Cloud Computing: Building ML Pipelines with Amazon SageMaker and XGBoost

Farid Afzali, Ph.D., P.Eng.

Introduction and Key Learning Outcomes

In this lecture, we will use real-world structured datasets (such as university admission and life expectancy) to explore machine learning workflows on AWS SageMaker.

You will learn how to:

- Train an XGBoost model in SageMaker using tabular data.
- Tune hyperparameters and interpret model performance.
- Deploy a model as a SageMaker endpoint.
- Perform inference in a cloud-native ML pipeline.

Case Study 1 – University admission project

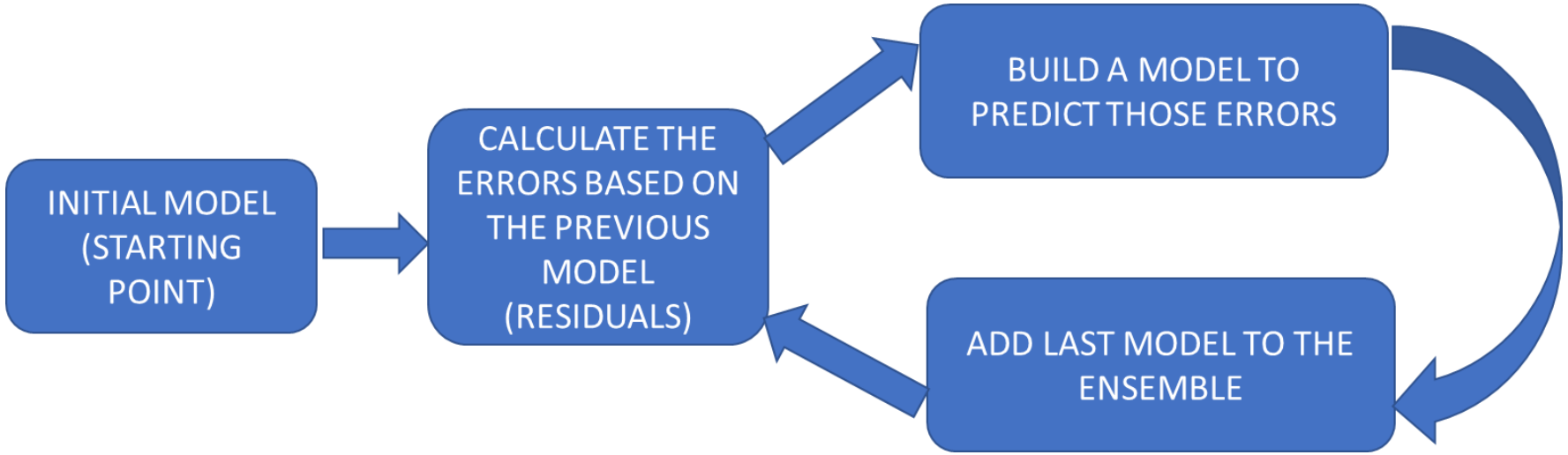
- **Goal**: The project's objective is to build, train, test, and deploy a machine learning model that can predict the likelihood of university admission based on a student's profile.
- **Tool**: AWS SageMaker is mentioned as the tool for launching a training job from the Management Console, which is a part of AWS that allows you to build, train, and deploy machine learning models at scale.
- **Practical Real-World Application**: The machine learning model is intended to be used by university admissions departments to identify the most qualified students.
- **Data**:
  - **Inputs (Features)**: These are the parameters the model will use to make its predictions. They include GRE scores, TOEFL scores, University Rating, Statement of Purpose (SOP) quality, Letter of Recommendation (LOR) strength, Undergraduate GPA, and Research Experience.
  - **Outputs**: The model will output the probability of admission, which will range from 0 to 1 (with 1 likely indicating certain admission and 0 indicating no chance of admission).
- The **Data Source** for the machine learning model is provided as a URL, which is a Kaggle link. Kaggle is a popular platform for data science competitions and datasets.

Case Study 1 – University admission project

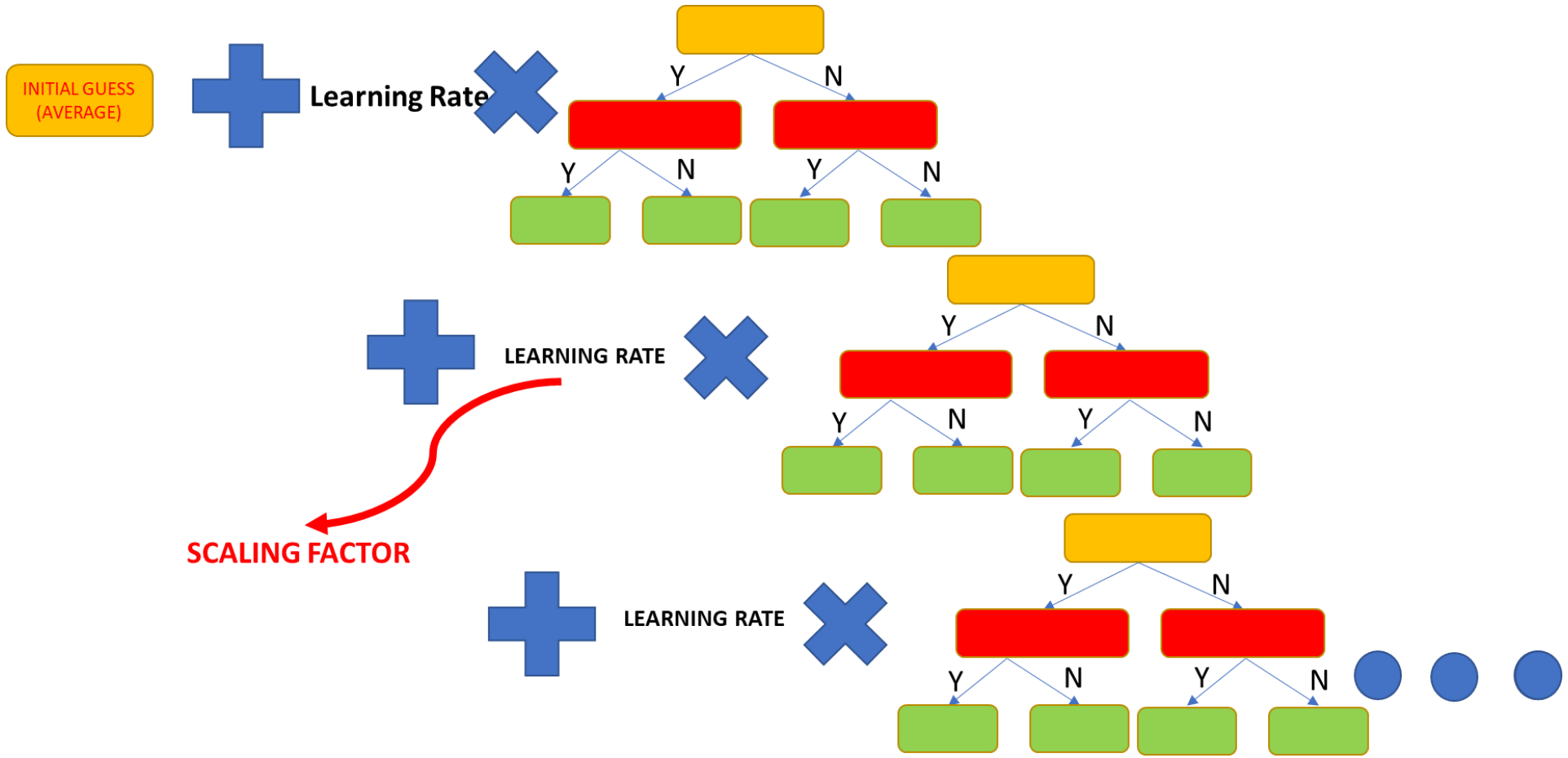
- **Inputs**: These are the variables or features the model will use to make its predictions. They include:
  - ❑ GRE Score
  - ❑ TOEFL Score
  - ❑ University Rating
  - ❑ Statement of Purpose (SOP)
  - ❑ Letter of Recommendation (LOR)
  - ❑ Cumulative Grade Point Average (CGPA)
  - ❑ Research (likely a binary indicator of research experience)
- **Machine Learning Model**: This is the core computational component that processes the inputs to make a prediction. This component represents the algorithms and statistical methods that learn from the input data to make predictions or decisions without being explicitly programmed to perform the task.
- **Output**: The result of the machine learning model's processing of the inputs, which in this case is the "Chance of Admission." This is likely a probability score between 0 and 1 where a higher score indicates a higher likelihood of admission.

XGBoost Recap: Applied in Cloud-Based Workflows

- XGBoost is an ensemble learning algorithm that builds models iteratively to improve prediction accuracy.
- In cloud computing environments like AWS, it is commonly used for regression and classification tasks due to scalability.
- The model starts by learning initial patterns and then creates successive models to minimize residual errors.
- Cloud platforms enable distributed training and automatic deployment pipelines for XGBoost models.
- Regularization and tuning options help prevent overfitting, making it suitable for real-world production use.



XGBoost: Gradient Boosting Algorithm



Overview of XGBoost in SageMaker

- **What is XGBoost?**
  - ❑ XGBoost is a fast and accurate gradient boosting algorithm for structured data.
  - ❑ It builds an ensemble of weak learners (typically decision trees) in sequence.
  - ❑ Each tree corrects the error of the previous one, improving the model iteratively.
  - ❑ XGBoost is widely used in industry and machine learning competitions
- **Why Use XGBoost with SageMaker?**
  - ❑ XGBoost is widely adopted in Kaggle and real-world applications due to its high performance.
  - ❑ Ensemble architecture makes it robust across diverse datasets.
  - ❑ Cloud services allow easy access to compute resources, automated training, and endpoint deployment.
  - ❑ Use cases in cloud include fraud detection, recommendation systems, and user behavior analysis.

Running XGBoost in AWS SageMaker: Input Formats & Compute Options

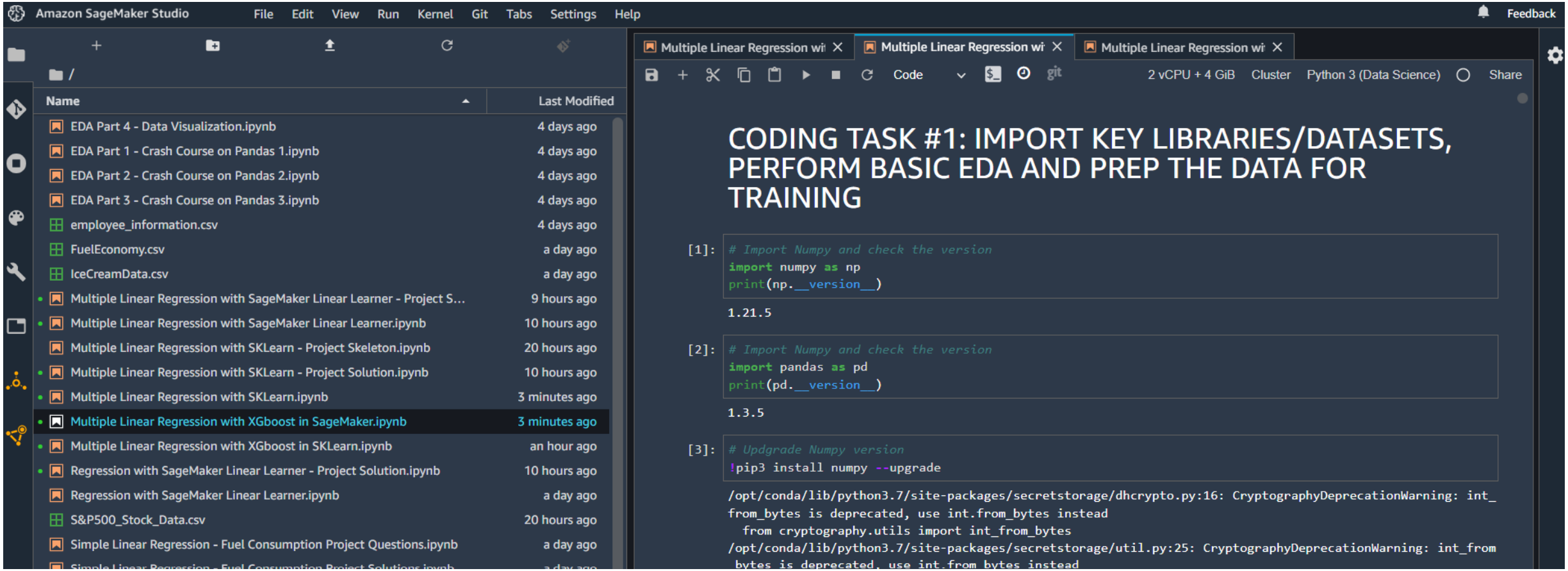
- **Data Formats for Training**
  - ❑ XGBoost in SageMaker uses tabular data (rows = samples, columns = features).
  - ❑ Supported input formats: CSV and LibSVM.
  - ❑ Unlike other SageMaker algorithms, XGBoost does not support protobuf format.

- **Compute Instance Considerations**
  - ❑ Optimized for CPU-based training — no GPU needed unless additional processing is required.
  - ❑ Recommended instance types: m4.xlarge or other memory-efficient EC2 types.
  - ❑ Ideal for cloud-based model training due to balanced performance and cost.

Key Hyperparameters in XGBoost

- max_depth: Maximum tree depth to control complexity.
- eta: Learning rate for conservative boosting steps.
- alpha: L1 regularization to prevent overfitting.
- lambda: L2 regularization to improve generalization.
- Other tunable parameters available in AWS documentation.

CODE DEMO: XG-BOOST IN SAGEMAKER



Optional Case Study – Life Expectancy Prediction

- In this hands-on project, we will train an XG-Boost regression model to predict life expectancy using built-in SageMaker Algorithms.
- This data was initially obtained from World Health Organization (WHO) and United Nations Website. Data contains features like year, status, life expectancy, adult mortality, infant deaths, percentage of expenditure, alcohol etc.
- Tasks:
  - ❑ Split the data into training, validation, testing and upload it to S3
  - ❑ Train a regression model using built-in SageMaker XG-boost algorithm
  - ❑ Assess trained model performance
  - ❑ Plot trained model predictions vs. ground truth output
  - ❑ What is R2?

Optional Case Study – Life Expectancy Prediction

INPUT FEATURES

YEAR
ADULT MORTALITY
STATUS
INFANT DEATHS
ALCOHOL
HEPATITIS B

PREDICTION

LIFE EXPECTANCY