafter(getcurrentdatetime(), "2050-11-30T01:05:01Z") == "False"
getepochmilliseconds(DateTime)	DateTime을 가져와서 해당 DateTime을 에포크 밀리초 단위로 반환합니다. 날짜에 수학 작업을 수행하는 데 유용합니다.	getepochmilliseconds("2019-11-30T01:01:01Z") == 1575032461

문자열 연산자

연산자	예제
문자열을 대문자로 변환	대문자($variable)
문자열을 소문자로 변환	소문자($variable)

기타

연산자	설명
설명 추가	# 내 의견

규칙 생성

Amazon Fraud Detector 콘솔에서 create-rule 명령을 사용하거나 CreateRule API를 사용하거나를 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 평가되어야 하며 길이가 4,000자 미만이어야 합니다. If-else 유형 조건은 지원되지 않

습니다. 표현식에 사용되는 모든 변수는 평가된 이벤트 유형에서 사전 정의되어야 합니다. 마찬가지로 표현식에 사용되는 모든 목록은 사전 정의되고, 변수 유형과 연결되고, 항목으로 채워져야 합니다.

다음 예제에서는 기존 감지기에 high_risk 대한 규칙을 생성합니다payments_detector. 규칙은 표현식과 결과를 규칙verify_customer과 연결합니다.

사전 조건

아래 언급된 단계를 따르려면 규칙 생성을 진행하기 전에 다음을 완료해야 합니다.

- 감지기 생성
- 결과 생성

사용 사례에 대한 감지기, 규칙 및 결과를 생성하는 경우 감지기 이름, 규칙 이름, 규칙 표현식 및 결과 이름 예를 사용 사례와 관련된 이름 및 표현식으로 바꿉니다.

Amazon Fraud Detector 콘솔에서 새 규칙 생성

1. AWS Management Console을 열고 계정에 로그인합니다. Amazon Fraud Detector로 이동합니다.
2. 왼쪽 탐색 창에서 탐지기를 선택하고 사용 사례에 대해 생성한 탐지기, 예를 들면 payment_detector를 선택합니다.
3. Payment_detector 페이지에서 연결된 규칙 탭을 선택한 다음 규칙 생성을 선택합니다.
4. 새 규칙 페이지에서 다음을 입력합니다.
    a. 이름에 규칙의 이름을 입력합니다. 예 **high_risk**
    b. 설명 - 선택 사항에서 선택적으로 규칙 설명, 예를 입력합니다. **This rule captures events with a high ML model score**
    c. 표현식 빠른 참조 가이드를 사용하여 표현식에서 사용 사례에 대한 규칙 표현식을 입력합니다. 예: `$sample_fraud_detection_model_insightscore >900`
    d. 결과에서 사용 사례에 대해 생성한 결과, 예를 들면 verify_customer를 선택합니다. 결과는 사기 예측의 결과이며 평가 중에 규칙이 일치하면 반환됩니다.
5. 규칙 저장을 선택합니다.

감지기에 대한 새 규칙을 생성했습니다. 이는 Amazon Fraud Detector가 감지기가 자동으로 사용할 수 있도록 하는 규칙의 버전 1입니다.

를 사용하여 규칙 생성 AWS SDK for Python (Boto3)

다음 예제 코드는 CreateRule API를 사용하여 기존 감지기에 high_risk 대한 규칙을 생성합니다payments_detector. 또한 예제 코드는 규칙 표현식과 결과를 verify_customer 규칙에 추가합니다.

사전 조건

예제 코드를 사용하려면 규칙 생성을 진행하기 전에 다음을 완료해야 합니다.

- 감지기 생성
- 결과 생성

사용 사례에 대한 감지기, 규칙 및 결과를 생성하는 경우 감지기 이름, 규칙 이름, 규칙 표현식 및 결과 이름 예를 사용 사례와 관련된 이름 및 표현식으로 바꿉니다.

```
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 Detector에서 자동으로 사용할 수 있도록 하는 규칙 버전 1을 생성했습니다.

규칙 업데이트

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 updateupdate-rule-version 명령을 사용하거나 UpdateRuleVersion API를 사용하거나 AWS SDK를 사용하여 규칙을 업데이트할 수 있습니다.

규칙을 업데이트한 후에는 새 규칙 버전을 사용하도록 감지기 버전을 업데이트해야 합니다.

Amazon Fraud Detector 콘솔에서 규칙 업데이트

규칙을 업데이트하려면

1. AWS Management Console을 열고 계정에 로그인합니다. Amazon Fraud Detector로 이동합니다.
2. 왼쪽 탐색 창에서 감지기를 선택합니다.
3. 감지기 창에서 업데이트하려는 규칙과 연결된 감지기를 선택합니다.
4. 감지기 페이지에서 연결된 규칙 탭을 선택하고 업데이트할 규칙을 선택합니다.
5. 규칙 페이지에서 작업을 선택하고 버전 생성을 선택합니다.
6. 버전이 변경되었습니다. 업데이트된 설명, 표현식 또는 결과를 입력합니다.
7. 새 버전 저장을 선택합니다.

를 사용하여 규칙 업데이트 AWS SDK for Python (Boto3)

다음 예제 코드는 UpdateRuleVersion API를 사용하여 규칙의 임계값을 900high_risk에서 950으로 업데이트합니다. 이 규칙은 감지기와 연결됩니다payments_detector.

```
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']
)
```

Lists

목록은 이벤트 데이터 세트의 변수에 대한 입력 데이터 세트입니다. 감지기와 연결된 규칙에서 입력 데이터를 사용합니다. 규칙은 사기 예측 중에 입력 데이터를 해석하는 방법을 Amazon Fraud Detector에 알려주는 조건입니다. 예를 들어 IP 주소 목록을 생성한 다음 특정 IP 주소가 목록에 있는 경우 액세스를 거부하는 규칙을 생성할 수 있습니다. 목록을 사용하는 규칙은 $ip_address_value @list_name 형식으로 표시됩니다.

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

목록에는 최대 100,000개의 고유 항목이 포함될 수 있으며 각 항목은 최대 320자까지 가능합니다. 규칙에 사용하는 모든 목록은 기본적으로 Amazon Fraud Detector의 변수 유형 FREE_FORM_TEXT와 연결됩니다. 언제든지 목록에 변수 유형을 할당할 수 있습니다. 규칙에 최대 3개의 목록을 사용할 수 있습니다.

목록을 생성하거나, 목록에 항목을 추가하거나, 목록을 삭제하거나, 목록에서 하나 이상의 항목을 삭제하거나, API를 사용하거나,를 사용하거나 AWS CLI, AWS SDK를 사용하여 Amazon Fraud Detector 콘솔의 목록에 변수 유형을 할당할 수 있습니다.

목록 생성

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

목록을 생성하려면 먼저 이름을 지정한 다음 선택적으로 Amazon Fraud Detector에서 변수 유형 지원하는에 목록을 연결해야 합니다. 기본적으로 Amazon Fraud Detector는 목록을 FREE_FORM_TEXT 변수 유형으로 가정합니다.

Amazon Fraud Detector 콘솔에서 API, AWS CLI또는 AWS SDK를 사용하여 목록을 생성할 수 있습니다.

Amazon Fraud Detector 콘솔을 사용하여 목록 생성

목록을 생성하려면

1. AWS Management Console을 열고 계정에 로그인합니다. Amazon Fraud Detector로 이동합니다.
2. 왼쪽 탐색 창에서 목록을 선택합니다.
3. 목록 세부 정보에서
    a. 목록 이름에 목록 이름을 입력합니다.
    b. 선택적으로 설명에 설명을 입력합니다.
    c. (선택 사항) 변수 유형에서 목록의 변수 유형을 선택합니다.

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

4. 목록 데이터 추가에서 목록 항목을 각 줄에 하나씩 추가합니다. 스프레드시트에서 항목을 복사하여 붙여 넣을 수도 있습니다.

> **Note**
>
> 항목이 쉼표를 사용하여 구분되지 않고 목록에서 고유해야 합니다. 두 개의 동일한 항목이 입력되면 하나만 추가됩니다.

5. 생성(Create)을 선택합니다.

를 사용하여 목록 생성 AWS SDK for Python (Boto3)

목록 이름을 지정하여 목록을 생성합니다. 선택적으로 설명을 제공하거나, 변수 유형을 연결하거나, 목록을 생성할 때 목록에 항목을 추가할 수 있습니다. 또는 나중에 항목 또는 설명을 추가하여 목록을 업데이트할 수 있습니다. 목록 생성 시 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 할당해야 합니다. 변수 유형을 할당하지 않으면 Amazon Fraud Detector는 목록을 FREE_FORM_TEXT 변수 유형으로 가정합니다.

다음 예제에서는 CreateList API 작업을 사용하여 설명, 변수 유형을 제공하고 4개의 `allow_email_ids` 목록 항목을 추가하여 목록을 생성합니다.

```
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 생성한 후에는 언제든지 목록에 항목을 추가하거나 추가할 수 있습니다. 목록에 항목을 추가하거나 추가할 때 목록이 연결된 규칙을 업데이트할 필요가 없습니다. 규칙은 새로 추가된 항목을 자동으로 통합합니다.

목록에는 최대 100,000개의 고유 항목이 포함될 수 있으며 각 항목은 최대 320자일 수 있습니다.

Amazon Fraud Detector 콘솔에서 API, AWS CLI또는 AWS SDK를 사용하여 항목을 추가할 수 있습니다.

Amazon Fraud Detector 콘솔을 사용하여 목록에 항목 추가

목록에 항목을 하나 이상 추가하려면

1. AWS Management Console을 열고 계정에 로그인합니다. Amazon Fraud Detector로 이동합니다.
2. 왼쪽 탐색 창에서 목록을 선택합니다.
3. 목록 페이지에서 항목을 추가할 목록을 선택합니다.
4. 목록 세부 정보 페이지에서 데이터 나열 탭을 선택하고 데이터 추가를 선택합니다.
5. 목록 데이터 추가 상자에서 각 줄에 하나의 항목을 추가하거나 스프레드시트에서 항목을 복사하여 붙여넣습니다. 쉼표를 사용하여 항목을 구분하지 마십시오.
6. 추가를 선택합니다.

를 사용하여 목록에 항목 추가 AWS SDK for Python (Boto3)

다음 예제에서는 UpdateList API 작업을 사용하여 `allow_email_ids` 목록에 두 개의 새 항목을 추가합니다. 추가하려는 항목이 목록에서 고유한지 확인합니다.

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

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

목록에 변수 유형 할당

규칙에 사용하는 모든 목록은 Amazon Fraud Detector의 변수 유형 변수 유형과 연결되어야 합니다. 기본적으로 Amazon Fraud Detector는 목록을 FREE_FORM_TEXT 변수 유형으로 가정합니다. IP 주소로 구성된 목록은 IP_ADDRESS 변수 유형과 연결되어야 합니다.

목록 생성 시 또는 나중에 언제든지 목록을 변수 유형과 연결할 수 있습니다. 목록을 변수 유형과 이미 연결하고 나중에 변경하려면 새 목록을 생성해야 합니다. 목록의 변수 유형은 변경할 수 없습니다.

Amazon Fraud Detector 콘솔에서 API, AWS CLI또는 AWS SDK를 사용하여 변수 유형을 할당할 수 있습니다.

Amazon Fraud Detector 콘솔을 사용하여 목록에 변수 유형 할당

목록에 변수 유형을 할당하려면

1. AWS Management Console을 열고 계정에 로그인합니다. Amazon Fraud Detector로 이동합니다.
2. 왼쪽 탐색 창에서 목록을 선택합니다.
3. 목록 페이지에서 변수 유형을 할당할 목록을 선택합니다.
4. 목록 세부 정보 페이지에서 작업을 선택하고 목록 편집을 선택합니다.
5. 목록 편집 상자에서 목록의 변수 유형을 선택합니다.
6. 저장(Save)을 선택합니다.

를 사용하여 목록에 변수 유형 할당 AWS SDK for Python (Boto3)

다음 예제에서는 UpdateList API 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 규칙에서도 사용되지 않는 목록을 삭제할 수 있습니다. 목록을 삭제하면 Amazon Fraud Detector가 해당 목록과 목록의 모든 항목을 영구적으로 삭제합니다.

AWS CLI 또는 AWS SDK를 사용하여 API를 사용하여 Amazon Fraud Detector 콘솔에서 목록을 삭제할 수 있습니다.

Amazon Fraud Detector 콘솔을 사용하여 목록 삭제

목록을 삭제하려면

1. AWS Management Console을 열고 계정에 로그인합니다. Amazon Fraud Detector로 이동합니다.
2. 왼쪽 탐색 창에서 목록을 선택합니다.
3. 목록 페이지에서 삭제할 목록을 선택합니다.
4. 목록 세부 정보 페이지에서 작업을 선택하고 목록 삭제를 선택합니다.
5. 목록 삭제를 선택합니다.

를 사용하여 목록 삭제 AWS SDK for Python (Boto3)

다음 예제에서는 DeleteList API 작업을 사용하여를 삭제합니다`allow_email_ids`.

```
import boto3
                        fraudDetector = boto3.client('frauddetector')
fraudDetector.delete_list(
   name = 'allow_email_ids'
)
```

목록에서 항목 삭제

언제든지 목록에서 항목을 하나 이상 삭제할 수 있습니다. 목록에서 항목을 삭제할 때 목록이 연결된 규칙을 업데이트할 필요가 없습니다. 규칙은 업데이트된 목록을 자동으로 통합합니다.

AWS CLI 또는 AWS SDK를 사용하여 API를 사용하여 Amazon Fraud Detector 콘솔의 목록에서 항목을 삭제할 수 있습니다.

Amazon Fraud Detector 콘솔을 사용하여 목록에서 항목 삭제

목록에서 하나 이상의 항목을 삭제하려면

1. AWS Management Console을 열고 계정에 로그인합니다. Amazon Fraud Detector로 이동합니다.
2. 왼쪽 탐색 창에서 목록을 선택합니다.
3. 목록 페이지에서 삭제하려는 항목이 포함된 목록을 선택합니다.
4. 목록 세부 정보 페이지에서 데이터 나열 탭을 선택하고 삭제할 항목을 선택합니다.
5. 삭제를 선택하고 다시 삭제를 선택하여 확인합니다.

를 사용하여 목록에서 항목 삭제 AWS SDK for Python (Boto3)

다음 예제에서 UpdateList API 작업은 `allow_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

목록에 규칙이 사용되지 않는 경우 목록의 모든 항목을 삭제할 수 있습니다. 목록에 있는 모든 항목을 삭제하고 나중에 동일한 목록에 항목을 추가할 수 있습니다.

AWS CLI 또는 AWS SDK를 사용하여 API를 사용하여 Amazon Fraud Detector 콘솔의 목록에서 항목을 삭제할 수 있습니다.

Amazon Fraud Detector 콘솔을 사용하여 목록에서 모든 항목 삭제

목록에서 모든 항목을 삭제하려면

1. AWS Management Console을 열고 계정에 로그인합니다. Amazon Fraud Detector로 이동합니다.

2. 왼쪽 탐색 창에서 목록을 선택합니다.
3. 목록 페이지에서 삭제하려는 항목이 포함된 목록을 선택합니다.
4. 목록 세부 정보 페이지에서 데이터 나열 탭을 선택하고 모두 삭제를 선택합니다.
5. 모두 삭제 상자에 `delete all`를 입력하여 확인한 다음 모든 목록 데이터 삭제를 선택합니다.

를 사용하여 목록에서 모든 항목 삭제 AWS SDK for Python (Boto3)

다음 예제에서 UpdateList API 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(높음_위험, 중간_위험 및 낮음_위험) 또는 작업(승인, 검토)을 나타내기를 원할 수 있습니다. 결과가 생성된 후 규칙에 하나 이상의 결과를 추가할 수 있습니다. GetEventPrediction 응답의 일부로 Amazon Fraud Detector는 일치하는 규칙에 대해 정의된 결과를 반환합니다.

결과 생성

Amazon Fraud Detector 콘솔에서 put-outcome 명령을 사용하거나 PutOutcome API를 사용하거나를 사용하여 결과를 생성할 수 있습니다 AWS SDK for Python (Boto3).

Amazon Fraud Detector 콘솔을 사용하여 결과 생성

하나 이상의 결과를 생성하려면

1. AWS Management Console을 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 선택적으로 설명을 입력합니다.
5. 결과 저장을 선택합니다.
6. 2~5단계를 반복하여 추가 결과를 생성합니다.

를 사용하여 결과 생성 AWS SDK for Python (Boto3)

다음 예제에서는 PutOutcome API를 사용하여 세 가지 결과를 생성합니다. `verify_customer`, `review`및 입니다`approve`. 결과가 생성된 후 규칙에 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 삭제하고 데이터는 더 이상 Amazon Fraud Detector에 저장되지 않습니다.

Amazon Fraud Detector 콘솔, delete-outcome 명령, DeleteOutcome API 또는를 사용하여 결과를 삭제할 수 있습니다. AWS SDK for Python (Boto3)

Amazon Fraud Detector 콘솔에서 결과 삭제

결과를 삭제하려면

1. 에 로그인 AWS Management Console 하고 https://console.aws.amazon.com/frauddetector Amazon Fraud Detector 콘솔을 엽니다.
2. Amazon Fraud Detector 콘솔의 왼쪽 탐색 창에서 리소스를 선택한 다음 결과를 선택합니다.
3. 삭제할 결과를 선택합니다.
4. 작업을 선택한 후 삭제를 선택합니다.
5. 결과 이름을 입력한 다음 결과 삭제를 선택합니다.

를 사용하여 결과 삭제 AWS SDK for Python (Boto3)

다음 예제에서는 DeleteOutcome API를 사용하여 `verify_customer` 결과를 삭제합니다. 결과가 삭제된 후에는 더 이상 규칙에 할당할 수 없습니다.

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

fraudDetector.delete_outcome(
name = 'verify_customer'
)
```

개체

개체는 이벤트를 수행하는 사람 또는 사물을 나타냅니다. 엔터티 유형에 따라 엔터티가 분류됩니다. 분류 예에는 고객, 판매자, 사용자 또는 계정이 포함됩니다. 이벤트를 수행한 특정 개체를 나타내기 위해 이벤트 데이터 세트의 일부로 개체 유형(ENTITY_TYPE)과 개체 식별자(ENTITY_ID)를 제공합니다.

Amazon Fraud Detector는 이벤트에 대한 사기 예측을 생성할 때 엔터티 유형을 사용하여 누가 이벤트를 수행했는지 나타냅니다. 사기 예측에 사용할 엔터티 유형은 먼저 Amazon Fraud Detector에서 생성한 다음 이벤트 유형을 생성할 때 이벤트에 추가해야 합니다.

개체 유형 생성

Amazon Fraud Detector 콘솔에서 put-entity-type 명령을 사용하거나 PutEntityType API를 사용하거나 를 사용하여 개체 유형을 생성할 수 있습니다 AWS SDK for Python (Boto3). 아래 예제에서는 Amazon

Fraud Detector 콘솔customer에서 SDK for Python(Boto3)을 사용하여 엔터티 유형을 생성합니다. 사기 탐지 모델을 훈련하기 위한 이벤트 유형과 연결할 엔터티 유형을 생성하는 경우 사용 사례에 적합한 이벤트 데이터 세트의 엔터티 유형을 사용합니다.

Amazon Fraud Detector 콘솔을 사용하여 개체 유형 생성

개체 유형을 생성하려면

1. AWS Management Console을 열고 계정에 로그인합니다.
2. Amazon Fraud Detector로 이동하여 왼쪽 탐색에서 엔터티를 선택한 다음 생성을 선택합니다.
3. 개체 생성 페이지에서 엔터티 유형 이름으로 고객을 입력합니다. 선택적으로 개체에 대한 설명을 입력합니다.
4. 개체 생성을 선택합니다.

를 사용하여 개체 유형 생성 AWS SDK for Python (Boto3)

다음 AWS SDK for Python (Boto3) 코드 예제에서는 PutEntityType API를 사용하여 엔터티 유형을 생성합니다customer. 사기 탐지 모델을 훈련하기 위한 이벤트 유형과 연결할 엔터티 유형을 생성하는 경우 사용 사례에 적합한 이벤트 데이터 세트의 엔터티를 사용합니다.

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

fraudDetector.put_entity_type(
name = 'customer',
description = 'customer'
)
```

개체 유형 삭제

Amazon Fraud Detector에서는 이벤트 유형에 포함된 엔터티 유형을 삭제할 수 없습니다. 먼저 엔터티가 연결된 이벤트 유형을 삭제한 다음 엔터티 유형을 삭제해야 합니다.

개체 유형을 삭제하면 Amazon Fraud Detector는 해당 개체 유형을 영구적으로 삭제하고 데이터는 더 이상 Amazon Fraud Detector에 저장되지 않습니다.

개체 유형은 Amazon Fraud Detector 콘솔, delete-entity-type 명령, DeleteEntityType API 또는를 사용하여 삭제할 수 있습니다. AWS SDK for Python (Boto3)

Amazon Fraud Detector 콘솔에서 개체 유형 삭제

개체 유형을 삭제하려면

1. 에 로그인 AWS Management Console 하고 https://console.aws.amazon.com/frauddetector Amazon Fraud Detector 콘솔을 엽니다.
2. Amazon Fraud Detector 콘솔의 왼쪽 탐색 창에서 리소스를 선택한 다음 엔터티를 선택합니다.
3. 삭제할 엔터티 유형을 선택합니다.
4. 작업을 선택한 후 삭제를 선택합니다.
5. 개체 유형 이름을 입력한 다음 개체 유형 삭제를 선택합니다.

를 사용하여 개체 유형 삭제 AWS SDK for Python (Boto3)

다음 AWS SDK for Python (Boto3) 예제 코드는 DeleteEntityType API를 사용하여 개체 유형 고객을 삭제합니다.

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

fraudDetector.delete_entity_type (

name = 'customer'

)
```

를 사용하여 Amazon Fraud Detector 리소스 관리 AWS CloudFormation

Amazon Fraud Detector는 Amazon Fraud Detector 리소스를 모델링하고 설정하는 데 도움이 되는 AWS CloudFormation서비스와 통합되어 리소스와 인프라를 생성하고 관리하는 데 소요되는 시간을 줄일 수 있습니다. 원하는 모든 Amazon Fraud Detector 리소스(예: Detector, Variables, EntityType, EventType, Outcome 및 Label)를 설명하고 해당 리소스를 AWS CloudFormation 프로비저닝하고 구성하는 템플릿을 생성합니다. 템플릿을 재사용하여 여러 AWS 계정 및 리전에서 리소스를 일관되고 반복적으로 프로비저닝하고 구성할 수 있습니다.

AWS CloudFormation 사용에 대한 추가 요금은 없습니다.

Amazon Fraud Detector 템플릿 생성

Amazon Fraud Detector 및 관련 서비스에 대한 리소스를 프로비저닝하고 구성하려면 AWS CloudFormation 템플릿을 이해해야 합니다. 템플릿은 JSON 또는 YAML로 서식 지정된 텍스트 파일입니다. 이러한 템플릿은 AWS CloudFormation 스택에서 프로비저닝하려는 리소스를 설명합니다. JSON 또는 YAML에 익숙하지 않은 경우 AWS CloudFormation Designer를 사용하여 AWS CloudFormation 템플릿을 시작할 수 있습니다. 자세한 내용은 AWS CloudFormation 사용 설명서의 AWS CloudFormation Designer란 무엇입니까?를 참조하세요.

AWS CloudFormation 템플릿을 사용하여 Amazon Fraud Detector 리소스를 생성, 업데이트 및 삭제할 수도 있습니다. 리소스에 대한 JSON 및 YAML 템플릿의 예를 포함하여 자세한 내용은 AWS CloudFormation 사용 설명서의 Amazon Fraud Detector 리소스 유형 참조를 참조하세요.

이미 CloudFormation을 사용하고 있는 경우 추가 IAM 정책 또는 CloudTrail 로깅을 관리할 필요가 없습니다.

Amazon Fraud Detector 스택 관리

CloudFormation 콘솔 또는 AWS CLI를 통해 Amazon Fraud Detector 스택을 생성, 업데이트 및 삭제할 수 있습니다.

스택을 생성하려면 AWS CloudFormation이 스택에 포함할 리소스를 설명하는 템플릿이 있어야 합니다. 이미 생성한 Amazon Fraud Detector 리소스를 새 스택 또는 기존 스택으로 가져와 CloudFormation 관리로 가져올 수도 있습니다. https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/resource-import.html

스택 관리에 대한 자세한 지침은 AWS CloudFormation 사용 설명서를 참조하여 스택을 생성, 업데이트 및 삭제하는 방법을 알아봅니다.

Amazon Fraud Detector 스택 구성

AWS CloudFormation 스택을 구성하는 방법은 전적으로 사용자에게 달려 있습니다. 일반적으로 수명 주기 및 소유권별로 스택을 구성하는 것이 가장 좋습니다. 즉, 리소스가 변경되는 빈도 또는 업데이트를 담당하는 팀별로 리소스를 그룹화합니다.

각 감지기 및 감지 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 스택할지 여부를 고려해야 합니다. 예를 들어 데이터를 수집하는 데 도움이 되는 Kinesis 리소스와 데이터를 처리하는 Amazon Fraud Detector 리소스가 포함된 스택을 생성할 수 있습니다. 이는 모든 사기 팀의 제품이 함께 작동하도록 하는 효과적인 방법일 수 있습니다.

Amazon Fraud Detector CloudFormation 파라미터 이해

Amazon Fraud Detector는 모든 CloudFormation 템플릿에서 사용할 수 있는 표준 파라미터 외에도 배포 동작을 관리하는 데 도움이 되는 두 가지 추가 파라미터를 도입합니다. 이러한 파라미터 중 하나 또는 둘 다를 포함하지 않는 경우 CloudFormation은 아래 표시된 기본값을 사용합니다.

파라미터	값	기본 값
DetectorVersionStatus	활성: 새/업데이트된 감지기 버전을 활성 상태로 설정 초안: 새/업데이트된 감지기 버전을 초안 상태로 설정	초안
인라인	TRUE: 스택을 create/update/delete할 때 CloudFormation이 리소스를 creating/updating/deleting 허용합니다. FALSE: CloudFormation이 객체가 존재하는지 검증하도록 허용하지만 객체를 변경하지는 않습니다.	TRUE

Amazon Fraud Detector 리소스용 샘플 AWS CloudFormation 템플릿

다음은 감지기 및 관련 감지기 버전을 관리하기 위한 샘플 AWS CloudFormation YAML 템플릿입니다.

```
# 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'
```

에 대해 자세히 알아보기 AWS CloudFormation

에 대해 자세히 알아보려면 다음 리소스를 AWS CloudFormation참조하세요.

- AWS CloudFormation
- AWS CloudFormation 사용 설명서
- AWS CloudFormation API Reference

- AWS CloudFormation 명령줄 인터페이스 사용 설명서

사기 예측

Amazon Fraud Detector를 사용하여 단일 이벤트에 대한 사기 예측을 실시간으로 가져오거나 일련의 이벤트에 대한 사기 예측을 오프라인으로 가져올 수 있습니다. 단일 이벤트 또는 이벤트 세트에 대한 사기 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 결과와 사기 예측 점수를 제공합니다. 다음 구성 요소를 사용하여 사기 예측 로직을 생성합니다.

- 이벤트 유형 - 이벤트의 구조를 정의합니다.
- 모델 - 사기를 예측하기 위한 알고리즘 및 데이터 요구 사항을 정의합니다.
- 변수 - 이벤트와 연결된 데이터 요소를 나타냅니다.
- 규칙 - 사기 예측 중에 변수 값을 해석하는 방법을 Amazon Fraud Detector에 알립니다.
- 결과 - 사기 예측에서 생성된 결과
- 감지기 버전 - 특정 이벤트에 대한 사기 예측 로직 포함

사기 탐지 로직을 생성하는 데 사용되는 구성 요소에 대한 자세한 내용은 Amazon Fraud Detector 개념을 참조하세요. 사기 예측을 생성하기 전에 사기 예측 로직이 포함된 감지기 버전을 생성하고 게시했는지 확인합니다. 사기 감지기 콘솔 또는 API를 사용하여 감지기 버전을 생성하고 게시할 수 있습니다. 콘솔 사용에 대한 지침은 시작하기(콘솔)를 참조하세요. API 사용에 대한 지침은 감지기 버전 생성을 참조하세요.

이벤트 메타데이터

이벤트 메타데이터는 평가 중인 이벤트의 세부 정보를 제공합니다. 평가하려는 각 이벤트에는 감지기 버전과 연결된 이벤트 유형에 각 변수의 값이 포함되어야 합니다. 또한 이벤트 메타데이터에는 다음이 포함되어야 합니다.

- EVENT_ID - 이벤트의 식별자입니다. 예를 들어 이벤트가 온라인 트랜잭션인 경우 EVENT_ID는 고객에게 제공된 트랜잭션 참조 번호일 수 있습니다.

  EVENT_ID에 대한 중요 참고 사항

  - 해당 이벤트에 대해 고유해야 합니다.
  - 비즈니스에 의미 있는 정보를 나타내야 합니다.
  - 정규식 패턴을 충족해야 합니다. `^[0-9a-z_-]+$.`
  - 저장해야 합니다. EVENT_ID는 이벤트의 참조이며 이벤트 삭제와 같은 이벤트 작업을 수행하는 데 사용됩니다.
  - EVENT_ID에 타임스탬프를 추가하면 나중에 이벤트를 업데이트하려고 할 때 문제가 발생할 수 있으므로 EVENT_ID에 타임스탬프를 추가하는 것은 권장되지 않습니다. 정확히 동일한 EVENT_ID를 제공해야 하기 때문입니다.
- ENTITY_TYPE - 판매자 또는 고객과 같이 이벤트를 수행하는 엔터티입니다.
- ENTITY_ID - 이벤트를 수행하는 개체의 식별자입니다. ENTITY_ID는 정규식 패턴인를 충족해야 합니다`^[0-9a-z_-]+$`. 평가 시 ENTITY_ID를 사용할 수 없는 경우 알 수 없음 문자열을 전달합니다.
- EVENT_TIMESTAMP - 이벤트가 발생한 시점의 타임스탬프입니다. 타임스탬프는 UTC의 ISO 8601 표준이어야 합니다.

실시간 예측

`GetEventPrediction` API를 호출하여 사기에 대한 온라인 활동을 실시간으로 평가할 수 있습니다. 각 요청에서 단일 이벤트에 대한 정보를 제공하고 지정된 감지기와 연결된 사기 예측 로직을 기반으로 모델 점수와 결과를 동기식으로 수신합니다.

실시간 사기 예측 작동 방식

`GetEventPrediction` API는 지정된 감지기 버전을 사용하여 이벤트에 제공된 이벤트 메타데이터를 평가합니다. 평가 중에 Amazon Fraud Detector는 먼저 감지기 버전에 추가된 모델에 대한 모델 점수를 생성한 다음 결과를 평가 규칙에 전달합니다. 규칙은 규칙 실행 모드에 지정된 대로 실행됩니다(감지기 버전 생성 참조). 응답의 일환으로 Amazon Fraud Detector는 모델 점수와 일치하는 규칙과 관련된 결과를 제공합니다.

실시간 사기 예측

실시간 사기 예측을 가져오려면 사기 예측 모델 및 규칙 또는 단순히 규칙 세트가 포함된 탐지기를 생성하고 게시했는지 확인합니다.

AWS 명령줄 인터페이스(AWS CLI) 또는 Amazon Fraud Detector SDK 중 하나를 사용하여 GetEventPrediction API 작업을 호출하여 이벤트에 대한 사기 예측을 실시간으로 얻을 수 있습니다. SDKs

API를 사용하려면 각 요청과 함께 단일 이벤트의 정보를 제공합니다. 요청의 일부로 Amazon Fraud Detector`detectorId`가 이벤트를 평가하는 데 사용하도록 지정해야 합니다. 선택적으로를 지정할 수 있습니다`detectorVersionId`. `detectorVersionId`를 지정하지 않으면 Amazon Fraud Detector는 감지기 `ACTIVE` 버전을 사용합니다.

선택적으로 필드에 데이터를 전달하여 SageMaker AI 모델을 호출하기 위해 데이터를 전송할 수 있습니다`externalModelEndpointBlobs`.

를 사용하여 사기 예측 가져오기 AWS SDK for Python (Boto3)

사기 예측을 생성하려면 `GetEventPrediction` API를 호출합니다. 아래 예제에서는를 완료했다고 가정합니다파트 B: 사기 예측 생성. 응답의 일부로 모델 점수와 일치하는 규칙 및 해당 결과를 받게 됩니다. aws-fraud-detector-samples GitHub 리포지토리에서 `GetEventPrediction` 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'
}
)
```

배치 예측

Amazon Fraud Detector에서 배치 예측 작업을 사용하여 실시간 점수가 필요하지 않은 이벤트 집합에 대한 예측을 가져올 수 있습니다. 예를 들어 배치 예측 작업을 생성하여 오프라인 proof-of-concept 수행하거나 시간별, 일별 또는 주별로 이벤트 위험을 소급 평가할 수 있습니다.

Amazon Fraud Detector 콘솔을 사용하거나 AWS 명령줄 인터페이스(AWS CLI) 또는 Amazon Fraud Detector SDK 중 하나를 사용하여 CreateBatchPredictionJob API 작업을 호출하여 배치 예측 작업을 생성할 수 있습니다. SDKs

주제

배치 예측 작동 방식

`CreateBatchPredictionJob` API 작업은 지정된 감지기 버전을 사용하여 Amazon S3 버킷에 있는 입력 CSV 파일에 제공된 데이터를 기반으로 예측합니다. 그러면 API가 결과 CSV 파일을 S3 버킷에 반환합니다.

배치 예측 작업은 `GetEventPrediction` 작업과 동일한 방식으로 모델 점수 및 예측 결과를 계산합니다. 와 마찬가지로 배치 예측 작업을 생성`GetEventPrediction`하려면 먼저 이벤트 유형을 생성하고, 선택적으로 모델을 훈련한 다음 배치 작업의 이벤트를 평가하는 감지기 버전을 생성합니다.

배치 예측 작업에서 평가한 이벤트 위험 점수의 요금은 `GetEventPrediction` API에서 생성한 점수의 요금과 동일합니다. 자세한 내용은 Amazon Fraud Detector 요금을 참조하세요.

배치 예측 작업은 한 번에 하나만 실행할 수 있습니다.

입력 및 출력 파일

입력 CSV 파일에는 선택한 감지기 버전과 연결된 이벤트 유형과 일치하는 헤더가 포함되어야 합니다. 입력 데이터 파일의 최대 크기는 1GB입니다. 이벤트 수는 이벤트 크기에 따라 다릅니다.

출력 데이터에 별도의 위치를 지정하지 않는 한 Amazon Fraud Detector는 입력 파일과 동일한 버킷에 출력 파일을 생성합니다. 출력 파일에는 입력 파일의 원본 데이터와 다음과 같은 추가된 열이 포함됩니다.

- `MODEL_SCORES` - 선택한 감지기 버전과 연결된 각 모델의 이벤트에 대한 모델 점수를 자세히 설명합니다.
- `OUTCOMES` - 선택한 감지기 버전 및 해당 규칙에 따라 평가된 이벤트 결과를 자세히 설명합니다.
- `STATUS` - 이벤트가 성공적으로 평가되었는지 여부를 나타냅니다. 이벤트가 성공적으로 평가되지 않은 경우이 열에는 실패의 이유 코드가 표시됩니다.
- `RULE_RESULTS` - 규칙 실행 모드에 따라 일치하는 모든 규칙의 목록입니다.

배치 예측 가져오기

다음 단계에서는 이벤트 유형을 이미 생성하고, 해당 이벤트 유형을 사용하여 모델을 훈련시키고(선택 사항), 해당 이벤트 유형에 대한 감지기 버전을 생성했다고 가정합니다.

배치 예측을 가져오려면

1. 에 로그인 AWS Management Console 하고 https://console.aws.amazon.com/frauddetector Amazon Fraud Detector 콘솔을 엽니다.
2. Amazon Fraud Detector 콘솔의 왼쪽 탐색 창에서 배치 예측을 선택한 다음 새 배치 예측을 선택합니다.
3. 작업 이름에서 배치 예측 작업의 이름을 지정합니다. 이름을 지정하지 않으면 Amazon Fraud Detector가 임의로 작업 이름을 생성합니다.
4. 감지기에서이 배치 예측을 위한 감지기를 선택합니다.
5. 감지기 버전에서이 배치 예측을 위한 감지기 버전을 선택합니다. 모든 상태의 감지기 버전을 선택할 수 있습니다. 감지기 버전이 `Active` 상태인 경우 해당 버전이 자동으로 선택되지만 필요한 경우이 선택을 변경할 수도 있습니다.
6. IAM 역할에서 입력 및 출력 Amazon S3 버킷에 대한 읽기 및 쓰기 액세스 권한이 있는 역할을 선택하거나 생성합니다. 자세한 내용은 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 예제를 참조하세요.
7. 입력 데이터 위치에서 입력 데이터의 Amazon S3 위치를 지정합니다. 출력 파일을 다른 S3 버킷에 저장하려면 출력을 위한 데이터 위치 분리를 선택하고 출력 데이터에 대한 Amazon S3 위치를 제공합니다.
8. (선택 사항) 배치 예측 작업에 대한 태그를 생성합니다.
9. 시작을 선택합니다.

   Amazon Fraud Detector는 배치 예측 작업을 생성하고 작업의 상태는 입니다`In progress`. 배치 예측 작업 처리 시간은 이벤트 수와 감지기 버전 구성에 따라 달라집니다.

진행 중인 배치 예측 작업을 중지하려면 배치 예측 작업 세부 정보 페이지로 이동하여 작업을 선택한 다음 배치 예측 중지를 선택합니다. 배치 예측 작업을 중지하면 해당 작업에 대한 결과가 수신되지 않습니다.

배치 예측 작업의 상태가 로 변경되면 지정된 출력 Amazon S3 버킷에서 작업의 출력을 검색할 Complete수 있습니다. 출력 파일의 이름은 형식입니다batch prediction job name_file creation timestamp_output.csv. 예를 들어 라는 작업의 출력 파일은 mybatchjob입니다mybatchjob_ 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 이동하여 작업을 선택한 다음 배치 예측 삭제를 선택합니다.

IAM 역할에 대한 지침

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 예제를 참조하세요. Amazon Fraud Detector 콘솔에는 배치 예측에 대한 IAM 역할을 선택하는 세 가지 옵션이 있습니다.

1. 새 배치 예측 작업을 생성할 때 역할을 생성합니다.
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 보안 주체에 대한 신뢰 정책이 있습니다`frauddetector.amazonaws.com`.

를 사용하여 배치 사기 예측 가져오기 AWS SDK for Python (Boto3)

다음 예제에서는 CreateBatchPredictionJob API에 대한 샘플 요청을 보여줍니다. 배치 예측 작업에는 감지기, 감지기 버전 및 이벤트 유형 이름 등의 기존 리소스가 포함되어야 합니다. 다음 예제에서는 이벤트 유형 `sample_registration`, 감지기 및 감지기 버전 `sample_detector`를 생성했다고 가정합니다1.

```
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-
**'
)
```

예측 설명

예측 설명은 각 이벤트 변수가 모델의 사기 예측 점수에 어떤 영향을 미쳤는지에 대한 통찰력을 제공하며 사기 예측의 일부로 자동으로 생성됩니다. 각 사기 예측에는 1~1000의 위험 점수가 포함됩니다. 예측 설명은 각 이벤트 변수가 위험 점수에 미치는 영향에 대한 세부 정보를 크기(0~5, 5가 가장 높음) 및 방향(점수를 더 높거나 낮게 유도) 측면에서 제공합니다. 다음 작업에 예측 설명을 사용할 수도 있습니다.

- 이벤트가 검토를 위해 플래그 지정될 때 수동 반향 중에 상위 위험 지표를 식별합니다.
- 거짓 긍정 예측으로 이어지는 근본 원인을 좁히려면(예: 합법적인 이벤트에 대한 고위험 점수).
- 이벤트 데이터 전반의 사기 패턴을 분석하고 데이터 세트에서 편향이 있는 경우 이를 탐지합니다.

> **Important**
>
> 예측 설명은 자동으로 생성되며 2021년 6월 30일 이후에 훈련된 모델에 대해서만 사용할 수 있습니다. 2021년 6월 30일 이전에 훈련된 모델에 대한 예측 설명을 받으려면 해당 모델을 재훈련하세요.

예측 설명은 모델을 훈련하는 데 사용된 각 이벤트 변수에 대해 다음과 같은 값 집합을 제공합니다.

상대적 영향

사기 예측 점수에 대한 변수의 영향을 시각적으로 참조합니다. 상대적 영향 값은 사기 위험의 별 등급(0~5, 5가 가장 높음)과 방향(증가/감소) 영향으로 구성됩니다.

- 사기 위험을 높이는 변수는 빨간색 별표로 표시됩니다. 빨간색 별의 수가 많을수록 변수가 사기 점수를 높이고 사기 가능성이 높아집니다.
- 사기 위험을 줄이는 변수는 녹색 별표로 표시됩니다. 녹색 시작 횟수가 많을수록 변수가 사기 위험 점수를 더 많이 낮추고 사기 가능성이 감소합니다.
- 모든 변수에 대해 별이 0이면 그 자체로 사기 위험을 크게 변경한 변수가 없음을 나타냅니다.

원시 설명 값

사기의 로그 오즈로 표시되는 해석되지 않은 원시 값을 제공합니다. 이러한 값은 일반적으로 -10~+10이지만 - 무한대~+ 무한대 범위입니다.

- 양수 값은 변수가 위험 점수를 높였음을 나타냅니다.
- 음수 값은 변수가 위험 점수를 낮추었음을 나타냅니다.

Amazon Fraud Detector 콘솔에서 예측 설명 값은 다음과 같이 표시됩니다. 색상이 지정된 별 등급과 해당 원시 숫자 값을 사용하면 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. AWS SDK의 APIs를 사용하여 예측 설명을 보려면 먼저 `ListEventPrediction` API를 호출하여 이벤트에 대한 예측 타임스탬프를 얻은 다음 `GetEventPredictionMetadata` API를 호출하여 예측 설명을 받아야 합니다.

Amazon Fraud Detector 콘솔을 사용하여 예측 설명 보기

콘솔을 사용하여 예측 설명을 보려면

1. AWS 콘솔을 열고 계정에 로그인합니다. Amazon Fraud Detector로 이동합니다.
2. 왼쪽 탐색 창에서 과거 예측 검색을 선택합니다.
3. 속성, 연산자 및 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 클릭하여 예측 설명을 봅니다.
6. 예측 설명 창까지 아래로 스크롤합니다.
7. 에서 원시 예측 설명 값 표시 버튼을 설정하여 모든 변수의 원시 예측 설명 값을 봅니다.

Python용 AWS SDK(Boto3)를 사용하여 예측 설명 보기

다음 예제에서는 AWS SDK의 `ListEventPredictions` 및 `GetEventPredictionMetadata` APIs를 사용하여 예측 설명을 보기 위한 샘플 요청을 보여줍니다.

예제 1: **`ListEventPredictions`** API를 사용하여 최신 예측 목록 가져오기

```
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'
    }
 )
```

예제 2: **`ListEventPredictions`** API를 사용하여 이벤트 유형 "등록"에 대한 과거 예측 목록 가져오기

```
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'
    }
 )
```

예제 3: **`GetEventPredictionMetadata`** API를 사용하여 지정된 기간에 생성된 지정된 이벤트 ID, 이벤트 유형, 감지기 ID 및 감지기 버전 ID에 대한 과거 예측의 세부 정보를 가져옵니다.

이 요청에 `predictionTimestamp` 지정된는 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 설명 계산 방법 이해

Amazon Fraud Detector는 SHAP(SHapeley Additive exPlanations)를 사용하여 모델 훈련에 사용되는 각 이벤트 변수의 원시 설명 값을 계산하여 개별 이벤트 예측을 설명합니다. 원시 설명 값은 예측을 생성할 때 분류 알고리즘의 일부로 모델에 의해 계산됩니다. 이러한 원시 설명 값은 사기 확률의 로그에 대한 각 입력의 기여도를 나타냅니다. 원시 설명 값(-infinity에서 +infinity로)은 매핑을 사용하여 상대 영향 값(-5~+5)으로 변환됩니다. 원시 설명 값에서 파생된 상대 영향 값은 사기(긍정) 또는 합법(부정)의 확률이 증가하는 횟수를 나타내므로 예측 설명을 더 쉽게 이해할 수 있습니다.

Amazon Fraud Detector의 보안

의 클라우드 보안 AWS 이 최우선 순위입니다. AWS 고객은 보안에 가장 민감한 조직의 요구 사항을 충족하도록 구축된 데이터 센터 및 네트워크 아키텍처의 이점을 누릴 수 있습니다.

보안은 AWS 와 사용자 간의 공동 책임입니다. 공동 책임 모델은 이 사항을 클라우드의 보안 및 클라우드 내 보안으로 설명합니다.

- 클라우드 보안 - AWS 는 AWS 클라우드에서 AWS 서비스를 실행하는 인프라를 보호할 책임이 있습니다. AWS 또한는 안전하게 사용할 수 있는 서비스를 제공합니다. 타사 감사자는 AWS 규정 준수 프로그램 일환으로 보안의 효과를 정기적으로 테스트하고 확인합니다. Amazon Fraud Detector에 적용되는 규정 준수 프로그램에 대한 자세한 내용은 규정 준수 프로그램 제공 범위 내 AWS 서비스규정 준수 프로그램 참조하세요.
- 클라우드의 보안 - 사용자의 책임은 사용하는 AWS 서비스에 따라 결정됩니다. 또한 귀하는 귀사의 데이터 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 목적에 맞게 Amazon Fraud Detector를 구성하는 방법을 보여줍니다. 또한 Amazon Fraud Detector 리소스를 모니터링하고 보호하는 데 도움이 되는 다른 AWS 서비스를 사용하는 방법을 알아봅니다.

주제

Amazon Fraud Detector의 데이터 보호

AWS 공동 책임 모델 Amazon Fraud Detector의 데이터 보호에 적용됩니다. 이 모델에 설명된 대로 AWS 는 모든를 실행하는 글로벌 인프라를 보호할 책임이 있습니다 AWS 클라우드. 사용자는 인프라

에서 호스팅되는 콘텐츠를 관리해야 합니다. 사용하는 AWS 서비스 의 보안 구성과 관리 태스크에 대한 책임도 사용자에게 있습니다. 데이터 프라이버시에 대한 자세한 내용은 데이터 프라이버시 FAQ를 참조하세요. 유럽의 데이터 보호에 대한 자세한 내용은 AWS 보안 블로그의 AWS 공동 책임 모델 및 GDPR 블로그 게시물을 참조하세요.

데이터 보호를 위해 자격 증명을 보호하고 AWS 계정 AWS IAM Identity Center 또는 AWS Identity and Access Management (IAM)를 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 다중 인증(MFA)을 사용하세요.
- SSL/TLS를 사용하여 AWS 리소스와 통신합니다. TLS 1.2는 필수이며 TLS 1.3을 권장합니다.
- 를 사용하여 API 및 사용자 활동 로깅을 설정합니다 AWS CloudTrail. CloudTrail 추적을 사용하여 AWS 활동을 캡처하는 방법에 대한 자세한 내용은 AWS CloudTrail 사용 설명서의 CloudTrail 추적 작업을 참조하세요.
- 내부의 모든 기본 보안 제어와 함께 AWS 암호화 솔루션을 사용합니다 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 통해에 액세스할 때 FIPS 140-3 검증 암호화 모듈이 필요한 경우 FIPS 엔드포인트를 사용합니다. 사용 가능한 FIPS 엔드포인트에 대한 자세한 내용은 Federal Information Processing Standard(FIPS) 140-3을 참조하세요.

고객의 이메일 주소와 같은 기밀 정보나 중요한 정보는 태그나 이름 필드와 같은 자유 형식 텍스트 필드에 입력하지 않는 것이 좋습니다. 여기에는 Amazon Fraud Detector 또는 기타 AWS 서비스 에서 콘솔 AWS CLI, API 또는 AWS SDKs를 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 자격 증명을 URL에 포함해서는 안 됩니다.

저장 데이터 암호화

Amazon Fraud Detector는 사용자가 선택한 암호화 키로 저장 데이터를 암호화합니다. 다음 중 하나를 선택할 수 있습니다.

- AWS 소유 KMS 키입니다. 암호화 키를 지정하지 않으면 기본적으로 이 키로 데이터가 암호화됩니다.

- 고객 관리형 KMS 키입니다. 키 정책을 사용하여 고객 관리형 KMS 키에 대한 액세스를 제어할 수 있습니다. 고객 관리형 KMS 키 생성 및 관리에 대한 자세한 내용은 섹션을 참조하세요키 관리.

전송 중 데이터 암호화

Amazon Fraud Detector는 계정에서 데이터를 복사하여 내부 AWS 시스템에서 처리합니다. 기본적으로 Amazon Fraud Detector는 AWS 인증서와 함께 TLS 1.2를 사용하여 전송 중인 데이터를 암호화합니다.

키 관리

Amazon Fraud Detector는 다음 두 가지 유형의 키 중 하나를 사용하여 데이터를 암호화합니다.

- AWS 소유 KMS 키입니다. 이 값이 기본값입니다.
- 고객 관리형 KMS 키입니다.

고객 관리형 KMS 키 생성

KMS 콘솔 또는 CreateKey API를 사용하여 고객 관리형 AWS KMS 키를 생성할 수 있습니다. 키를 생성할 때 다음을 확인하세요.

- 대칭 암호화 고객 관리형 KMS 키를 선택하면 Amazon Fraud Detector는 비대칭 KMS 키를 지원하지 않습니다. 자세한 내용은 Key Management Service 개발자 안내서의 비대칭 AWS KMS AWS 키를 참조하세요.
- 단일 리전 KMS 키를 생성합니다. Amazon Fraud Detector는 다중 리전 KMS 키를 지원하지 않습니다. 자세한 내용은 Key Management Service 개발자 안내서의 의 다중 리전 AWS KMS AWS 키를 참조하세요.
- Amazon Fraud Detector에 키를 사용할 수 있는 권한을 부여하려면 다음 키 정책을 제공합니다.

```
    {
"Effect": "Allow",
"Principal": {
    "Service": "frauddetector.amazonaws.com"
},
"Action": [
    "kms:Encrypt",
    "kms:Decrypt",
```

```
            "kms:ReEncrypt*",
            "kms:GenerateDataKey*",
            "kms:DescribeKey",
            "kms:CreateGrant",
            "kms:RetireGrant"
        ],
        "Resource": "*"
}
```

키 정책에 대한 자세한 내용은 Key Management Service 개발자 안내서의 AWS KMS에서 키 정책 사용을 참조하세요. AWS

고객 관리형 KMS 키를 사용하여 데이터 암호화

Amazon Fraud Detector의 PutKMSEncryptionKey API를 사용하여 고객 관리형 KMS 키를 사용하여 Amazon Fraud Detector 저장 데이터를 암호화합니다. `PutKMSEncryptionKey` API를 사용하여 언제든지 암호화 구성을 변경할 수 있습니다.

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 변경되면 이전 암호화 구성을 사용하여 암호화된 데이터는 다시 암호화되지 않습니다.

데이터 보기

고객 관리형 KMS 키를 사용하여 Amazon Fraud Detector 데이터를 암호화하는 경우이 방법을 사용하여 암호화된 데이터는 Amazon Fraud Detector 콘솔의 과거 예측 검색 영역에 있는 필터를 사용하여 검색할 수 없습니다. 전체 검색 결과를 얻으려면 다음 속성 중 하나 이상을 사용하여 결과를 필터링합니다.

- 이벤트 ID
- 평가 타임스탬프
- 감지기 상태
- Detector 버전
- 모델 버전

- 모델 유형
- 규칙 평가 상태
- 규칙 실행 모드
- 규칙 일치 상태
- 규칙 버전
- 변수 데이터 소스

고객 관리형 KMS 키가 삭제되었거나 삭제가 예약된 경우 데이터를 사용하지 못할 수 있습니다. 자세한 내용은 KMS 키 삭제를 참조하세요.

Amazon Fraud Detector 및 인터페이스 VPC 엔드포인트(AWS PrivateLink)

인터페이스 VPC 엔드포인트를 생성하여 VPC와 Amazon Fraud Detector 간에 프라이빗 연결을 설정할 수 있습니다. 인터페이스 엔드포인트는 인터넷 게이트웨이AWS PrivateLink, NAT 디바이스, VPN 연결 또는 AWS Direct Connect 연결 없이 Amazon Fraud Detector APIs에 비공개로 액세스할 수 있는 기술로 구동됩니다. VPC의 인스턴스는 Amazon Fraud Detector APIs. VPC와 Amazon Fraud Detector 간의 트래픽은 Amazon 네트워크를 벗어나지 않습니다.

각 인터페이스 엔드포인트는 서브넷에서 하나 이상의 탄력적 네트워크 인터페이스로 표현됩니다.

자세한 내용은 Amazon VPC 사용 설명서에서 인터페이스 VPC 종단점(AWS PrivateLink)을 참조하세요.

Amazon Fraud Detector VPC 엔드포인트에 대한 고려 사항

Amazon Fraud Detector에 대한 인터페이스 VPC 엔드포인트를 설정하기 전에 Amazon VPC 사용 설명서의 인터페이스 엔드포인트 속성 및 제한 사항을 검토해야 합니다.

Amazon Fraud Detector는 VPC에서 모든 API 작업을 호출할 수 있도록 지원합니다.

VPC 엔드포인트 정책은 Amazon Fraud Detector에서 지원됩니다. 기본적으로 엔드포인트를 통해 Amazon Fraud Detector에 대한 전체 액세스가 허용됩니다. 자세한 정보는 Amazon VPC 사용 설명서의 VPC 엔드포인트를 통해 서비스에 대한 액세스 제어를 참조하세요.

Amazon Fraud Detector에 대한 인터페이스 VPC 엔드포인트 생성

Amazon VPC 콘솔 또는 AWS Command Line Interface ()를 사용하여 Amazon Fraud Detector 서비스에 대한 VPC 엔드포인트를 생성할 수 있습니다AWS CLI. 자세한 내용은 Amazon VPC 사용 설명서의 인터페이스 엔드포인트 생성을 참조하세요.

다음 서비스 이름을 사용하여 Amazon Fraud Detector용 VPC 엔드포인트를 생성합니다.

- com.amazonaws.*region*.frauddetector

엔드포인트에 대해 프라이빗 DNS를 활성화하는 경우 리전의 기본 DNS 이름을 사용하여 Amazon Fraud Detector에 API 요청을 할 수 있습니다. 예: `frauddetector.us-east-1.amazonaws.com`.

자세한 내용은 Amazon VPC 사용 설명서의 인터페이스 엔드포인트를 통해 서비스 액세스를 참조하세요.

Amazon Fraud Detector에 대한 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 라는 Amazon Fraud Detector 감지기에 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 Amazon Fraud Detector API 작업

- Amazon Fraud Detector `GetEventPrediction` API 호출

> **Note**
>
> 이 예제에서 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 자격 증명 기반 정책을 참조하세요.

서비스 개선을 위한 데이터 사용 선택 해제

모델을 훈련하고 예측을 생성하기 위해 제공하는 과거 이벤트 데이터는 서비스를 제공하고 유지 관리하는 데에만 사용됩니다. 이 데이터는 Amazon Fraud Detector의 품질을 개선하는 데에도 사용될 수 있습니다. 콘텐츠의 신뢰, 개인 정보 보호 및 보안은 당사의 최우선 사항이며 당사의 사용이 고객에 대한 당사의 약속을 준수하도록 보장합니다. 자세한 내용은 데이터 프라이버시 FAQ를 참조하세요.

AWS Organizations 사용 설명서의 AI 서비스 옵트아웃 정책 페이지를 방문하여 여기에 설명된 프로세스에 따라 Amazon Fraud Detector의 품질을 개발하거나 개선하는 데 이벤트 데이터를 사용하지 않도록 선택할 수 있습니다.

> **Note**
>
> 옵트아웃 정책을 사용하려면 AWS Organizations에서 AWS 계정을 중앙에서 관리해야 합니다. AWS 계정에 대한 조직을 아직 생성하지 않은 경우 조직 생성 및 관리 페이지를 방문하여 여기에 설명된 프로세스를 따르세요.

Amazon Fraud Detector의 자격 증명 및 액세스 관리

AWS Identity and Access Management (IAM)는 관리자가 AWS 리소스에 대한 액세스를 안전하게 제어하는 데 도움이 AWS 서비스 되는 입니다. IAM 관리자는 Amazon Fraud Detector 리소스를 사용할 수 있는 인증(로그인) 및 권한(권한 있음)을 받을 수 있는 사용자를 제어합니다. IAM은 추가 비용 없이 사용할 수 AWS 서비스 있는 입니다.

주제

- 정책을 사용하여 액세스 관리
- Amazon Fraud Detector가 IAM과 작동하는 방식
- Amazon Fraud Detector 자격 증명 기반 정책 예제
- 혼동된 대리자 방지
- Amazon Fraud Detector 자격 증명 및 액세스 문제 해결

대상

AWS Identity and Access Management (IAM)를 사용하는 방법은 Amazon Fraud Detector에서 수행하는 작업에 따라 다릅니다.

서비스 사용자 - Amazon Fraud Detector 서비스를 사용하여 작업을 수행하는 경우 필요한 자격 증명과 권한을 관리자가 제공합니다. 더 많은 Amazon Fraud Detector 기능을 사용하여 작업을 수행하게 되면 추가 권한이 필요할 수 있습니다. 액세스 권한 관리 방법을 이해하면 관리자에게 올바른 권한을 요청하는 데 도움이 됩니다. Amazon Fraud Detector의 기능에 액세스할 수 없는 경우 섹션을 참조하세요Amazon Fraud Detector 자격 증명 및 액세스 문제 해결.

서비스 관리자 - 회사에서 Amazon Fraud Detector 리소스를 책임지고 있는 경우 Amazon Fraud Detector에 대한 전체 액세스 권한을 가지고 있을 것입니다. 서비스 관리자는 서비스 사용자가 액세스해야 하는 Amazon Fraud Detector 기능과 리소스를 결정합니다. 그런 다음 IAM 관리자에게 요청을 제출하여 서비스 사용자의 권한을 변경해야 합니다. 이 페이지의 정보를 검토하여 IAM의 기본 개념을 이해하세요. 회사가 Amazon Fraud Detector에서 IAM을 사용하는 방법에 대한 자세한 내용은 섹션을 참조하세요Amazon Fraud Detector가 IAM과 작동하는 방식.

IAM 관리자 - IAM 관리자라면 Amazon Fraud Detector에 대한 액세스 관리 정책 작성 방법을 자세히 알고 싶을 것입니다. IAM에서 사용할 수 있는 Amazon Fraud Detector 자격 증명 기반 정책 예제를 보려면 섹션을 참조하세요Amazon Fraud Detector 자격 증명 기반 정책 예제.

ID를 통한 인증

인증은 자격 증명 자격 증명을 AWS 사용하여에 로그인하는 방법입니다. IAM 사용자 또는 AWS 계정 루트 사용자 IAM 역할을 수임하여 로 인증(로그인 AWS)되어야 합니다.

자격 증명 소스를 통해 제공된 자격 증명을 사용하여 페더레이션 자격 증명 AWS 으로에 로그인할 수 있습니다. AWS IAM Identity Center (IAM Identity Center) 사용자, 회사의 Single Sign-On 인증 및 Google 또는 Facebook 자격 증명은 페더레이션 자격 증명의 예입니다. 페더레이션형 ID로 로그인할 때 관리자가 이전에 IAM 역할을 사용하여 ID 페더레이션을 설정했습니다. 페더레이션을 사용하여 AWS 에 액세스하면 간접적으로 역할을 수임하게 됩니다.

사용자 유형에 따라 AWS Management Console 또는 AWS 액세스 포털에 로그인할 수 있습니다. 로그인에 대한 자세한 내용은 AWS 로그인 사용 설명서의에 로그인하는 방법을 AWS참조하세요. AWS 계정

AWS 프로그래밍 방식으로에 액세스하는 경우는 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 직접 요청에 서명해야 합니다. 권장 방법을 사용하여 요청에 직접 서명하는 자세한 방법은 IAM 사용 설명서에서 API 요청용AWS Signature Version 4를 참조하세요.

사용하는 인증 방법에 상관없이 추가 보안 정보를 제공해야 할 수도 있습니다. 예를 들어는 멀티 팩터 인증(MFA)을 사용하여 계정의 보안을 강화할 것을 AWS 권장합니다. 자세한 내용은 AWS IAM Identity Center 사용 설명서에서 다중 인증 및 IAM 사용 설명서에서 IAM의AWS 다중 인증을 참조하세요.

AWS 계정 루트 사용자

를 생성할 때 계정의 모든 AWS 서비스 및 리소스에 대한 완전한 액세스 권한이 있는 하나의 로그인 자격 증명으로 AWS 계정시작합니다. 이 자격 증명을 AWS 계정 루트 사용자라고 하며 계정을 생성하는 데 사용한 이메일 주소와 암호로 로그인하여 액세스합니다. 일상적인 작업에 루트 사용자를 사용하지 않을 것을 강력히 권장합니다. 루트 사용자 자격 증명을 보호하고 루트 사용자만 수행할 수 있는 작업을 수행하는 데 사용합니다. 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 권한이 AWS 계정 있는 내의 자격 증명입니다. 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 내용은 IAM 사용 설명서에서 IAM 사용자 사용 사례를 참조하세요.

IAM 역할

IAM 역할은 특정 권한이 AWS 계정 있는 내의 자격 증명입니다. IAM 사용자와 유사하지만, 특정 개인과 연결되지 않습니다. 에서 IAM 역할을 일시적으로 수임하려면 사용자에서 IAM 역할(콘솔)로 전환할 AWS Management Console수 있습니다. https://docs.aws.amazon.com/IAM/latest/UserGuide/id_roles_use_switch-role-console.html 또는 AWS API 작업을 호출하거나 사용자 지정 URL을 AWS CLI 사용하여 역할을 수임할 수 있습니다. 역할 사용 방법에 대한 자세한 내용은 IAM 사용 설명서의 역할 수임 방법을 참조하세요.

임시 보안 인증이 있는 IAM 역할은 다음과 같은 상황에서 유용합니다.

- 페더레이션 사용자 액세스 - 페더레이션 ID에 권한을 부여하려면 역할을 생성하고 해당 역할의 권한을 정의합니다. 페더레이션 ID가 인증되면 역할이 연결되고 역할에 정의된 권한이 부여됩니다. 페더레이션 관련 역할에 대한 자세한 내용은 IAM 사용 설명서의 Create a role for a third-party identity provider (federation)를 참조하세요. IAM Identity Center를 사용하는 경우, 권한 집합을 구성합니다. 인증 후 ID가 액세스할 수 있는 항목을 제어하기 위해 IAM Identity Center는 권한 집합을 IAM의 역할과 연관짓습니다. 권한 집합에 대한 자세한 내용은 AWS IAM Identity Center 사용 설명서의 권한 집합을 참조하세요.
- 임시 IAM 사용자 권한 - IAM 사용자 또는 역할은 IAM 역할을 수임하여 특정 작업에 대한 다양한 권한을 임시로 받을 수 있습니다.
- 교차 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 에서는 정책을 리소스에 직접 연결할 AWS 서비스수 있습니다(역할을 프록시로 사용하는 대신). 교차 계정 액세스에 대한 역할과 리소스 기반 정책의 차이점을 알아보려면 IAM 사용 설명서의 IAM의 교차 계정 리소스 액세스를 참조하세요.
- 교차 서비스 액세스 - 일부는 다른에서 기능을 AWS 서비스 사용합니다 AWS 서비스. 예를 들어, 서비스에서 호출하면 일반적으로 해당 서비스는 Amazon EC2에서 애플리케이션을 실행하거나 Amazon S3에 객체를 저장합니다. 서비스는 직접적으로 호출하는 위탁자의 권한을 사용하거나, 서비스 역할을 사용하거나, 또는 서비스 연결 역할을 사용하여 이 작업을 수행할 수 있습니다.
  - 전달 액세스 세션(FAS) - IAM 사용자 또는 역할을 사용하여에서 작업을 수행하는 경우 AWS보안 주체로 간주됩니다. 일부 서비스를 사용하는 경우, 다른 서비스에서 다른 작업을 시작하는 작업을 수행할 수 있습니다. FAS는를 호출하는 보안 주체의 권한을 다운스트림 서비스에 AWS 서비스 대한 요청과 AWS 서비스함께 사용합니다. FAS 요청은 서비스가 완료하기 위해 다른 AWS 서비스 또는 리소스와의 상호 작용이 필요한 요청을 수신할 때만 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 작업을 수행하기 위해 맡는 IAM 역할입니다. IAM 관리자는 IAM 내에서 서비스 역할을 생성, 수정 및 삭제할 수 있습니다. 자세한 정보는 IAM 사용 설명서의 Create a role to delegate permissions to an AWS 서비스를 참조하세요.
  - 서비스 연결 역할 - 서비스 연결 역할은에 연결된 서비스 역할의 한 유형입니다 AWS 서비스. 서비스는 사용자를 대신하여 작업을 수행하기 위해 역할을 수임할 수 있습니다. 서비스 연결 역할은에 표시 AWS 계정 되며 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 AWS CLI 또는 AWS API 요청을 수행하는 애플리케이션의 임시 자격 증명을 관리할 수 있습니다. 이는 EC2 인스턴스 내에 액세스 키를 저장할 때 권장되는 방법입니다. EC2 인스턴스에 AWS 역할을 할당하고 모든 애플리케이션에서 사용할 수 있도록 하려면 인스턴스에 연결된 인스턴스 프로파일을 생성합니다. 인스턴스 프로필에는 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.

정책을 사용하여 액세스 관리

정책을 AWS 생성하고 자격 증명 또는 리소스에 연결하여 AWS 에서 액세스를 제어합니다. 정책은 자격 증명 또는 리소스와 연결될 AWS 때 권한을 정의하는의 객체입니다.는 보안 주체(사용자, 루트 사용자 또는 역할 세션)가 요청할 때 이러한 정책을 AWS 평가합니다. 정책에서 권한은 요청이 허용되거나 거부되는 지를 결정합니다. 대부분의 정책은에 JSON 문서 AWS 로 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, 사용자 및 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` 작업을 허용하는 정책이 있다고 가정합니다. 해당 정책이 있는 사용자는 AWS Management Console AWS CLI, 또는 API에서 역할 정보를 가져올 수 있습니다 AWS .

ID 기반 정책

ID 기반 정책은 IAM 사용자, 사용자 그룹 또는 역할과 같은 ID에 연결할 수 있는 JSON 권한 정책 문서입니다. 이러한 정책은 사용자 및 역할이 어떤 리소스와 어떤 조건에서 어떤 작업을 수행할 수 있는지

를 제어합니다. 자격 증명 기반 정책을 생성하는 방법을 알아보려면 IAM 사용 설명서에서 고객 관리형 정책으로 사용자 지정 IAM 권한 정의를 참조하세요.

ID 기반 정책은 인라인 정책 또는 관리형 정책으로 한층 더 분류할 수 있습니다. 인라인 정책은 단일 사용자, 그룹 또는 역할에 직접 포함됩니다. 관리형 정책은의 여러 사용자, 그룹 및 역할에 연결할 수 있는 독립 실행형 정책입니다 AWS 계정. 관리형 정책에는 AWS 관리형 정책 및 고객 관리형 정책이 포함됩니다. 관리형 정책 또는 인라인 정책을 선택하는 방법을 알아보려면 IAM 사용 설명서의 관리형 정책 및 인라인 정책 중에서 선택을 참조하세요.

리소스 기반 정책

리소스 기반 정책은 리소스에 연결하는 JSON 정책 설명서입니다. 리소스 기반 정책의 예제는 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 위탁자가 해당 리소스와 어떤 조건에서 어떤 작업을 수행할 수 있는지를 정의합니다. 리소스 기반 정책에서 위탁자를 지정해야 합니다. 보안 주체에는 계정, 사용자, 역할, 페더레이션 사용자 또는이 포함될 수 있습니다 AWS 서비스.

리소스 기반 정책은 해당 서비스에 있는 인라인 정책입니다. 리소스 기반 정책에서는 IAM의 AWS 관리형 정책을 사용할 수 없습니다.

액세스 제어 목록(ACL)

액세스 제어 목록(ACL)은 어떤 보안 주체(계정 멤버, 사용자 또는 역할)가 리소스에 액세스할 수 있는 권한을 가지고 있는지를 제어합니다. ACL은 JSON 정책 문서 형식을 사용하지 않지만 리소스 기반 정책과 유사합니다.

Amazon S3 AWS WAF및 Amazon VPC는 ACLs. ACL에 관한 자세한 내용은 Amazon Simple Storage Service 개발자 가이드의 액세스 제어 목록(ACL) 개요를 참조하세요.

기타 정책 타입

AWS 는 덜 일반적인 추가 정책 유형을 지원합니다. 이러한 정책 타입은 더 일반적인 정책 유형에 따라 사용자에게 부여되는 최대 권한을 설정할 수 있습니다.

- 권한 경계 – 권한 경계는 ID 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 객체의 ID 기반 정책과 그 권한 경계의 교집합입니다. `Principal` 필드에서 사용자나 역할을 지정하는 리소스 기반 정책은 권한 경계를 통해 제한되지 않습니다. 이러한 정책 중 하나에 포

함된 명시적 거부는 허용을 재정의합니다. 권한 경계에 대한 자세한 정보는 IAM 사용 설명서의 IAM 엔티티에 대한 권한 경계를 참조하세요.

- 서비스 제어 정책(SCPs) - SCPs는의 조직 또는 조직 단위(OU)에 대한 최대 권한을 지정하는 JSON 정책입니다 AWS Organizations. AWS Organizations 는 기업이 소유한 여러 AWS 계정 을 그룹화하고 중앙에서 관리하기 위한 서비스입니다. 조직에서 모든 기능을 활성화할 경우, 서비스 제어 정책(SCP)을 임의의 또는 모든 계정에 적용할 수 있습니다. SCP는 각각을 포함하여 멤버 계정의 엔터티에 대한 권한을 제한합니다 AWS 계정 루트 사용자. 조직 및 SCP에 대한 자세한 내용은 AWS Organizations 사용 설명서에서 Service control policies을 참조하세요.
- 리소스 제어 정책(RCP) - RCP는 소유한 각 리소스에 연결된 IAM 정책을 업데이트하지 않고 계정의 리소스에 대해 사용 가능한 최대 권한을 설정하는 데 사용할 수 있는 JSON 정책입니다. RCP는 멤버 계정의 리소스에 대한 권한을 제한하며 조직에 속하는지 여부에 AWS 계정 루트 사용자관계없이를 포함한 자격 증명에 대한 유효 권한에 영향을 미칠 수 있습니다. RCP를 AWS 서비스 지원하는 목록을 포함하여 조직 및 RCPs에 대한 자세한 내용은 AWS Organizations 사용 설명서의 리소스 제어 정책(RCPs)을 참조하세요.
- 세션 정책 – 세션 정책은 역할 또는 페더레이션 사용자에 대해 임시 세션을 프로그래밍 방식으로 생성할 때 파라미터로 전달하는 고급 정책입니다. 결과적으로 얻는 세션의 권한은 사용자 또는 역할의 ID 기반 정책의 교차와 세션 정책입니다. 또한 권한을 리소스 기반 정책에서 가져올 수도 있습니다. 이러한 정책 중 하나에 포함된 명시적 거부는 허용을 재정의합니다. 자세한 정보는 IAM 사용 설명서의 세션 정책을 참조하세요.

여러 정책 유형

여러 정책 유형이 요청에 적용되는 경우, 결과 권한은 이해하기가 더 복잡합니다. 에서 여러 정책 유형이 관련될 때 요청을 허용할지 여부를 AWS 결정하는 방법을 알아보려면 IAM 사용 설명서의 정책 평가 로직을 참조하세요.

Amazon Fraud Detector가 IAM과 작동하는 방식

IAM을 사용하여 Amazon Fraud Detector에 대한 액세스를 관리하기 전에 Amazon Fraud Detector에서 사용할 수 있는 IAM 기능을 이해해야 합니다. Amazon Fraud Detector 및 기타 AWS 서비스에서 IAM을 사용하는 방법을 전체적으로 알아보려면 IAM 사용 설명서의 AWS IAM으로 작업하는 서비스를 참조하세요.

주제

- Amazon Fraud Detector 자격 증명 기반 정책
- Amazon Fraud Detector 리소스 기반 정책

- Amazon 사기 탐지기 태그 기반 권한 부여
- Amazon Fraud Detector IAM 역할

Amazon Fraud Detector 자격 증명 기반 정책

IAM ID 기반 정책을 사용하면 허용되거나 거부되는 작업과 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 작업 및 필요한 권한으로 액세스가 제한된 사용자를 생성하는 것이 좋습니다. 필요하다면 그 밖의 권한을 추가할 수 있습니다. 다음 정책은 Amazon Fraud Detector를 사용하는 데 필요한 권한을 제공합니다. `AmazonFraudDetectorFullAccessPolicy` 및 `AmazonS3FullAccess`. 이러한 정책을 사용하여 Amazon Fraud Detector를 설정하는 방법에 대한 자세한 내용은 섹션을 참조하세요Amazon Fraud Detector 설정.

작업

관리자는 AWS JSON 정책을 사용하여 누가 무엇에 액세스할 수 있는지 지정할 수 있습니다. 즉, 어떤 위탁자가 어떤 리소스와 어떤 조건에서 작업을 수행할 수 있는지를 지정할 수 있습니다.

JSON 정책의 `Action` 요소는 정책에서 액세스를 허용하거나 거부하는 데 사용할 수 있는 작업을 설명합니다. 정책 작업은 일반적으로 연결된 AWS API 작업과 이름이 동일합니다. 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 포함하세요.

Amazon Fraud Detector의 정책 작업은 작업 앞에 접두사를 사용합니다`frauddetector:`. 예를 들어 Amazon Fraud Detector `CreateRule` API 작업으로 규칙을 생성하려면 정책에 `frauddetector:CreateRule` 작업을 포함합니다. 정책 문에는 `Action` 또는 `NotAction` 요소가 포함되어야 합니다. Amazon Fraud Detector는이 서비스로 수행할 수 있는 작업을 설명하는 고유한 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 참조하세요.

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 있는지를 지정할 수 있습니다.

`Resource` JSON 정책 요소는 작업이 적용되는 하나 이상의 객체를 지정합니다. 문에는 `Resource`또는 `NotResource`요소가 반드시 추가되어야 합니다. 모범 사례에 따라 Amazon 리소스 이름(ARN)을 사용하여 리소스를 지정합니다. 리소스 수준 권한이라고 하는 특정 리소스 유형을 지원하는 작업에 대해 이를 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 ARNs 나열됩니다.

예를 들어 문에 `my_detector` 감지기를 지정하려면 다음 ARN을 사용합니다.

```
"Resource": "arn:aws:frauddetector:us-east-1:123456789012:detector/my_detector"
```

ARN 형식에 대한 자세한 내용은 Amazon 리소스 이름(ARNs) 및 AWS 서비스 네임스페이스를 참조하세요.

특정 계정에 속하는 모든 탐지기를 지정하려면 와일드카드(*)를 사용합니다.

```
"Resource": "arn:aws:frauddetector:us-east-1:123456789012:detector/*"
```

리소스를 생성하기 위한 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 ARNs 목록을 보려면 IAM 사용 설명서의 Amazon Fraud Detector에서 정의한 리소스를 참조하세요. 각 리소스의 ARN을 지정할 수 있는 작업을 알아보려면 Amazon Fraud Detector에서 정의한 작업을 참조하세요.

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 있는지를 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` 작업을 사용하여 평가합니다. 단일 조건 키에 여러 값을 지정하는 경우는 논리적 `OR` 작업을 사용하여 조건을 AWS 평가합니다. 문의 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 내용은 IAM 사용 설명서의 IAM 정책 요소: 변수 및 태그를 참조하세요.

AWS 는 전역 조건 키와 서비스별 조건 키를 지원합니다. 모든 AWS 전역 조건 키를 보려면 IAM 사용 설명서의 AWS 전역 조건 컨텍스트 키를 참조하세요.

Amazon Fraud Detector는 자체 조건 키 세트를 정의하고 일부 전역 조건 키 사용을 지원합니다. 모든 AWS 전역 조건 키를 보려면 IAM 사용 설명서의 AWS 전역 조건 컨텍스트 키를 참조하세요.

Amazon Fraud Detector 조건 키 목록을 보려면 IAM 사용 설명서의 Amazon Fraud Detector에 사용되는 조건 키를 참조하세요. 조건 키를 사용할 수 있는 작업과 리소스를 알아보려면 Amazon Fraud Detector에서 정의한 작업을 참조하세요.

예시

Amazon Fraud Detector 자격 증명 기반 정책의 예를 보려면 섹션을 참조하세요Amazon Fraud Detector 자격 증명 기반 정책 예제.

Amazon Fraud Detector 리소스 기반 정책

Amazon Fraud Detector는 리소스 기반 정책을 지원하지 않습니다.

Amazon 사기 탐지기 태그 기반 권한 부여

Amazon Fraud Detector 리소스에 태그를 연결하거나 Amazon Fraud Detector에 대한 요청으로 태그를 전달할 수 있습니다. 태그에 근거하여 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 권한이 있는 엔터티입니다.

Amazon Fraud Detector에서 임시 자격 증명 사용

임시 보안 인증을 사용하여 페더레이션을 통해 로그인하거나, IAM 역할을 맡거나, 교차 계정 역할을 맡을 수 있습니다. AssumeRole 또는 GetFederationToken과 같은 AWS STS API 작업을 호출하여 임시 보안 자격 증명을 얻습니다.

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 자격 증명 기반 정책 예제

기본적으로 사용자 및 IAM 역할에는 Amazon Fraud Detector 리소스를 생성하거나 수정할 수 있는 권한이 없습니다. 또한 AWS Management Console AWS CLI또는 AWS API를 사용하여 작업을 수행할 수 없습니다. 관리자는 지정된 리소스에서 특정 API 태스크를 수행할 수 있는 권한을 사용자와 역할에게 부여하는 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 참조하세요.

주제

정책 모범 사례

자격 증명 기반 정책에 따라 계정에서 사용자가 Amazon Fraud Detector 리소스를 생성, 액세스 또는 삭제할 수 있는지 여부가 결정됩니다. 이 작업으로 인해 AWS 계정에 비용이 발생할 수 있습니다. ID 기반 정책을 생성하거나 편집할 때는 다음 지침과 권장 사항을 따릅니다.

- AWS 관리형 정책을 시작하고 최소 권한으로 전환 - 사용자 및 워크로드에 권한 부여를 시작하려면 많은 일반적인 사용 사례에 대한 권한을 부여하는 AWS 관리형 정책을 사용합니다. 에서 사용할 수 있습니다 AWS 계정. 사용 사례에 맞는 AWS 고객 관리형 정책을 정의하여 권한을 추가로 줄이는 것이 좋습니다. 자세한 정보는 IAM 사용 설명서의 AWS 관리형 정책 또는 AWS 직무에 대한 관리형 정책을 참조하세요.
- 최소 권한 적용 – IAM 정책을 사용하여 권한을 설정하는 경우, 작업을 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 IAM의 정책 및 권한을 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. AWS 서비스와 같은 특정를 통해 사용되는 경우 조건을 사용하여 서비스 작업에 대한 액세스 권한을 부여할 수도 있습니다 AWS CloudFormation. 자세한 정보는 IAM 사용 설명서의 IAM JSON 정책 요소: 조건을 참조하세요.
- IAM Access Analyzer를 통해 IAM 정책을 확인하여 안전하고 기능적인 권한 보장 - IAM Access Analyzer에서는 IAM 정책 언어(JSON)와 모범 사례가 정책에서 준수되도록 새로운 및 기존 정책을 확인합니다. IAM Access Analyzer는 100개 이상의 정책 확인 항목과 실행 가능한 추천을 제공하여 안전하고 기능적인 정책을 작성하도록 돕습니다. 자세한 내용은 IAM 사용 설명서의 IAM Access Analyzer에서 정책 검증을 참조하세요.
- 다중 인증(MFA) 필요 -에서 IAM 사용자 또는 루트 사용자가 필요한 시나리오가 있는 경우 추가 보안을 위해 MFA를 AWS 계정켭니다. API 작업을 직접 호출할 때 MFA가 필요하면 정책에 MFA 조건을 추가합니다. 자세한 내용은 IAM 사용 설명서의 MFA를 통한 보안 API 액세스를 참조하세요.

IAM의 모범 사례에 대한 자세한 내용은 IAM 사용 설명서의 IAM의 보안 모범 사례를 참조하세요.

Amazon Fraud Detector에 대한 AWS 관리형(미리 정의된) 정책

AWS 는에서 생성하고 관리하는 독립 실행형 IAM 정책을 제공하여 많은 일반적인 사용 사례를 처리합니다 AWS. 이러한 AWS 관리형 정책은 일반적인 사용 사례에 필요한 권한을 부여하므로 필요한 권한을 조사할 필요가 없습니다. 자세한 내용은 AWS Identity and Access Management 관리 사용 설명서의 AWS 관리형 정책을 참조하세요.

계정의 사용자에게 연결할 수 있는 다음 AWS 관리형 정책은 Amazon Fraud Detector에만 해당됩니다.

`AmazonFraudDetectorFullAccess`: Amazon Fraud Detector 리소스, 작업 및 다음을 포함한 지원되는 작업에 대한 전체 액세스 권한을 부여합니다.

- Amazon SageMaker AI의 모든 모델 엔드포인트 나열 및 설명
- 계정의 모든 IAM 역할 나열
- 모든 Amazon S3 버킷 나열
- IAM 역할 전달이 Amazon Fraud Detector에 역할을 전달하도록 허용

이 정책은 무제한 S3 액세스를 제공하지 않습니다. 모델 훈련 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. 자체 사용자 지정 IAM 정책을 생성하여 Amazon Fraud Detector 작업 및 리소스에 대한 권한을 필요에 따라 허용할 수도 있습니다. 정책이 필요한 사용자 또는 그룹에 이러한 사용자 지정 정책을 연결할 수 있습니다.

사용자가 자신의 고유한 권한을 볼 수 있도록 허용

이 예제는 IAM 사용자가 자신의 사용자 ID에 연결된 인라인 및 관리형 정책을 볼 수 있도록 허용하는 정책을 생성하는 방법을 보여줍니다. 이 정책에는 콘솔에서 또는 AWS CLI 또는 AWS API를 사용하여 프로그래밍 방식으로이 작업을 완료할 수 있는 권한이 포함됩니다.

```
{
    "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 예제에서는의 사용자에게 모든 Amazon Fraud Detector 리소스 및 작업에 대한 AWS 계정 전체 액세스 권한을 부여합니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "frauddetector:*"
            ],
            "Resource": "*"
        }
    ]
}
```

Amazon Fraud Detector 리소스에 대한 읽기 전용 액세스 허용

이 예제에서는의 사용자에게 Amazon Fraud Detector 리소스에 대한 AWS 계정 읽기 전용 액세스 권한을 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

리소스 수준 정책의이 예제에서는 특정 감지기 리소스 하나를 제외한 모든 작업 및 리소스에 대한 AWS 계정 액세스 권한을의 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}"
        }
    ]
}
```

듀얼 모드 API를 사용할 때 특정 리소스에 대한 액세스 허용

Amazon Fraud Detector는 목록 및 설명 작업으로 모두 작동하는 듀얼 모드 가져오기 APIs 제공합니다. 파라미터 없이 호출될 때 듀얼 모드 API는와 연결된 지정된 리소스 목록을 반환합니다 AWS 계정. 파라미터와 함께 호출될 때 듀얼 모드 API는 지정된 리소스의 세부 정보를 반환합니다. 리소스는 모델, 변수, 이벤트 유형 또는 엔터티 유형일 수 있습니다.

듀얼 모드 APIs는 IAM 정책에서 리소스 수준 권한을 지원합니다. 그러나 리소스 수준 권한은 요청의 일부로 하나 이상의 파라미터가 제공된 경우에만 적용됩니다. 예를 들어 사용자가 GetVariables API를 호출하고 변수 이름을 제공하고 변수 리소스 또는 변수 이름에 연결된 IAM 거부 정책이 있는 경우 사용자에게 `AccessDeniedException` 오류가 발생합니다. 사용자가 `GetVariables` API를 호출하고 변수 이름을 지정하지 않으면 모든 변수가 반환되어 정보 유출이 발생할 수 있습니다.

사용자가 특정 리소스의 세부 정보만 볼 수 있도록 허용하려면 IAM 거부 `NotResource` 정책의 IAM 정책 요소를 사용합니다. IAM 거부 정책에이 정책 요소를 추가한 후 사용자는 `NotResource` 블록에 지정된 리소스의 세부 정보만 볼 수 있습니다. 자세한 내용은 IAM 사용 설명서의 IAM JSON 정책 요소: NotResource를 참조하세요.

다음 예제 정책은 사용자가 모든 Amazon Fraud Detector의 리소스에 액세스할 수 있도록 허용합니다. 그러나 `NotResource` 정책 요소는 GetVariables API 호출을 접두사가 `user*`, `job_*`및 인 변수 이름으로만 제한하는 데 사용됩니다`var*`.

```
{
 "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 다음 동작을 나타냅니다.

- 변수 이름을 포함하지 않는 GetVariables 호출은 요청이 거부 문에 매핑되기 때문에 `AccessDeniedException` 오류가 발생합니다.

- 허용되지 않는 변수 이름이 포함된 GetVariables 호출은 변수 이름이 NotResource 블록의 변수 이름에 매핑되지 않기 때문에 AccessDeniedException 오류가 발생합니다. 예를 들어 변수 이름을 사용하여 GetVariables를 호출email_address하면 AccessDeniedException 오류가 발생합니다.
- NotResource 블록의 변수 이름과 일치하는 변수 이름이 포함된 GetVariables 호출이 예상대로 반환됩니다. 예를 들어 변수 이름을 포함하는 GetVariables 호출은 job_cpa 변수의 세부 정보를 job_cpa 반환합니다.

태그를 기반으로 액세스 제한

이 예제 정책은 리소스 태그를 기반으로 Amazon Fraud Detector에 대한 액세스를 제한하는 방법을 보여줍니다. 이 예제에서는 다음을 가정합니다.

- 에서 Team1과 Team2라는 두 개의 서로 다른 그룹을 정의 AWS 계정 했습니다.
- 감지기 4개를 생성했습니다.
- Team1의 구성원이 2개의 탐지기에서 API를 호출하도록 허용하려고 함
- Team2 구성원이 다른 2개의 탐지기에서 API를 호출하도록 허용하려고 함

API 직접 호출에 대한 액세스를 제어하려면(예)

1. Team1에서 사용하는 감지A기에 키Project와 값이 포함된 태그를 추가합니다.
2. Team2에서 사용하는 감지B기에 키Project와 값이 포함된 태그를 추가합니다.
3. 키 B및 Project 값이 인 태그가 있는 감지기에 대한 액세스를 거부하는 ResourceTag 조건으로 IAM 정책을 생성하고 해당 정책을 Team1에 연결합니다.
4. 키 A및 Project 값이 인 태그가 있는 감지기에 대한 액세스를 거부하는 ResourceTag 조건으로 IAM 정책을 생성하고 해당 정책을 Team2에 연결합니다.

다음은 키Project와 값이 인 태그가 있는 Amazon Fraud Detector 리소스에 대한 특정 작업을 거부하는 정책의 예입니다. 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

혼동된 대리자 문제는 작업을 수행할 권한이 없는 엔터티가 더 많은 권한이 있는 엔터티에 작업을 수행하도록 강요할 때 발생합니다.는 계정의 리소스에 대한 타사(교차 계정이라고 함) 또는 기타 AWS 서비스(교차 서비스라고 함) 액세스를 제공하는 경우 계정을 보호하는 데 도움이 되는 도구를 AWS 제공합니다.

교차 서비스 혼동된 대리자 문제는 한 서비스(호출 서비스)가 다른 서비스(호출 서비스)를 호출할 때 발생할 수 있습니다. 직접적으로 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 보안 주체를 사용하여 모든 서비스에 대한 데이터를 보호하는 데 도움이 되는 정책을 생성할 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 정

책의 sts:AssumeRole 작업을 수행하도록 허용되어야 합니다. 서비스가를 호출하면 서비스 보안 주체가 역할의 권한 정책에서 허용하는 리소스에 액세스하는 데 사용하는 임시 보안 자격 증명 세트를 sts:AssumeRole AWS STS 반환합니다.

교차 서비스 혼동된 대리자 문제를 방지하기 위해 Amazon Fraud Detector는 역할 신뢰 정책의 aws:SourceArn 및 aws:SourceAccount 전역 조건 컨텍스트 키를 사용하여 역할에 대한 액세스를 예상 리소스에서 생성된 요청으로만 제한할 것을 권장합니다.

는 계정 ID를 aws:SourceAccount 지정하고는 교차 서비스 액세스와 연결된 리소스의 ARN을 aws:SourceArn 지정합니다. 는 ARN 형식을 사용하여 지정해야 aws:SourceArn 합니다. 동일한 정책 문에서 사용할 때 aws:SourceAccount 및 aws:SourceArn가 모두 동일한 계정 ID를 사용하고 있는지 확인합니다.

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 전역 컨텍스트 조건 키를 사용합니다. 예를 들어 arn:aws:*servicename*:*:*123456789012*:*입니다. 권한 정책에 사용할 수 있는 Amazon Fraud Detector 리소스 및 작업에 대한 자세한 내용은 Amazon Fraud Detector에 사용되는 작업, 리소스 및 조건 키를 참조하세요.

다음 역할 신뢰 정책 예제에서는 aws:SourceArn 조건 키에서 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 Detector가 `external-model` 리소스에만 액세스할 수 있도록 허용합니다. 조건 블록의 `aws:SourceArn` 파라미터를 확인합니다. 리소스 한정자는 `PutExternalModel` API 호출을 위해 제공되는 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 자격 증명 및 액세스 문제 해결

다음 정보를 사용하여 Amazon Fraud Detector 및 IAM으로 작업할 때 발생할 수 있는 일반적인 문제를 진단하고 수정할 수 있습니다.

주제

- Amazon Fraud Detector에서 작업을 수행할 권한이 없음

- iam:PassRole을 수행하도록 인증되지 않음
- 내 AWS 계정 외부의 사람이 내 Amazon Fraud Detector 리소스에 액세스하도록 허용하고 싶습니다.
- Amazon Fraud Detector가 지정된 역할을 수임할 수 없음

Amazon Fraud Detector에서 작업을 수행할 권한이 없음

에서 작업을 수행할 권한이 없다는 AWS Management Console 메시지가 표시되면 관리자에게 문의하여 도움을 받아야 합니다. 관리자는 로그인 보안 인증 정보를 제공한 사람입니다.

다음 예제 오류는 `mateojackson` 사용자가 콘솔을 사용하여 *##*기에 대한 세부 정보를 보려고 하지만 `frauddetector:`*`GetDetectors`* 권한이 없는 경우에 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.

iam:PassRole을 수행하도록 인증되지 않음

`iam:PassRole` 작업을 수행할 권한이 없다는 오류가 수신되면 Amazon Fraud Detector에 역할을 전달할 수 있도록 정책을 업데이트해야 합니다.

일부 AWS 서비스 에서는 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 라는 IAM 사용자가 콘솔을 사용하여 Amazon Fraud Detector에서 작업을 수행하려고 `marymajor` 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 문의하세요. 관리자는 로그인 자격 증명을 제공한 사람입니다.

내 AWS 계정 외부의 사람이 내 Amazon Fraud Detector 리소스에 액세스하도록 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 여부를 알아보려면 섹션을 참조하세요Amazon Fraud Detector가 IAM과 작동하는 방식.
- 소유 AWS 계정 한의 리소스에 대한 액세스 권한을 제공하는 방법을 알아보려면 IAM 사용 설명서의 소유한 다른의 IAM 사용자에게 액세스 권한 제공을 참조 AWS 계정 하세요.
- 타사에 리소스에 대한 액세스 권한을 제공하는 방법을 알아보려면 IAM 사용 설명서의 타사가 AWS 계정 소유한에 대한 액세스 권한 제공을 AWS 계정참조하세요.
- ID 페더레이션을 통해 액세스 권한을 제공하는 방법을 알아보려면 IAM 사용 설명서의 외부에서 인증된 사용자에게 액세스 권한 제공(ID 페더레이션)을 참조하세요.
- 크로스 계정 액세스에 대한 역할과 리소스 기반 정책 사용의 차이점을 알아보려면 IAM 사용 설명서의 IAM의 크로스 계정 리소스 액세스를 참조하세요.

Amazon Fraud Detector가 지정된 역할을 수임할 수 없음

Amazon Fraud Detector가 지정된 역할을 수임할 수 없다는 오류가 수신되면 지정된 역할에 대한 신뢰 관계를 업데이트해야 합니다. Amazon Fraud Detector를 신뢰할 수 있는 엔터티로 지정하면 서비스가 역할을 수임할 수 있습니다. Amazon Fraud Detector를 사용하여 역할을 생성하면이 신뢰 관계가 자동으로 설정됩니다. Amazon Fraud Detector에서 생성하지 않은 IAM 역할에 대해서만이 신뢰 관계를 설정하면 됩니다.

Amazon Fraud Detector에 대한 기존 역할에 대한 신뢰 관계를 설정하려면

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 Detector의 로깅 및 모니터링

AWS는 Amazon Fraud Detector를 모니터링하고, 이상이 있을 때 보고하고, 적절한 경우 자동 조치를 취할 수 있는 다음과 같은 모니터링 도구를 제공합니다.

- Amazon CloudWatch는 AWS 리소스와 AWS 에서 실행하는 애플리케이션을 실시간으로 모니터링합니다. CloudWatch에 대한 자세한 내용은 Amazon CloudWatch 사용 설명서를 참조하세요.
- AWS CloudTrail는 AWS 계정에서 또는 계정을 대신하여 수행한 API 호출 및 관련 이벤트를 캡처하고 사용자가 지정한 Amazon S3 버킷에 로그 파일을 전송합니다. CloudTrail에 대한 자세한 내용은 AWS CloudTrail 사용 설명서를 참조하세요.

Amazon Fraud Detector 모니터링에 대한 자세한 내용은 섹션을 참조하세요Amazon Fraud Detector 모니터링.

Amazon Fraud Detector에 대한 규정 준수 검증

타사 감사자는 SOC, PCI, FedRAMP 및 HIPAA와 같은 여러 규정 준수 프로그램의 일환으로 AWS 서비스의 보안 및 AWS 규정 준수를 평가합니다.

AWS 서비스 가 특정 규정 준수 프로그램의 범위 내에 있는지 알아보려면 AWS 서비스 규정 준수 프로그램 제공 범위 섹션을 참조하고 관심 있는 규정 준수 프로그램을 선택합니다. 일반 정보는 AWS 규정 준수 프로그램.

를 사용하여 타사 감사 보고서를 다운로드할 수 있습니다 AWS Artifact. 자세한 내용은 에서 보고서 다운로드 AWS Artifact에서 .

사용 시 규정 준수 책임은 데이터의 민감도, 회사의 규정 준수 목표 및 관련 법률과 규정에 따라 AWS 서비스 결정됩니다.는 규정 준수를 지원하기 위해 다음 리소스를 AWS 제공합니다.

- 보안 규정 준수 및 거버넌스 - 이러한 솔루션 구현 가이드에서는 아키텍처 고려 사항을 설명하고 보안 및 규정 준수 기능을 배포하는 단계를 제공합니다.
- HIPAA 적격 서비스 참조 - HIPAA 적격 서비스가 나열되어 있습니다. 모두 HIPAA 자격이 AWS 서비스 있는 것은 아닙니다.
- AWS 규정 준수 리소스 -이 워크북 및 가이드 모음은 산업 및 위치에 적용될 수 있습니다.
- AWS 고객 규정 준수 가이드 - 규정 준수의 관점에서 공동 책임 모델을 이해합니다. 이 가이드에는 여러 프레임워크(미국 국립표준기술연구소(NIST), 결제 카드 산업 보안 표준 위원회(PCI), 국제표준화기구(ISO) 포함)의 보안 제어에 대한 지침을 보호하고 AWS 서비스 매핑하는 모범 사례가 요약되어 있습니다.
- 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 - 내 보안 상태에 대한 포괄적인 보기를 AWS 서비스 제공합니다 AWS. Security Hub는 보안 컨트롤을 사용하여 AWS 리소스를 평가하고 보안 업계 표준 및 모범 사례에 대한 규정 준수를 확인합니다. 지원되는 서비스 및 제어 목록은 Security Hub 제어 참조를 참조하세요.
- Amazon GuardDuty - 의심스러운 활동과 악의적인 활동이 있는지 환경을 모니터링하여 사용자, AWS 계정워크로드, 컨테이너 및 데이터에 대한 잠재적 위협을 AWS 서비스 탐지합니다. GuardDuty는 특정 규정 준수 프레임워크에서 요구하는 침입 탐지 요구 사항을 충족하여 PCI DSS와 같은 다양한 규정 준수 요구 사항을 따르는 데 도움을 줄 수 있습니다.
- AWS Audit Manager - 이를 AWS 서비스 통해 AWS 사용량을 지속적으로 감사하여 위험과 규정 및 업계 표준 준수를 관리하는 방법을 간소화할 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 보안으로 보호됩니다. AWS 보안 서비스 및가 인프라를 AWS 보호하는 방법에 대한 자세한 내용은 AWS 클라우드 보안을 참조하세요. 인프라 보안 모범 사례를 사용하여 AWS 환경을 설계하려면 보안 원칙 AWS Well-Architected Framework의 인프라 보호를 참조하세요.

AWS 에서 게시한 API 호출을 사용하여 네트워크를 통해 Amazon Fraud Detector에 액세스합니다. 고객은 다음을 지원해야 합니다.

- Transport Layer Security(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 위탁자와 관련된 보안 암호 액세스 키를 사용하여 서명해야 합니다. 또는 AWS Security Token Service(AWS STS)를 사용하여 임시 자격 증명을 생성하여 요청에 서명할 수 있습니다.

Amazon Fraud Detector 모니터링

모니터링은 Amazon Fraud Detector 및 기타 AWS 솔루션의 안정성, 가용성 및 성능을 유지하는 데 중요한 부분입니다. AWS는 Amazon Fraud Detector를 모니터링하고, 이상이 있을 때 보고하고, 적절한 경우 자동 조치를 취할 수 있는 다음과 같은 모니터링 도구를 제공합니다.

- Amazon CloudWatch는 AWS 리소스와 AWS 실행 중인 애플리케이션을 실시간으로 모니터링합니다. 지표를 수집 및 추적하고, 사용자 지정 대시보드를 생성할 수 있으며, 지정된 지표가 지정한 임곗값에 도달하면 사용자에게 알리거나 조치를 취하도록 경보를 설정할 수 있습니다. 자세한 내용은 Amazon CloudWatch 사용 설명서를 참조하십시오.
- AWS CloudTrail는 AWS 계정에 의해 또는 계정을 대신하여 수행된 API 호출 및 관련 이벤트를 캡처하고 사용자가 지정한 Amazon S3 버킷에 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

- Amazon CloudWatch를 사용하여 Amazon Fraud Detector 모니터링
- 를 사용하여 Amazon Fraud Detector API 호출 로깅 AWS CloudTrail

Amazon CloudWatch를 사용하여 Amazon Fraud Detector 모니터링

원시 데이터를 수집하여 읽기 가능하며 실시간에 가까운 지표로 처리하는 CloudWatch를 사용하여 Amazon Fraud Detector를 모니터링할 수 있습니다. 이러한 통계는 15개월간 보관되므로 기록 정보에 액세스하고 웹 애플리케이션 또는 서비스가 어떻게 실행되고 있는지 전체적으로 더 잘 파악할 수 있습니다. 특정 임곗값을 주시하다가 해당 임곗값이 충족될 때 알림을 전송하거나 조치를 취하도록 경보를 설정할 수도 있습니다. 자세한 내용은 Amazon CloudWatch 사용 설명서를 참조하세요.

주제

- Amazon Fraud Detector에 CloudWatch 지표 사용.
- Amazon 사기 탐지기 지표

Amazon Fraud Detector에 CloudWatch 지표 사용.

측정치를 사용하려면 다음 정보를 지정해야 합니다.

- 지표 네임스페이스입니다. 네임스페이스는 Amazon Fraud Detector가 지표를에 게시하는 데 사용하는 CloudWatch 컨테이너입니다. CloudWatch ListMetrics API 또는 list-metrics 명령을 사용하여 Amazon Fraud Detector에 대한 지표를 보는 경우 네임스페이스`AWS/FraudDetector`에를 지정합니다.
- 지표 측정기준. 차원은 지표를 고유하게 식별하는 데 도움이 되는 이름-값 쌍입니다. 예를 들어, `DetectorId`가 차원 이름일 수 있습니다. 지표 차원을 지정하는 것은 선택 사항입니다.
- `GetEventPrediction`와 같은 지표 이름.

AWS Management Console AWS CLI, 또는 CloudWatch API를 사용하여 Amazon Fraud Detector에 대한 모니터링 데이터를 가져올 수 있습니다. 또한 Amazon AWS 소프트웨어 개발 키트(SDK) 또는 CloudWatch API 도구 중 하나를 통해 CloudWatch API를 사용할 수 있습니다. 콘솔에는 CloudWatch API의 원시 데이터를 기초로 하는 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 합니까?	`GetEventPrediction` 지표를 모니터링합니다.
`GetEventPrediction` 오류를 모니터링하려면 어떻게 해야 합니까?	`GetEventPrediction5xxError` 및 `GetEventPrediction4xxError` 지표를 사용합니다.
`GetEventPrediction` 호출의 지연 시간은 어떻게 모니터링할 수 있습니까?	`GetEventPredictionLatency` 측정치를 사용합니다.

CloudWatch를 사용하여 Amazon Fraud Detector를 모니터링하려면 적절한 CloudWatch 권한이 있어야 합니다. 자세한 내용은 Amazon CloudWatch에 대한 인증 및 액세스 제어를 참조하세요.

Amazon Fraud Detector 지표에 액세스

다음 단계에서는 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 사기 탐지기를 선택합니다.
3. 지표 차원을 선택합니다.
4. 목록에서 원하는 지표를 선택하고 그래프의 기간을 선택합니다.

경보 만들기

경보 때문에 상태가 변경되면 Amazon Simple Notification Service(SNS) 메시지를 보내는 CloudWatch 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 대해서만 작업을 호출합니다. CloudWatch 경보는 특정 상태에 있다고 해서 작업을 호출하지는 않습니다. 상태가 변경되어 지정된 기간 동안 유지되어야 합니다.

경보를 설정하려면(콘솔)

1. 에 로그인 AWS Management Console 하고 https://console.aws.amazon.com/cloudwatch/ CloudWatch 콘솔을 엽니다.
2. 탐색 창에서 경보를 선택한 다음 경보 생성을 선택합니다. 그러면 경보 생성 마법사가 열립니다.
3. 지표 선택을 선택하세요.
4. 모든 지표 탭에서 사기 탐지기를 선택합니다.
5. 감지기 ID별을 선택한 다음 GetEventPrediction 지표를 선택합니다.
6. 그래프로 표시된 지표 탭을 선택합니다.
7. Statistic(통계)에서 Sum(합계)를 선택합니다.
8. 지표 선택을 선택하세요.
9. 조건에는 임곗값 유형으로 정적을 선택하고 항상...으로 크게를 선택한 다음 선택한 최대값을 입력합니다. 다음을 선택합니다.
10. 기존 SNS 주제에 경보를 전송하려면 다음 주소로 알림 전송:에서 기존 SNS 주제를 선택합니다. 새 이메일 구독 목록에 대한 이름 및 이메일 주소를 설정하려면, 새 목록을 선택합니다. CloudWatch는 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 전에이 경보 상태 변경이 발생하면 의도한 수신자는 알림을 받지 않습니다.

11. 다음을 선택합니다. 경보에 이름과 설명(선택 사항)을 추가합니다. 다음을 선택합니다.
12. 경보 생성을 선택합니다.

Amazon 사기 탐지기 지표

Amazon Fraud Detector는 CloudWatch에 다음 지표를 전송합니다. 모든 지표는 `Average`, , `Minimum`, 통계`Maximum`를 지원합니다`Sum`.

지표	설명
`GetEventPrediction`	GetEventPrediction API 요청 수입니다. 유효한 차원: `DetectorID`
`GetEventPredictionLatency`	GetEventPrediction 요청의 클라이언트 요청에 응답하는 데 걸리는 시간 간격입니다. 유효한 차원: `DetectorID` 단위: 밀리초
`GetEventPrediction4XXError`	Amazon Fraud Detector가 4xx HTTP 응답 코드를 반환한 GetEventPrediction 요청 수입니다. 각 4xx 응답에 대해 1이 전송됩니다. 유효한 차원: `DetectorID`
`GetEventPrediction5XXError`	Amazon Fraud Detector가 5xx HTTP 응답 코드를 반환한 GetEventPrediction 요청 수입니다. 각 5xx 응답에 대해 1이 전송됩니다.

지표	설명
	유효한 차원: DetectorID
Prediction	예측 수입니다. 성공하면 1이 전송됩니다. 유효한 차원: DetectorID , DetectorVersionID
PredictionLatency	예측 작업에 소요된 시간 간격입니다. 유효한 차원: DetectorID , DetectorVersionID 단위: 밀리초
PredictionError	Amazon Fraud Detector에서 오류가 발생한 예측 수입니다. 오류가 발생하면 1이 전송됩니다. 유효한 차원: DetectorID , DetectorVersionID
VariableUsed	변수가 평가의 일부로 사용된 GetEventPrediction 요청 수입니다. 유효한 차원: DetectorID , DetectorVersionID , VariableName
VariableDefaultReturned	변수가 이벤트 속성의 일부로 존재하지 않아 변수의 기본값이 평가 중에 사용된 GetEventPrediction 요청 수입니다. 유효한 차원: DetectorID , DetectorVersionID , VariableName
RuleNotEvaluated	이전 규칙이 일치하여 규칙이 평가되지 않은 GetEventPrediction 요청 수입니다. 유효한 차원: DetectorID , DetectorVersionID , RuleID

지표	설명
RuleEvaluateTrue	규칙이 True로 트리거되고 규칙 결과가 반환된 GetEventPrediction 요청 수입니다. 유효한 차원: DetectorID , DetectorVersionID , RuleID
RuleEvaluateFalse	규칙이 False로 평가된 GetEventPrediction 요청 수입니다. 유효한 차원: DetectorID , DetectorVersionID , RuleID
RuleEvaluateError	규칙이 오류로 평가되는 GetEventPrediction 요청 수 유효한 차원: DetectorID , DetectorVersionID , RuleID
OutcomeReturned	지정된 결과가 반환된 GetEventPrediction 호출 수입니다. 유효한 차원: DetectorID , DetectorVersionID , OutcomeName
ModelInvocation (Amazon SageMaker model endpoint)	평가의 일부로 SageMaker 모델 엔드포인트가 호출된 GetEventPrediction 요청 수입니다. 유효한 차원: DetectorID , DetectorVersionID , ModelEndpoint
ModelInvocationError (Amazon SageMaker model endpoint)	간접 호출된 SageMaker 모델 엔드포인트가 평가 중에 오류를 반환한 GetEventPrediction 요청 수입니다. 유효한 차원: DetectorID , DetectorVersionID , ModelEndpoint

지표	설명
`ModelInvocationLatency (Amazon SageMaker model endpoint)`	Amazon Fraud Detector에서 볼 때 가져온 모델이 응답하는 데 걸리는 시간 간격입니다. 이 간격에는 모델 호출만 포함됩니다. 유효한 차원: `DetectorID` , `DetectorVersionID` , `ModelEndpoint` 단위: 밀리초
`ModelInvocation`	평가의 일부로 모델이 호출된 GetEventPrediction 요청 수입니다. 유효한 차원: `DetectorID` , `DetectorVersionID` , `ModelType` , `ModelID`
`ModelInvocationError`	Amazon Fraud Detector 모델이 평가 중에 오류를 반환한 GetEventPrediction 요청 수입니다. 유효한 차원: `DetectorID` , `DetectorVersionID` , `ModelType` , `ModelID`
`ModelInvocationLatency`	Amazon Fraud Detector 모델이 Amazon Fraud Detector에서 볼 때 응답하는 데 걸리는 시간 간격입니다. 이 간격에는 모델 호출만 포함됩니다. 유효한 차원: `DetectorID` , `DetectorVersionID` , `ModelType` , `ModelID` 단위: 밀리초

를 사용하여 Amazon Fraud Detector API 호출 로깅 AWS CloudTrail

Amazon Fraud Detector는 Amazon Fraud Detector에서 사용자 AWS CloudTrail, 역할 또는 서비스가 수행한 작업에 대한 레코드를 제공하는 AWS 서비스와 통합됩니다. CloudTrail은 Amazon Fraud Detector 콘솔의 호출 및 Amazon Fraud Detector API에 대한 코드의 호출을 포함하여 Amazon Fraud Detector에 대한 모든 APIs.

추적을 생성하면 Amazon Fraud Detector 이벤트를 포함하여 CloudTrail 이벤트를 지속적으로 Amazon S3 버킷에 배포할 수 있습니다. Amazon S3 트레일을 구성하지 않은 경우에도 CloudTrail 콘솔의 이벤트 기록에서 최신 이벤트를 볼 수 있습니다. CloudTrail에서 수집한 정보를 사용하여 Amazon Fraud Detector에 수행된 요청, 요청이 수행된 IP 주소, 요청을 수행한 사람, 요청이 수행된 시간 및 추가 세부 정보를 확인할 수 있습니다.

CloudTrail에 대한 자세한 내용은 AWS CloudTrail 사용 설명서를 참조하세요.

CloudTrail의 Amazon 사기 탐지기 정보

AWS 계정을 생성할 때 계정에서 CloudTrail이 활성화됩니다. Amazon Fraud Detector에서 활동이 발생하면 해당 활동이 이벤트 기록의 다른 AWS 서비스 이벤트와 함께 CloudTrail 이벤트에 기록됩니다. AWS 계정에서 최근 이벤트를 보고 검색하고 다운로드할 수 있습니다. 자세한 설명은 CloudTrail 이벤트 기록으로 이벤트 보기를 참조하세요.

Amazon Fraud Detector에 대한 이벤트를 포함하여 AWS 계정에 이벤트를 지속적으로 기록하려면 추적을 생성합니다. CloudTrail은 추적을 사용하여 Amazon S3 버킷으로 로그 파일을 전송할 수 있습니다. 콘솔에서 추적을 생성하면 기본적으로 모든 리전에 추적이 적용됩니다. 추적은 AWS 파티션의 모든 리전에서 이벤트를 로깅하고 지정한 Amazon S3 버킷으로 로그 파일을 전송합니다. 또는 CloudTrail 로그에서 수집된 이벤트 데이터를 추가 분석 및 처리하도록 다른 AWS 서비스를 구성할 수 있습니다. 자세한 내용은 다음 자료를 참조하세요.

- 트레일 생성 개요
- CloudTrail 지원 서비스 및 통합
- CloudTrail에서 Amazon SNS 알림 구성
- 여러 리전으로부터 CloudTrail 로그 파일 받기 및 여러 계정으로부터 CloudTrail 로그 파일 받기

Amazon Fraud Detector는 모든 작업(API 작업)을 CloudTrail 로그 파일에 이벤트로 로깅할 수 있도록 지원합니다. 자세한 내용은 작업을 참조하십시오.

모든 이벤트 또는 로그 항목에는 요청을 생성했던 사용자에 대한 정보가 포함됩니다. 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 여부입니다.

자세한 설명은 CloudTrail userIdentity 요소를 참조하세요.

Amazon Fraud Detector 로그 파일 항목 이해

추적이란 지정한 Amazon S3 버킷에 이벤트를 로그 파일로 입력할 수 있게 하는 구성입니다. CloudTrail 로그 파일에는 하나 이상의 로그 항목이 포함될 수 있습니다. 이벤트는 모든 소스로부터의 단일 요청을 나타내며 요청 작업, 작업 날짜와 시간, 요청 파라미터 등에 대한 정보가 들어 있습니다. CloudTrail 로그 파일은 퍼블릭 API 직접 호출의 주문 스택 트레이스가 아니므로 특정 순서로 표시되지 않습니다.

다음 예는 `GetDetectors`작업을 보여주는 CloudTrail 로그 항목입니다.

```
{
    "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 섹션에서는 Amazon Fraud Detector로 작업할 때 발생할 수 있는 문제를 해결하는 데 도움이 됩니다.

훈련 데이터 문제 해결

이 섹션의 정보를 사용하여 모델을 훈련할 때 Amazon Fraud Detector 콘솔의 모델 훈련 진단 창에 표시될 수 있는 문제를 진단하고 해결할 수 있습니다.

모델 훈련 진단 창에 표시되는 문제는 다음과 같이 분류됩니다. 문제를 해결하기 위한 요구 사항은 문제의 범주에 따라 다릅니다.

- 오류 - 모델 훈련이 실패합니다. 모델이 성공적으로 훈련하려면 이러한 문제를 해결해야 합니다.
- 경고 -는 모델 훈련을 계속하게 하지만 훈련 프로세스에서 일부 변수가 제외될 수 있습니다. 데이터 세트의 품질을 개선하려면이 단원의 관련 지침을 확인하세요.
- 정보(Info) -는 모델 훈련에 영향을 주지 않으며 모든 변수가 훈련에 사용됩니다. 데이터 세트의 품질과 모델 성능을 더욱 개선하려면이 섹션의 관련 지침을 확인하는 것이 좋습니다.

주제

- 고유 개체 부족

지정된 데이터 세트의 불안정한 사기 비율

문제 유형: 오류

설명

지정된 데이터의 사기율이 시간이 지나도 너무 불안정합니다. 사기 및 합법적인 이벤트는 시간이 지남에 따라 균일하게 샘플링되어야 합니다.

원인

이 오류는 데이터 세트의 사기 및 합법적인 이벤트가 균등하지 않게 배포되고 서로 다른 시간대에서 발생하는 경우에 발생합니다. Amazon Fraud Detector 모델 훈련은 EVENT_TIMESTAMP를 기반으로 데이터 세트를 샘플링하고 분할합니다. 예를 들어 데이터 세트가 지난 6개월에서 가져온 사기 이벤트로 구성되지만 합법적인 이벤트의 마지막 달만 포함된 경우 데이터 세트는 불안정한 것으로 간주됩니다. 불안정한 데이터 세트는 모델 성능 평가에 편향을 초래할 수 있습니다.

솔루션

동일한 시간대의 사기 및 합법적인 이벤트 데이터를 제공해야 하며 사기 비율은 시간이 지남에 따라 크게 변경되지 않습니다.

데이터 부족

1. 문제 유형: 오류

   설명

   50개 미만의 행에는 사기 이벤트로 레이블이 지정됩니다. 사기 이벤트와 합법적인 이벤트가 모두 최소 수인 50개를 초과하는지 확인하고 모델을 재훈련합니다.

   원인

   이 오류는 데이터 세트에 모델 훈련에 필요한 것보다 사기로 레이블이 지정된 이벤트가 적은 경우에 발생합니다. Amazon Fraud Detector는 모델을 훈련하기 위해 최소 50개의 사기 이벤트가 필요합니다.

   솔루션

데이터 세트에 최소 50개의 사기 이벤트가 포함되어 있는지 확인합니다. 필요한 경우 더 긴 기간을 포함시켜 이를 보장할 수 있습니다.

2. 문제 유형: 오류

   설명

   50개 미만의 행에는 합법적인 이벤트로 레이블이 지정됩니다. 사기 및 합법적인 이벤트가 최소 $threshold 수를 초과하는지 확인하고 모델을 재훈련합니다.

   원인

   이 오류는 데이터 세트에 모델 훈련에 필요한 것보다 합법적인 것으로 레이블이 지정된 이벤트가 적은 경우에 발생합니다. Amazon Fraud Detector는 모델을 훈련하기 위해 최소 50개의 합법적인 이벤트가 필요합니다.

   솔루션

   데이터 세트에 최소 50개의 합법적인 이벤트가 포함되어 있는지 확인합니다. 필요한 경우 더 긴 기간을 포함시켜 이를 보장할 수 있습니다.

3. 문제 유형: 오류

   설명

   사기와 관련된 고유 엔터티의 수가 100개 미만입니다. 성능을 개선하기 위해 사기 엔터티의 더 많은 예를 포함하는 것이 좋습니다.

   원인

   이 오류는 데이터 세트에 모델 훈련에 필요한 것보다 사기 이벤트가 있는 개체가 적은 경우에 발생합니다. TFI(Transaction Fraud Insights) 모델은 사기 이벤트가 있는 최소 100개의 엔터티가 사기 공간을 최대한 포함하도록 요구합니다. 소규모 엔터티 그룹에서 모든 사기 이벤트를 수행하는 경우 모델이 제대로 일반화되지 않을 수 있습니다.

   솔루션

   데이터 세트에 사기 이벤트가 있는 엔터티가 100개 이상 포함되어 있는지 확인합니다. 필요한 경우 이 기간이 더 긴지 확인할 수 있습니다.

4. 문제 유형: 오류

   설명

합법적인와 연결된 고유 엔터티의 수가 100개 미만입니다. 성능을 개선하기 위해 합법적인 엔터티의 더 많은 예를 포함하는 것이 좋습니다.

원인

이 오류는 데이터 세트에 모델 훈련에 필요한 것보다 합법적인 이벤트가 있는 개체가 적은 경우에 발생합니다. TFI(Transaction Fraud Insights) 모델은 사기 공간을 최대한 활용하려면 합법적인 이벤트가 있는 최소 100개의 개체가 필요합니다. 소규모 엔터티 그룹에서 모든 합법적인 이벤트를 수행하는 경우 모델이 제대로 일반화되지 않을 수 있습니다.

솔루션

데이터 세트에 합법적인 이벤트가 있는 엔터티가 100개 이상 포함되어 있는지 확인합니다. 필요한 경우이 기간이 더 긴지 확인할 수 있습니다.

5. 문제 유형: 오류

   설명

   데이터 세트에 100개 미만의 행이 있습니다. 전체 데이터 세트에 100개 이상의 행이 있고 50개 이상의 행에 사기로 레이블이 지정되어 있는지 확인합니다.

   원인

   이 오류는 데이터 세트에 레코드가 100개 미만인 경우에 발생합니다. Amazon Fraud Detector는 모델 훈련을 위해 데이터 세트에 있는 최소 100개의 이벤트(레코드)의 데이터가 필요합니다.

   솔루션

   데이터 세트에 100개가 넘는 이벤트의 데이터가 있는지 확인합니다.

EVENT_LABEL 값이 누락되었거나 다릅니다.

1. 문제 유형: 오류

   설명

   EVENT_LABEL 열의 1% 이상이 null이거나 모델 구성에 정의된 값 이외의 값입니다**$label_values**. EVENT_LABEL 열에 누락된 값이 1% 미만이고 값이 모델 구성에 정의된 값인지 확인합니다**$label_values**.

원인

이 오류는 다음 이유 중 하나로 인해 발생합니다.

- 훈련 데이터가 포함된 CSV 파일의 레코드 중 1% 이상이 EVENT_LABEL 열에 누락된 값이 있습니다.
- 훈련 데이터가 포함된 CSV 파일의 레코드 중 1% 이상이 EVENT_LABEL 열에 이벤트 유형과 연결된 값과 다른 값이 있습니다.

온라인 사기 인사이트(OFI) 모델에서는 각 레코드의 EVENT_LABEL 열을 이벤트 유형과 연결된 레이블 중 하나로 채워야 합니다(또는에 매핑됨`CreateModelVersion`).

솔루션

이 오류가 EVENT_LABEL 값 누락으로 인한 경우 해당 레코드에 적절한 레이블을 할당하거나 데이터 세트에서 해당 레코드를 삭제하는 것이 좋습니다. 일부 레코드의 레이블이에 속하지 않기 때문에 이 오류가 발생하는 경우 EVENT_LABEL 열의 모든 값을 이벤트 유형의 레이블에 추가하고 모델 생성 시 사기 또는 합법적인(사기, 적법)에 매핑해야 **`label_values`**합니다.

2. 문제 유형: 정보

설명

EVENT_LABEL 열에는 모델 구성에 정의된 값 이외의 null 값 또는 레이블 값이 포함되어 있습니다**`$label_values`**. 이러한 일관되지 않은 값은 훈련 전에 '사기 아님'으로 변환되었습니다.

원인

다음 이유 중 하나로 인해이 정보를 얻을 수 있습니다.

- 훈련 데이터가 포함된 CSV 파일의 레코드 중 EVENT_LABEL 열에 누락된 값이 있는 레코드는 1% 미만입니다.
- 훈련 데이터가 포함된 CSV 파일의 레코드 중 1% 미만에는 EVENT_LABEL 열의 값이 이벤트 유형과 연결된 값과 다릅니다.

두 경우 모두 모델 훈련이 성공합니다. 그러나 레이블 값이 누락되거나 매핑되지 않은 이벤트의 레이블 값은 합법적인 것으로 변환됩니다. 이것이 문제라고 생각되면 아래 제공된 솔루션을 따르세요.

솔루션

데이터 세트에 EVENT_LABEL 값이 누락된 경우 데이터 세트에서 해당 레코드를 삭제하는 것이 좋습니다. 해당 EVENT_LABELS에 대해 제공된 값이 매핑되지 않은 경우 이러한 모든 값이 각 이벤트에 대해 사기 또는 합법적(사기, 적법)으로 매핑되어야 합니다.

EVENT_TIMESTAMP 값이 누락되었거나 잘못되었습니다.

1. 문제 유형: 오류

   설명

   훈련 데이터 세트에는 허용되는 형식을 준수하지 않는 타임스탬프가 있는 EVENT_TIMESTAMP가 포함되어 있습니다. 형식이 허용되는 날짜/타임스탬프 형식 중 하나인지 확인합니다.

   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 형식을 준수하는지 확인합니다. EVENT_TIMESTAMP 열에 누락된 값이 있는 경우 지원되는 타임스탬프 형식을 사용하여 값을 채우거나 , 또는와 같은 문자열을 입력하는 대신 이벤트를 완전히 삭제하는 것을 고려할 수 있습니다`nonenullmissing`.
2. 문제 유형: 오류

   훈련 데이터 세트에는 누락된 값이 있는 EVENT_TIMESTAMP가 포함되어 있습니다. 누락된 값이 없는지 확인합니다.

   원인

   이 오류는 데이터 세트의 EVENT_TIMESTAMP 열에 누락된 값이 있는 경우 발생합니다. Amazon Fraud Detector를 사용하려면 데이터 세트의 EVENT_TIMESTAMP 열에 값이 있어야 합니다.

   솔루션

   데이터 세트의 EVENT_TIMESTAMP 열에 값이 있고 해당 값이 지원되는 타임스탬프 형식을 준수하는지 확인합니다. EVENT_TIMESTAMP 열에 누락된 값이 있는 경우 지원되는 타임스탬프 형식을

사용하여 값을 채우거나 , 또는와 같은 문자열을 입력하는 대신 이벤트를 완전히 삭제하는 것을 고려할 수 있습니다`nonenullmissing`.

수집되지 않은 데이터

문제 유형: 오류

설명

훈련에 대해 수집된 이벤트를 찾을 수 없습니다. 훈련 구성을 확인하세요.

원인

이 오류는 Amazon Fraud Detector에 저장된 이벤트 데이터가 있는 모델을 생성하지만 모델 훈련을 시작하기 전에 Amazon Fraud Detector로 데이터 세트를 가져오지 않은 경우에 발생합니다.

솔루션

Amazon Fraud Detector 콘솔에서 `SendEvent` API 작업, `CreateBatchImportJob` API 작업 또는 배치 가져오기 기능을 사용하여 먼저 이벤트 데이터를 가져온 다음 모델을 학습합니다. 자세한 내용은 저장된 이벤트 데이터 세트를 참조하세요.

> **Note**
> 데이터 가져오기를 완료한 후 10분 후에 모델을 훈련하는 데 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 참조하세요.

변수 부족

문제 유형: 오류

설명

데이터세트에는 훈련에 적합한 변수가 2개 이상 포함되어야 합니다.

원인

이 오류는 데이터 세트에 모델 훈련에 적합한 변수가 2개 미만인 경우에 발생합니다. Amazon Fraud Detector는 모든 검증을 통과한 경우에만 모델 훈련에 적합한 변수를 고려합니다. 변수가 검증에 실패하면 모델 훈련에서 제외되고 모델 훈련 진단에 메시지가 표시됩니다.

솔루션

데이터 세트에 값으로 채워지고 모든 데이터 검증을 통과한 변수가 두 개 이상 있는지 확인합니다. 열 헤더(EVENT_TIMESTAMP, EVENT_ID, ENTITY_ID, EVENT_LABEL 등)를 제공한 이벤트 메타데이터 행은 변수로 간주되지 않습니다.

누락되거나 잘못된 변수 유형

문제 유형: 경고

설명

의 예상 데이터 형식은 NUMERIC**$variable_name**입니다. 데이터 세트**$variable_name**에서 검토 및 업데이트하고 모델을 재학습합니다.

원인

변수가 NUMERIC 변수로 정의되어 있지만 데이터 세트에 NUMERIC로 변환할 수 없는 값이 있는 경우 이 경고가 표시됩니다. 따라서 해당 변수는 모델 훈련에서 제외됩니다.

솔루션

NUMERIC 변수로 유지하려면 제공한 값을 부동 소수점 숫자로 변환할 수 있는지 확인합니다. 변수에 누락된 값이 포함된 경우 , nonene `null`또는와 같은 문자열로 채우지 마세요`missing`. 변수에 숫자가 아닌 값이 포함된 경우 CATEGORICAL 또는 FREE_FORM_TEXT 변수 유형으로 다시 생성합니다.

누락된 변수 값

문제 유형: 경고

설명

의 보다 큰 **$threshold** 값이 훈련 데이터 세트에서 누락**$variable_name**되었습니다. 데이터 세트**$variable_name**에서를 수정하고 성능을 개선하기 위해 재학습하는 것이 좋습니다.

원인

누락된 값이 너무 많아 지정된 변수가 삭제되는 경우이 경고가 표시됩니다. Amazon Fraud Detector는 변수의 누락된 값을 허용합니다. 그러나 한 변수에 누락된 값이 너무 많으면 모델에 큰 영향을 주지 않으며 해당 변수는 모델 훈련에서 삭제됩니다.

솔루션

먼저 누락된 값이 데이터 수집 및 준비의 실수로 인한 것이 아닌지 확인합니다. 실수인 경우 모델 훈련에서 삭제할 수 있습니다. 그러나 이러한 누락 값이 가치가 있다고 생각하고 여전히 해당 변수를 유지하려는 경우 모델 훈련과 실시간 추론 모두에서 누락된 값을 상수로 수동으로 채울 수 있습니다.

고유 변수 값 부족

문제 유형: 경고

설명

의 고유 값 수가 100보다 작**$variable_name**습니다. 데이터 세트**$variable_name**에서 검토 및 업데이트하고 모델을 재학습합니다.

원인

지정된 변수의 고유 값 수가 100보다 작으면이 경고가 표시됩니다. 임계값은 변수 유형에 따라 다릅니다. 고유한 값이 거의 없는 경우 데이터 세트가 해당 변수의 특성 공간을 덮을 만큼 일반적이지 않을 위험이 있습니다. 따라서 모델은 실시간 예측에서 잘 일반화되지 않을 수 있습니다.

솔루션

먼저 변수 분포가 실제 비즈니스 트래픽을 나타내는지 확인합니다. 그런 다음 `first_name` 및 `last_name` 별도로 `full_customer_name` 대신를 사용하는 등 카디널리티가 더 높은 보다 세분화된 변수를 채택하거나 카디널리티를 낮출 수 있도록 변수 유형을 CATEGORICAL로 변경할 수 있습니다.

잘못된 변수 표현식

1. 문제 유형: 정보

   설명

   $email_variable_name 값의 50% 이상이 예상 정규식 http://emailregex.com 일치하지 않습니다. 데이터 세트**$email_variable_name**에서를 수정하고 성능을 개선하기 위해 재학습하는 것이 좋습니다.

원인

이 정보는 데이터 세트의 레코드가 50%를 초과하는 경우 일반 이메일 표현식을 준수하지 않아 검증에 실패하는 이메일 값이 있는 경우에 표시됩니다.

솔루션

정규식을 준수하도록 이메일 변수 값의 형식을 지정합니다. 누락된 이메일 값이 있는 경우 , none null또는와 같은 문자열로 채우는 대신 빈 상태로 두는 것이 좋습니다missing.

2. 문제 유형: 정보

설명

$IP_variable_name 값의 50% 이상이 IPv4 또는 IPv6 주소 https://digitalfortress.tech/tricks/top-15-commonly-used-regex/ 정규식과 일치하지 않습니다. 데이터 세트**$IP_variable_name**에서를 수정하고 성능을 개선하기 위해 재학습하는 것이 좋습니다.

원인

이 정보는 데이터 세트의 레코드가 50%를 초과하여 IP 값이 정규 IP 표현식을 준수하지 않아 검증에 실패하는 경우에 표시됩니다.

솔루션

정규식을 준수하도록 IP 값의 형식을 지정합니다. 누락된 IP 값이 있는 경우 , none null또는와 같은 문자열로 채우는 대신 빈 상태로 두는 것이 좋습니다missing.

3. 문제 유형: 정보

설명

$phone_variable_name 값의 50% 이상이 기본 전화 정규식 /$pattern/과 일치하지 않습니다. 데이터 세트**$phone_variable_name**에서를 수정하고 성능을 개선하기 위해 재학습하는 것이 좋습니다.

원인

이 정보는 데이터 세트의 레코드가 50%를 초과하는 경우 일반 전화번호 표현식을 준수하지 않아 검증에 실패하는 전화번호로 표시됩니다.

솔루션

정규식을 준수하도록 전화번호의 형식을 지정합니다. 전화번호가 누락된 경우 , none null또는와 같은 문자열로 채우지 말고 비워 두는 것이 좋습니다missing.

고유 개체 부족

문제 유형: 정보

설명

고유 엔터티 수가 1500개 미만입니다. 성능을 개선하기 위해 더 많은 데이터를 포함하는 것이 좋습니다.

원인

이 정보는 데이터 세트의 고유 개체 수가 권장 숫자보다 적을 때 표시됩니다. 트랜잭션 사기 인사이트(TFI) 모델은 시계열 집계와 일반 트랜잭션 기능을 모두 사용하여 최상의 성능을 제공합니다. 데이터 세트에 고유한 개체가 너무 적은 경우 IP_ADDRESS, EMAIL_ADDRESS와 같은 대부분의 일반 데이터에 고유한 값이 없을 수 있습니다. 그런 다음이 데이터 세트가 해당 변수의 특성 공간을 충당할 만큼 일반적이지 않을 위험도 있습니다. 따라서 모델은 새로운 개체의 트랜잭션에 대해 잘 일반화되지 않을 수 있습니다.

솔루션

더 많은 개체를 포함합니다. 필요한 경우 훈련 데이터 시간 범위를 확장합니다.

할당량

AWS 계정 에는 각 Amazon Web Service에 대해 이전에 제한이라고 하는 기본 할당량이 있습니다. 다르게 표시되지 않는 한 리전별로 각 할당량이 적용됩니다. 아래 표에 언급된 조정 가능한 모든 할당량에 대해 할당량 증가를 요청할 수 있습니다. 자세한 내용은 할당량 증가 요청을 참조하세요.

다음 표에는 구성 요소별 Amazon Fraud Detector 할당량이 요약되어 있습니다.

Amazon Fraud Detector 모델

할당량 이름	기본 할당량	조정 가능
학습 데이터 크기	5GB	아니요
계정당 모델	50	아니요
모델당 버전 수	200	아니요
계정당 배포된 모델 버전	5	아니요
계정당 동시 훈련 작업	3	아니요
모델별 동시 교육 작업	1	아니요

Amazon 사기 탐지기 / 변수 / 결과 / 규칙

할당량 이름	기본 할당량	조정 가능
계정당 변수	5000	아니요
계정당 규칙	5000	아니요
규칙당 목록	3	아니요
계정당 결과	5000	아니요
계정당 감지기	100	아니요

할당량 이름	기본 할당량	조정 가능
감지기당 목록	30	아니요
감지기당 초안 버전	100	아니요
감지기 버전당 모델	10	아니요
계정당 라벨	100	아니요
계정당 이벤트 유형	100	아니요
계정당 개체 유형	100	아니요

Amazon Fraud Detector API

할당량 이름	기본 할당량	조정 가능
초당 GetEventPrediction API 호출	200TPS	예
GetEventPrediction API 호출당 페이로드 크기	256KB	아니요
GetEventPrediction API 호출당 입력 수	5000	아니요

문서 기록

다음 표에서는 Amazon Fraud Detector 사용 설명서의 중요한 변경 사항에 대해 설명합니다. 또한 보내주신 피드백을 처리하기 위해 Amazon Fraud Detector 사용 설명서를 자주 업데이트합니다.

변경 사항	설명	날짜
새로운 변수 및 데이터 형식	Amazon Fraud Detector는 유용한 정보를 추출하는 데 사용할 수 있는 새로운 변수 유형과 데이터 유형을 도입합니다.	2023년 6월 5일
이벤트 오케스트레이션	이벤트 오케스트레이션을 사용하면 Amazon EventBridge를 사용하여 다운스트림 처리를 AWS 서비스 위해 이벤트를에 쉽게 보낼 수 있습니다.	2023년 5월 30일
Lists	Lists 리소스를 사용하면 규칙의 일부로 IP 주소 또는 이메일 주소와 같은 값 집합을 참조할 수 있습니다. 규칙의 목록을 사용하여 액세스 또는 트랜잭션을 허용하거나 거부합니다.	2023년 2월 14일
데이터 모델 탐색기	Data Models Explorer는 Amazon Fraud Detector가 사기 탐지 모델을 생성하는 데 필요한 데이터 요소에 대한 인사이트를 제공합니다. 이벤트 데이터 세트를 준비하기 전에 데이터 모델 탐색기를 사용합니다.	2022년 12월 15일
계정 탈취 인사이트 모델	계정 탈취 인사이트(ATI) 모델을 사용하여 악의적인 탈취, 피싱 또는 도용된 자격 증명으로	2022년 7월 21일

	인해 손상된 계정을 탐지합니다.	
장 업데이트	Amazon Fraud Detector에 대한 추가 정보로 소개 장을 업데이트했습니다.	2022년 4월 11일
가변 보강	이러한 데이터 요소를 사용하고 2022년 2월 8일 이전에 훈련된 모델의 성능을 높이기 위해 제공하는 일부 원시 데이터를 보강할 수 있습니다.	2022년 2월 8일
옵트아웃 정책	옵트아웃 정책을 사용하여 이벤트 데이터를 사용하여 Amazon Fraud Detector의 품질을 개발하거나 개선하는 것을 옵트아웃합니다.	2022년 1월 6일
혼동된 대리자 방지	타사 또는 교차 서비스 엔터티가 계정의 리소스에 액세스하기 위해 타사 또는 교차 서비스 엔터티를 대신하여 작업할 수 있는 권한이 있는 엔터티를 조작하지 못하도록 정책을 생성합니다.	2021년 12월 6일
이벤트 데이터 세트 생성	이벤트 데이터 세트 생성에 제공된 지침을 사용하여 모델 훈련을 위한 데이터를 준비하고 수집합니다.	2021년 11월 22일
예측 설명	예측 설명을 사용하여 각 이벤트 변수가 모델의 사기 예측 점수에 어떤 영향을 미쳤는지 파악할 수 있습니다.	2021년 11월 10일

문제 해결	훈련 데이터 문제 해결의 정보를 사용하여 모델을 훈련할 때 Amazon Fraud Detector 콘솔에서 볼 수 있는 문제를 진단하고 해결할 수 있습니다.	2021년 10월 11일
트랜잭션 사기 인사이트 모델	트랜잭션 사기 인사이트(TFI) 모델을 사용하여 온라인 또는 card-not-present 트랜잭션 사기를 탐지합니다.	2021년 10월 11일
저장된 이벤트	이벤트 데이터를 Amazon Fraud Detector에 저장하고 저장된 데이터를 사용하여 나중에 모델을 훈련합니다. Amazon Fraud Detector에 이벤트 데이터를 저장하면 자동 계산 변수를 사용하는 모델을 훈련하여 성능을 개선하고, 모델 재학습을 간소화하고, 사기 레이블을 업데이트하여 기계 학습 피드백 루프를 닫을 수 있습니다.	2021년 10월 11일
모델 변수 중요도	모델 변수 중요도를 사용하여 모델의 성능을 높이거나 낮추는 요인과 가장 많이 기여하는 모델 변수에 대한 인사이트를 얻습니다. 그런 다음 모델을 조정하여 전반적인 성능을 개선합니다.	2021년 7월 9일
AWS CloudFormation과 통합	AWS CloudFormation 를 사용하여 Amazon Fraud Detector 리소스를 관리합니다.	2021년 5월 10일

배치 예측	배치 예측을 사용하여 실시간 채점이 필요하지 않은 이벤트 집합에 대한 예측을 가져옵니다.	2021년 3월 31일
챕터 재작업	시작하기 및 기타 섹션의 재작업	2020년 7월 17일
최초 릴리스	초기 릴리스	2019년 12월 2일

기계 번역으로 제공되는 번역입니다. 제공된 번역과 원본 영어의 내용이 상충하는 경우에는 영어 버전이 우선합니다.