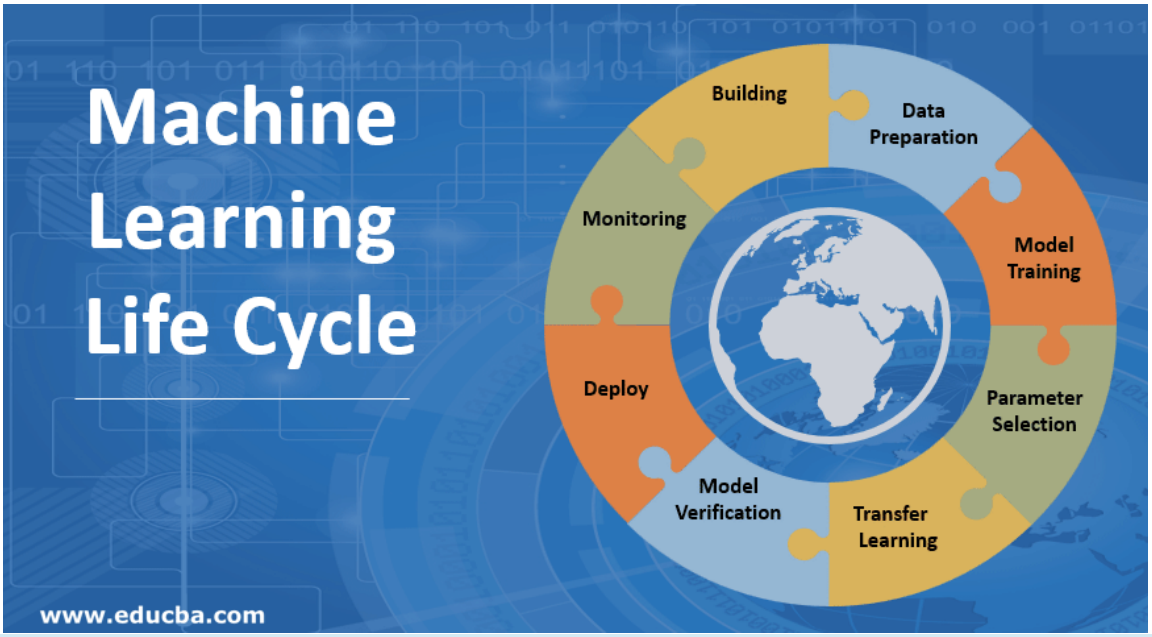
MLFLOW

MACHINE LEARNING LIFE CYCLE MANAGEMENT

Date: 04/21/2022



ABSTRACT

Data science (DS) projects are intrinsically interdisciplinary in nature, including teams with varied degrees of computational knowledge and experience with data management. The underlying Machine Learning (ML) methods and analysis workflows utilized in DS projects are frequently composed of constantly growing open-source software stacks, with analysis activities executed on diverse computational infrastructure (workstations, HPC, Cloud, etc.). Managing the data and analysis lifecycle is critical for the reproducibility and long-term sustainability of any DS project. Any DS project must maintain a record of the results of thorough testing, as well as the parameters, metrics, artifacts, source code, and package dependencies connected with them. Additionally, many team members must be able to navigate this data, which necessitates the usage of a platform-independent framework and a robust model provenance system (storage, versioning, reproducibility). We've all had to construct or access an older model in order to react to a colleague or reviewer's question, but do we know the fundamental parameters? Model development, iterative experimentation, and

deployment can be accelerated with machine learning lifecycle management such as MLFlow. They enable individuals or teams of data scientists to develop robust and repeatable machine learning pipelines with platform-agnostic model packaging, deployment, and versioning, as well as quality assurance, all without sacrificing their preferred programming language or library, which is critical for the productivity of any DS project.

CHALLENGES IN (TRADITIONAL) ML DEVELOPMENT

Typical machine learning projects need to track a diverse set of inputs and results. We frequently conduct a huge number of experiments and must keep track of not only the outputs but also the parameters, code, models, and artifacts. Additionally, the majority of machine learning projects involve numerous team members who require access to both experiments and the most recent code versions. This demands a significant degree of coordination amongst all team members.

HOW MLFLOW CAN HELP

Custom machine learning platforms, such as Facebook's FBLearner and Uber's Michelangelo, exist to address the challenges, but are not publicly available. Additionally, they tailor their frameworks to match their unique requirements. MLFlow's project component provides a platform-independent environment that enables any team member, regardless of the target system, to access the project and, if necessary, replicate any experiment. This is designed to be readily integrated into current applications due to its extensive support for a range of machine learning frameworks and languages, including Python, R, and Java. The project component's user-friendly design and accessibility will appeal to a diverse range of teams. The tracking function enables developers to save every piece associated with an experiment or model, from the code version to the model's parameters and metrics. Its model registry simplifies the process of deploying or accessing already-deployed production models. Additionally, developers can utilize this model registry to monitor the performance of deployed models over time. It is compatible with widely used environment management frameworks such as CONDA and DOCKER, the latter of which can be quickly deployed to multi-cluster production sites with a few Docker filechanges.