**AWS Sagemaker- Detracting the ML heavy lifting**

Nowadays, the focus on machine learning in cloud has risen due to several advantages like pay-per-use model, high computation power and availability. The best part about ML in cloud is that we can begin testing with smaller projects and then scale up in case of high demand, this prevents starting off in full-fledged mode and keeps the window to experiment open. ML on AWS cloud is comprising of multiple services which are used for pattern identification and responsive application development.

**What is Sagemaker?**

AWS Sagemaker is service belonging to the ML stack, it is fully managed service which allows development and training of ML models in production-ready environment. The need of server management is diminished as it has an integrated Jupyter notebook which is used to explore and analyse datasets. Popular ML algorithms are optimized to handle heavy workloads, also bring-your-own-algorithms are well supported by offering customisable distributions options. Generally, any ML model will learn from millions of data items in production environment and the observations are produced in time-gap as less as 20 ms so let’s explore below that how AWS Sagemaker makes it work end-to-end.

* **Working with ML**
* **Exploration and pre-processing of data**
* **Training ML models //checkpointing**
* **Deploying the ML model**
* **Monitoring the model deployed in production**

**How local training & deployment waste time in ML related tasks**

* Setting up or maintaining notebook servers
* Bargaining with co-workers for compute resources
* Troubleshooting conflicting package installations
* Not knowing how your model is performing while training
* Getting your cluster back online
* Stalling in front of your customers while your inferencing starts
* Not being able to replicate model training jobs
* Not having training materials for new hire or for new projects
* Grid, random or manual hyperparameter searching or none at all
* Not having a continuous integration pipeline

**Sagemaker USP - a game changing solution for the enterprise**

* In-built Jupyter noteBooks running R/Python kernels with a compute instance as per on demand data engineering requirements.
* Visualization, processing, cleaning and transforming of the data into required forms using the traditional methods (Pandas + Matplotlib, R +ggplot2 or other combinations).
* Post data engineering, the models are trained using a different compute instance on the basis of compute demand, like GPU enabled, optimized memory.
* Leverages smart default high-performance hyperparameter tuning setting
* Leverage performance-optimized algorithms from AWS library
* Supports bring our own algorithms through industry standard containers.
* Deploys the trained model as an API with a few lines of code
* Cost effective

**Best practices**

**Debugger point**

An *inference pipeline* is an Amazon SageMaker model that is composed of a linear sequence of two tofive containers that process requests for inferences on data. You use an inference pipeline to define and deploy any combination of pretrained Amazon SageMaker built-in algorithms and your own custom algorithms packaged in Docker containers. You can use an inference pipeline to combine pre-processing, predictions, and post-processing data science tasks. Inference pipelines are fully managed.

**Few evident use-cases**

**Image analysis, performance analysis, forecasting, fraud and anomaly detection, predictive analysis,**

* **Medical image analysis:** Medical fraternityspends considerable time searching X-rays to identify diseases and illnesses. There is increased use of machine learning models to compare patient images to identify anomalies and suggest diagnoses which provides earlier and more accurate detection.
* **Sports performance analysis:** Major games and events now use machine learning on AWS to drive stats for their live broadcasts and other applications. For eg: SageMaker processes pitcher and hitter stats as per current conditions to suggest future pitch selection, weakest batsmen selection etc.
* **Product forecasting:** Variouscompanies now use ML on AWS for analyzing sales and detailed manufacturing data which helps to predict upcoming demand on the basis of economic, seasonal and other factors. SageMaker provides detailed predictions and recommendations for the product supply chain.
* **Bank or credit card fraud detection:** The finance industry has made very good use of machine learning technology by using SageMaker to prepare models which monitor banking and credit related activities for signs of fraud.
* **Marketing predictions:** SageMaker models are used to take accurate marketing decisions, on the basis of past behaviors and other parameters that the models have identified. For eg., a marketing person analyses customer's browsing habits based on which tailored ads ads are rendered to boost sales.
* **Data analysis:** Machine learning helps data analysts spot correlations, cause and effect, and other relationships in complex and disparate data sets. For example, SageMaker could support models for population analysis and use census data to identify population trends that help a government better allocate funding.

Some other market alternatives for Sagemaker are IBM Watson Studio, Google Cloud AI Platform, Azure Machine Learning Studio, TensorFlow, Google Cloud AutoML, DataRobot, Dataiku DSS, H2O. In conclusion, the future of any kind of application development will be supported by machine learning, so services like AWS Sagemaker will revolutionize the adaptability of ML in day-to-day applications. It is highly flexible and cost-effective and it offers easy integration with MapReduce and other AWS services which makes it easier to deploy ML models as compared to open-source tools. It also has sample notebooks to get you started with image classification, text classification etc and is a great service for visually seeing the ML models develop step-by-step.

**Relevant links:**

[**https://aws.amazon.com/blogs/machine-learning/bring-your-own-model-with-amazon-sagemaker-script-mode/**](https://aws.amazon.com/blogs/machine-learning/bring-your-own-model-with-amazon-sagemaker-script-mode/)

[**https://aws.amazon.com/blogs/machine-learning/building-algorithmic-trading-strategies-with-amazon-sagemaker/**](https://aws.amazon.com/blogs/machine-learning/building-algorithmic-trading-strategies-with-amazon-sagemaker/)

[**https://aws.amazon.com/blogs/machine-learning/deploy-shadow-ml-models-in-amazon-sagemaker/**](https://aws.amazon.com/blogs/machine-learning/deploy-shadow-ml-models-in-amazon-sagemaker/)