MLOps：机器学习中的持续交付和自动化流水线

本文档讨论了实现和自动执行机器学习系统的持续集成 (CI)、持续交付 (CD) 和持续训练 (CT) 的技术。

数据科学和机器学习正逐渐成为解决复杂现实问题、转变行业以及在所有领域创造价值的核心功能。现在，有效运用机器学习技术的各种要素都已齐备：

- 大型数据集
- 经济实惠的按需计算资源
- 适用于各种云平台的机器学习专用加速器
- 不同机器学习研究领域（例如计算机视觉、自然语言理解和推荐 AI 系统）的快速发展。

因此，许多企业正在投资打造数据科学团队和机器学习功能，以开发能够为用户带来商业价值的预测模型。

本文档适用于希望将 DevOps (/devops) 原则运用于机器学习系统 (MLOps) 的数据科学家和机器学习工程师。MLOps 是一种机器学习工程文化和做法，旨在统一机器学习系统开发 (Dev) 和机器学习系统运营 (Ops)。实施 MLOps 意味着您将在机器学习系统构建流程的所有步骤（包括集成、测试、发布、部署和基础架构管理）中实现自动化和监控。

数据科学家可以实现并训练一个机器学习模型，该模型在给定用例的相关训练数据的情况下，可在留出的离线数据集上实现出色的预测性能。但是，真正的挑战不是构建机器学习模型，而是构建集成的机器学习系统以及在生产环境中持续运行该系统。由于 Google 的生产型机器学习服务已运行多年，因此我们了解到在生产环境中运行基于机器学习的系统时可能会遇到许多问题。《机器学习：技术债务的高息信用卡》 (https://ai.google/research/pubs/pub43146)总结了其中的一些问题。

如下图所示，在实际的机器学习系统中，只有一小部分是由机器学习代码组成的。所需的相关元素既庞大又复杂。

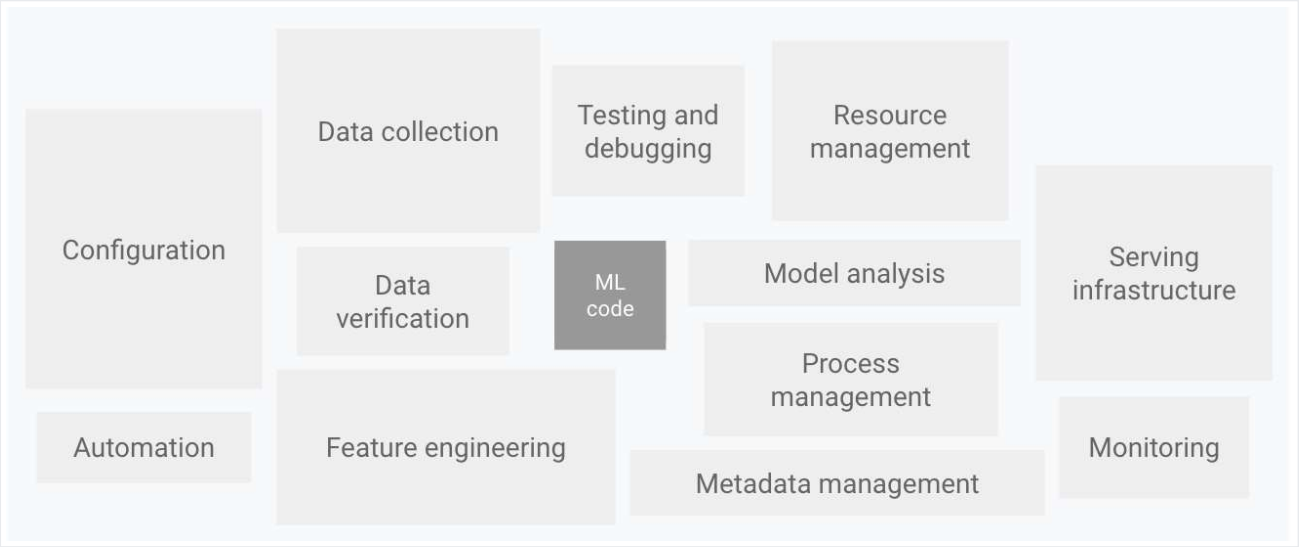

图 1.机器学习系统的元素。改编自机器学习系统中的隐藏技术债务 (https://papers.nips.cc/paper/5656-hidden-technical-debt-in-machine-learning-systems.pdf)。

在此图中，系统的其余部分包括配置、自动化、数据收集、数据验证、测试和调试、资源管理、模型分析、过程和元数据管理、服务基础架构和监控。

如需开发和运行类似的复杂系统，您可以将 DevOps 原则应用于机器学习系统 (MLOps)。本文档介绍了在为数据科学做法（例如机器学习中的 CI、CD 和 CT）设置 MLOps 环境时要考虑的概念。

本文档讨论了以下主题：

- DevOps 与 MLOps
- 开发机器学习模型的步骤
- MLOps 成熟度级别

DevOps 与 MLOps

DevOps (/devops) 是开发和运行大规模软件系统的一种常见做法。这种做法具有诸多优势，例如缩短开发周期、提高部署速度、实现可靠的发布。如需获得这些优势，您需要在软件系统开发中引入两个概念：

- 持续集成 (CI) (https://wikipedia.org/wiki/Continuous_integration)
- 持续交付 (CD) (https://wikipedia.org/wiki/Continuous_delivery)

机器学习系统是一种软件系统，因此类似的做法有助于确保您能够可靠地大规模构建和运行机器学习系统。

但是，机器学习系统在以下方面与其他软件系统不同：

- 团队技能：在机器学习项目中，团队通常包括数据科学家或机器学习研究人员，他们主要负责进行探索性数据分析、模型开发和实验。这些成员可能不是经验丰富的、能够构建生产级服务的软件工程师。
- 开发：机器学习在本质上具有实验性。您应该尝试不同的特征、算法、建模技术和参数配置，以便尽快找到问题的最佳解决方案。您所面临的挑战在于跟踪哪些方案有效、哪些方案无效，并在最大程度提高代码重复使用率的同时维持可重现性。
- 测试：测试机器学习系统比测试其他软件系统更复杂。除了典型的单元测试和集成测试之外，您还需要验证数据、评估经过训练的模型质量以及验证模型。
- 部署：在机器学习系统中，部署不是将离线训练的机器学习模型部署为预测服务那样简单。机器学习系统可能会要求您部署多步骤流水线以自动重新训练和部署模型。此流水线会增加复杂性，并要求您自动执行部署之前由数据科学家手动执行的步骤，以训练和验证新模型。
- 生产：机器学习模型的性能可能会下降，不仅是因为编码不理想，而且也因为数据资料在不断演变。换句话说，与传统的软件系统相比，模型可能会通过更多方式衰退，而您需要考虑这种降级现象。因此，您需要跟踪数据的摘要统计信息并监控模型的在线性能，以便系统在值与预期不符时发送通知或回滚。

机器学习和其他软件系统在源代码控制的持续集成、单元测试、集成测试以及软件模块或软件包的持续交付方面类似。但是，在机器学习中，有一些显著的差异：

- CI 不再仅仅测试和验证代码及组件，而且还会测试和验证数据、数据架构和模型。
- CD 不再针对单个软件包或服务，而会针对应自动部署其他服务（模型预测服务）的系统（机器学习训练流水线）。
- CT 是机器学习系统特有的一个新属性，它主要涉及自动重新训练和提供模型。

下一部分讨论了训练和评估要用作预测服务的机器学习模型的典型步骤。

机器学习的数据科学步骤

在任何机器学习项目中，定义业务用例并确定成功标准后，将机器学习模型交付给生产环境的过程涉及以下步骤。这些步骤可以手动完成，也可以由自动流水线完成。

1. 数据提取：您可以为机器学习任务选择和集成来自各种数据源的相关数据。
2. 数据分析：您可以执行探索性数据分析 (https://wikipedia.org/wiki/Exploratory_data_analysis) (EDA) 以了解可用于构建机器学习模

型的数据。此过程将产生以下结果：

- 了解模型预期的数据架构和特性。
- 识别模型所需的数据准备和特征工程。

3. 数据准备：为机器学习任务准备数据。此准备工作涉及数据清理，即将数据拆分为训练集、验证集和测试集。您还可以将数据转换和特征工程应用于解决目标任务的模型。此步骤的输出是准备格式的数据分片。
4. 模型训练：数据科学家使用准备好的数据实现不同的算法，以训练各种机器学习模型。此外，您还需要对实现的算法进行超参数调节，以获得最佳机器学习模型。此步骤的输出是经过训练的模型。
5. 模型评估：在保留测试集 (https://wikipedia.org/wiki/Training,_validation,_and_test_sets#Holdout_dataset)上评估模型，以评估模型质量。此步骤的输出是一组用于评估模型质量的指标。
6. 模型验证：模型已确定适合部署 - 它的预测性能优于特定基准。
7. 提供模型：经过验证的模型会部署到目标环境以提供预测。此部署可以是以下其中一项：
   - 具有 REST API 的微服务，可用于提供在线预测。
   - 边缘设备或移动设备的嵌入式模型。
   - 批量预测系统的一部分。
8. 模型监控：监控模型预测性能，以便在机器学习过程中潜在调用新的迭代。

这些步骤的自动化级别决定了机器学习过程的成熟度，成熟度反映了使用新数据训练新模型或者使用新实现训练新模型的速度。以下部分介绍了三个级别的 MLOps，从最常见的级别（该级别不涉及自动化）开始，一直到自动执行机器学习和 CI/CD 流水线。

MLOps 级别 0：手动过程

许多团队都有数据科学家和机器学习研究人员，他们可以构建领先的模型，但他们构建和部署机器学习模型的过程完全是手动的。这样的级别被视为基本成熟度级别（级别 0）。下图展示了此过程的工作流。

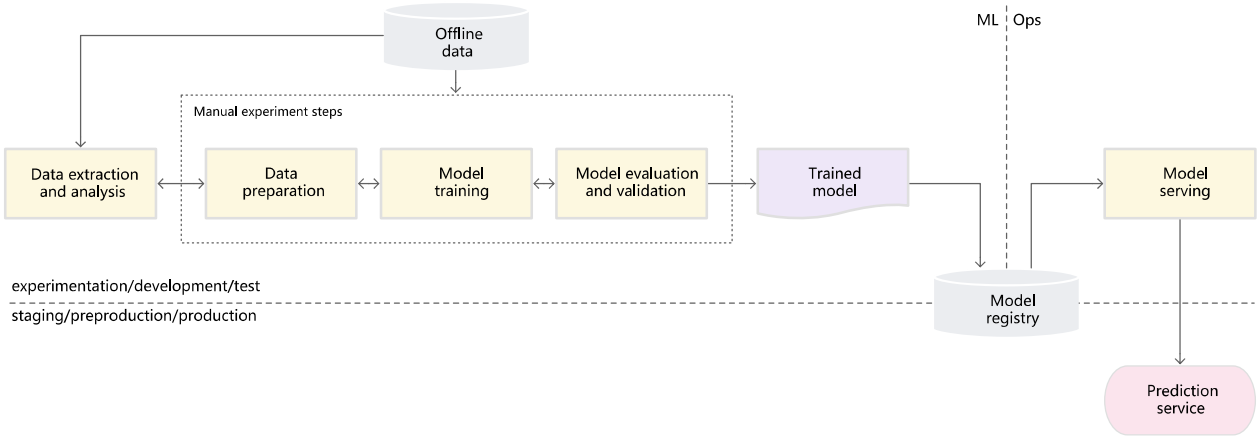

图 2：将模型用作预测服务的手动机器学习步骤。

特性

以下列表突出显示了 MLOps 级别 0 过程的特性，如图 2 所示：

- 脚本驱动的交互式手动过程：每个步骤（包括数据分析、数据准备、模型训练和验证）都是手动的。该过程需要手动执行每个步骤，并且手动从一个步骤转到另一个步骤。此过程通常由数据科学家以交互方式在笔记本中编写和执行的实验性代码驱动，直到生成有效的模型为止。
- 机器学习与操作分离：该过程会将创建模型的数据科学家与将模型用作预测服务的工程师分开。数据科学家将经过训练的模型作为工件移交给工程团队，以便在其 API 基础架构上进行部署。此移交工作可能包括将经过训练的模型放在存储位置、将模型对象签入代码库，或者将其上传到模型注册表。然后，部署模型的工程师需要在生产环境中提供所需的功能以实现低延时服务，这可能会导致训练-应用偏差 (https://developers.google.com/machine-learning/guides/rules-of-ml/#training-serving_skew)。
- 不频繁发布迭代：该过程假定您的数据科学团队管理一些不会频繁更改（更改模型实现或使用新数据重新训练模型）的模型。新模型版本每年仅部署几次。
- 无 CI：由于假定几乎不更改实现，因此 CI 已被忽略。通常，测试代码是笔记本或脚本执行的一部分。实现实验步骤的脚本和笔记本由源代码控制，并生成经过训练的模型、评估指标和可视化等工件。
- 无 CD：由于不会频繁部署模型版本，因此不考虑 CD。
- 部署指的是预测服务：该过程仅涉及将经过训练的模型部署为预测服务（例如，具有 REST API 的微服务），而不是部署整个机器学习系统。
- 缺少主动性能监控：该过程不会跟踪或记录模型预测和操作，模型预测和操作是检测模型性能下降和其他模型行为偏移所必需的信息。

工程团队可能会对 API 配置、测试和部署（包括安全、回归以及负载测试和 Canary 版测试）进行自己的复杂设置。此外，在升级模型以处理所有预测请求流量之前，新版机器学习模型的生产部署通常会进行 A/B 测试或在线实验。

挑战

MLOps 级别 0 在许多开始将机器学习应用于其用例的企业中很常见。如果难得更改或训练模型，则由数据科学家驱动的此手动过程可能就足够了。实际上，在现实环境中部署模型时，模型通常会失效。模型无法适应环境的动态变化或描述环境的数据的变化。如需了解详情，请参阅为什么机器学习模型会在生产环境中崩溃和失效 (https://www.forbes.com/sites/forbestechcouncil/2019/04/03/why-machine-learning-models-crash-and-burn-in-production/)

。

如需战胜这些挑战并保持模型在生产环境中的准确性，您需要执行以下操作：

- 在生产环境中主动监控模型的质量：Monitoring 可让您检测性能下降和模型过时情况。它会提示您执行新的实验迭代以及（手动）针对新数据重新训练模型。
- 频繁地重新训练生产模型：如需捕获不断演变和新兴的模式，您需要使用最新数据重新训练模型。例如，如果您的应用使用机器学习推荐时尚产品，则其建议应适应最新的趋势和产品。
- 不断尝试新的实现以生成模型：如需利用最新的技术理念和技术进步，您需要尝试新的实现，例如特征工程、模型架构和超参数。例如，如果您在人脸检测中使用计算机视觉，则人脸模式是固定的，但更好的新技术可以提高检测准确度。

为了战胜此手动过程的挑战，CI/CD 和 CT 的 MLOps 做法很有用。通过部署机器学习训练流水线，您可以启用 CT，并且可以设置 CI/CD 系统以快速测试、构建和部署机器学习流水线的新实现。后续部分将详细讨论这些功能。

MLOps 级别 1：机器学习流水线自动化

级别 1 的目标是通过自动执行机器学习流水线来持续训练模型；这样可以持续交付模型预测服务。如需自动执行在生产环境中使用新数据重新训练模型的过程，您需要在流水线中引入自动化数据和模型验证步骤，以及流水线触发器和元数据管理。

下图是针对 CT 的自动化机器学习流水线的示意图。

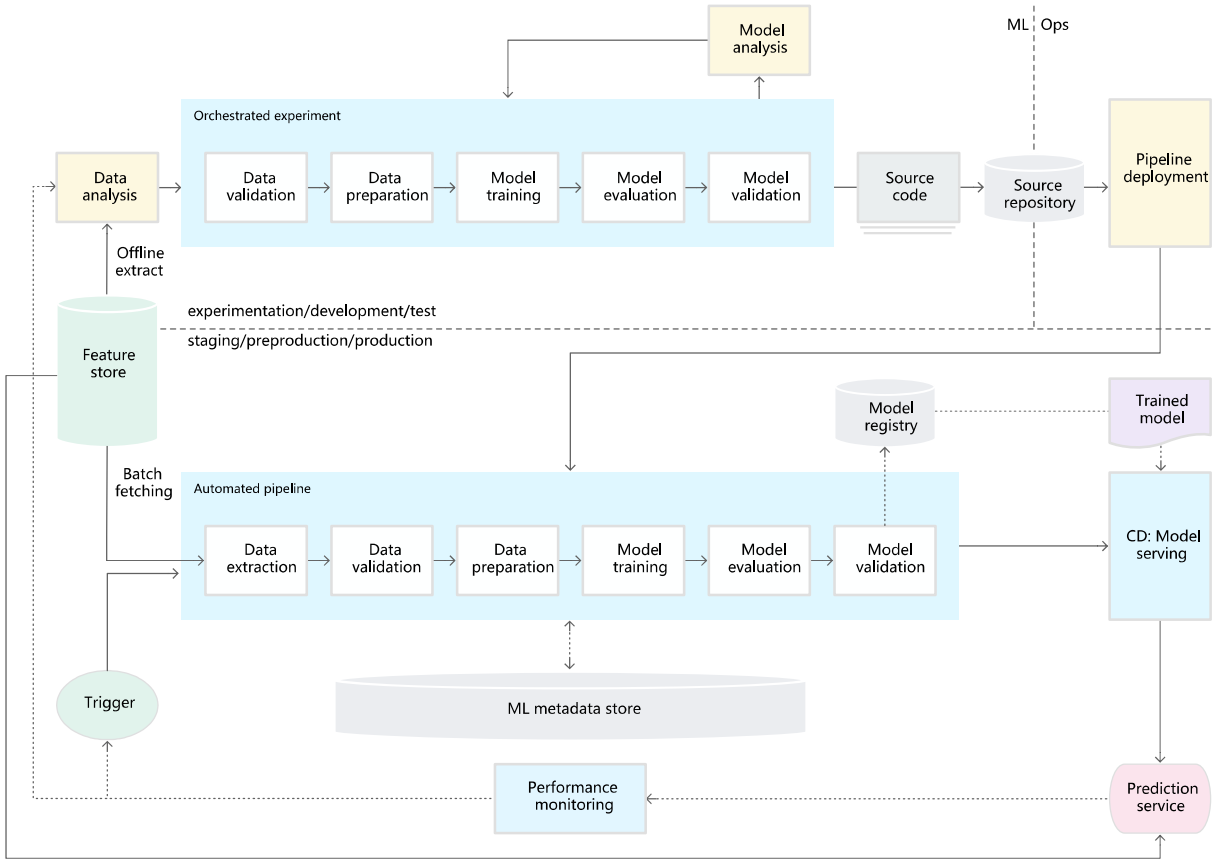

图 3：针对 CT 的机器学习流水线自动化。

特性

以下列表突出显示了 MLOps 级别 1 设置的特性，如图 3 所示：

- 快速实验：机器学习实验的步骤已经过编排。各个步骤之间的转换是自动执行的，这样可以快速迭代实验，并更好地准备将整个流水线移至生产环境。
- 生产环境中的模型 CT：系统会在生产环境中根据实时流水线触发器使用新数据自动训练模型，下一部分将对此进行讨论。
- 实验与操作之间的对称性：在开发或实验环境中使用的流水线实现会在预生产和生产环境中使用，这是 MLOps 做法的主要方面，用于统一 DevOps。
- 组件和流水线的模块化代码：如需构建机器学习流水线，组件必须可重复使用、可组合，并且有望跨机器学习流水线共享。因此，虽然 EDA 代码仍可在笔记本中使用，但组件的源代码必须模块化。此外，组件最好应容器化，以执行以下操作：
    - 将执行环境与自定义代码运行时环境分离。
    - 使代码在开发和生产环境之间可重现。

    - 隔离流水线中的每个组件。组件可以有自己的运行时环境版本，并且可以有不同的语言和库。
- 持续交付模型：生产环境中的机器学习流水线会向使用新数据进行训练的新模型持续交付预测服务。模型部署步骤是自动执行的，该步骤将经过训练和验证的模型用作在线预测的预测服务。
- 流水线部署：在级别 0 中，您可以将经过训练的模型作为预测服务部署到生产环境。对于级别 1，您可以部署整个训练流水线，该流水线会自动重复运行以将经过训练的模型用作预测服务。

其他组件

本部分讨论您需要添加到架构以实现机器学习持续训练的组件。

数据和模型验证

将机器学习流水线部署到生产环境时，机器学习流水线触发器 (#ml_pipeline_triggers)部分中讨论的一个或多个触发器会自动执行流水线。流水线需要新的实时数据来生成使用新数据进行训练的新模型版本（如图 3 所示）。因此，生产流水线需要自动化数据验证和模型验证步骤，以确保实现以下预期行为：

- 数据验证：在模型训练之前需要执行此步骤，以确定您应该重新训练模型还是停止执行流水线。如果流水线识别到以下情况，则系统会自动做出此决策。
    - 数据架构偏差：这些偏差被视为输入数据中的异常情况，这意味着下游流水线步骤（包括数据处理和模型训练）接收的数据不符合预期架构。在这种情况下，您应该停止流水线，以便数据科学团队进行调查。团队可能会发布对流水线的修复或更新，以处理架构中的这些更改。架构偏差包括接收意外特征、未接收所有预期特征或接收具有意外值的特征。
    - 数据值偏差：这些偏差是数据统计属性的重大变化，这意味着数据模式正在变化，您需要触发模型的重新训练才能捕获这些变化。
- 模型验证：此步骤发生在您使用新数据成功训练模型之后。您可以在模型投入生产环境之前对其进行评估和验证。此离线模型验证步骤包含以下操作。
    - 在测试数据集上使用经过训练的模型生成评估指标值，以评估模型的预测质量。
    - 将新训练的模型生成的评估指标值与当前模型（例如生产模型、基准模型或其他业务需求模型）进行比较。在将新模型投入生产环境之前，请确保新模型的性能比当前模型更好。

- 确保模型的性能对于数据的各个细分来说是一致的。例如，新训练的客户流失模型的总体预测准确率可能比先前模型高，但每个客户区域的准确率值可能存在较大的偏差。
- 确保针对部署测试模型，包括测试基础架构与预测服务 API 的兼容性和一致性。

除了离线模型验证之外，新部署的模型在为在线流量提供预测之前会进行在线模型验证（在 Canary 版部署或 A/B 测试设置中进行）。

特征存储区

级别 1 机器学习流水线自动化的一个可选附加组件是特征存储区。特征存储区是一个集中式存储区，您可以在其中对特征的定义、存储和访问进行标准化处理，以方便训练和提供服务。特征存储区需要为特征值提供高吞吐量批量服务和低延时实时服务的 API，以及支持训练和服务工作负载。

特征存储区可帮助数据科学家执行以下操作：

- 发现并重复使用可用于实体的特征集，而不是重新创建相同或类似的特征集。
- 通过保留特征及其相关元数据来避免使用具有不同定义的类似特征。
- 从特征存储区提供最新的特征值。
- 通过将特征存储区用作实验、持续训练和在线服务的数据源，避免训练-应用偏差。此方法可确保用于训练的特征与服务期间使用的特征相同：
  - 对于实验，数据科学家可以从特征存储区中提取离线数据以运行实验。
  - 对于持续训练，自动化机器学习训练流水线可以提取用于训练任务的数据集的一批最新特征值。
  - 对于在线预测，预测服务可以提取与所请求实体相关的一批特征值，例如客户受众特征、产品特征和当前的会话聚合特征。

元数据管理

系统会记录有关机器学习流水线每次执行情况的信息，以帮助实现数据和工件沿袭、可重现性以及比较。这些信息还有助于您调试错误和异常情况。每次执行流水线时，机器学习元数据存储区都会记录以下元数据：

- 执行的流水线和组件版本。
- 开始和结束日期、时间以及流水线完成每个步骤所花费的时间。
- 流水线的执行者。

- 传递给流水线的参数。
- 指向流水线每个步骤生成的工件的指针，例如准备好的数据的位置、验证异常情况、计算的统计信息以及从分类特征中提取的词汇。跟踪这些中间输出有助于您在流水线因某个步骤失败而停止时，从最近的步骤继续执行流水线，而不必重新执行已完成的步骤。
- 指向之前训练的模型的指针（如果您需要回滚到之前的模型版本，或者需要在流水线在模型验证步骤中获得新的测试数据时为之前的模型版本生成评估指标）。
- 在模型评估步骤中为训练集和测试集生成的模型评估指标。这些指标可帮助您在模型验证步骤中将新训练的模型的性能与之前模型的记录性能进行比较。

机器学习流水线触发器

您可以自动执行机器学习生产流水线，以根据您的用例使用新数据重新训练模型：

- 按需：临时手动执行流水线。
- 按时间表：系统每天、每周或每月向机器学习系统提供带有标签的新数据。重新训练的频率还取决于数据模式的更改频率以及重新训练模型的费用。
- 根据新训练数据的可用性：新数据不会系统地提供给机器学习系统，而是在系统收集新数据并在源数据库中提供新数据时临时提供给机器学习系统。
- 在模型性能下降时：模型在性能明显下降时会重新训练。
- 在数据分布发生重大变化（概念偏移 (https://wikipedia.org/wiki/Concept_drift)）时。您很难评估在线模型的完整性能，但会注意到用于执行预测的特征的数据分布发生重大变化。这些变化表明您的模型已过时，需要使用新数据重新训练。

挑战

假设流水线的新实现不会频繁部署，并且您只管理几条流水线，则您通常需要手动测试流水线及其组件。此外，您需要手动部署新的流水线实现。您还需要将经过测试的流水线源代码提交给 IT 团队，以部署到目标环境。如果您根据新数据（而不是新的机器学习理念）部署新模型，则此设置是合适的。

但是，您需要尝试新的机器学习理念，并快速部署机器学习组件的新实现。如果您在生产环境中管理多条机器学习流水线，则需要设置 CI/CD 以自动构建、测试和部署机器学习流水线。

MLOps 级别 2：CI/CD 流水线自动化

如需在生产环境中快速、可靠地更新流水线，您需要一个可靠的自动化 CI/CD 系统。此自动化 CI/CD 系统可让您的数据科学家快速探索有关特征工程、模型架构和超参数的新理念。他们可以实现这些理念，并自动构建、测试新的流水线组件，以及将其部署到目标环境。

下图显示了使用 CI/CD 的机器学习流水线的实现，它具有自动化机器学习流水线设置的特性以及自动化 CI/CD 例程。

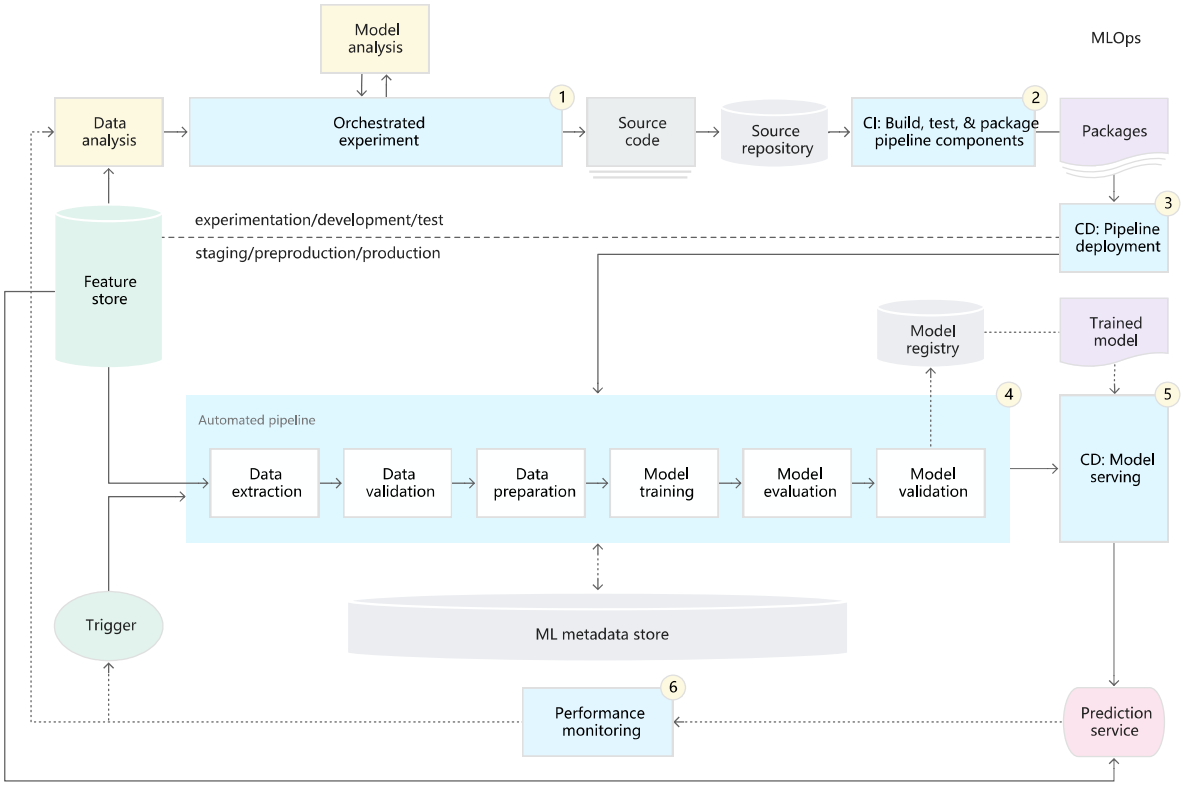

图 4：CI/CD 和自动化机器学习流水线。

此 MLOps 设置包含以下组件：

- 源代码控制
- 测试和构建服务
- 部署服务
- 模型注册表
- 特征存储区
- 机器学习元数据存储区
- 机器学习流水线编排者

特性

下图展示了机器学习 CI/CD 自动化流水线的各个阶段：

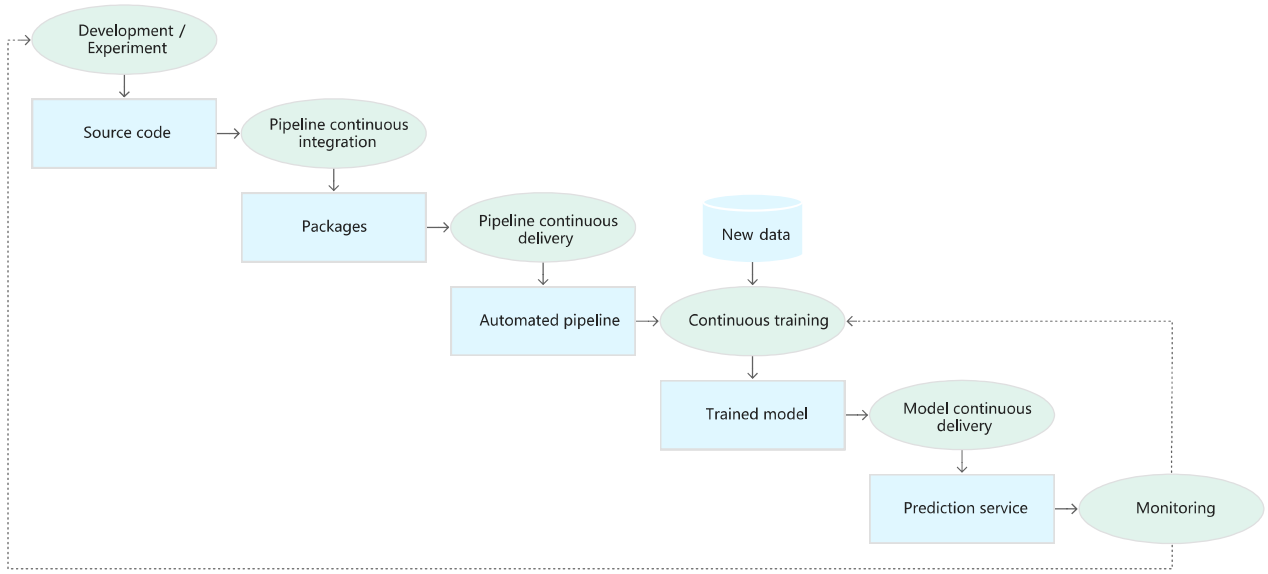

图 5：CI/CD 自动化机器学习流水线的各个阶段。

流水线包括以下阶段：

1. 开发和实验：在编排了实验步骤的阶段，您可以反复尝试新的机器学习算法和新的建模。此阶段的输出是机器学习流水线步骤的源代码，该源代码随后会被推送到源代码库。
2. 流水线持续集成：您可以构建源代码并运行各种测试。此阶段的输出是要在后续阶段部署的流水线组件（软件包、可执行文件和工件）。
3. 流水线持续交付：您可以将 CI 阶段生成的工件部署到目标环境。此阶段的输出是已部署的流水线，其中包含模型的新实现。
4. 自动触发：流水线会根据时间表或响应触发器而自动在生产环境中执行。此阶段的输出是已推送到模型注册表并且经过训练的模型。
5. 模型持续交付：您可以将经过训练的模型用作预测服务。此阶段的输出是已部署的模型预测服务。
6. 监控：您可以根据实时数据收集模型性能的统计信息。此阶段的输出是用于执行流水线或执行新实验周期的触发器。

在流水线开始实验的新迭代之前，数据分析步骤仍然是数据科学家手动执行的过程。模型分析步骤也是手动执行的过程。

持续集成

在此设置中，当新代码提交或推送到源代码库时，您会构建、测试和封装流水线及其组件。除了构建软件包、容器映像和可执行文件之外，CI 过程还可以包含以下测试：

- 对特征工程逻辑进行单元测试。
- 对模型中实现的不同方法进行单元测试。例如，您有一个接受分类数据列的函数，并将该函数编码为独热（https://wikipedia.org/wiki/One-hot)功能。
- 测试您的模型训练是否会收敛（即模型的损失会因迭代而下降，并且会过拟合（https://wikipedia.org/wiki/Overfitting)一些示例记录）。
- 测试模型训练是否不会因为除以零或者处理小值或大值而产生 NaN（https://wikipedia.org/wiki/NaN) 值。
- 测试流水线中的每个组件都会生成预期的工件。
- 测试流水线组件之间的集成。

持续交付

在此级别中，您的系统会向目标环境持续交付新的流水线实现，从而交付新训练的模型的预测服务。如需快速、可靠地持续交付流水线和模型，您应考虑以下事项：

- 在部署模型之前，验证模型与目标基础架构的兼容性。例如，您需要验证模型所需的软件包是否已安装到服务环境中，以及内存、计算和加速器资源是否可用。
- 测试预测服务，方法是：使用预期输入调用服务 API，并确保获得预期响应。此测试通常会捕获您在更新模型版本时可能会出现的问题，它需要您提供不同的输入。
- 测试预测服务性能，包括对服务进行负载测试以捕获每秒查询次数 (QPS)（https://wikipedia.org/wiki/Queries_per_second) 和模型延迟时间等指标。
- 验证数据以便重新训练或者进行批量预测。
- 在部署模型之前，验证其是否达到预测性能目标。
- 自动部署到测试环境，例如通过将代码推送到开发分支而触发的部署。
- 半自动部署到预生产环境，例如通过在审核者批准更改后将代码合并到主分支而触发的部署。
- 在预生产环境中多次成功运行流水线后，手动部署到生产环境。

总而言之，在生产环境中实现机器学习并不仅意味着将模型部署为用于预测的 API。相反，它意味着部署可自动重新训练和部署新模型的机器学习流水线。通过设置 CI/CD 系统，您可以自动测试和部署新的流水线实现。此系统可让您应对数据和业务环境的快速变化。您不必

立即将所有过程从一个级别迁移到另一个级别。您可以逐步实现这些做法，以帮助改进机器学习系统的开发和生产自动化。

后续步骤

- 详细了解使用 Kubeflow 和 Cloud Build 的 CI/CD 和机器学习流水线架构 (/solutions/machine-learning/architecture-for-mlops-using-tfx-kubeflow-pipelines-and-cloud-build)。
- 详细了解如何通过 Cloud Build 实现 GitOps 式持续交付 (/kubernetes-engine/docs/tutorials/gitops-cloud-build)。
- 详细了解如何为数据处理工作流设置 CI/CD 流水线 (/solutions/cicd-pipeline-for-data-processing)。
- 在 YouTube 上观看 Google Cloud 上的 MLOps 最佳做法 (Cloud Next '19) (https://www.youtube.com/watch?v=20h_RTHEtZI)。
- 探索有关 Google Cloud 的参考架构、图表、教程和最佳做法。查看我们的云架构中心 (/architecture)。

Last updated 2021-07-19 UTC.