Machine Learning Cloud Platforms

There are a number of machine learning cloud platforms, we provide *more* details about *a few* **below**. In the *next* few lessons, you will learn how to use **Amazon's SageMaker** to *deploy* machine learning models. Therefore, we *focused* on providing *more* information on Amazon's SageMaker. To allow for a comparison of features offered by SageMaker, we also provide detailed information about Google's ML Engine because it's *most* similar to **SageMaker**.

Amazon Web Services (AWS)

Amazon Web Services (AWS) SageMaker is Amazon's cloud service that allows you to *build*, *train*, and *deploy* machine learning models. Some advantages to using Amazon's SageMaker service are the following:

- **Flexibility in Machine Learning Software**: SageMaker has the flexibility to enable the use of ***any*** programming language or software framework for building, training, and deploying machine learning models in **AWS**. For the details see the three methods of modeling within **SageMaker below**.
    - Built-in Algorithms - There are at least fifteen built-in algorithms that are easily used within SageMaker. Specifically, built-in algorithms for discrete classification or quantitative analysis using linear learner or XGBoost, item recommendations using factorization machine, grouping based upon

attributes using K-Means, an algorithm for image classification, and many other algorithms.

    - **Custom Algorithms** - There are different programming languages and software frameworks that can be used to develop custom algorithms which include: PyTorch, TensorFlow, Apache MXNet, Apache Spark, and Chainer.
    - Your Own Algorithms - Regardless of the programming language or software framework, you can use your own algorithm when it **isn't** included within the *built-in* or *custom algorithms* **above**.
- **Ability to Explore and Process Data within SageMaker**: SageMaker enables the use of Jupyter Notebooks to explore and process data, along with creation, training, validation, testing, and deployment of machine learning models. This notebook interface makes data exploration and documentation easier.
- **Flexibility in Modeling and Deployment**: SageMaker provides a number of features and automated tools that make ***modeling*** and ***deployment*** easier. For the details on these features within **SageMaker** see **below**.
    - Automatic Model Tuning: SageMaker provides a feature that allows hyperparameter tuning to find the **best** version of the model for *built-in* and *custom algorithms*. For built-in algorithms **SageMaker** also provides evaluation metrics to evaluate the performance of your models.
    - Monitoring Models: SageMaker provides features that allow you to monitor your *deployed* models. Additionally with *model deployment*, one can choose *how much* traffic to route to *each* deployed model (model variant). More information on routing traffic to model variants can be found here and here .
    - **Type of Predictions**: SageMaker by *default* allows for On-demand type of predictions where *each* prediction *request* can contain *one* to *many* requests.

SageMaker also allows for Batch predictions, and request *data* size limits are based upon S3 object size limits.

Google Cloud Platform (GCP)

Google Cloud Platform (GCP) ML Engine is Google's cloud service that allows you to *build*, *train*, and *deploy* machine learning models. Below we have highlighted some of the ***similarities*** and ***differences*** between these two cloud service platforms.

- **Prediction Costs**: The ***primary difference*** between the two is how they handle predictions. With ***SageMaker*** *predictions*, you must leave resources running to provide predictions. This enables *less* latency in providing predictions at the *cost* of paying for running *idle* services, if there are no (or few) prediction requests made while services are running. With ***ML Engine*** *predictions*, one has the option to *not* leave resources running which reduces cost associated with *infrequent* or *periodic* requests. Using this has *more* latency associated with predictions because the resources are in a offline state until they receive a prediction request. The *increased* latency is associated to bringing resources back online, but one *only* pays for the time the resources are *in use*. To see more about ML Engine pricing and SageMaker pricing.
- **Ability to Explore and Process Data**: Another ***difference*** between **ML Engine** and **SageMaker** is the fact that *Jupyter Notebooks* are not available within **ML Engine**. To use *Jupyter Notebooks* within **Google's Cloud Platform** (GCP), one would use Datalab. **GCP** separates data exploration, processing, and transformation into other services. Specifically, Google's Datalab can be used for data exploration and data

processing, Dataprep can be used to explore and transform raw data into clean data for analysis and processing, and DataFlow can be used to deploy batch and streaming data processing pipelines. Noting that **Amazon Web Services** (AWS), also have data processing and transformation pipeline services like AWS Glue and AWS Data Pipeline.

- **Machine Learning Software**: The final ***difference*** is that Google's ML Engine has *less* flexibility in available software frameworks for building, training, and deploying machine learning models in **GCP** as compared to **Amazon's SageMaker**. For the details regarding the two available software frameworks for modeling within **ML Engine** see **below**.
    - Google's TensorFlow is an open source machine learning framework that was originally developed by the Google Brain team. TensorFlow can be used for creating, training, and deploying machine learning and deep learning models. Keras is a higher level API written in Python that runs on top of TensorFlow, that's easier to use and allows for faster development. GCP provides both TensorFlow examples and a Keras example.
    - Google's Scikit-learn is an open source machine learning framework in Python that was originally developed as a Google Summer of Code project. Scikit-learn and an XGBoost Python package can be used together for creating, training, and deploying machine learning models. In the in Google's example, XGBoost is used for modeling and Scikit-learn is used for processing the data.
- **Flexibility in Modeling and Deployment**: Google's ML Engine provides a number of features and automated tools that make ***modeling*** and ***deployment*** easier, ***similar*** to the those provided by **Amazon's SageMaker**. For the details on these features within **ML Engine** see **below**.

- Automatic Model Tuning: Google's ML Engine provides a feature that enables hyperparameter tuning to find the **best** version of the model.
- Monitoring Models: Google's ML Engine provides features that allow you to monitor your models. Additionally ML Engine provides methods that enable managing runtime versions and managing models and jobs.
- **Type of Predictions**: ML Engine allows for Online(*or On-demand*) type of predictions where *each* prediction *request* can contain *one* to *many* requests. **ML Engine** also allows for Batch predictions. More information about **ML Engine's** Online and Batch predictions.

Microsoft Azure

Similar to Amazon's SageMaker and Google's ML Engine, Microsoft offers Azure AI. **Azure AI** offers an open and comprehensive platform that includes AI software frameworks like: TensorFlow, PyTorch, scikit-learn, MxNet, Chainer, Caffe2, and other software like their Azure Machine Learning Studio. For more details see Azure AI and Azure Machine Learning Studio.

Paperspace

Paperspace simply provides GPU-backed virtual machines with industry standard software tools and frameworks like: TensorFlow, Keras, Caffe, and Torch for machine learning, deep learning, and data science. **Paperspace** claims to provide more powerful and less expensive virtual machines than are offered by **AWS**, **GCP**, or **Azure**.

Cloud Foundry

Cloud Foundry is an open source cloud application platform that's backed by companies like: Cisco, Google, IBM, Microsoft, SAP, and more. Cloud Foundry provides a faster and easier way to build, test, deploy, and scale applications by providing a choice of clouds, developer frameworks, and applications services to it's users. Cloud Foundry Certified Platforms provide a way for an organization to have their cloud applications portable across platforms including IBM and SAP cloud platforms.

NEXT