Monitoring Machine Learning Solutions

The machine learning lifecycle presents a structured method to prepare data, develop and evaluate algorithms and models, and create analytic solutions to solve business problems. When a machine learning or analytic solution is developed to meet these business needs, you should consider the following questions:

1. How can you ensure that you are truly providing an end-to-end machine learning development process and providing your clients with the best experience and solution?
2. How can an ML Engineer ensure that they have developed a model that can continue to improve itself?

Let’s answer these questions by learning about monitoring machine learning solutions in production.

At the end of this module, you will be able:

* Define Machine Learning Operations a.k.a. MLOps
* Use Datadog to monitor a deployed machine learning solution.

**The Machine Learning Lifecycle**

The job of a Machine Learning Professional involves an estimated 70% data cleansing and 30% model development and analysis. The process of developing a machine learning solution is often structured and follows a methodology called the *machine learning lifecycle.*

The lifecycle as shown below in *figure 1*, can be modified to suit the structure of the machine learning team and the needs of the client. The typical ML lifecycle includes:

* Data Gathering- the collection of data of interest including variables and observations to enable a data profession to answer a stated question.
* Data Preparation- involves transforming raw data by cleaning and reformatting it to provide an enriched dataset used to fit a model.
* Data Analysis- involves inspecting and modeling data to uncover initial insights that can help inform decision making.
* Model Training- The ML Engineer will develop a model at this stage. The model will learn from a dataset called the training set. The training set is a subset of the overall dataset.
* Model Testing- involves the evaluation of a trained model using a subset of the dataset called test data.
* Deployment- models are integrated into an existing production environment to meet the defined business needs.

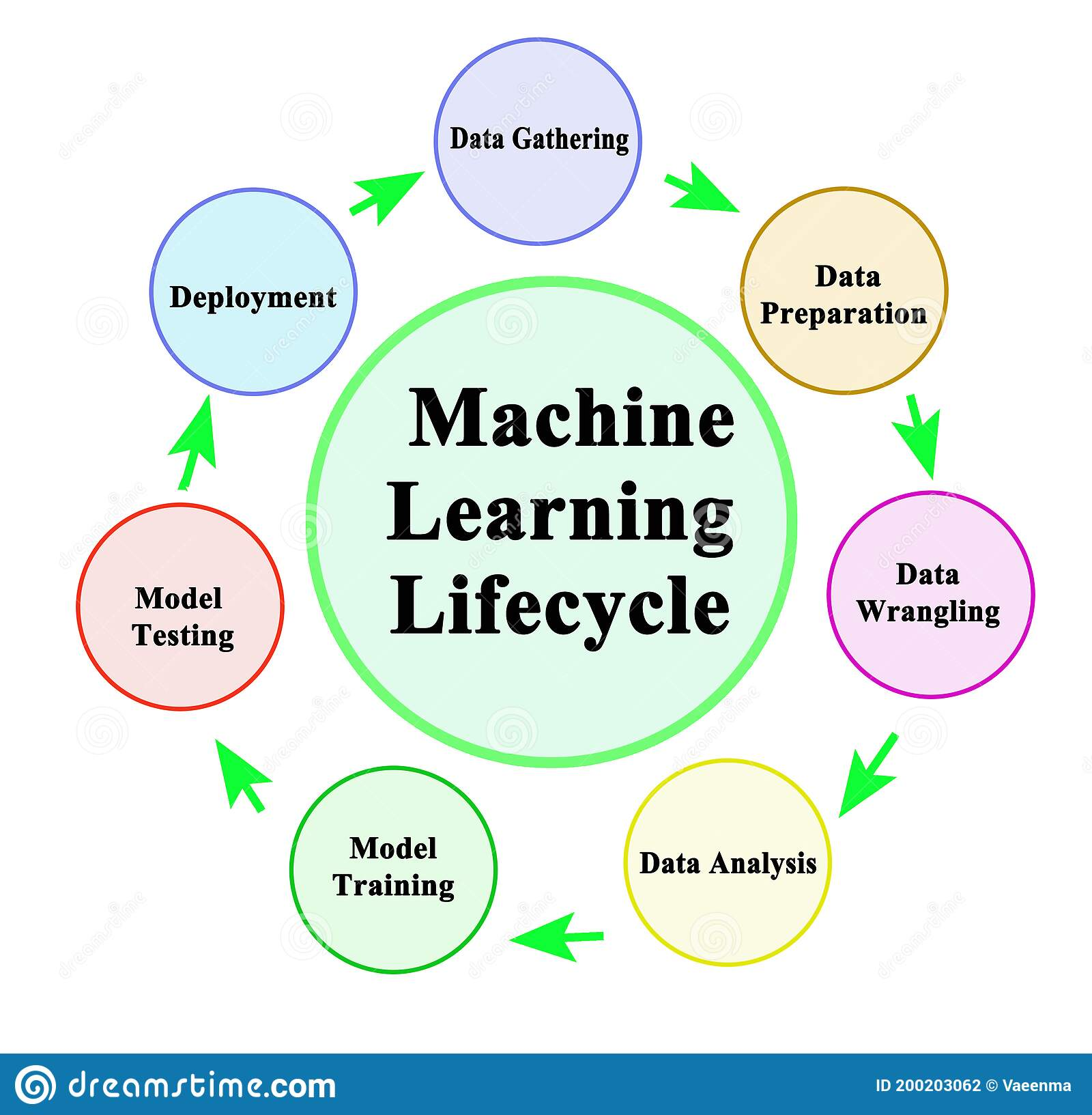


Figure 1: Machine Learning Lifecycle-*Vae Enma*

**Machine Learning Operations**

The ML solution, although successfully deployed, must be continuously monitored to ensure that the solution is not degraded, when new circumstances such as data and environment changes are introduced. This monitoring process is known as **Machine Learning Operations** or **MLOps**. MLOps takes into account the processes, frameworks, and tools needed to ensure that machine learning models and solutions remain relevant after deployment to production. MLOps introduces monitoring and improvement to the ML lifecycle.

MLOps is essential to the success of a machine learning solution and this is reflected in the revised ML Lifecycle as shown below. The monitoring process ensures that models can continue to learn from data by retraining itself.

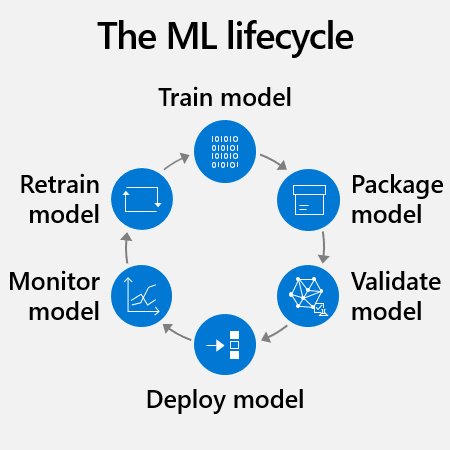


Figure 2: Machine Learning Lifecycle-Microsoft

**MLOps Using Datadog**

Datadog provides multiple tools and features that can be used in measuring, monitoring, detecting and securing deployed machine learning models.

Let us learn about select platforms and features used for MLOps.

**Algorithmia**

Datadog’s Algorithmia provides a platform that is suitable for deploying, monitoring, and securing machine learning models in production. This platform was launched in 2020. Algorithmia can also be used to monitor metrics including data and model drift and model bias.

**Watchdog**

As ML Engineers often develop applications and other solutions, it is important to also monitor the performance and overall infrastructure of those applications. Datadog’s Watchdog is a feature that can be used for Application Performance Management (APM).

**Practical Application of MLOps in Datadog**

Now that we have learned about the fundamentals of MLOps, it is time for you to practice how to monitor your machine learning models.

Detecting Data Drift

Data drift can be defined as a variation in data that was used to validate a model and production data. Data drift happens because the time when data is collected, used to train and validate a model, and when a model interacts with real time data can have a significant gap (in terms of weeks, months, and sometimes years). Data drift can also happen if there is a change in circumstances.

***Scenario***

First Bank of South Dakota’s (FBSD) customer retention unit has discovered that there has been a reduction in its customer base year over year from 2000-2010. The Bank has grown its customer base by over 30% through promotions and offering online only banking options. These efforts have garnered customer interest from people and businesses across the country. FBSD has set a creditworthiness FICO 7 score of 560 minimum to allow customers to sign up for an account. In 2014, FICO recently introduced the FICO 10 scoring. The FICO 10 equivalent of FBSD’s minimum creditworthiness is 90.

You have been tasked to develop a model to determine the right type of customers to send promotions and advertisements and whether they will churn or not. Assume that you have developed a Customer Churn Prediction model using the Random Forest(RF) technique using customer data from 2000-2010.

**Problem:** You have discovered that your model is not recognizing the new FICO 10 score that has been introduced to the model. The model is seeing customers with a FICO 10 score of 90 and above and rejecting those customers. The model has assumed that customers with a 90 score are well below the 560 threshold.

**Question:** How can we monitor this model to help it learn the real world data?

**Proposed Solution:** We will use MLOps techniques discussed above to monitor the model in production. We will use Algorithmia to monitor the model for data drift.

Follow the steps below to create your first metric monitor to ensure that your team is aware of any data drifts.

**Setup Insights**

* Follow this tutorial to set up your data drift monitor: [How to: Algorithmia Insights](https://www.youtube.com/watch?v=pdKwtp-_n2M)**.**
* Configure Algorithmia to a Kafka Broker.
* You want to set metrics from the RF algorithm and connect your Kafka pipeline output.
* Ensure that you have included the source code (in github) for the Customer Churn model.

***View Metrics in Datadog***

* In the Datadog interface, navigate to the Metrics summary page.
* Verify that the metrics set above from Insights are present in Datadog by filtering for “algorithmia”