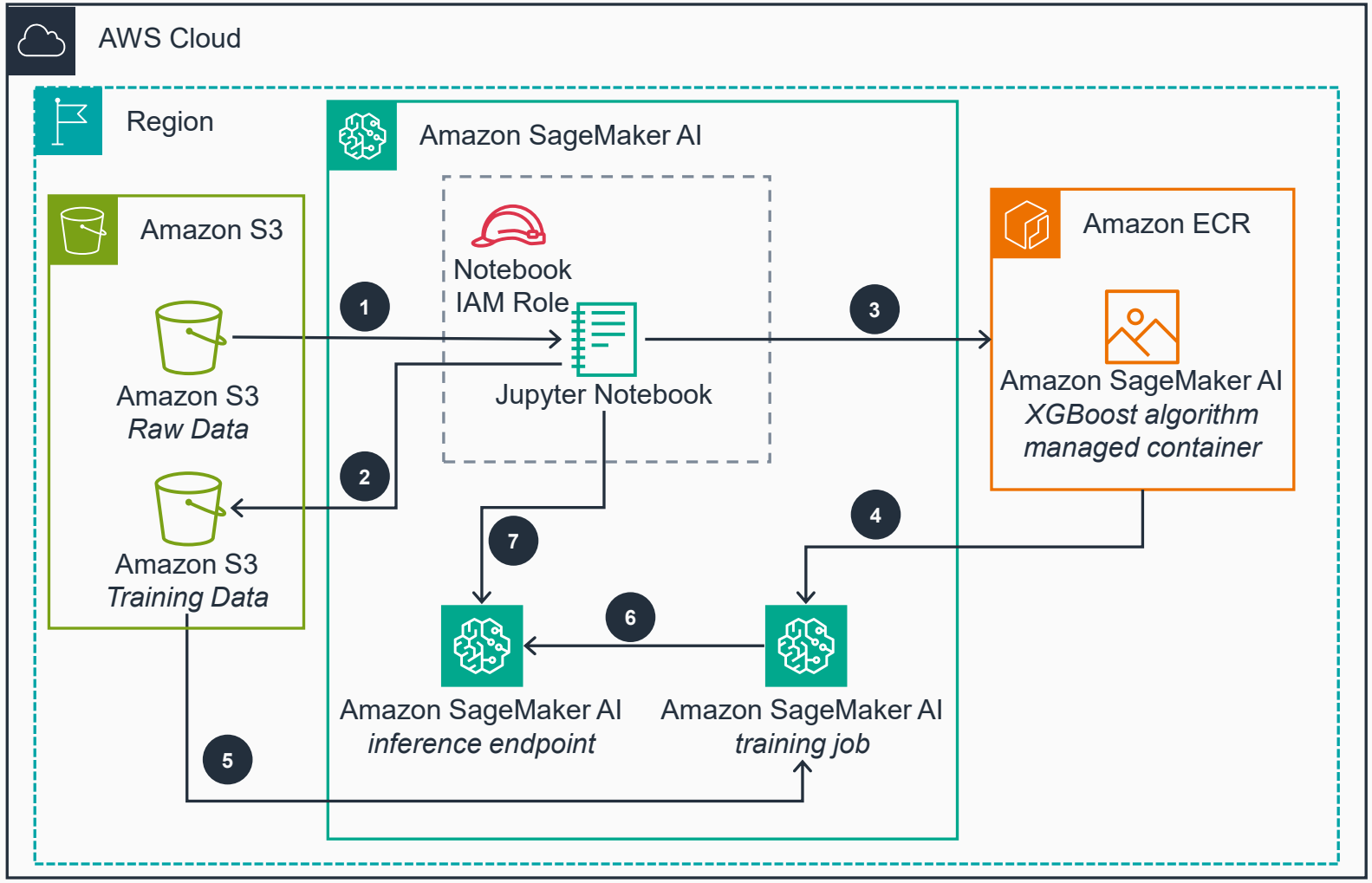
Guidance for a Predictive Responsible Gaming Model Using Amazon SageMaker AI

This architecture diagram shows how to build and train an ML model using Amazon SageMaker AI to predict problematic gambling behavior for player protection. This slide shows Steps 1-6.



1. The Jupyter notebook retrieves raw training and test data, which includes betting activity metrics, from the **Amazon Simple Storage Service (Amazon S3)** bucket using the Notebook **AWS Identity and Access Management (IAM)** role. The notebook visualizes the data to help understand the requirements and expected predictions of the training data.

2. Use the Jupyter notebook to explore and analyze the data. Preprocess the data to encode categorical data and address multicollinearity, data leakage, and oddly distributed, missing, or unbalanced data. Split the data into training, validation, and test datasets, and upload the datasets to the **Amazon S3** training data bucket.

3. Specify the **Amazon Elastic Container Registry (Amazon ECR)** location for the **Amazon SageMaker AI** implementation of XGBoost, an open source implementation of the gradient boosted trees algorithm. The AWS built-in implementation has a smaller memory footprint, better logging, improved hyperparameter validation, and a bigger set of metrics than the original versions.

4. Create a training job using the **Amazon SageMaker AI** XGBoost algorithm managed container.

5. During training, the **Amazon SageMaker AI** managed training job downloads the input datasets from the **S3** bucket to each training instance.

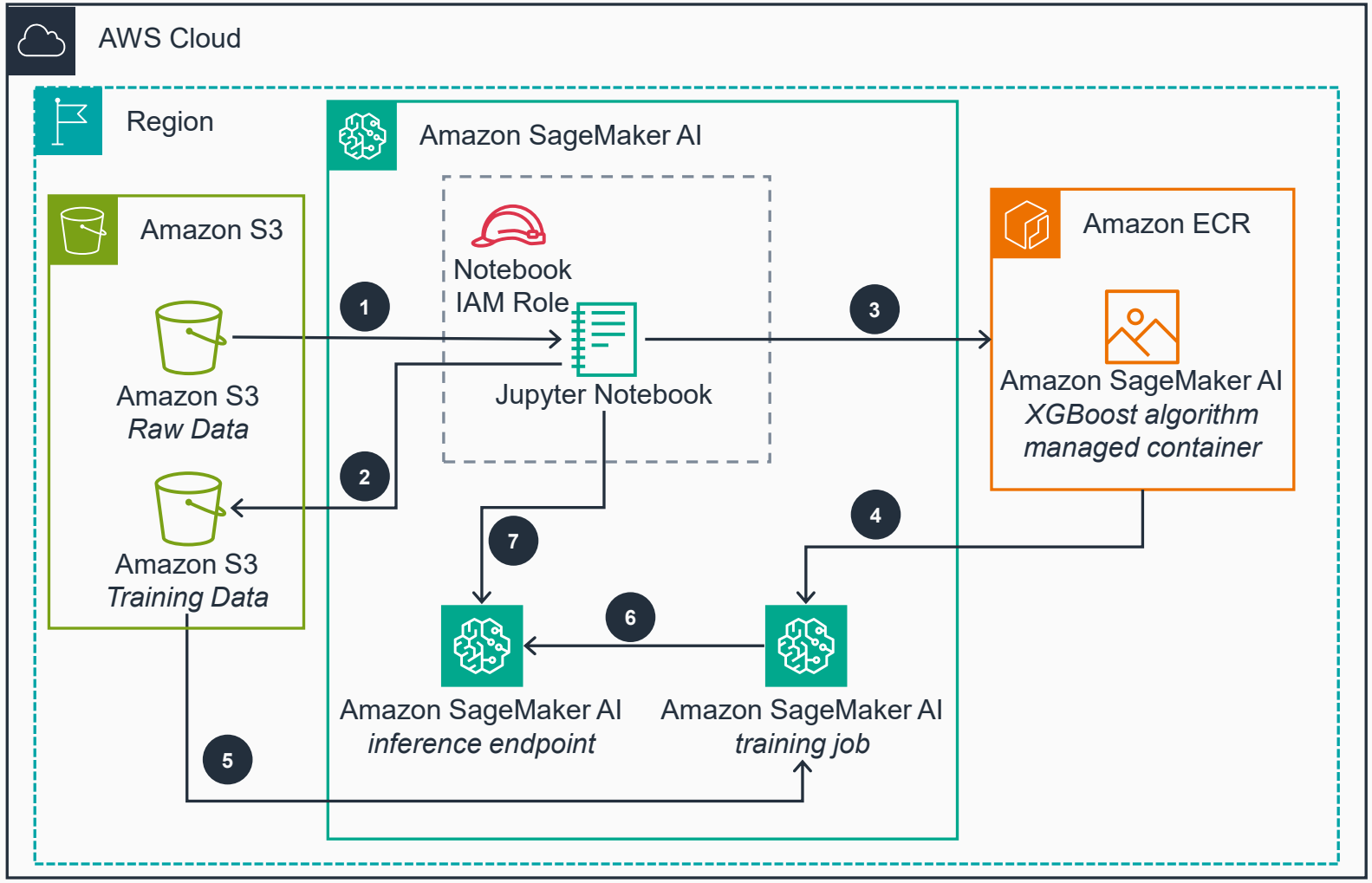
6. After the training job finishes, validate the model based on the validation dataset. In case of successful validation, the model is deployed as an **Amazon SageMaker AI** inference endpoint.

Reviewed for technical accuracy July 17, 2025

AWS Reference Architecture

Guidance for a Predictive Responsible Gaming Model Using Amazon SageMaker AI

This architecture diagram shows how to build and train an ML model using Amazon SageMaker AI to predict problematic gambling behavior for player protection. This slide shows Steps 7.



7 Invoke the deployed model to determine the probability that a player's gambling behavior is problematic. Validate predictions against the test data with a customer-configured threshold. Identification of problematic behavior may require further action from the customer to protect the player.

AWS Reference Architecture