Model Deployment

Introduction to Model Deployment

dφ

Democratizing Data Science Learning

DPhi

Learning Objectives

Environments

What Is Model Deployment?

Why Model Deployment?

Where does it fit in the ML Pipeline?

Different modes of Model Deployment

Methods and Techniques of Model Deployment

DPhi

Environments

There are three different environments that you'll probably deal with at some point. Each environment has its own properties and uses. Thus, it is important to use them accordingly.

1. **Development**

This is the environment that's on your computer. Here is where you'll do all of your code updates. Nothing you do in the development environment affects what users currently see when they open the website. This is just for you and the other web devs to see how new features will work and to try out improvements.

2. **Stage**

Think of the stage environment as the place you do the last checks and you polish things up. It is as similar to the production environment as it can be. If you have a client, this is when you would be able to give them a demo of how things work and look. They will be able to see how things will work when they make it live and they will be able to give you any feedback you need.

3. Production Environment

Finally, While learning about Model Deployment, you'll commonly hear a word - 'production' or 'production environment'.

Every time you talk about making your project live, this is the environment you are talking about.

A production environment is a term used to describe the setting where software and other products are actually put into operation for their intended uses by end users.

Of all the environments, this one is the most important.

Imagine

- You have spent several weeks or months building a machine learning model that would identify if a person has put a mask on his/her face or not with a very good accuracy score.
- Sounds great, right? Is this all you wanted?
- No. Building a model is generally not the end of the project.
- You would want your model to be used in real time where it could identify people in parks, at bus stations, streets, etc. with no mask and immediately inform the people nearby to maintain sufficient distance.
- This is where model deployment comes into picture.

What Is Model Deployment?

- The concept of deployment in data science refers to the application of a model for prediction using new data.
- In other words, when you use your trained ML model to make a prediction of new data available to users or other systems, it is called model deployment.
- In technical terms, model deployment means to integrate a machine learning model into an existing production environment where it can take in an input and return an output.

Why Model Deployment?

- A machine learning model can begin to add value to an organization **only when that model's insight is timely available to the users for which it was built**.
- Even if the aim of the model is to increase the knowledge of the data, the **knowledge gained should be organized and presented in a way that a customer will use it.**
- **If you directly present the model's code, the customers cannot understand.** You need to provide an user interface.
- To achieve this, you need to deploy the model on the web.

Why Model Deployment?

Where does it fit in the ML Pipeline?

Let's have a broad overview of the whole pipeline for a ML project:

1. Collecting data
2. Exploratory Data analysis
3. Feature engineering
4. Feature selection
5. Training the machine learning model, and
6. Model deployment

So as you can see, model deployment is usually one of the last steps of the ML Pipeline.

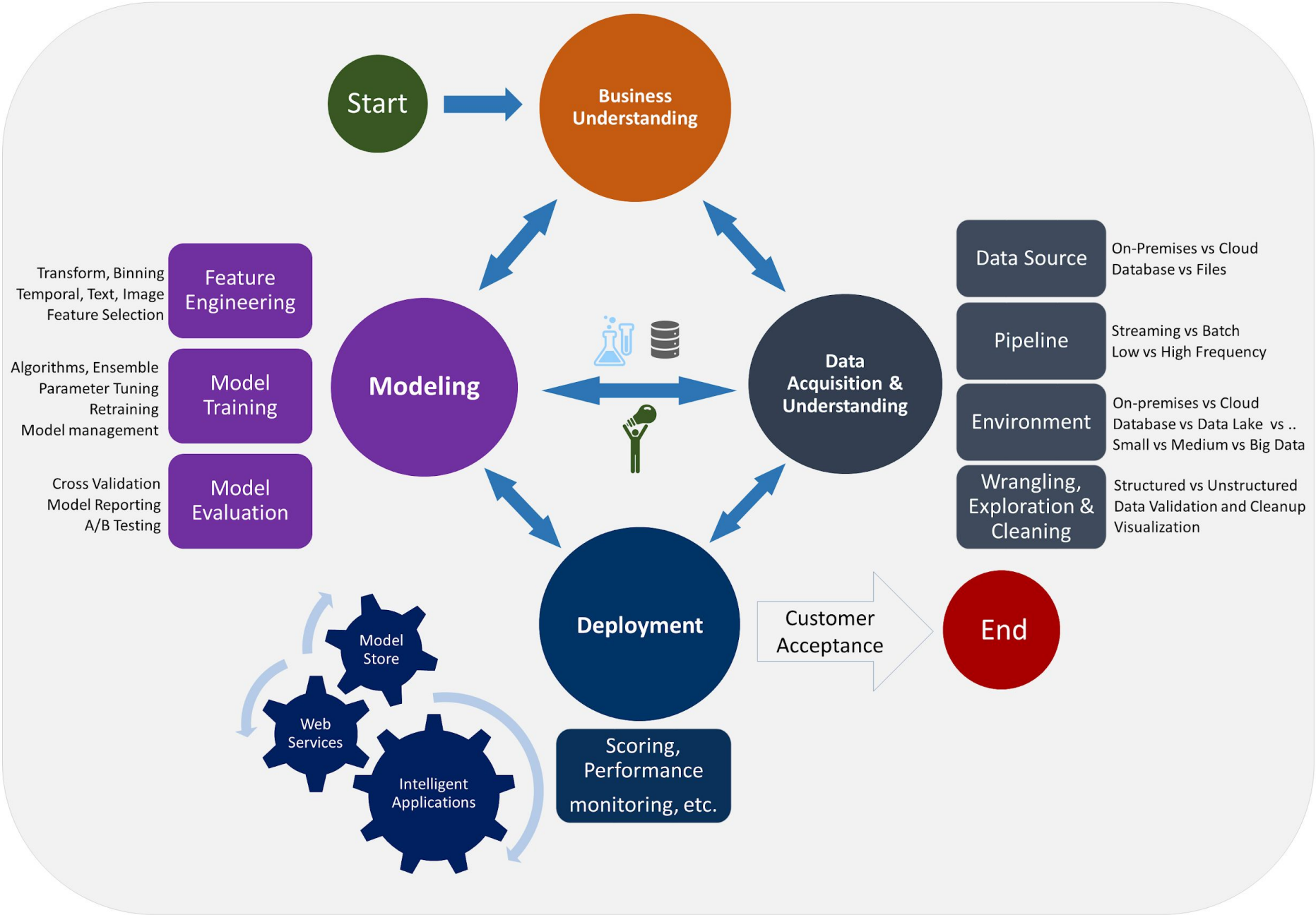
Is Model Deployment the end?

Surprisingly, deploying a model is not the end. When your model is deployed in a setup where it is being consistently used, it needs monitoring as well!

Think of monitoring an ML model the same way as you would think about getting your annual physical check-up or getting periodic oil changes for your car. Model modeling is an operational task that allows you to check that your model is performing to the best of its abilities.

Have a look at the beautifully elaborated Data Science Lifecycle in the next slide.

Data Science Lifecycle



Some criterion to consider

When designing an ML system architecture, the following should be borne in mind in order to make your deployment process easier:

- **Modularity:** the code used in preprocessing/feature engineering should be arranged in comprehensive pipelines.
- **Reproducibility:** the output of each component must be replicable for any version in time.
- **Portability:** this refers to the ability of your software to be transferred from one machine or system to another.
- **Scalability:** the model must be serveable to a large number of customers with minimal response time.
- **Extensibility:** it should be easy to modify for future tasks.
- **Testing:** the ability to test variation between model versions.
- **Automation:** eliminating manual steps wherever possible to reduce error chances.

Different modes of Model Deployment

1. **One-off**

It's not always that you need to continuously train a machine learning model in order for it to be deployed. Sometimes a model is only needed once or periodically. In this case, the model can simply be trained ad-hoc when it's needed and pushed to production until it deteriorates enough to need some fixing.

2. **Batch**

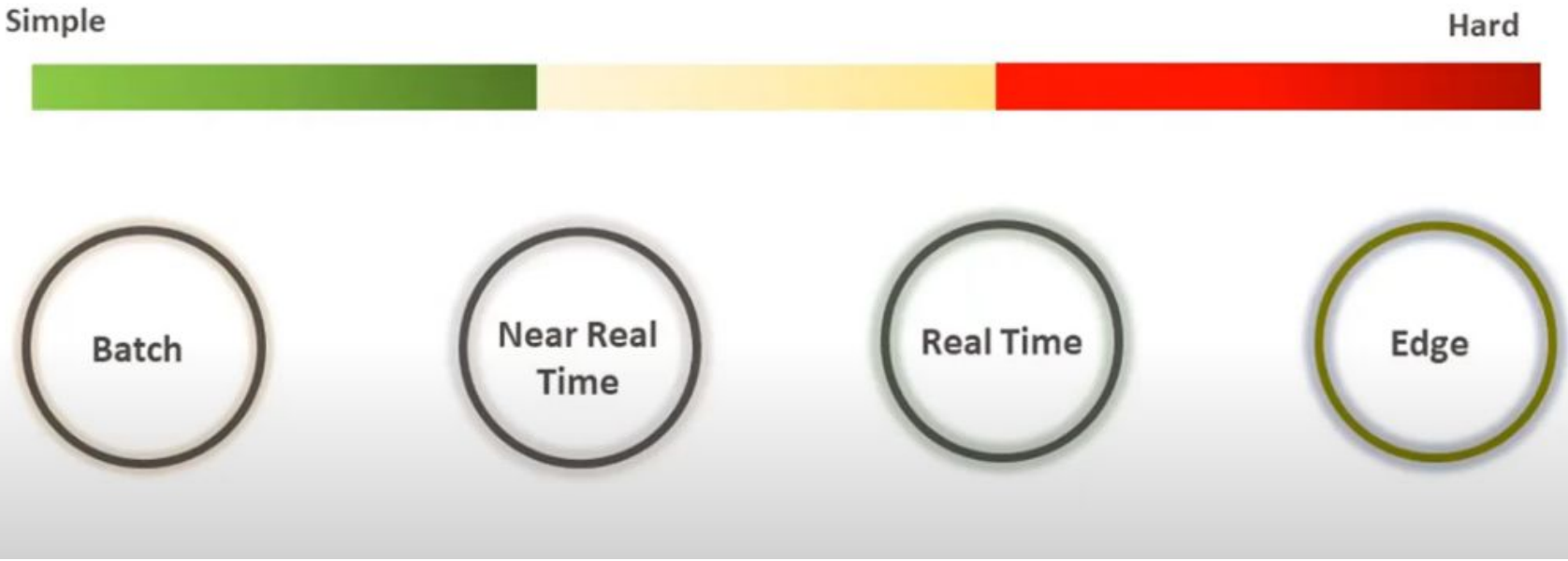
Batch training allows you to constantly have an up-to-date version of your model. It is a scalable method that takes a subsample of data at a time, eliminating the need to use the full data set for each update. This is good if you use the model on a consistent basis, but don't necessarily require the predictions in real-time.

3. **Real-time**

In some cases you'll want a prediction in real-time, for example, to determine if a transaction is fraudulent or not. This is possible by using online machine learning models, like linear regression using stochastic gradient descent.

Apart from this, deployment can also be Near Real Time or Edge. Have a look at the image in the next slide.

Different modes of Model Deployment



Different modes of Model Deployment

Methods and Techniques of Model Deployment

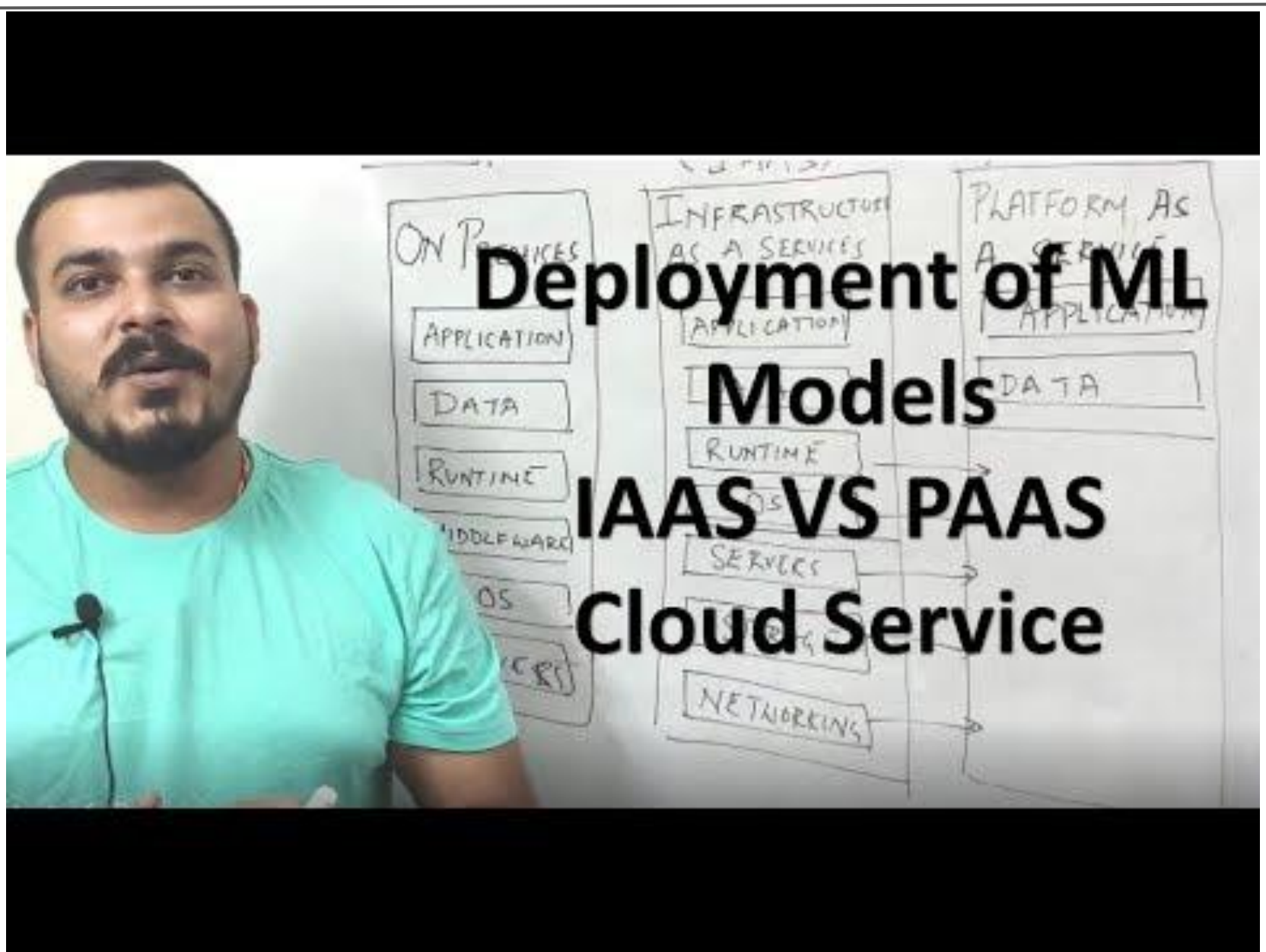
When it comes to deployments, you need to decide if you're going to go with a Platform as a Service (PaaS) or Infrastructure as a Service (IaaS).

SaaS (Software as a Service), PaaS, and IaaS are simply three ways to describe how you can use the cloud for your business. Simply put:

- ➔ IaaS: cloud-based services, pay-as-you-go for services such as storage, networking, and virtualization.
- ➔ PaaS: hardware and software tools available over the internet.
- ➔ SaaS: software that's available via a third-party over the internet.

- A PaaS can be great for prototyping and businesses with lower traffic.
- Eventually, once the business grows and/or traffic increases, you're going to need to embrace more complexity with IaaS. There are plenty of solutions from the usual suspects (AWS, Google, Microsoft), as well as an army of niche players.
- If you've never deployed anything before, the recommended way is to start with Heroku and that's what we're going to do in this course!

Methods and Techniques of Model Deployment



DPhi

References

- https://dev.to/flippedcoding/difference-between-development-stage-and-production-d0p
- https://towardsdatascience.com/what-does-it-mean-to-deploy-a-machine-learning-model-dddb983ac416
- https://christophergs.com/machine%20learning/2019/03/17/how-to-deploy-machine-learning-models/

DPhi

Slide Download Link

You can download this unit from the below link:

https://docs.google.com/presentation/d/1f9NmE_nczJ87y8aMaAWjrjm_meN2SEbPea5TGy3jqEk/edit?usp=sharing

DPhi

That’s it for this unit. Thank you!

Feel free to post any queries on Discuss.

DPhi