

ORACLE

머신 러닝 모델의 수명 주기



서론

머신 러닝을 성장의 잠재적인 영역으로서 인식하거나 투자하는 기업이 늘고 있습니다. 데이터를 활용하여 고객에 대한 통찰력을 확보할 수 있는 것부터 프로세스를 보다 효율적으로 만드는 데 이르기까지 기업이 머신 러닝에 투자하는 이유는 많습니다. 이 책에서는 데이터 액세스 및 수집, 데이터 준비 및 탐색, 모델 구축 및 교육, 모델 평가, 모델 도입, 모델 모니터링의 6단계로 머신 러닝 모델이 구축되는 방법을 자세히 설명합니다.

머신 러닝 모델 구축은 반복적인 프로세스입니다. 머신 러닝 모델 구축에 필요한 많은 단계는 데이터 과학자가 모델 성능에 만족할 때까지 반복되고 수정됩니다. 이 프로세스에서 각 단계가 독립적으로 탐색, 수정 및 감사되어야 하므로 많은 양의 데이터 탐색, 시각화 및 실험이 필요합니다.

머신 러닝 모델 구축 단계



I. 데이터 액세스 및 수집

머신 러닝 문제의 첫 번째 단계는 데이터에 액세스하는 것입니다. 일반적으로 데이터 과학자는 회사에서 데이터를 저장하는 데이터베이스를 쿼리하여 작업 대상인 비즈니스 문제에 대한 데이터를 얻을 수 있습니다. 또한 관계형 데이터베이스에 잘 맞지 않는 구조화되지 않은 데이터 세트에 많은 값이 있습니다(예: 로그, 원시 텍스트, 이미지, 비디오 등). 해당 데이터 세트는 데이터 엔지니어 및 데이터 과학자가 작성한 ETL(추출, 변환, 로드) 파이프라인을 통해 많은 양이 처리됩니다. 이러한 데이터 세트는 관계형이든 그렇지 않든 데이터 레이크 또는 데이터베이스에 있습니다.

데이터 과학자에게 문제 해결에 필요한 데이터가 없을 경우 웹 사이트에서 데이터를 스크래핑하거나, 데이터 제공 업체로부터 데이터를 구매하거나, 설문조사, 클릭스트림 데이터, 센서, 카메라 등에서 데이터를 수집하여 데이터를 얻을 수 있습니다.

II. 데이터 준비 및 탐색

데이터를 확보한 후 데이터 과학자는 원시 데이터를 준비하고, 데이터 탐색을 수행하고, 데이터를 시각화하고, 데이터를 변환하며, 모델링에 사용할 준비가 될 때까지 해당 단계를 반복해야 합니다. 데이터 준비는 분석 전에 원시 데이터를 정리하고 처리하는 작업입니다. 머신러닝 모델을 구축하기 전에 데이터 과학자는 사용 가능한 데이터를 파악해야 합니다. 원시 데이터는 복잡하거나 중복되거나 부정확할 수 있습니다. 데이터 과학자는 사용 가능한 데이터를 탐색한 다음 손상되고 부정확하고 불완전한 데이터를 식별하고 바꾸거나 삭제하여 데이터를 정리합니다.

또한 데이터 과학자는 데이터에 레이블이 있는지 여부를 확인해야 합니다. 예를 들어, 일련의 이미지가 있고 이미지에 자동차가 있는지 확인하는 감지 모델을 개발하려는 경우 이미지에 자동차가 있는지 여부를 나타내는 라벨이 지정된 세트가 있어야 하고, 이미지의 자동차 주위에 경계 상자가 필요할 가능성이 높습니다. 이미지에 레이블이 없으면 데이터 과학자가 레이블을 지정해야 합니다. 오픈 소스 도구 및 상용 벤더가 있어 데이터 레이블 지정 플랫폼을 제공하며, 라벨 지정을 위한 인력을 고용할 수도 있습니다.

데이터를 정리한 후 데이터 과학자는 데이터 세트의 특성(또는 변수)을 탐색하고 특성 변환 간의 관계를 식별합니다. 데이터 과학자는 다양한 도구를 사용하여 오픈 소스 라이브러리 및 분석/데이터 과학 플랫폼에서 탐색형 데이터 분석을 할 수 있습니다. 이 단계에서는 데이터 세트의 통계 분석을 수행하고 데이터 시각화를 만들어 특성의 플롯을 생성하는 도구가 유용합니다.

데이터 세트에 포함된 특성 유형을 확인하는 것이 중요합니다. 특성은 부동 소수점이나 정수인 숫자일 수 있습니다. 범주적 특성에는 일반적으로 데이터를 그룹에 할당하는 한정된 수의 값이 있습니다. 예를 들어 고객 설문조사의 데이터 세트가 있는 경우 응답자의 성별(남성 또는 여성)은 범주형 특성입니다. 서수 특성은 설정된 순서나 배율이 있는 범주형 특성입니다. 예를 들어, 고객 만족도 응답: 매우 만족, 만족, 무관심, 불만족, 매우 불만족은 설정된 순서를 갖습니다. 이 순서를 정수 배율(1->5)로 변환할 수 있습니다. 특성의 종류를 결정한 후 각 특성이 가진 값의 분포를 구하고 각 특성의 요약 통계를 얻는 것이 다음 작업입니다. 그렇게 하면 데이터 세트에 대한 다음 질문에 대답하는 데 도움이 됩니다.

- 데이터 세트가 값 범위 또는 범주 하위 집합에 대해 비대칭인가?
- 특성의 최소값, 최대값, 평균값, 중간값 및 모드 값은 무엇인가?

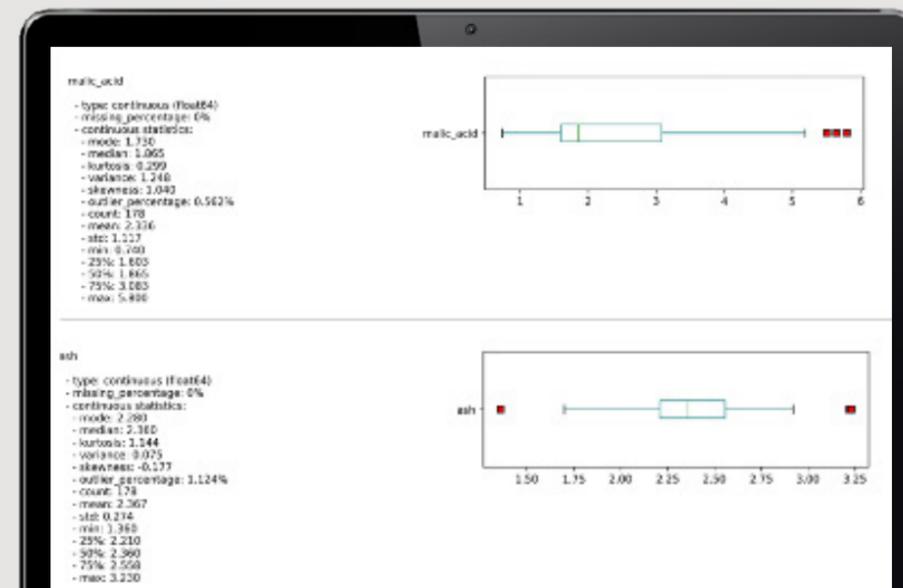
- 누락 값이나 Null과 같은 부적합한 값이 있는가? 있다면 몇 개인가?
- 데이터 세트에 이상값이 있는가?

데이터 탐색 단계에서는 특성을 플롯으로 그리고 특성 간에 서로에 대한 플롯도 그리면 데이터 세트의 패턴을 식별하는 데 유용하며 데이터 변형의 필요성을 확인할 수 있습니다. 대답해야 할 몇 가지 질문은 다음과 같습니다.

- 누락된 값을 어떻게 처리할 것인가? 값을 채운다면, 누락 값을 채우기 위해 취하려는 접근법은 무엇인가? 평균값, 중앙값, 모드 값, 인근 항목의 값 및 인근 항목 값의 평균을 취하는 방법 등이 있습니다.

- 이상치를 어떻게 처리할 것인가?
- 일부 특성은 서로 관련이 있는가?
- 데이터 세트를 정규화하거나 다른 변환을 수행하여 데이터의 스케일을 조정해야 하는가(예: 로그 변환)?
- 범주별 값의 롱테일 현상에 대한 접근 방식은 무엇인가? 그대로 사용할 것인가, 어떤 의미 있는 방법으로 그룹화하거나 그 하위집합을 모두 무시할 것인가?

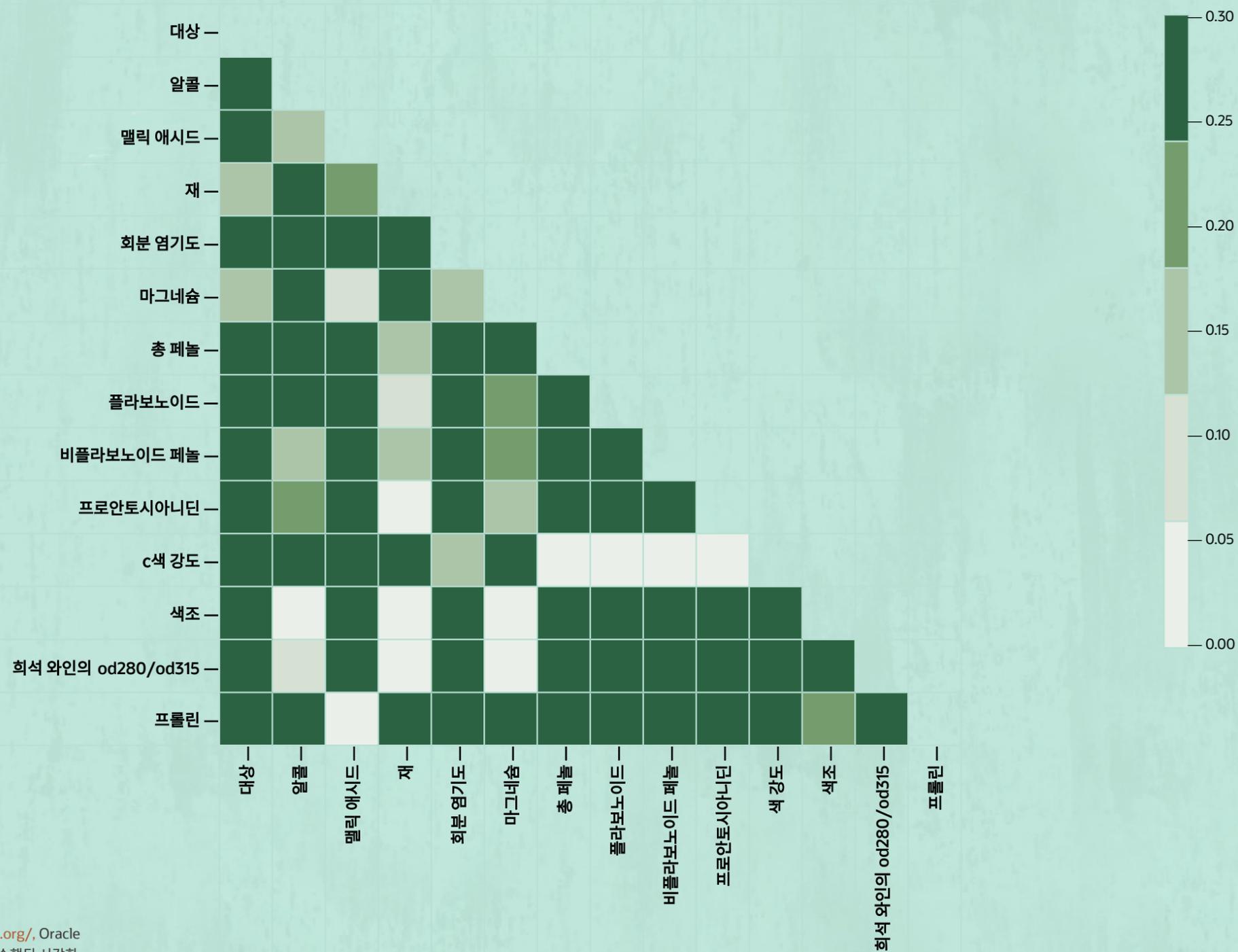
세 가지 유형의 와인 및 각 와인 기능의 데이터 세트에서 기능을 요약하고 시각화한 것입니다.



출처: Scikit Learn <https://scikit-learn.org/>, Oracle Cloud Infrastructure Data Science로 수행된 시각화 <https://docs.cloud.oracle.com/en-us/iaas/data-science/using/data-science.htm>



세 가지 유형의 와인 및 각 와인 특성을 포함하는 데이터세트에서 특성이 서로 얼마나 상관관계가 있는지에 대한 히트맵입니다.



데이터 탐색 단계 중에 데이터 세트의 패턴을 식별하여 데이터 세트를 더 잘 나타내는 새로운 특성을 개발하는 방법에 대한 아이디어를 얻을 수 있습니다. 이를 특성 엔지니어링이라고 합니다. 예를 들어, 매시간 주요 교차점을 통과하는 차량 수에 대한 트래픽 데이터 세트가 있는 경우 이른 오전, 오전 나절, 이른 오후, 늦은 오후 및 밤 시간과 같이 시간을 다른 부분으로 분류하는 새 특성을 만들 수 있습니다.

범주별 특성의 경우 특성을 원 핫 인코딩(one hot encode)하는 것이 필요한 경우가 많습니다. 원 핫 인코딩이란 범주별 특성을 각 범주마다 하나씩 이진 특성으로 변환하는 것을 의미합니다. 예를 들어, 고객의 데이터 세트가 있고 워싱턴, 오리건, 캘리포니아 등 고객의 출신인 주라는 특성이 있다고 가정해 보겠습니다. 원 핫 인코딩은 두 개의 이진 특성을 생성합니다. 여기서 한 특성은 고객이 워싱턴 주 출신인지 여부이고 두 번째 특성은 고객이 오리건 출신인지 여부입니다. 고객이 워싱턴 또는 오리건 출신이 아닌 경우 캘리포니아 출신이므로 세 번째 특성은 필요하지 않다고 가정합니다.

출처: Scikit Learn <https://scikit-learn.org/>, Oracle Cloud Infrastructure Data Science로 수행된 시각화 <https://docs.cloud.oracle.com/en-us/iaas/data-science/using/data-science.htm>



III. 모델 빌드 및 교육

모델 빌드는 문제를 해결할 적절한 머신 러닝 모델과 모델에 들어가는 특성을 선택하는 것으로 구성됩니다. 모델 빌드의 첫번째 단계에서 데이터 과학자는 문제를 해결하기 위해 적합한 머신 러닝 모델을 결정해야 합니다. 두 가지 주요 유형의 머신 러닝 모델에는 지도형 및 비지도형이 있습니다. 지도형 학습에서는 입력 데이터 세트를 출력 또는 레이블로 모델링합니다. 분류 및 회귀는 지도형 학습 문제입니다. 비지도형 학습에는 레이블 없이 입력 데이터 세트를 모델링하는 과정이 포함됩니다. 예를 들어, 고객 세분화는 비지도형 학습 문제입니다. 고객이 속한 고객 세그먼트의 우선순위를 알 수 없습니다. 모델에 의해 세그먼트에 할당됩니다.

모델 빌드는 문제를 해결할 적절한 머신 러닝 모델과 모델에 들어가는 특성을 선택하는 것으로 구성됩니다. 모델 빌드의 첫번째 단계에서 데이터 과학자는 문제를 해결하기 위해 적합한 머신 러닝 모델을 결정해야 합니다. 두 가지 주요 유형의 머신 러닝 모델에는 지도형 및 비지도형이 있습니다. 지도형 학습에서는 입력 데이터 세트를 출력 또는 레이블로 모델링합니다. 분류 및 회귀는 지도형 학습 문제입니다. 비지도형 학습에는 레이블 없이 입력 데이터 세트를 모델링하는 과정이 포함됩니다. 예를 들어, 고객 세분화는 비지도형 학습 문제입니다. 고객이 속한 고객 세그먼트의 우선순위를 알 수 없습니다. 모델에 의해 세그먼트에 할당됩니다.

모델 교육 중에 데이터 세트는 교육 세트 및 테스트 세트로 분할됩니다. 교육 데이터 세트는 모델을 교육하는 데 사용되며 테스트 데이터 세트는 본 적 없는 데이터에 대해 모델의 성능이 얼마나 좋은지 확인하는 데 사용됩니다. 모델 평가에 대해서는 아래에서 자세히 설명합니다.

모델 하이퍼매개변수 튜닝은 모델 교육 프로세스의 주요 작업입니다. 모델은 알고리즘이며 하이퍼매개변수는 데이터 과학자가 모델의 성능을 향상시키기 위해 조정할 수 있는 손잡이와 같습니다. 예를 들어, 결정 트리의 심도는 하이퍼매개변수입니다.

매우 깊거나 매우 얇은 결정 트리를 선택할 수 있습니다. 이는 모델의 편향과 분산에 영향을 줍니다. 편향(Bias)은 특성과 출력 간의 관계를 포착하지 않아서 발생하는 오류 또는 과소 적합입니다. 분산은 모델이 학습 데이터 세트에서 잘 작동하지만 본 적 없는 데이터에 대해 제대로 작동하지 않는 과적합 오류입니다. 모델의 하이퍼매개변수 튜닝을 부분적으로 자동화할 수는 있지만 데이터 과학자는 항상 프로세스에 관여해야 합니다.

데이터 과학자들은 또한 모델을 교육하는 데 필요한 컴퓨팅 리소스의 종류를 결정해야 합니다. 데이터를 준비하고 컴퓨터에서 로컬로 모델을 교육할 수 있습니다. 그러나 얼마나 많은 데이터를 준비하고 모델을 훈련하는 데 사용하는지에 따라 컴퓨터로는 충분하지 않을 수 있습니다. GPU를 포함한 광범위한 컴퓨팅 리소스에 대한 액세스를 제공할 수 있는 클라우드 워크로드를 전환해야 할 수도 있습니다.

일부 모델은 전문화된 하드웨어에서 더 빠르게 교육될 수 있습니다(예: GPU에서 퍼셉트론/심층 신경망 모델 교육). 또한 프로세스를 가속화할 수 있는 분산된 교육 환경을 탐색할 수 있습니다. 특히 여러 시스템에 걸쳐 데이터를 분할하고 분산하여 사용할 수 있는 가장 큰 시스템 메모리에서도 데이터 양을 맞출 수 없는 경우 또는 여러 모델 후보를 별도의 시스템에서 동시에 교육하려는 경우가 특히 그렇습니다.





AutoML

AutoML은 머신 러닝을 더 많은 대중에게 더 쉽게 접근할 수 있도록 만들 수 있다는 장래성 때문에 지난 몇 년 동안 상당한 관심을 끌었습니다. AutoML은 자동화된 머신 러닝을 의미합니다. 특성 선택, 모델/알고리즘 선택 및 하이퍼파라미터 조정 프로세스를 자동화합니다. 이는 모든 주요 데이터 과학 플랫폼이 갖는 기능입니다. 사용자가 데이터 세트를 AutoML에 공급하면, 여러 머신 러닝 모델을 교육하고, 해당 모델의 하이퍼매개변수를 조정하고, 서로에 대한 성능을 평가합니다.

AutoML은 교육 프로세스를 자동화하여 데이터 과학자의 생산성을 향상시킬 수 있습니다. 또한 데이터 분석가와 개발자는 데이터 과학 전문 지식을 동원해 모델 교육 프로세스의 모든 측면을 조정하지 않고도 머신 러닝 모델을 빌드할 수 있습니다. 대부분의 AutoML 기능은 분류 및 회귀 문제에 대한 표 형식 데이터를 지원하며, 시계열 예측뿐만 아니라 이미지 및 텍스트 데이터를 지원하는 고급 제품도 있습니다.

AutoML이나 모든 복잡한 모델의 단점이라면 사용자가 모델이 예측에 도달하는 방식을 이해하기 어려운 블랙박스 솔루션처럼 보일 수 있다는 것입니다. 사용자는 AutoML 시스템이 모델 설명 가능성을 제공하는지 찾아보고, 사용자가 모델을 해석하고 선택한 모델이 예측에 도달하는 방법을 이해하도록 지원하는 기능이 있는지 확인해야 합니다.



모델 설명은 일반적으로 글로벌 설명 및 로컬 설명으로 나뉩니다. 글로벌 설명은 머신 러닝 모델의 일반적인 동작을 전체적으로 이해하는 것입니다. 여기에는 각 특성이 모델 예측에 기여하는 중요한 방법에 대한 설명이 포함됩니다. 로컬 설명은 머신 러닝 모델이 하나의 데이터 샘플에 대해 특정 예측을 수행하는 이유를 설명합니다. (예: 사기 감지 알고리즘이 특정 트랜잭션을 사기 항목으로 예측한 이유는 무엇인가?)

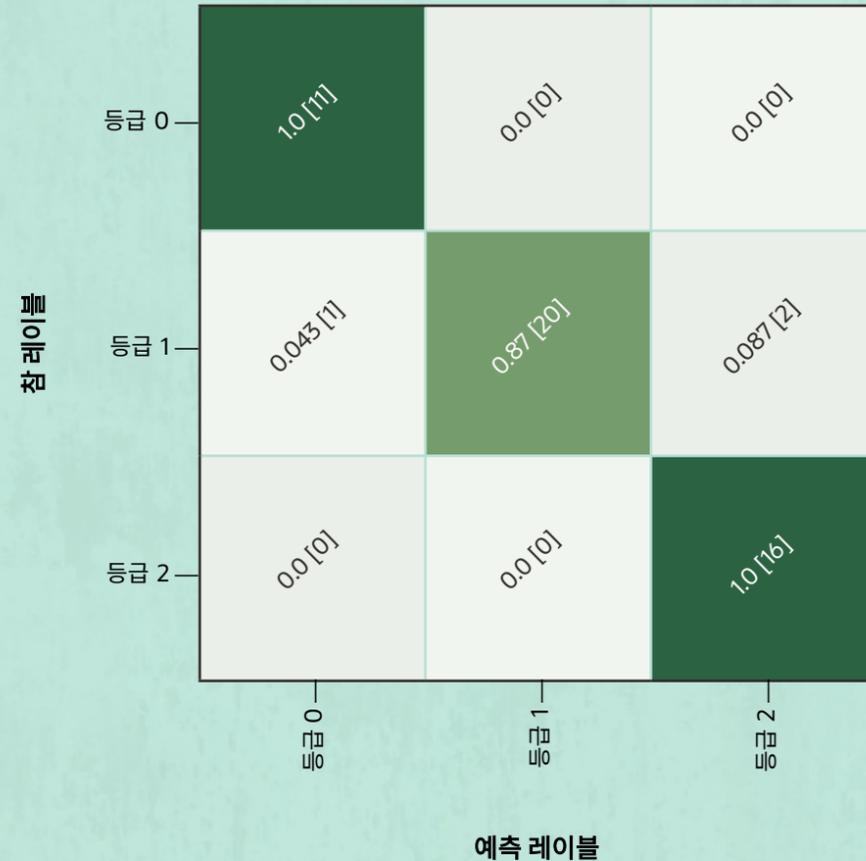


IV. 모델 평가

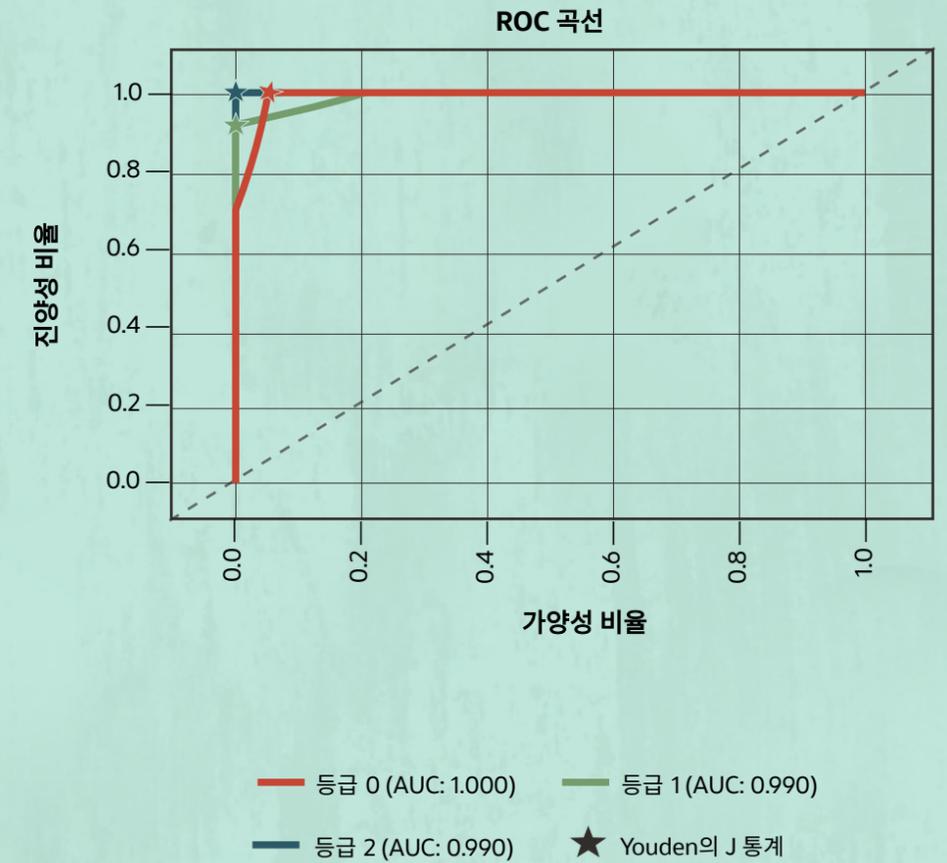
데이터 과학자가 머신 러닝 모델을 평가하기 위한 측정 항목을 계산하고 시각화하는 데 도움이 되는 오픈 소스 도구가 많이 있습니다(예: AUC-ROC 곡선, 계인 및 리프트 차트). 머신 러닝 모델을 평가할 때 데이터 과학자는 해결하려는 비즈니스 문제에 중요한 측정항목을 결정해야 합니다.

분류 문제의 경우 모델 평가에 정확도를 사용할 수 있지만 최적의 측정 항목이 아닐 수도 있습니다. 문제가 희귀한 질병이 있는지 여부를 감지하는 것과 관련된 경우 더 나은 측정 항목은 질병이 있는 것으로 정확하게 진단받은 사람 수를 질병이 있는 모든 사람 수로 나눈 것일 수 있습니다. 이 경우 진양성, 진음성, 가양성 및 가음성의 수를 표시하고 정밀도와 리콜을 계산하는 혼동행렬을 살펴보는 것이 더 유용할 것입니다. 회귀 문제의 경우 제곱평균제곱근 오류, 평균 절대 오류 등의 측정항목을 사용하거나 결정 계수 r2를 계산할 수 있습니다. 비지도형 문제의 경우 클러스터 내에서 높은 응집을 가진 클러스터 세트와 클러스터 간 분리가 이상적인 측정항목으로 간주됩니다. 이는 실루엣 점수 및 Calinski-Harabasz 계수와 같은 측정항목으로 측정할 수 있습니다.

세 가지 유형의 와인 및 각 와인 특성을 포함하는 데이터 세트에서 와인 특성을 기반으로 와인 유형을 예측하는 랜덤 포레스트 모델의 결과에 대한 다중 클래스 분류의 혼동 행렬



세 가지 유형의 와인 및 각 와인 특성을 포함하는 데이터 세트에서 와인 유형을 예측하는 랜덤 포레스트 모델의 결과에 대해 구성된 다중 클래스 분류의 ROC 곡선



출처: Scikit Learn Library <https://scikit-learn.org/>, Oracle Cloud Infrastructure Data Science로 수행된 시각화 <https://docs.cloud.oracle.com/en-us/iaas/data-science/using/data-science.htm>



V. 모델 도입

모델 교육 및 평가 프로세스가 완료되면 최적의 후보 모델이 저장됩니다. 모델은 일반적으로 Pickle, ONNX 및 PMML 형식으로 저장됩니다. 목적에 따라 데이터 과학자는 개념 증명, 실험 또는 생산에 도입하기 위해 머신러닝 문제에 대한 작업을 할 수 있습니다. 모델 도입은 머신러닝 모델에 의해 수행된 예측을 어떤 방식으로든 소비하는 것입니다. 데이터 변환 파이프라인도 도입되어야 할 가능성이 큼니다. 일반적으로 데이터 과학자는 모델 도입에 대해 엔지니어와 협력합니다.

예측을 소비하는 방법에 따라 배치 소비 또는 실시간 소비를 위해 도입할 수 있습니다. 배치 소비의 경우 예측을 스케줄링할 수 있습니다(예: 매 시간, 매일). 그런 다음 예측 작업을 데이터베이스에 저장하고 다른 애플리케이션에서 사용할 수 있습니다. 이 경우 일반적으로 처리하는 데이터의 양이 실시간 예측보다 큼니다. 사용 사례로는 전자 상거래 사이트를 실행하는데 과거 구매를 기준으로 고객에게 권장 제품에 대해 매주 이메일을 전송하려는 경우를 들 수 있습니다. 머신러닝 모델을 미리 실행하도록 일정을 잡을 수 있습니다.

실시간 소비의 경우라면 트리거는 예측을 제공하기 위해 지속 모델을 사용하는 프로세스를 시작할 것입니다. 예를 들어, 결제가 시작될 때 트랜잭션이 사기인지 여부를 결정하는 경우 실시간 예측이 필요합니다. 예측 서비스 속도(밀리초, 초 단위?), 서비스에 대한 수요량, 예측을 실행할 데이터 크기 등을 고려해야 합니다. 예측을 구현하기 위한 대기시간을 최소화하는 것이 중요합니다. GPU와 같은 가속기를 사용하여 더 작은 크기의 모델을 사용하고 개체와 관련된 특성을 실시간 예측을 위해 검색하는 방법을 개선함으로써 제공 대기 시간을 개선할 수 있습니다(예를 들어 사용자가 사이트를 검색할 때 사용자에게 제품을 추천하는 경우 사용자의 과거 구매에 대한 정보를 가져오는 방식을 개선하면 대기 시간이 향상될 수 있습니다.)

FaaS(Functions-as-a-Service) 플랫폼, HTTP 엔드포인트로서의 완전 관리형 모델 도입, 플라스크를 사용한 DIY, 컨테이너 통합관리 플랫폼(예: k8 및 도커 스왐 등) 등 모델 도입을 위한 다양한 도구와 클라우드 플랫폼 제공 서비스가 있습니다.





VI. 모델 모니터링

모델 모니터링은 성숙한 머신 러닝 및 데이터 과학 프로그램이 없다면 조직에서 종종 잊혀지는 까다로운 단계입니다. 모델 재교육 및 재도입에는 데이터 과학과 엔지니어링 팀의 시간과 컴퓨팅 리소스가 필요합니다. 모델 모니터링을 통해 팀은 모델을 재교육하고 재도입해야 할지의 여부와 시기를 결정할 수 있습니다. 모델 모니터링은 모델 성능에 대한 드리프트/통계 모니터링 및 운영 모니터링의 두 구성 요소로 나눌 수 있습니다.

모델이 도입된 후 모델을 측정하고 교육하는 측정항목은 생산 단계까지 내려갑니다. 데이터가 고정적이지 않기 때문입니다. 비고정성은 여러 가지 방식으로 표현될 수 있습니다. 즉, 생산 데이터의 특성은 데이터 세트의 범위를 벗어나는 값을 가질 수 있으며, 값의 분포에서 느린 드리프트가 있을 수 있습니다.

모델 성능 저하로 인해 모델 품질을 모니터링하여 모델을 재교육하고 재도입해야 하는지 여부 및 시기를 결정해야 합니다. 이 경우 운영 시스템으로 이동하는 라이브 데이터의 예측 정확도를 즉시 얻을 수 없을 때가 있습니다. 예를 들어 이탈 예측 모델 또는 사기 감지 모델이 정확한 예

측을 제공했는지 여부를 결정할 수 있으려면 다소 시간이 걸릴 수 있습니다. 그러나 실시간 데이터와 비교하여 교육 데이터의 통계 및 분포를 확인하고 모델 예측의 분포를 교육 및 실시간 데이터와 비교할 수도 있습니다. 예를 들어 고객 이탈 모델로 작업하는 경우 생산 시스템의 고객 특성과 비교되는 모델 교육에 사용된 고객의 특성을 비교할 수 있습니다. 또한 라이브 프로덕션과 비교하여 교육 샘플에서 이탈 예측 고객의 비율을 확인할 수 있습니다.

머신 러닝 시스템을 모니터링하려면 데이터 과학자와 엔지니어링 팀 간의 파트너십이 필요합니다. 모니터링할 항목에는 서비스 대기 시간, 메모리/CPU 사용량, 처리량 및 시스템 안정성이 포함됩니다. 추적 및 모니터링을 위해 로그 및 측정항목을 설정해야 합니다. 로그에는 발생한 시간이 포함된 이벤트 기록이 포함됩니다. 특정 사건을 조사하고 사건의 원인을 파악하는 데 사용할 수 있습니다. Kibana는 로그 검색 및 보기에 사용되는 오픈 소스 도구입니다. 측정항목은 머신 러닝 시스템의 사용량 및 동작을 측정합니다. Prometheus 및 Grafana는 측정항목을 모니터링하기 위한 도구입니다.



결론

이 문서가 머신 러닝 모델을 데 필요한 단계에 대한 유용한 가이드가 되었기를 바랍니다.

머신 러닝은 매우 반복적인 과정이며, 본서에 요약된 단계는 여러 번 반복되고 개선될 것입니다. 본서에서 다루는 각 단계에 대해 자세히 살펴볼 수 있는 자료가 많으며, 기업의 데이터 과학 전략에 대한 결정을 내릴 때 이런 자료를 자세히 알아보시기 바랍니다. 시작할 준비가 되었다면 Oracle이 제공하는 **실전 연습** 을 통해 자신만의 데이터 과학 모델 빌드를 실험해볼 수 있습니다.



Oracle corporation

전세계 본사

500 Oracle Parkway, Redwood Shores, CA 94065, USA

전 세계 문의처

전화 + 1.650.506.7000 + 1.800.ORACLE1

팩스 + 1.650.506.7200

oracle.com

문의처

전화 +1.800.ORACLE1 또는 oracle.com. 북미 지역 이외 국가인 경우 oracle.com/contact에서 현지 지사를 찾으십시오.

 blogs.oracle.com/oracle

 facebook.com/oracle

 twitter.com/oracle

저자

Wendy Yip, 데이터 과학자.

Copyright © 2020, Oracle and/or its affiliates. All rights reserved. 본 문서는 정보 제공의 목적으로만 제공되며 본 문서의 내용은 사전 공지 없이 변경될 수 있습니다. Oracle은 본 문서에 오류가 존재하지 않음을 보증하지 않으며, 상업성 또는 특정 목적의 적합성에 대한 암시적 보증이나 조건을 포함하여 구두로 표현했거나 법적으로 암시되었든 관계 없이 어떠한 보증이나 조건도 제시하지 않습니다. Oracle은 본 문서와 관련하여 어떠한 책임도 지지 않으며 이 문서로 인해 직접적 또는 간접적인 계약 상의 의무가 발생하지 않습니다. 본 문서는 Oracle의 사전 서면 승인 없이 전자적, 기계적 또는 어떠한 형태나 수단으로도 복제되거나 전송될 수 없습니다.

Oracle과 Java는 Oracle Corporation 및/또는 그 자회사의 등록상표입니다. 기타 명칭들은 각 해당 명칭을 소유한 회사의 상표일 수 있습니다.

Intel 및 Intel Xeon은 Intel Corporation의 상표 내지 등록상표입니다. SPARC 상표 일체는 라이센스에 의거하여 사용되며 SPARC International, Inc.의 상표 내지 등록상표입니다. AMD, Opteron, AMD 로고 및 AMD Opteron 로고는 Advanced Micro Devices의 상표 내지 등록상표입니다. UNIX는 The Open Group 05.10.19의 등록상표입니다.