Domain 4: Monitor Model

4.1 Monitor Model Performance and Data Quality

4.1.1 Monitoring Machine Learning Solutions

Importance of Monitoring in ML

a) **Machine Learning Lens**: *AWS Well-Architected Framework: Best practices and design principles*

Best practice*: When*

Optimize resources	***Resource pooling***	*Sharing compute, storage, and networking resources*
	Caching	
	Data management	*data compression, partitioning, and lifecycle management*
Scale ML workloads based on demand	***AWS Auto Scaling***	*SageMaker built-in scaling. AWS Auto-Scaling*
	Lambda	
Reduce Cost	***Monitor usage and costs***	*resource tagging*
	monitor ROI	
Enable continuous improvement	***Establish Feedback Loops***	
	Monitor Performance	*SageMaker Model Monitor (Drift)* *CloudWatch alerts (deviations)*
	Automate Retraining	

Detecting Drift in Monitoring

a) Drift Types

Drift Type	***Description***	***Causes***	***Implications***
Data Quality Drift	Production data distribution differs from training data distribution	• Real-world data not as curated as training data • Changes in data collection processes • Shifts in real-world conditions	• Model accuracy decreases • Predictions become less reliable
Model Quality Drift	Model predictions differ from actual ground truth labels	• Changes in the underlying relationship between features and target • Model decay over time • Concept drift	• Decreased model performance • Inaccurate predictions
Bias Drift	Increase in bias affecting model predictions over time	• Training data too small or not representative • Incorporation of societal assumptions in training data • Exclusion of important data points • Changes in real-world data distribution	• Model overgeneralization • Unfair or discriminatory predictions • Ethical concerns • New groups in production
Feature Attribution Drift	Changes in the contribution of individual features to model predictions	• Shifts in feature importance over time • Changes in the underlying problem domain • Introduction of new, more predictive features	• Model may rely on less relevant features • Decreased interpretability • Potentially reduced performance

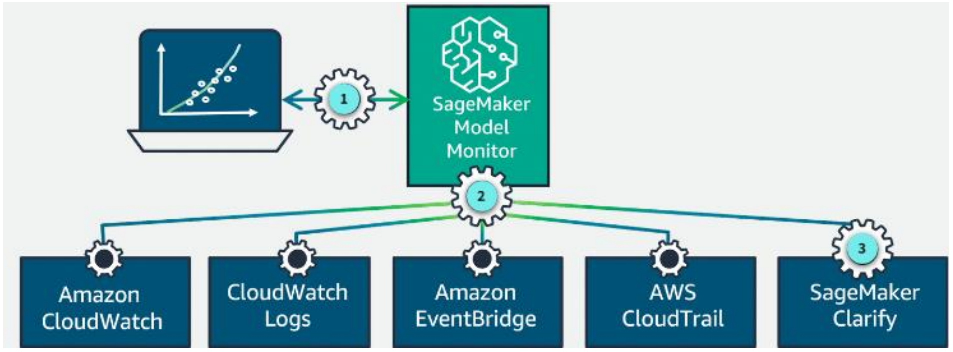
Note: *Bias inverse of variance, which is the level of small fluctuations or noise common in complex data sets. Bias tends to cause model predictions to overgeneralize, and variance tends to cause models to undergeneralize. Increasing variance is one method for reducing the impact of bias.*

b) Monitoring Drift

Monitoring Type	What It Monitors	How It Monitors
Data Quality Monitoring	• Missing values • Outliers • Data types • Statistical metrics (mean, std dev, etc.) • Data distribution	• Implement data validation checks • Calculate statistical metrics • Compare metrics with baseline values • Use data drift detection techniques (e.g., Kolmogorov-Smirnov tests, Maximum Mean Discrepancy)
Model Quality Monitoring	• Evaluation metrics (accuracy, precision, recall, F1, AUC, etc.) • Prediction confidence • Performance across different subpopulations	• Calculate evaluation metrics on held-out test set or production data sample • Implement confidence thresholding or uncertainty estimation • Flag low-confidence predictions • Monitor performance on different data subsets
Model Bias Drift Monitoring	• Bias metrics (disparate impact, fairness, etc.) • Performance across sensitive groups	• Calculate bias metrics for different sensitive groups • Compare bias metrics with baseline values or thresholds • Implement bias mitigation techniques (e.g., adversarial debiasing, calibrated equalized odds)
Feature Attribution Drift Monitoring	• Feature importance scores • Statistical metrics of feature attributions	• Use interpretability techniques (e.g., SHAP) to calculate feature attributions • Calculate statistical metrics on feature attributions • Compare metrics with baseline values • Identify features with significantly changed attributions

SageMaker Model Monitor

Integration



SageMaker - Monitoring for Data Quality Drift

STEPS

1. **Initiate data capture on the endpoint**
2. **Create a baseline**

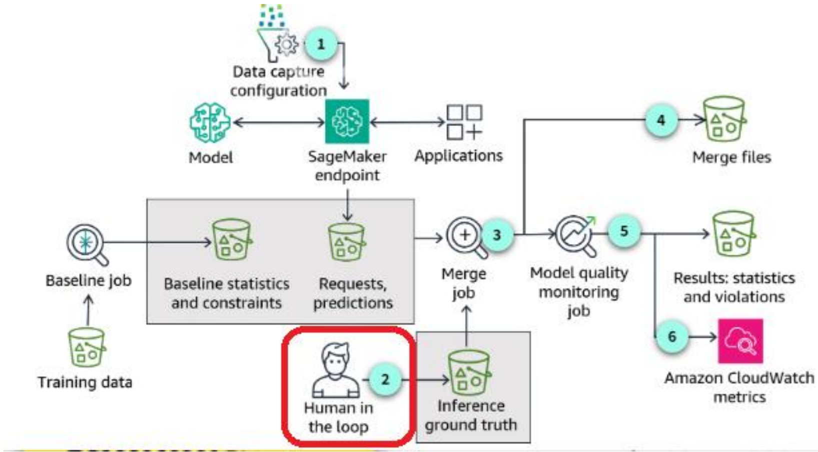
start a baseline processing job with the suggest_baseline method of the ModelQualityMonitor object using the SageMaker Python SDK.

3. **Schedule data quality monitoring jobs**
4. **Integrate data quality monitoring with Cloudwatch**
5. **Interpret results and analyze findings**

   The report is generated as the **constraint_violations.json** file. The SageMaker Model Monitor prebuilt container provides the following violation checks.

   - data_type_check
   - completeness_check
   - baseline_drift_check
   - missing_column_check
   - extra_column_check
   - categorical_values_check

SageMaker - Monitoring for Model Quality Drift using Model Monitor



To monitor model quality, SageMaker Model Monitor requires the following inputs:

1. Baseline data
2. Inference input and predictions made by the deployed model
3. Amazon SageMaker Ground Truth associated with the inputs to the model

SageMaker - Monitoring for Bias using Clarify

Statistical bias drift occurs when the data used for training differs from the data encountered during prediction, leading to potentially biased outcomes. This is prominent when training data changes over time. In this lesson, you will learn about AWS services that help you monitor for statistical bias drift.

Post-training bias metrics in SageMaker Clarify help us answer two key questions:

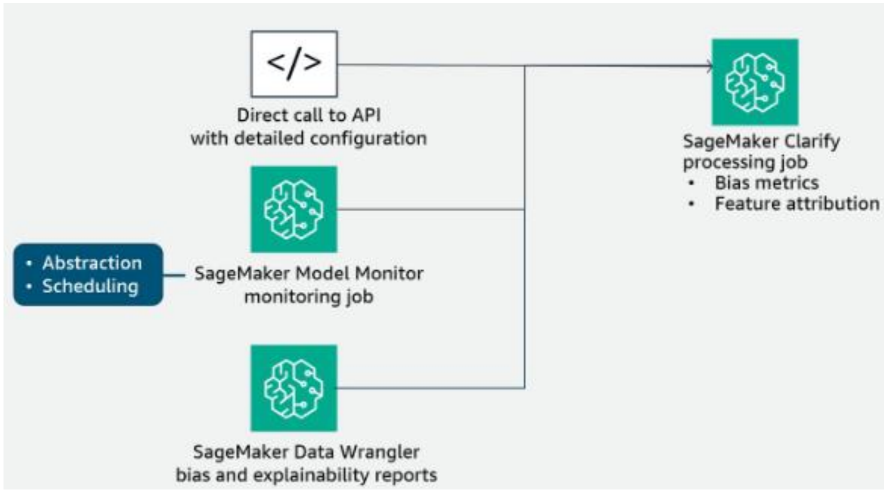
- Are all facet values represented at a similar rate in positive (favorable) model predictions?
- Does the model have similar predictive performance for all facet values?

SageMaker Model Monitor automatically does the following:

- **Merges** the prediction data with SageMaker Ground Truth labels
- **Computes** baseline statistics and constraints
- **Inspects** the merged data and generates bias metrics and violations
- **Emits** CloudWatch metrics to set up alerts and triggers
- **Reports and alerts** on bias drift detection
- **Provides** reports for visual analysis

How it works: It quantifies the contribution of each input feature (for example, audio characteristics) to the model's predictions, helping to explain how the model arrives at its decisions.

Options for using SageMaker Clarify

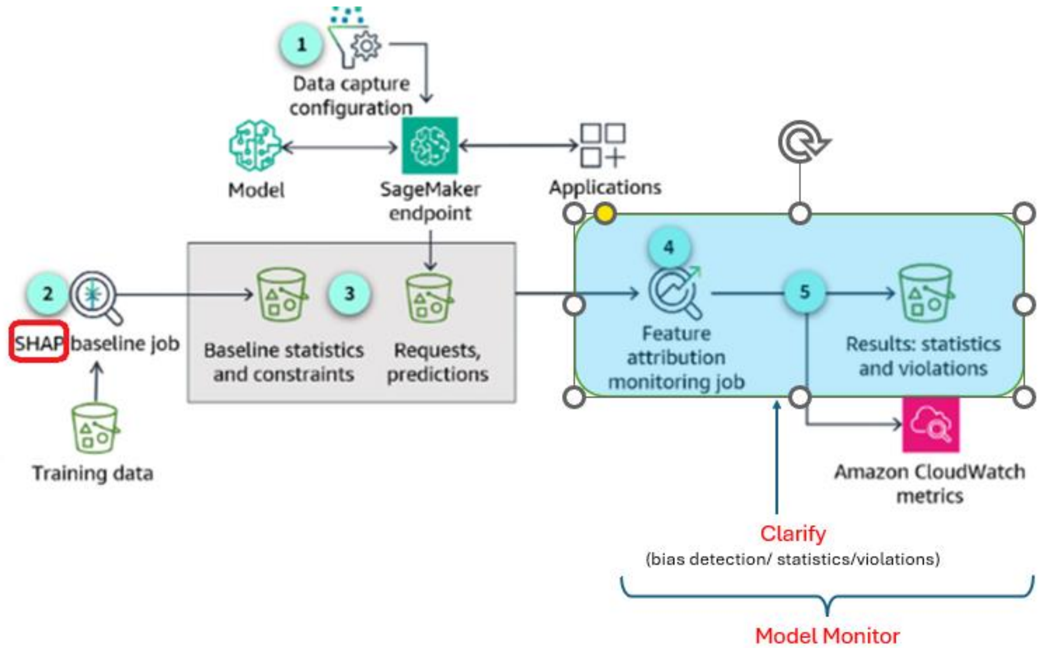


When to use which

Method	Description	When to Use
SageMaker Clarify Directly	Configure and run Clarify processing job using SageMaker Python SDK API	• For one-time or ad-hoc bias analysis • When you need full control over the analysis configuration • For integrating bias analysis into custom workflows
SageMaker Model Monitor + Clarify	Integrate Clarify with Model Monitor for continuous bias monitoring	• When you want to automate bias detection in production • If you need to set up alerts for bias drift
SageMaker Data Wrangler	Utilize Clarify within Data Wrangler during data preparation	• During the data preparation phase • When you want to identify potential bias early in the ML pipeline • If you're already using Data Wrangler for data preprocessing

SageMaker - Monitoring for Feature Attribution Drift (Model Monitor + Clarify)

Feature attribution refers to understanding and quantifying the contribution or influence of each feature on the model's predictions or outputs. It helps to identify the most relevant features and their relative importance in the decision-making process of the model.



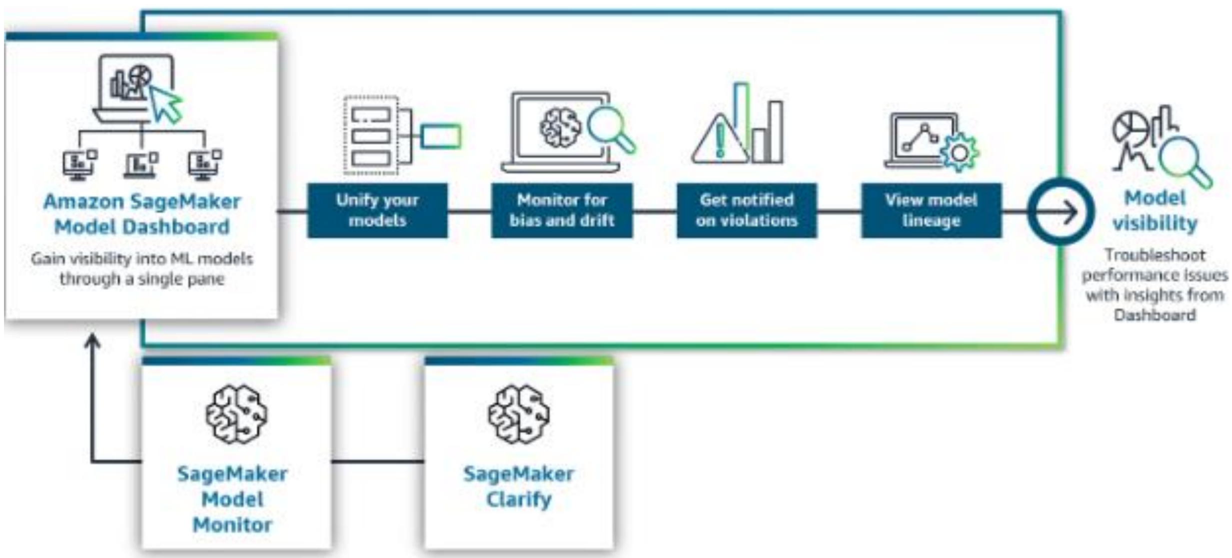
Uses SHAP

SageMaker Clarify provides feature attributions based on the concept of Shapley value. This is a game-theoretic approach that assigns an importance value (SHAP value) to each feature for a particular prediction.

Here's how it works:

1. **SageMaker Clarify:** This is the core component that performs the actual bias detection and generate quality metrics and violations
2. **SageMaker Model Monitor:** This is the framework that can use Clarify's capabilities to perform continuous monitoring of deployed models.

SageMaker Model Dashboard



Features

1. **Alerts** :

   How it helps: The dashboard provides a record of all activated alerts, allowing the data scientist to review and analyze past issues.

   Alert criteria depend upon two parameters:

   - **Datapoints to alert**: Within the evaluation period, how many runtime failures raise an alert?
   - **Evaluation period:** The # of most recent monitoring executions to consider when evaluating alert status.

2. **Risk rating**

   A user-specified parameter from the model card with a low, medium, or high value.

3. **Endpoint performance**

   You can select the endpoint column to view performance metrics, such as:

   - **CpuUtilization**: The sum of each individual CPU core's utilization from 0%-100%.
   - **MemoryUtilization**: The % of memory used by the containers on an instance, 0%-100%.
   - **DiskUtilization**: The % of disk space used by the containers on an instance, 0%-100%.

4. **Most recent batch transform job**

   This information helps you determine if a model is actively used for batch inference.

5. **Model lineage graphs**

   When training a model, SageMaker creates a **model lineage graph**, a visualization of the entire ML workflow from data preparation to deployment.

6. **Links to model details**

The dashboard links to a model details page where you can explore an individual model.

Model Monitor vs SageMaker Dashboard vs Clarify: When to use which one

Tool	*Description*	*Why to use*	*When to Use*
Model Monitor	Continuous monitoring of ML models in production	• data and model quality issues • model drift	• To set up automated alerts for performance degradation • When you need to monitor resource utilization • Monitor real-time endpoints, batch transform, On-demand monitoring job
SageMaker Dashboard	Centralized view of SageMaker resources and jobs	•	• For a high-level overview of all SageMaker activities • To track training jobs, endpoints, and notebook instances
SageMaker Clarify	Bias detection and model explainability tool	• Detecting Bias • Triggers statistics and Violations report	• To detect bias in training data and model predictions • When you need to explain model decisions • For regulatory compliance requiring model transparency • To improve model fairness and accountability

4.1.2 Remediating Problems Identified by Monitoring

Automated remediations and notifications

- **Stakeholder notifications:** When monitoring metrics indicate changes that impact business KPIs or the underlying problem
- **Data Scientist notification:** You can use automated notifications to **data scientists** when your monitoring detects data drift or when expected data is missing.
- **Model retraining:** Configure your model training pipeline to **automatically retrain models when monitoring detects drift, bias, or performance degradation**.
- **Autoscaling:** You use resource utilization metrics gathered by infrastructure monitoring to initiate autoscaling actions.

Model retraining strategies

Strategy	When to Use	Advantages	Considerations
Event-driven	• When drift is detected above a certain threshold • In response to significant changes in data or performance	• Timely response to changes • Efficient use of resources	• May be frequent if thresholds are too sensitive • Retraining can be expensive and time-consuming
On-demand	• When market conditions change significantly • In response to new competitors or strategies	• Allows for human judgment in decision-making • Can incorporate business context	• Requires constant monitoring by data scientists or stakeholders • May lead to delayed responses
Scheduled	• When there are known seasonal patterns • For maintaining model accuracy over time	• Predictable maintenance schedule • Can anticipate and prepare for retraining periods	• May retrain unnecessarily if no significant changes occur • Might miss sudden, unexpected changes

4.2 Monitor and Optimize Infrastructure and Costs

4.2.1 Monitor Infrastructure

***Monitor Performance Metrics* - CloudWatch vs Model Monitor**

Feature	***SageMaker Model Monitor***	***CloudWatch Logs***
Purpose	*Continuous monitoring of ML models in production*	*Monitoring, storing, and accessing log files*
Key Capabilities	(all 4 ML monitoring types) • Data quality monitoring • Model quality monitoring • Bias drift monitoring • Feature attribution drift monitoring	• Log collection from various sources • Log storage in S3 • Pattern recognition • Log anomaly detection
Monitoring Types	• Real-time endpoint monitoring • Batch transform job monitoring • On-schedule monitoring for async batch jobs	• EC2 instances • CloudTrail • Amazon Route 53 • Other sources
Alert System	Set alerts for deviations in model quality	Notifications based on preset thresholds
Customization	• Pre-built monitoring capabilities (no coding) • Custom analysis options	Customizable log patterns and anomaly detection

Monitoring vs. Observability

	Monitoring	**Observability**
Definition	Continuous collection and analysis of metrics	Deep insights into internal state and behavior of ML systems
Focus	Detecting anomalies and deviations	Understanding complex interactions and dependencies
Key Activities	• Collecting metrics • Logging • Alerting	• Analyzing system behavior • Identifying root causes • Reasoning about system health
Techniques	• • Metric collection • Threshold-based alerting • Basic log analysis	• Distributed tracing • Structured logging • Advanced data visualization
Outcome	Detect issues and invoke alerts or automated actions	Provide deeper insights for troubleshooting and optimization
Scope	Primarily focused on predefined metrics and thresholds	Enables asking and answering questions about system behavior

Monitoring Tools (for Performance and Latency)

Feature	AWS X-Ray	CloudWatch Lambda Insights	CloudWatch Logs Insights	QuickSight
Purpose	Trace information about responses and calls in applications	In-depth performance monitoring for Lambda fns only	Interactive log analytics service	BI and data visualization service
Key Features	• Works across AWS and third-party services • Generates detailed service graphs • Identifies performance bottlenecks	• Monitors metrics (memory, duration, invocation count) • Provides detailed logs and traces • Helps identify bottlenecks in Lambda functions	• Interactive querying and analysis of log data • Correlates log data from different sources • Visualizes time series data • Supports aggregations, filters, and regex	• Interactive dashboards • ML-powered insights • Supports various data sources
Compatible Services	EC2, ECS, Lambda, Elastic Beanstalk	Lambda	Any service that generates logs in CloudWatch	Various AWS services and external data sources
ML Use Cases	• Analyze bottlenecks in ML systems • Trace requests in ML applications (e.g., chatbot inference)	• Monitor and optimize ML models deployed as Lambda functions • Identify root causes of Lambda function issues	• Analyze logs from ML workloads • Identify patterns and anomalies in ML system behavior	• Create dashboards for ML experiment results • Analyze and present insights from ML predictions
Visualization	Service maps, Trace views	Performance dashboards, Trace details	Time series graphs, Log event views	Interactive dashboards, Charts, Graphs
Primary Benefit	End-to-end request tracing and bottleneck identification	Detailed Lambda function performance insights	Flexible, interactive log analysis and visualization	Comprehensive data visualization and business intelligence

SageMaker w/ EventBridge

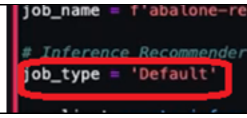
Actions that can be automatically invoked using EventBridge:

a) Invoking an AWS Lambda function
b) Invoking Amazon EC2 run command (not create or deploy)
c) Relaying event to Kinesis Data Streams
d) Activating an AWS Step Functions state machine.
e) **Notifying** SNS topic or an AWS Server Migration Service (AWS SMS) queue.

4.2.2 Optmize Infrastructure

Inference Recommender types

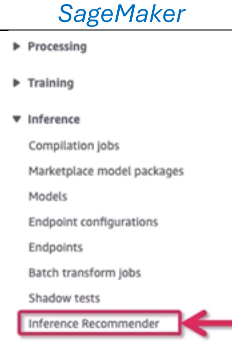
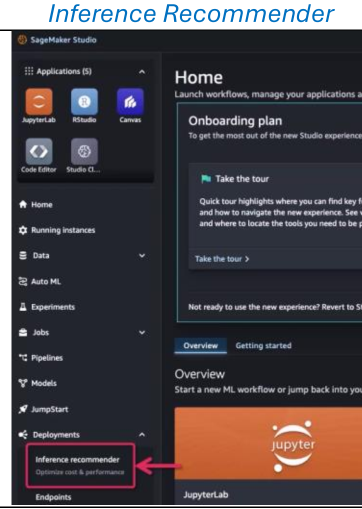
a) Inference Recommendation Types

<table>
<tr><th>Default</th><th>Advanced</th></tr>
<tr><td>Endpoint Recommender</td><td>Endpoint Recommender + Inference Recommender</td></tr>
<tr><td>45 mins</td><td>2 hours</td></tr>
<tr><td></td><td></td></tr>
<tr><td></td><td></td></tr>
</table>

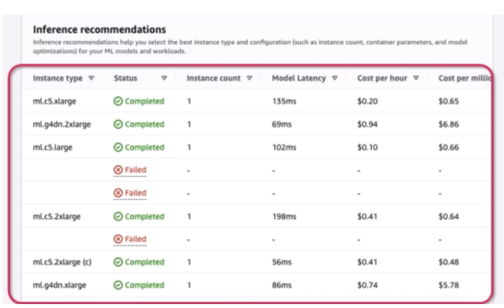
b) Endpoint Recommender vs Inference Recommender

	Endpoint Recommender	*Inference Recommender*
Output	list (or ranking) of prospective instances	Same
	run a set of load tests.	based on a custom load test.
What you need to do	**-N/A -**	your desired ML instances or a serverless endpoint, provide a custom traffic pattern, and provide requirements for latency and throughput

c) How to start

<table>
<tr><th>SageMaker</th><th>Inference Recommender</th></tr>
<tr><td></td><td></td></tr>
</table>

d) Sample Recommender output



Inference recommendations

Inference recommendations help you select the best instance type and configuration (such as instance count, container parameters, and model optimizations) for your ML models and workloads.

Instance type	Status	Instance count	Model Latency	Cost per hour	Cost per millio
ml.c5.xlarge	Completed	1	135ms	$0.20	$0.65
ml.g4dn.2xlarge	Completed	1	69ms	$0.94	$6.86
ml.c5.large	Completed	1	102ms	$0.10	$0.66
	Failed	-	-	-	-
	Failed	-	-	-	-
ml.c5.2xlarge	Completed	1	198ms	$0.41	$0.64
	Failed	-	-	-	-
ml.c5.2xlarge (c)	Completed	1	56ms	$0.41	$0.48
ml.g4dn.xlarge	Completed	1	86ms	$0.74	$5.78

4.2.3 Optmize Costs

Inference Recommender types

Option	Description	Best For	Cost Savings	Example Use Case
Spot Instances	Spare EC2 capacity at lower prices; can be interrupted	Interruptible workloads	Up to 90% vs On-Demand	Data preprocessing or batch processing
On-Demand Instances	Pay-per-use with no long-term commitment	Short-term, unpredictable workloads	None (baseline)	Real-time inference services
Reserved Instances	Discounted rates for 1 or 3-year commitments	Steady-state, predictable workloads	Up to 72% vs On-Demand	Long-running ML training jobs
Capacity Blocks	Reserved capacity for AWS Outposts or Wavelength Zones	Ensuring capacity during peak demand	Varies	ML workloads in on-premises environments
Savings Plans for SageMaker	Commit to a specific compute usage for 1 or 3 years	Flexible, recurring SageMaker usage	Up to 64% vs On-Demand	Regular model training and deployment

4.3 Secure AWS ML Resources

4.3.1 Securing ML Resources

Access Control using IAM

a) Roles vs Policies

Category	Type	Description	Key Responsibilities/Features
User Roles	**Data Scientist/ ML Engineer**	Provides access for experimentation	Access to S3, Athena , SageMaker Studio
	Data Engineer	Provides access for data management	Access to S3, Athena, AWS Glue, EMR
	MLOps Engineer	Provides access for ML operations	Access to SageMaker, CodePipeline, CodeBuild, CloudFormation, ECR, Lambda, Step Functions
Service Roles	**SageMaker Execution**	Allows SageMaker to perform tasks on behalf of users	General SageMaker operations
	Processing Job	Specific to SageMaker processing jobs	Data processing tasks
	Training Job	Specific to SageMaker training jobs	Model training tasks
	Model	Specific to SageMaker model deployment	Model deployment and hosting
IAM Policies	**Identity-based**	Attached to IAM users, groups, or roles	Define actions allowed on specific resources
	Resource-based	Attached to resources (e.g., S3 buckets)	Control who can access specific resources

IAM Policy – Examples for ML workflows

ID	Purpose	Key Permissions	Resource Scope	Notes
1	Least privilege access for ML workflow	• SageMaker: CreateTrainingJob, CreateModel • S3: GetObject, PutObject • ECR: BatchGetImage • CloudWatch: PutMetricData "Effect": "Allow", "Action": ["sagemaker:CreateTrainingJob", "sagemaker:DescribeTrainingJob", "sagemaker:StopTrainingJob"], "Resource": "*"	Specific ARNs for each service	Adheres to principle of least privilege

2	Read metadata of ML resources	• machinelearning:Get*
		• machinelearning:Describe*
		"Version": "2012-10-17",
		"Statement": [{
		"Effect": "Allow",
		"Action": [
		"machinelearning:Get*"
		],
		"Resource": [
		"arn:aws:machinelearning:us-east-1:555555555555:mlmodel/-ML-5555"
		"arn:aws:machinelearning:us-east-1:666666666666:mlmodel/-ML-6666"
		"arn:aws:machinelearning:us-east-1:777777777777:mlmodel/-ML-7777"
		"arn:aws:machinelearning:us-east-1:888888888888:mlmodel/-ML-8888"
		"arn:aws:machinelearning:us-east-1:555555555555:mlmodel/-ML-5555"
		]
		},
		{
		"Effect": "Allow",
		"Action": [
		"machinelearning:Describe*"
		],
		"Resource": [
		"*"
			Specific MLModel ARNs for Get*
			* (all) for Describe*
				Allows reading metadata but not modifying resources

3	Create ML resources	• machinelearning:CreateDataSourceFrom* • machinelearning:CreateMLModel • machinelearning:CreateBatchPrediction • machinelearning:CreateEvaluation	* (all)	Cannot be restricted to specific resources
4	Manage real-time endpoints and predictions	• machinelearning:CreateRealtimeEndpoint • machinelearning:DeleteRealtimeEndpoint • machinelearning:Predict	Specific MLModel ARN	Allows management of endpoints for a specific model

Detailed examples

1. *identity-based policy used in a machine learning use case*

```
{
"Version": "2012-10-17",
"Statement": [
{
"Effect": "Allow",
"Action": [
"s3:GetObject",
"s3:PutObject"
],
"Resource": [
"arn:aws:s3:::ml-data-bucket/*"
]
},
{
"Effect": "Allow",
"Action": [
"sagemaker:CreateTrainingJob",
"sagemaker:DescribeTrainingJob",
"sagemaker:StopTrainingJob"
],
```

```
{
"Effect": "Allow",
"Action": [
"cloudwatch:PutMetricData"
],
"Resource": "*",
"Condition": {
"StringEquals": {
"cloudwatch:namespace": "SageMaker"
}
}
},
{
"Effect": "Allow",
"Action": [
"logs:CreateLogGroup",
"logs:CreateLogStream",
"logs:PutLogEvents"
],
"Resource": [
"arn:aws:logs:*:*:log-group:/aws/sagemaker/*"
```

2. *Allow users to read machine learning resources metadata*

```
"Version": "2012-10-17",
"Statement": [{
  "Effect": "Allow",
  "Action": [
    "machinelearning:Get*"
  ],
  "Resource": [
    "arn:aws:machinelearning:us-east-1:555555555555:mlmodel/-ML-5555"
    "arn:aws:machinelearning:us-east-1:666666666666:mlmodel/-ML-6666"
    "arn:aws:machinelearning:us-east-1:777777777777:mlmodel/-ML-7777"
    "arn:aws:machinelearning:us-east-1:888888888888:mlmodel/-ML-8888"
    "arn:aws:machinelearning:us-east-1:555555555555:mlmodel/-ML-5555"
  ]
},
{
  "Effect": "Allow",
  "Action": [
    "machinelearning:Describe*"
  ],
  "Resource": [
    "*"
  ]
```

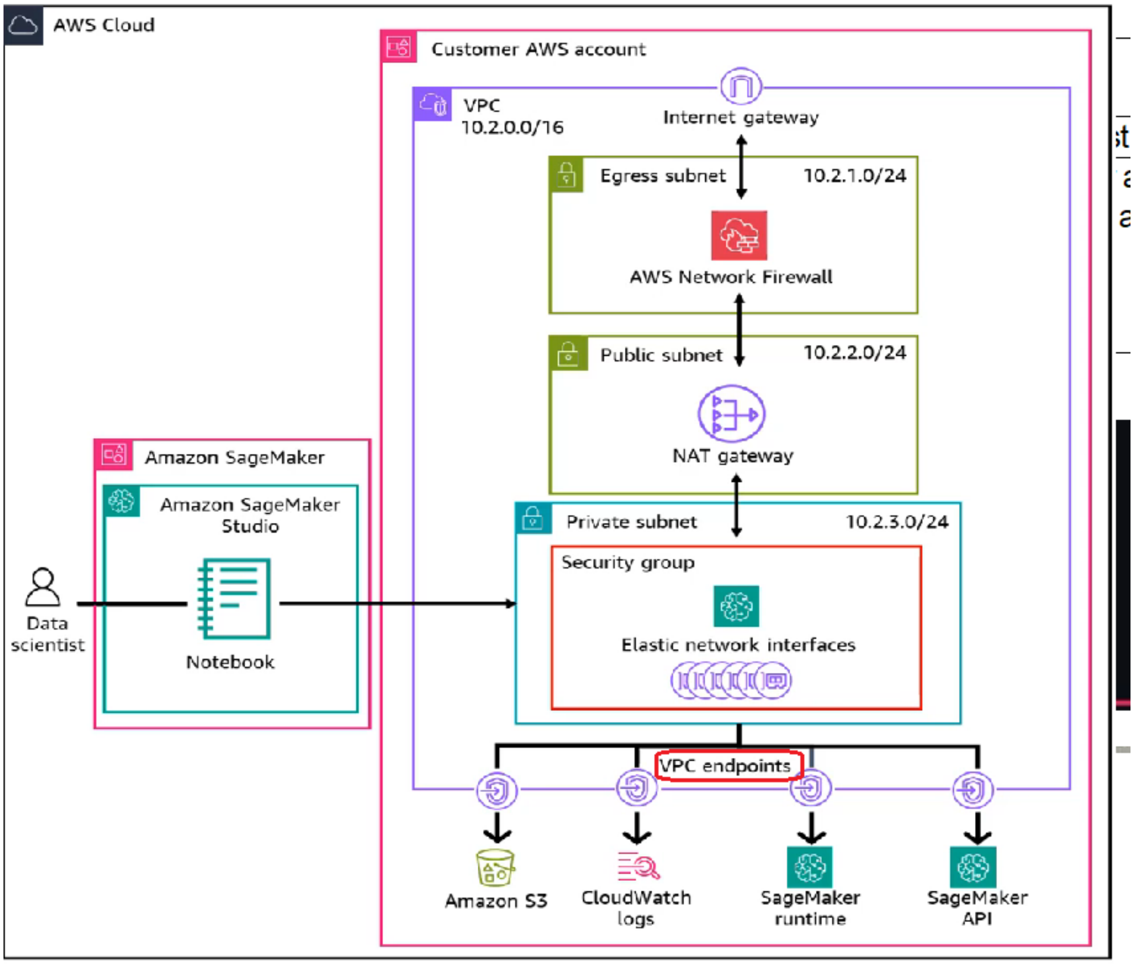
3. *Allow users to create machine learning resources*

```
"Version": "2012-10-17",
"Statement": [{
  "Effect": "Allow",
  "Action": [
    "machinelearning:CreateDataSourceFrom*",
    "machinelearning:CreateMLModel",
    "machinelearning:CreateBatchPrediction",
    "machinelearning:CreateEvaluation"
  ],
  "Resource": [
    "*"
  ]
```

4. *Allow users to create /delete real-time endpoints and perform real-time predictions on an ML model*

```
"Version": "2012-10-17",
"Statement": [{
  "Effect": "Allow",
  "Action": [
    "machinelearning:CreateRealtimeEndpoint",
    "machinelearning:DeleteRealtimeEndpoint",
    "machinelearning:Predict"
  ],
  "Resource": [
        "arn:aws:machinelearning:us-east-1:555555555555:mlmodel/-ML-5555'
  ]
```

Detailed examples



To ensure access only from VPC, **use VPC Endpoints for:**

- S3
- CloudWatch Logs
- SageMaker runtime
- SageMaker API

4.3.3 SageMaker Compliance & Governance

AWS Services for Compliance and Governance

Service	Purpose	Key Features	ML-Related Use Case
AWS Artifact	Provide on-demand access to AWS compliance reports and agreements	• Self-service portal • Access to compliance documentation	Access HIPAA compliance reports for healthcare ML projects
AWS Config	Monitor & Evaluate AWS resource configurations	• Continuous monitoring • Automated configuration evaluation	Monitor SageMaker resource configurations for compliance with security policies
Audit Manager	Continuously audit AWS usage for risk and compliance assessment	Streamlined auditing process, against regulations and standards	Assess compliance of ML workflows with industry standards
Security Hub	View of security alerts and posture	Centralized security alerts	Monitor security posture across ML workflows and resources
Amazon Inspector	Automated vulnerability management	Continuous scanning for vulnerabilities	Scan container images in ECR for ML model deployments
AWS Service Catalog	Create and manage catalogs of pre-approved resources	Governance-compliant resource catalogs	Create catalogs of compliant SageMaker resources and ML models

Amazon SageMaker Governance Tools Summary

Tool	*Purpose*	*Key Features*
SageMaker Role Manager	*Simplify access control*	• Define minimum permissions for ML activities • Quick setup & Streamlined access management
SageMaker Model Cards	*Document and share model information*	• Record intended uses • Document risk ratings
SageMaker Model Dashboard	*Provide overview of models*	• Unified view of all models in account • Monitor model behavior in production
SageMaker Assets	*Streamline ML asset management*	• Publish ML and data assets • Share assets across teams
Model Governance and Explainability	*Ensure compliance and transparency*	• Protect data and workloads • Ensure compliance with standards • Enhance model interpretability

Compliance certifications and regulatory Frameworks

Governance /Framework	***Description***	***AWS Services to Use***
ISO 27001	Information Security Management System standard	• AWS Config • AWS Security Hub
SOC 2	Service Organization Control for service organizations	• AWS Artifact • AWS Config • SageMaker Model Cards
PCI-DSS	Payment Card Industry Data Security Standard	• AWS Config • AWS WAF • Amazon Inspector
HIPAA	Health Insurance Portability and Accountability Act	• AWS Artifact • AWS Security Hub • AWS Config
FedRAMP	Federal Risk and Authorization Management Program	• AWS CloudTrail • AWS Config

Note: AWS Config common to all

4.3.3 Security Best Practices for CI/CD Pipelines

CI/CD pipeline stages

Best practice: *When*

,

CI/CD Stage	*Security Tools/Practices*

Pre-Commit	• pre-commit hooks (scripts)
	• IDE plugins to
	  ○ analyze code, detect issues
	  ○ provide recommendations for improvements.
	  ○ handle linting, formatting, beautifying, and securing *code*.

Commit	*Static Application Security Testing (SAST),*

Build	*Software Composition Analysis (SCA)*
	  ○ identifies the open-source packages used in code
	  ○ defining vulnerabilities and potential compliance-based issues
	  ○ scan infrastructure as code *(IaC) manifest files*

Test	• Dynamic Application Security Testing (DAST) • Interactive Application *Security Testing (IAST)* ○ *Combine the advantages of SAST and DAST* tools.
Deploy	• Penetration testing
Monitor	• Red/Blue/Purple teaming

4.3.4 Implement Security & Compliance w/ Monitoring, Logging and Auditing

CloudTrail for ML Resource Monitoring and Logging

Use Case	Description	Benefits
Compliance Auditing	Generate audit trails using CloudWatch Logs and CloudTrail	• Demonstrate compliance with regulations • Meet internal policy requirements
Resource Optimization	Monitor resource utilization metrics	• Optimize ML workloads • Prevent resource abuse and DoS attacks
Incident Response	Investigate and respond to security incidents	• Identify unauthorized access attempts • Detect and respond to data breaches
Anomaly Detection	Implement ML models to detect unusual patterns	• Identify potential security threats • Detect deviations in monitoring data

SageMaker Security Troubleshooting and Debugging Summary

Tool/Feature	Purpose	Key Information Provided	Use Case
CloudTrail Logs	Monitor API calls	• Caller identity • Timestamps • API details	Identify unauthorized API calls to SageMaker resources
Data Event Logs	Monitor data plane operations	Input/output data for training and inference	Verify if unauthorized entities accessed model data
IAM Policies	Manage access control	Permissions granted for SageMaker resources and operations	Identify overly permissive policies, ensure least privilege
VPC Flow Logs	Monitor network traffic	Network traffic to/from SageMaker resources	Identify suspicious IP addresses or communication patterns
Encryption Settings	Ensure data protection	• Encryption status (at rest and in transit) • AWS KMS key configurations	Verify proper data encryption and key management
AWS PrivateLink	Enhance network security	Private connections between VPC and SageMaker	Ensure traffic remains within AWS network