



Fundamentals of Accelerated Data Science

NVIDIA

Workshop Overview

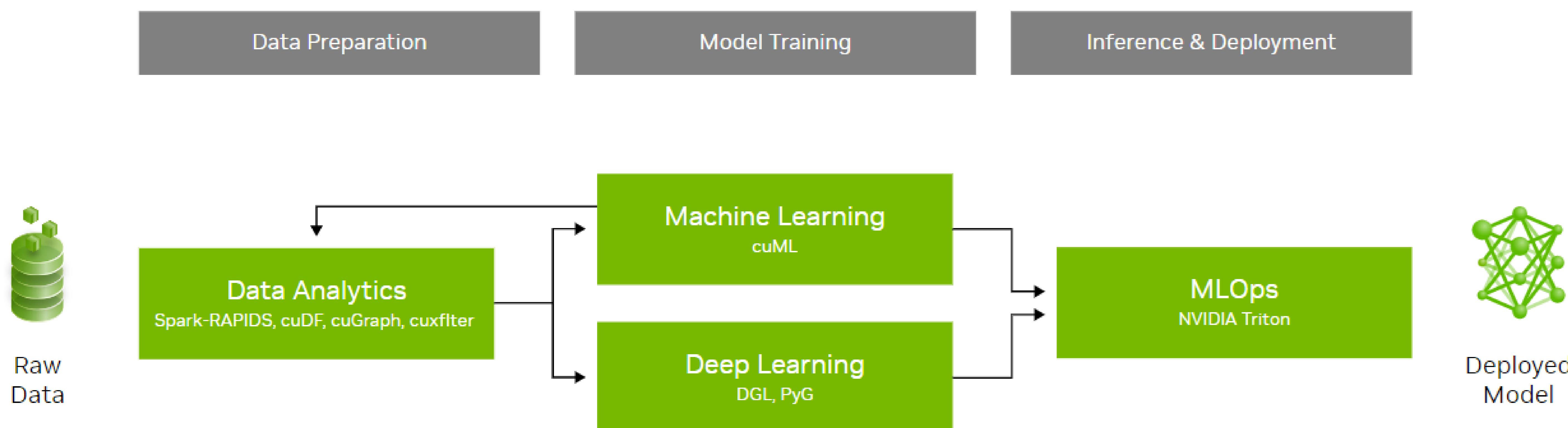
An introduction to data science with a focus on speed and efficiency

Learning objectives:

- Understand the fundamental concepts of data science and parallel computing
- Explore practical examples of accelerated data science pipelines
- Examine the methods used to achieve acceleration in data science and discuss their broader implications

This workshop will not cover statistical analysis, neural networks, and distributed computing

	Workshop Outline
Task 1	Data science overview Data manipulation
Task 2	Graph analytics
Task 3	Machine Learning
Coding Assessment	Biodefense





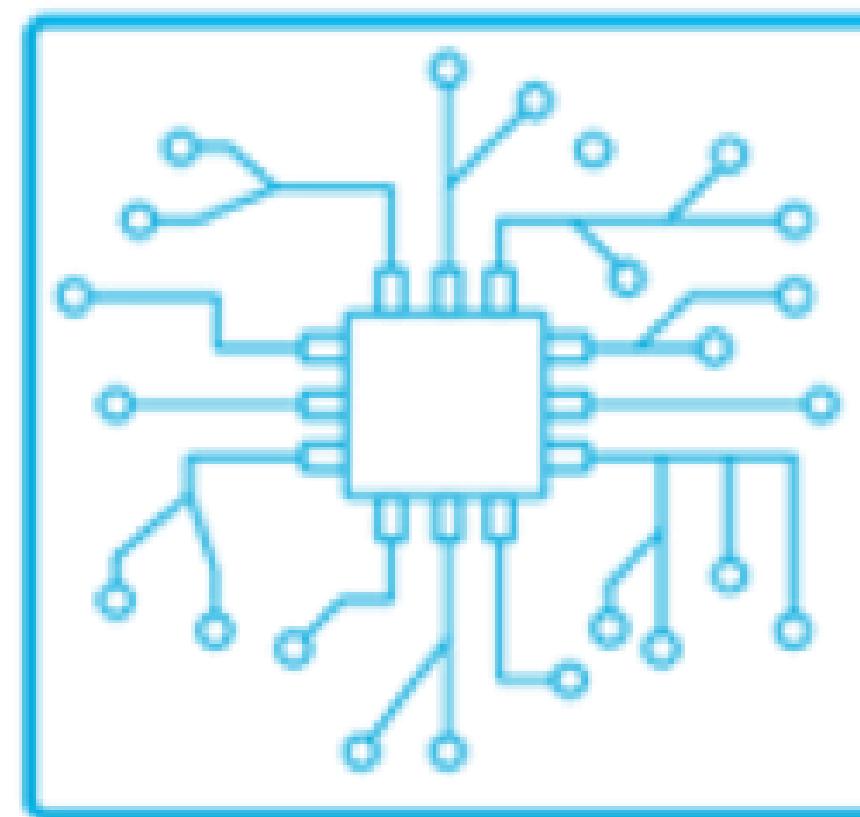
Section 3 Agenda

- Machine Learning Fundamentals
- End-To-End Machine Learning
- Accelerating Machine Learning
- Hands-On Lab

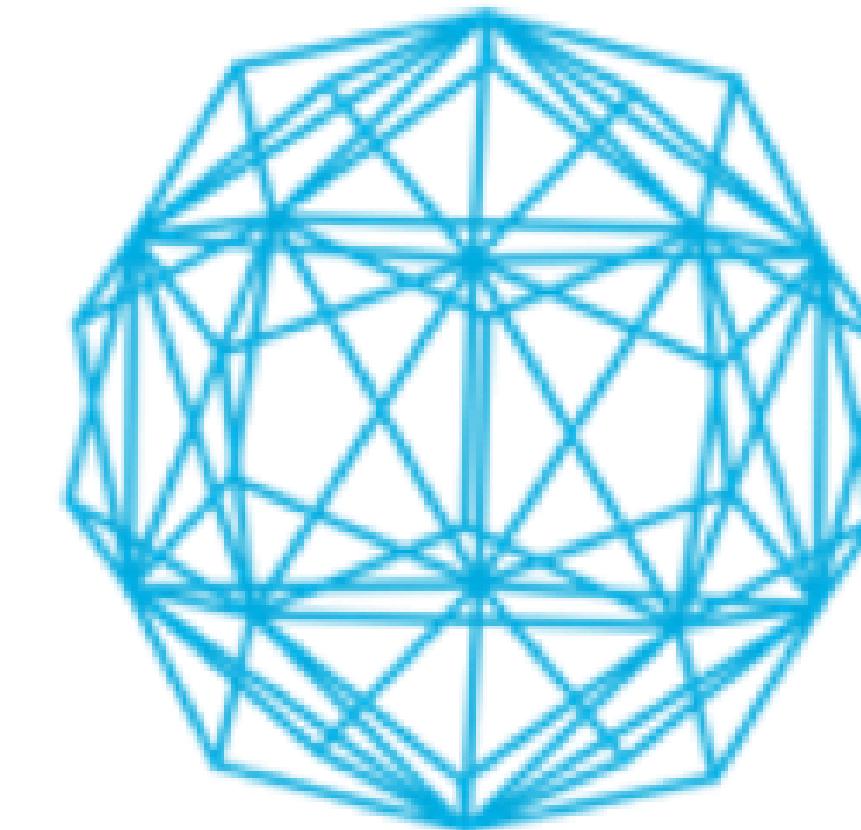
Machine Learning

Data, algorithm, and compute

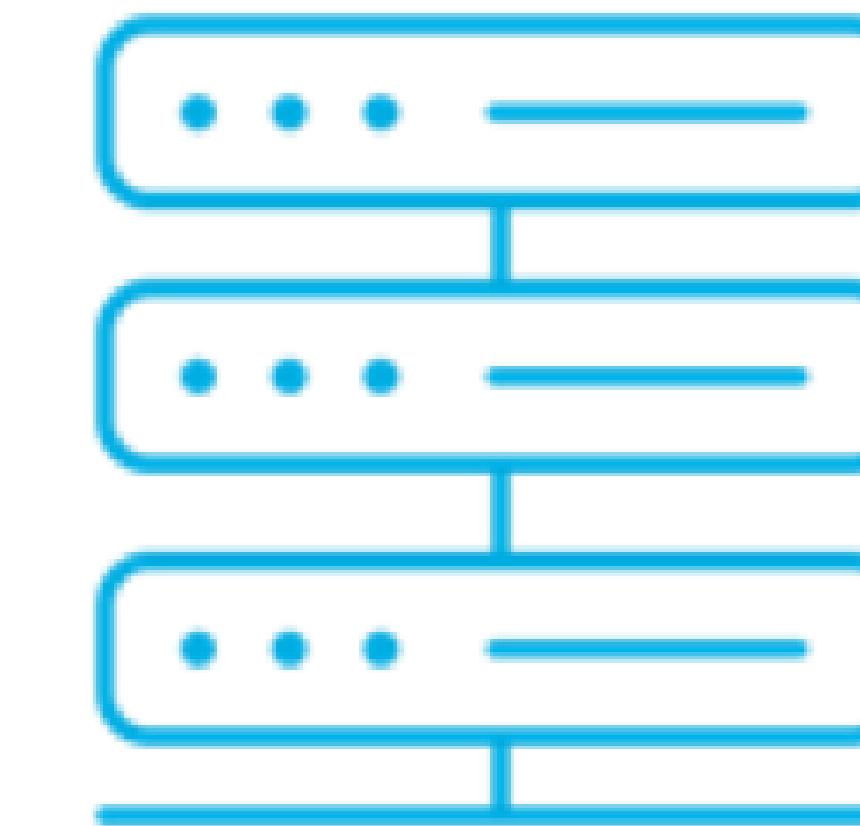
- Data science has a broader focus that encompasses various techniques for extracting insights and meaning from data, including statistical analysis and data visualization. In contrast, machine learning primarily focuses on building algorithms that enable computers to learn from data and make predictions.
- Pillars of machine learning:



Computing Power



Algorithm Power



Data Availability

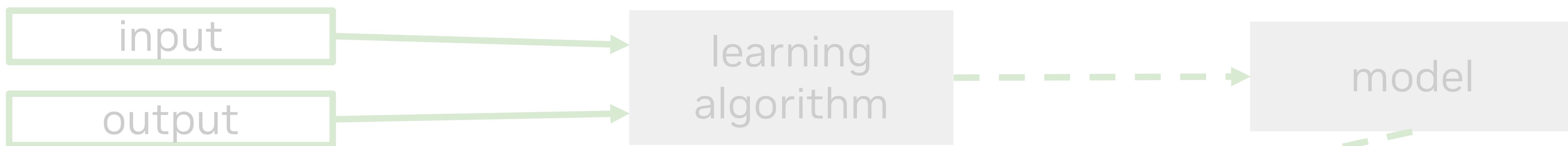
Traditional Programming: Rules-Based

Explicitly define rules and logic to follow

Rules-based



Training



Inference



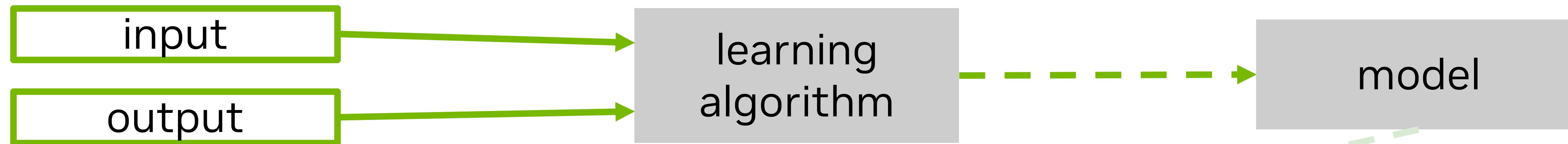
Model Training

Models learn patterns from data to make predictions or decisions without explicit programming

Rules-based



Training



Inference



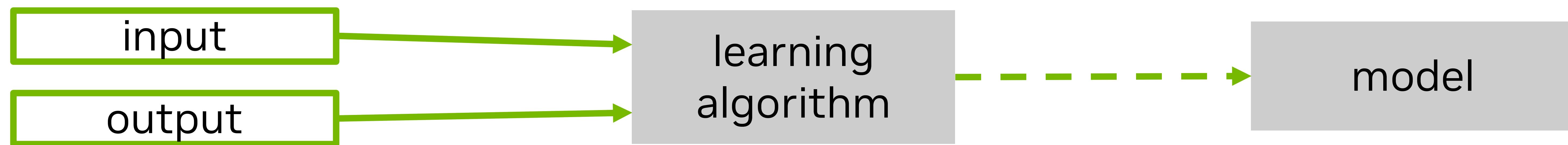
Supervised vs. Unsupervised

Trained on labeled data to make predictions or classifications on new, unseen data

Rules-based



Training



Supervised

- **Regression:** predicting a continuous value
- **Classification:** predicting a discrete label

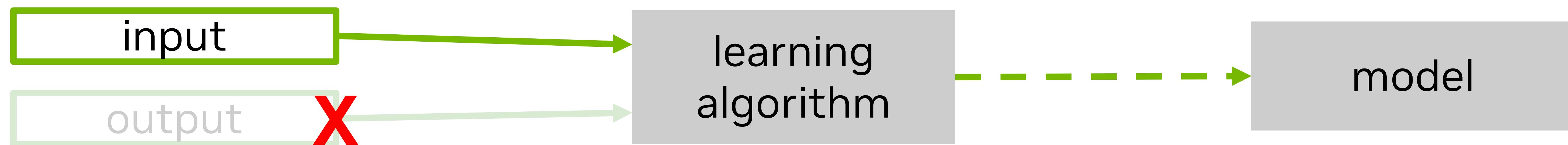
Supervised vs. Unsupervised

Identifying unknown patterns and structures in data without predefined categories

Rules-based



Training



Supervised

- **Regression:** predicting a continuous value
- **Classification:** predicting a discrete label

Unsupervised

- **Clustering:** grouping data points together
- **Dimensionality reduction** while retaining information

Example Dataset

Multiple machine learning tasks can be applicable to the same dataset

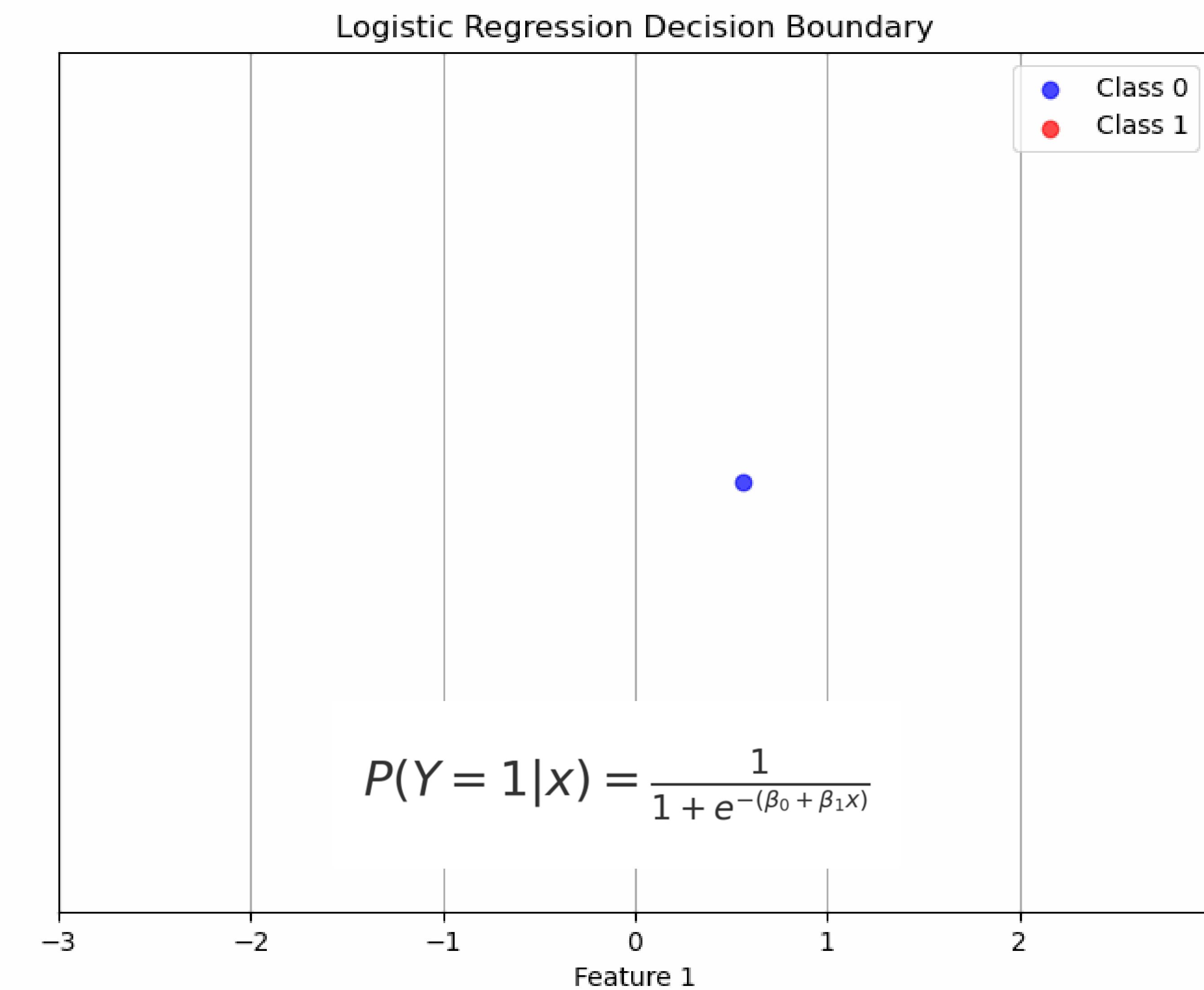
- Features
- Target

	Feature 1	Feature 2	Class
0	0.559	2.389	0
1	1.312	-0.717	1
2	-1.56	-1.925	0
3	-2.281	-0.137	0
4	1.561	-0.428	1
5	-0.808	1.197	0
6	-0.271	-2.256	1
7	0.481	0.549	1
8	-1.208	-1.269	0
9	...		

Example Dataset

Classification assigns data to known categories

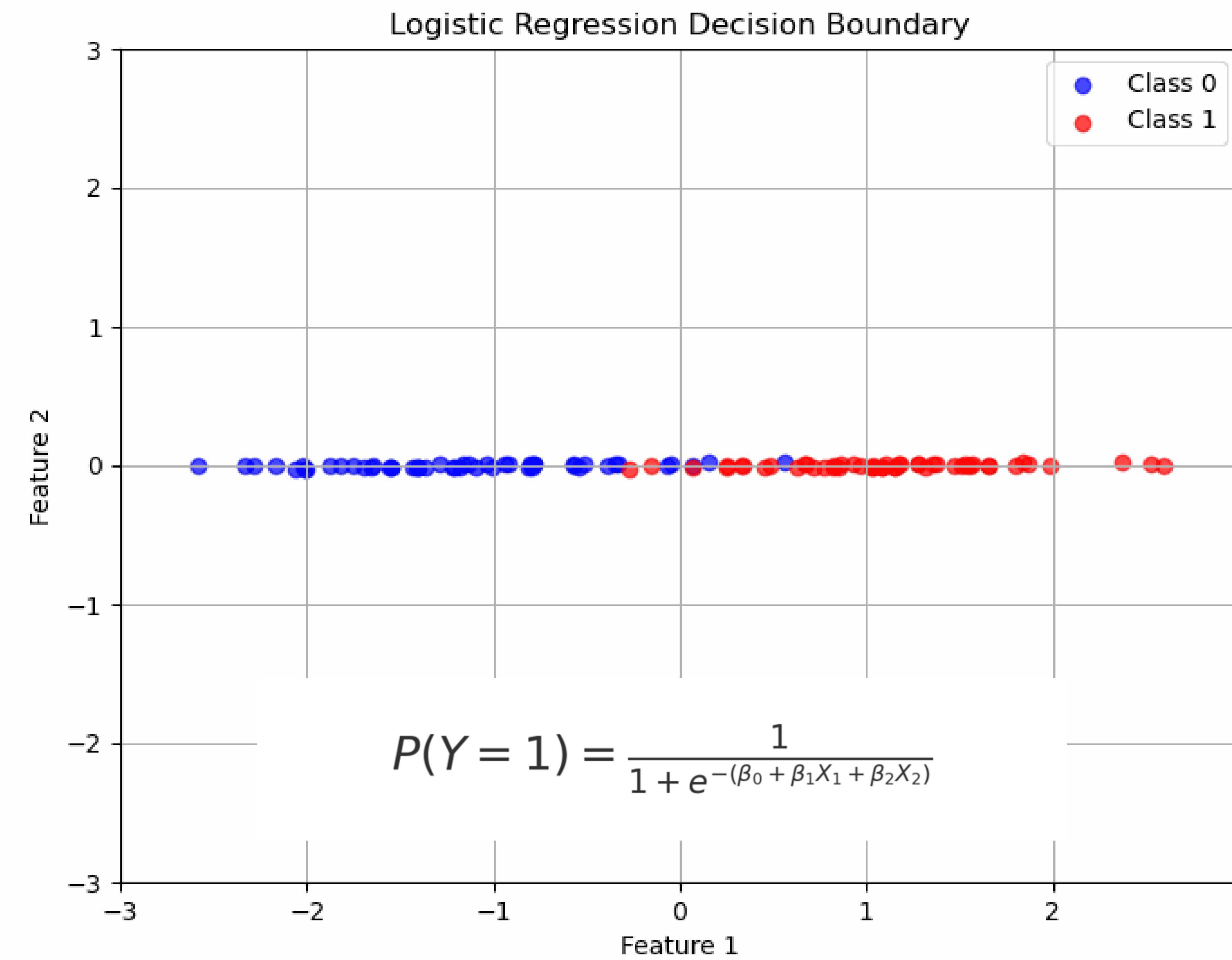
	Feature 1		Class
0	0.559		0
1	1.312		1



Example Dataset

Classification assigns data to known categories

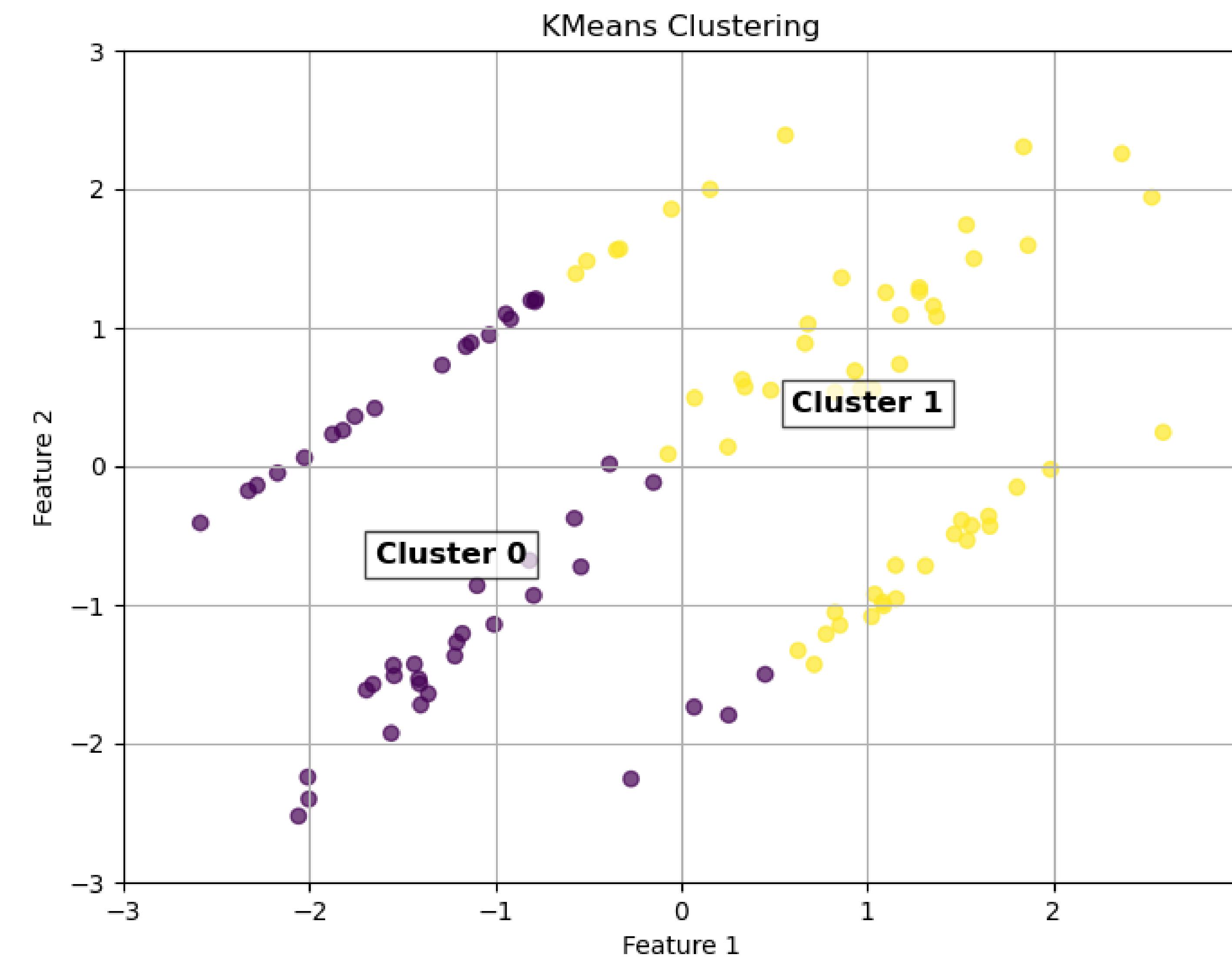
	Feature 1	Feature 2	Class
0	0.559	2.389	0
1	1.312	-0.717	1
2	-1.56	-1.925	0
3	-2.281	-0.137	0
4	1.561	-0.428	1
5	-0.808	1.197	0
6	-0.271	-2.256	1
7	0.481	0.549	1
8	-1.208	-1.269	0
9	...		



Example Dataset

Clustering discovers inherent groups in data without prior labeling

	Feature 1	Feature 2
0	0.559	2.389
1	1.312	-0.717
2	-1.56	-1.925
3	-2.281	-0.137
4	1.561	-0.428
5	-0.808	1.197
6	-0.271	-2.256
7	0.481	0.549
8	-1.208	-1.269
9	...	



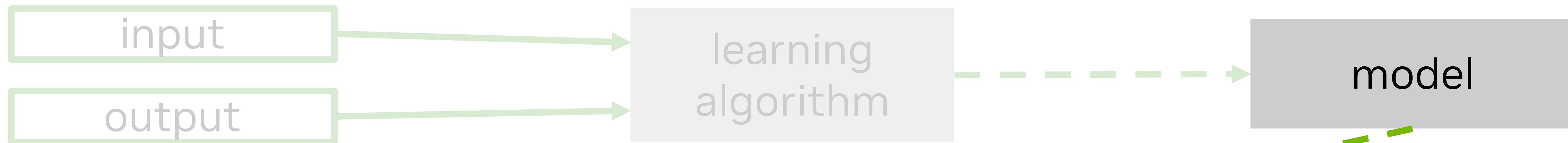
Inference

Process incoming data and producing results based on patterns learned during training

Rules-based



Training

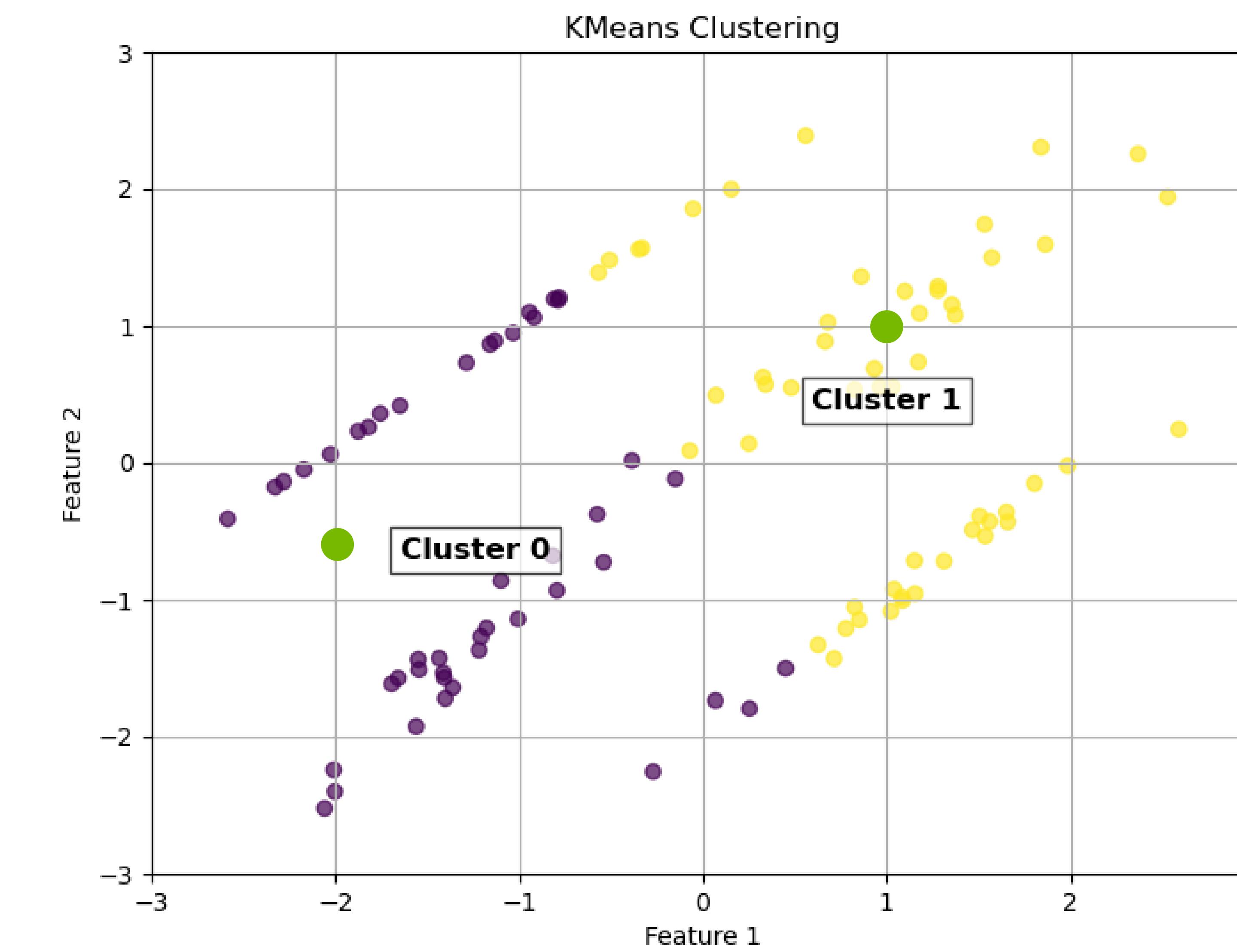
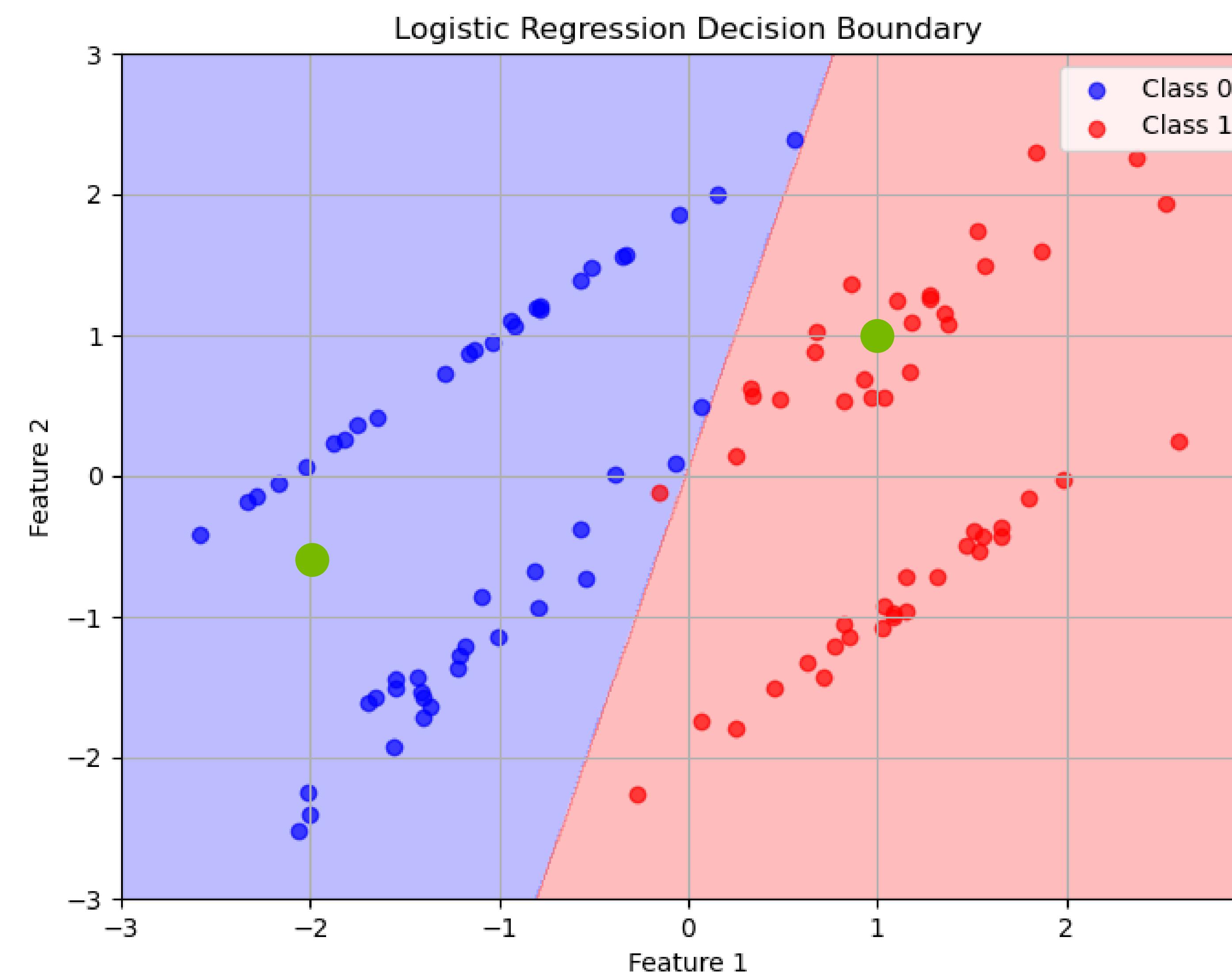


Inference



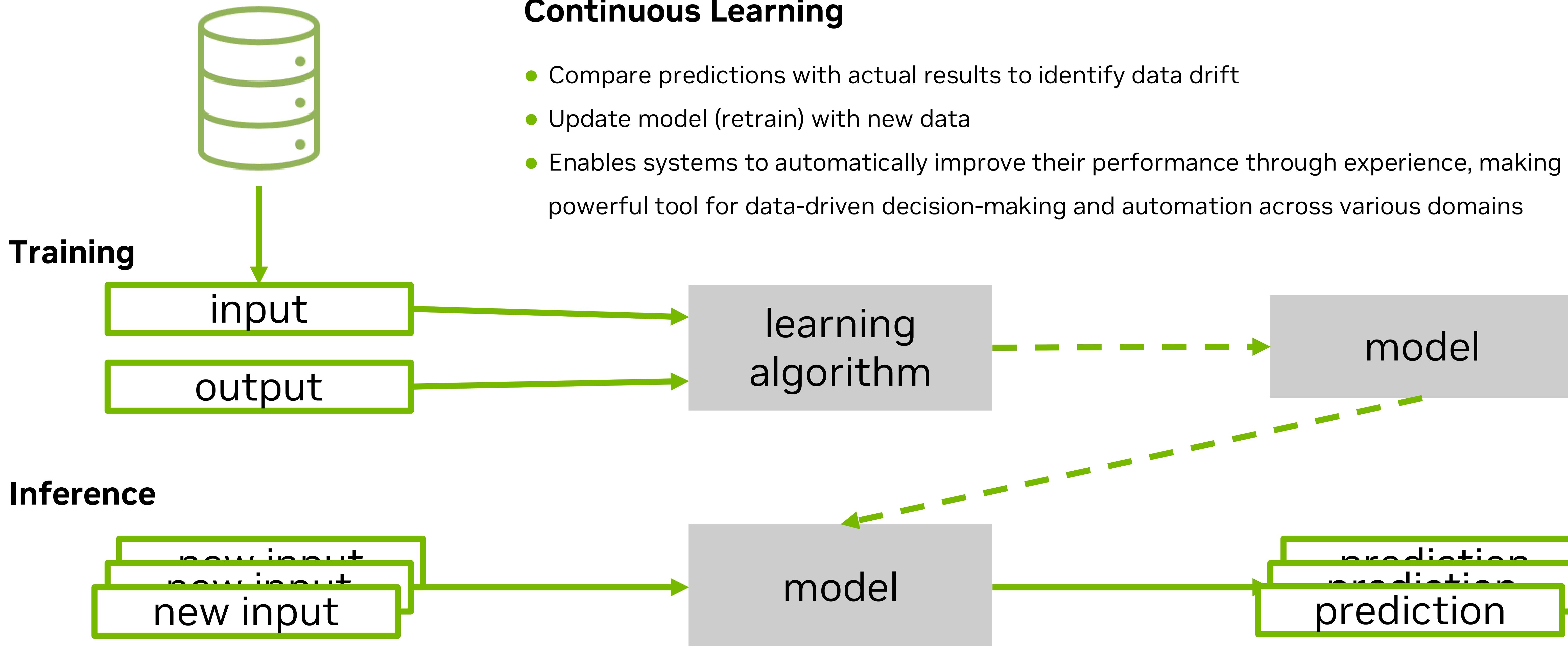
Example Dataset

Process incoming data and producing results based on patterns learned during training



Lifelong Learning

Allows models to update and improve their performance as new data becomes available



Common Use Cases

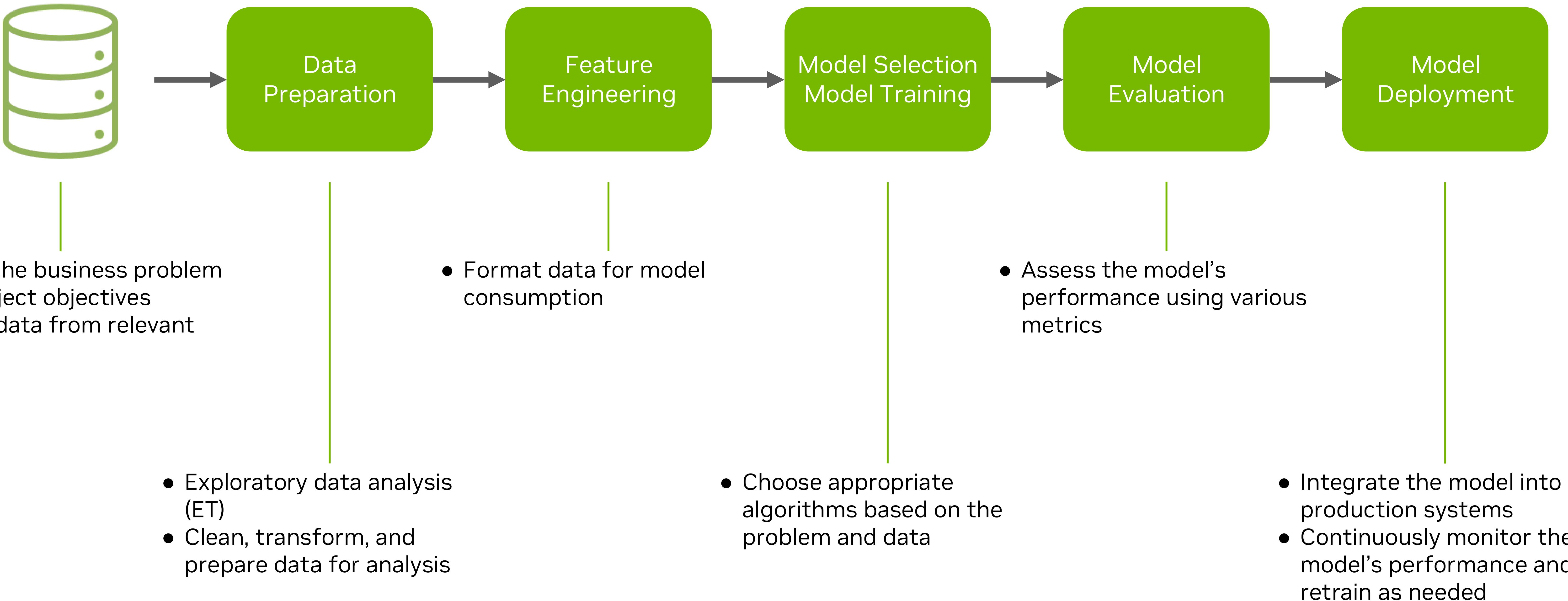
	Supervised	Unsupervised
Online Services	<ul style="list-style-type: none">• Spam / Not Spam• Predict Probability Customer Will Click on Ad	<ul style="list-style-type: none">• Group Articles into Categories• Similar Search Results
Financial Services	<ul style="list-style-type: none">• Credit Card Fraud	<ul style="list-style-type: none">• Anomaly Detection
Telecom	<ul style="list-style-type: none">• Predict Customer Churn	
Health Care	<ul style="list-style-type: none">• Probability of Readmission• Predict Days of Hospital Stay	<ul style="list-style-type: none">• Patient Similarity
Real Estate	<ul style="list-style-type: none">• Predict House Price	
Retail	<ul style="list-style-type: none">• Prejudice Price• Forecast Sales• Sentiment Analysis	<ul style="list-style-type: none">• Similar Customers• Product Similarity• Customer Group• Products Which Are Purchased Together

Benefits:

- **Problem solving:** excels at complex problems where patterns are not easily discernible
- **Adaptability:** models can adapt to new data, new tasks, and improve over time with retraining

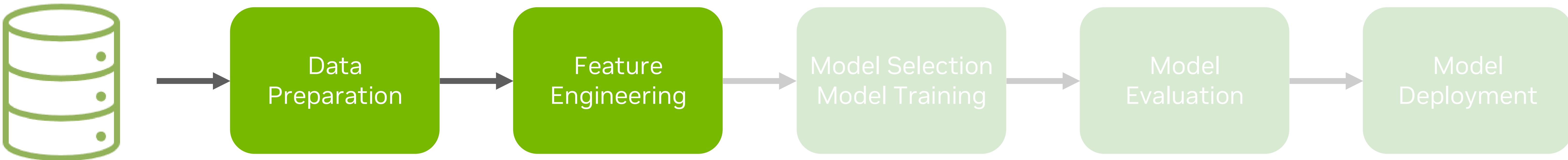
Machine Learning Model Development Workflow

Systematic process of developing, training, evaluating, and deploying machine learning models



Machine Learning Model Development Workflow

Systematic process of developing, training, evaluating, and deploying machine learning models



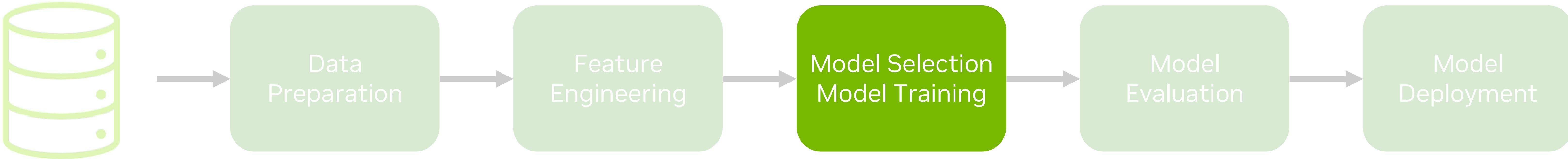
- **Data preparation** – clean, transform, and prepare the data
- **Feature engineering** is often considered one of the most crucial and impactful steps in the machine learning process to improve a model's predictive power and ability to generalize to new data
 - Transforms raw data into a format that better represents the underlying problem, making it easier for algorithms to learn patterns and make predictions
 - Feature engineering is an iterative process that allows for continuous refinement and improvement of model inputs

Use cuDF to create features for downstream ML tasks:

- cuML
- XGBoost
- cuDNN

Machine Learning Model Development Workflow

Systematic process of developing, training, evaluating, and deploying machine learning models



- **Model selection** – choosing the appropriate algorithm and configuration(s)
- **Model training** – fitting the model to the dataset
 - Faster runtime shortens development time and improves productivity

Task	Goal	Algorithm / Model
Classification	Assign input data to predefined categories or labels	Logistic Regression, Decision Trees, Random Forest, SVM, KNN, Naïve Bayes, Neural Networks
Regression	Predict continuous numerical values based on input features	Linear Regression, Ridge & Lasso Regression, Decision Trees, Random Forest, SVR, Neural Networks, XGBoost
Clustering	Group similar data points together without predefined labels	k-Means, Hierarchical Clustering, DBSCAN, Gaussian Mixture Models (GMM), Mean Shift
Decomposition & Dimensionality Reduction	Reduce the number of input features while preserving important information	PCA, SVD, t-SNE, UMAP, Autoencoders

Classification

Categorize input data into predefined classes

	K-Nearest Neighbors	Logistic Regression	Decision Tree
Approach	KNN makes predictions based on the similarity of new data points to known examples in the training set. The algorithm calculates the distance between the new point and all points in the training set. It identifies the K nearest neighbors and uses majority voting among the K neighbors	Logistic regression is a statistical method used for binary classification. It predicts the probability of an instance belonging to a particular class, using the logistic function (sigmoid) to map predictions to probabilities between 0 and 1	Creates a tree-like model of decisions based on features in the data. From the root node, it splits the data into subsets based on the most significant attributes. The splitting criteria is determined by maximizing information gain or minimizing impurity
Pros	<ul style="list-style-type: none">Makes no assumption about data distribution“Lazy” learning algorithmAdapts quickly to new data	<ul style="list-style-type: none">Generally efficientLess prone to overfitting	<ul style="list-style-type: none">Does not require feature scalingRobust to outliersCan handle categorical variablesMakes no assumption about data distribution
Cons	<ul style="list-style-type: none">Prone to overfittingSensitive to outliersRequires feature scalingHigh computational cost for large datasets, especially during predictionHigh memory requirement (needs to store the entire training dataset)	<ul style="list-style-type: none">Sensitive to outliersAssumes linear relationship between features and log-odds of outcomeSuitable for binary outcomeAssumes no correlation between featuresRequires irrelevant features to be excludedCan overfit on high-dimensional datasets	<ul style="list-style-type: none">Prone to overfittingComputationally expensive for large datasets

Decision Tree

Decision trees are prone to overfitting

Recursively splitting the data based on features to create a tree-like structure

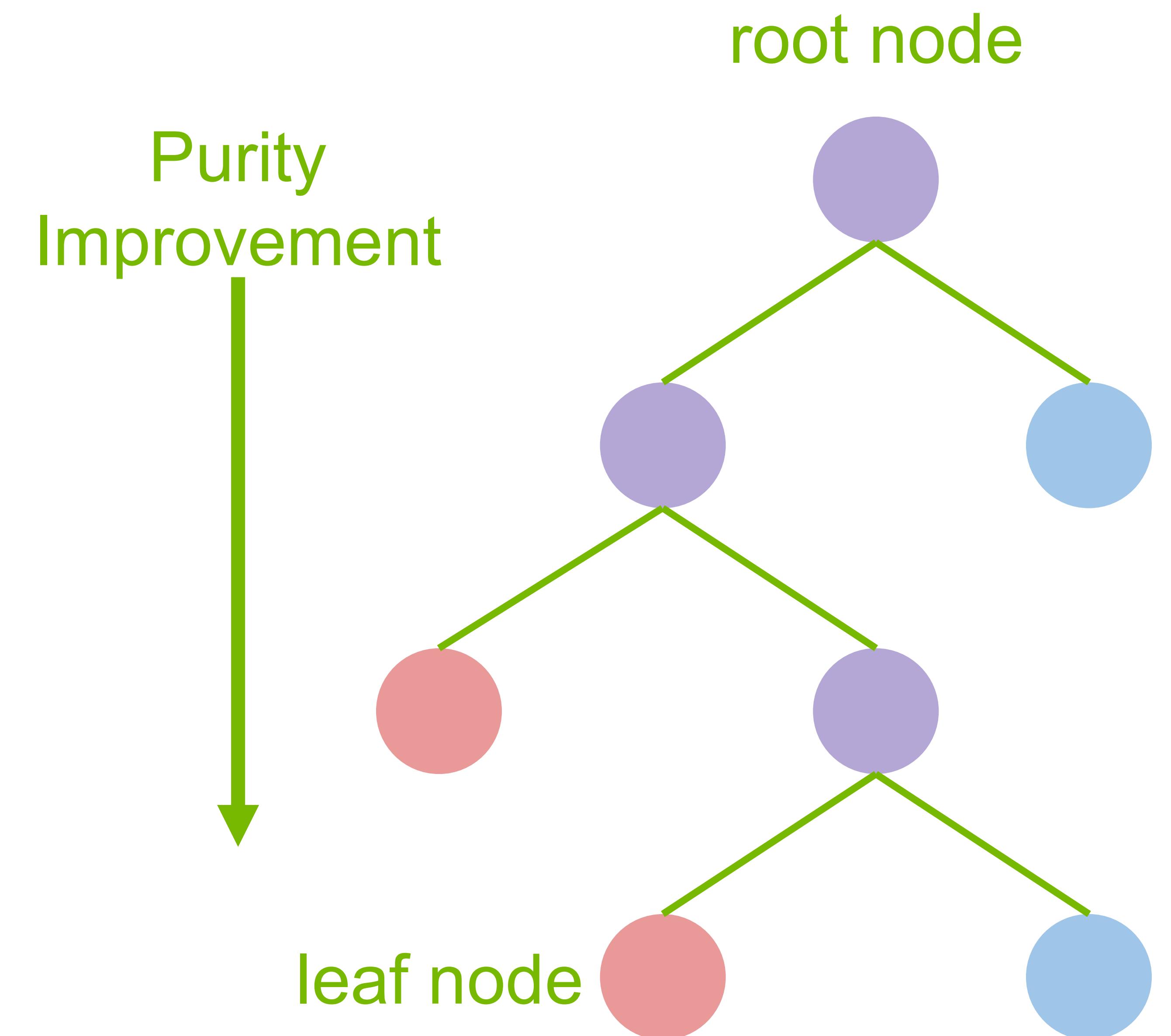
- Select the best feature to split the data to increase the purity of the resulting subsets with respect to the target variable

Challenges:

- Prone to overfitting

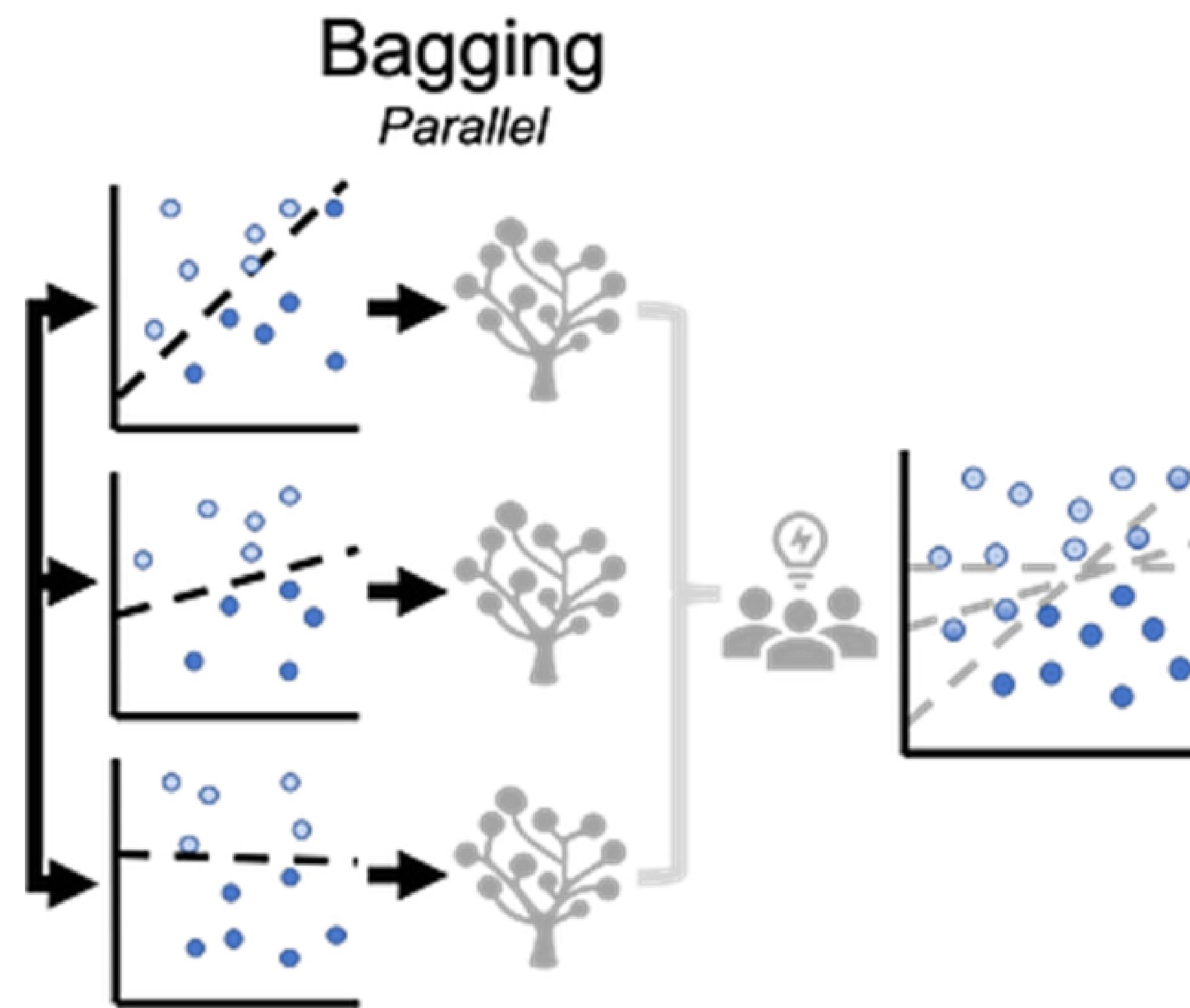
Solution:

- Pruning
- Ensemble

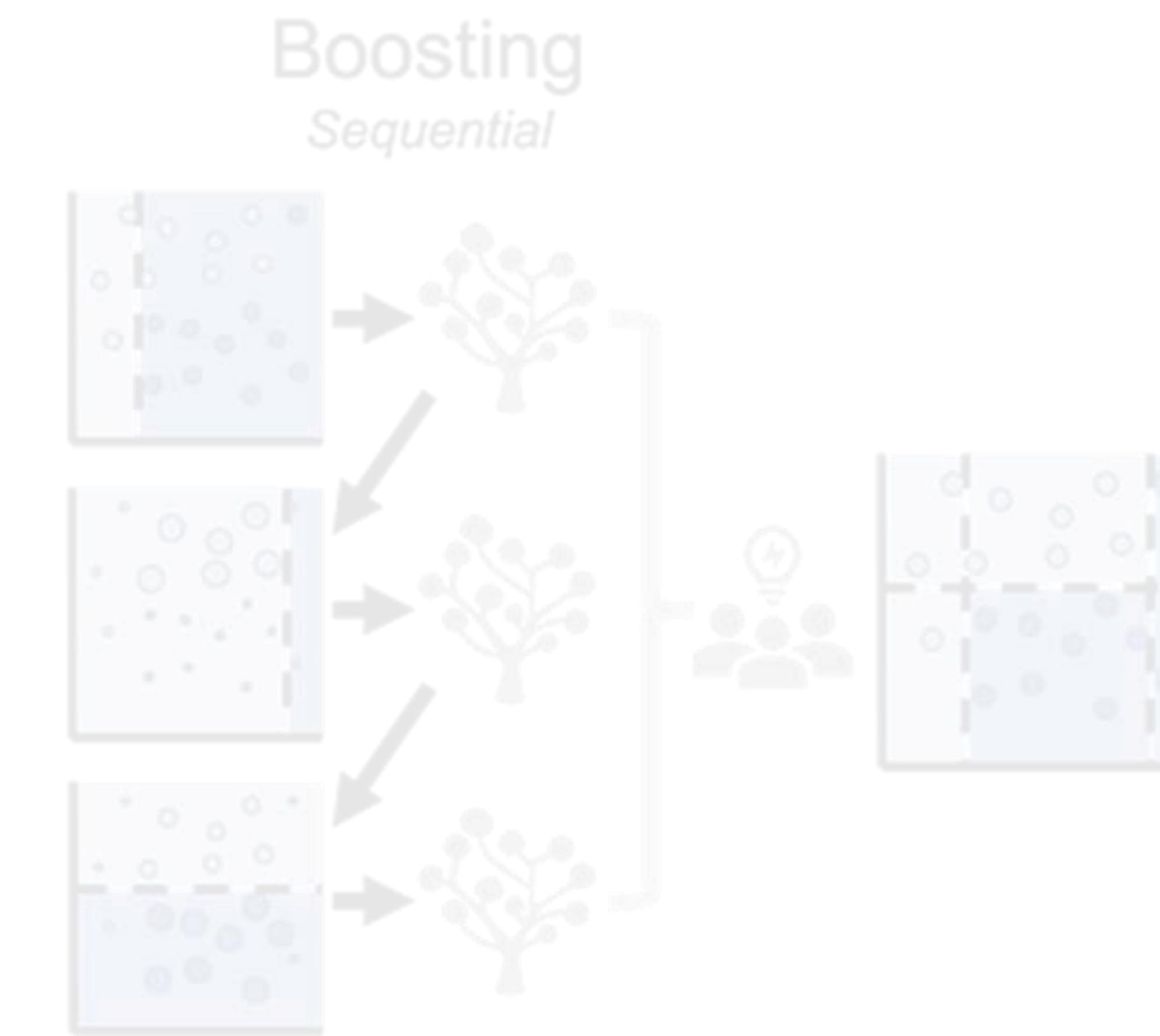


Ensemble

Using multiple decision trees



(Random Forest) Bagging is an ensemble technique used to reduce the variance of predictions by considering results from multiple decision tree models that were trained on different sub-samples of the same data set



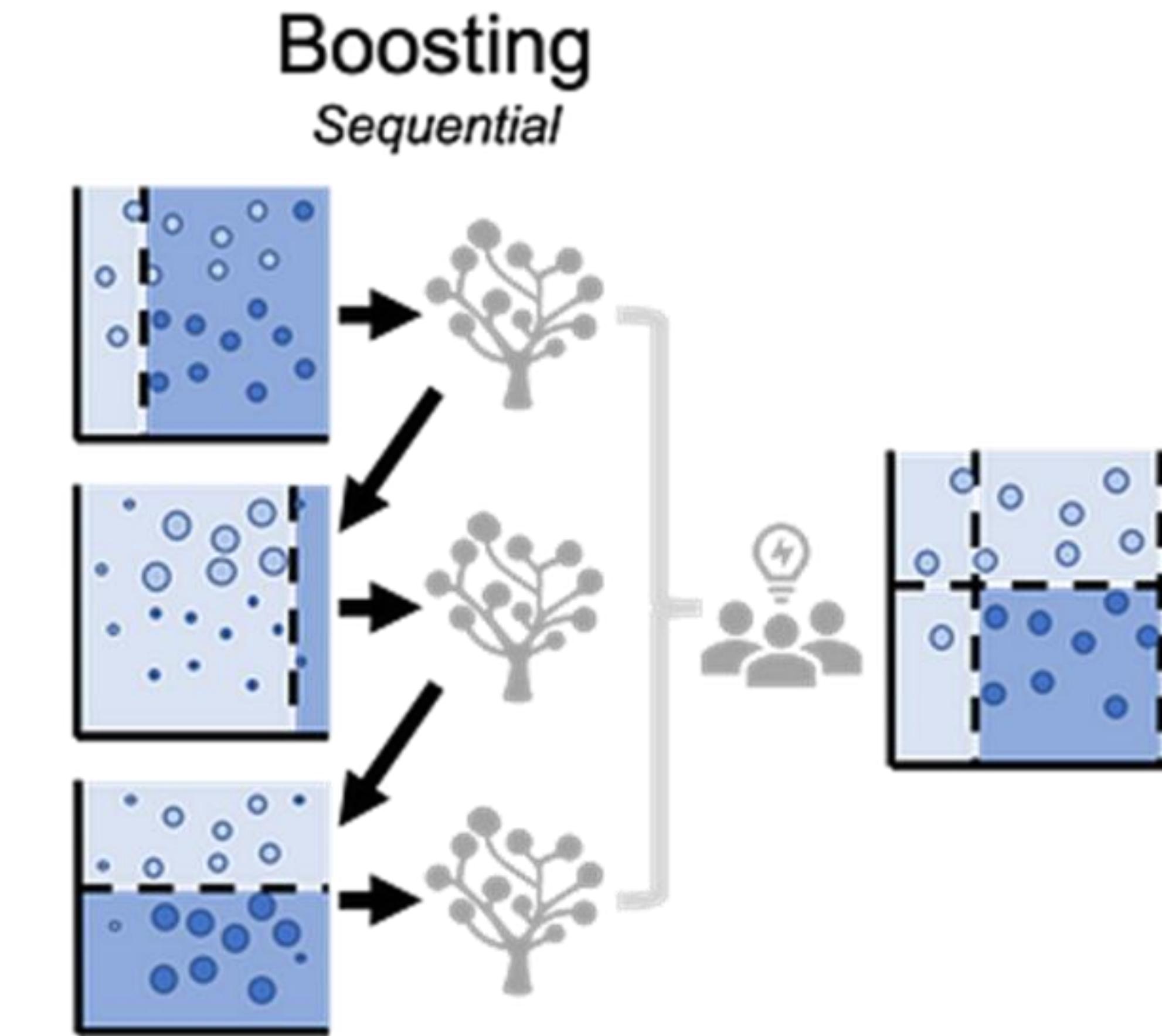
(Gradient Boosting Decision Tree) Boosting is another type of ensemble learning that combines weak learners to achieve improved model performance. Weak learners are simple models that predict relatively poorly. The concept of boosting is to sequentially train models, each time trying to fit better than before using residual errors

Ensemble

Using multiple decision trees



(Random Forest) Bagging is an ensemble technique used to reduce the variance of predictions by considering results from multiple decision tree models that were trained on different sub-samples of the same data set



(Gradient Boosting Decision Tree) Boosting is another type of ensemble learning that combines weak learners to achieve improved model performance. Weak learners are simple models that predict relatively poorly. The concept of boosting is to sequentially train models, each time trying to fit better than before using residual errors

Clustering

Group unlabeled examples based on their similarity to each other



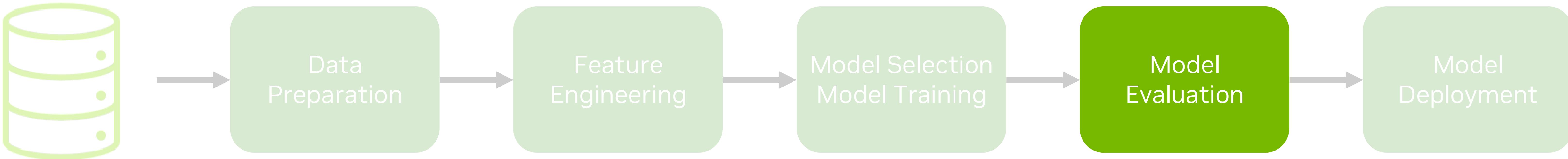
	K-Means	DBSCAN
Approach	Partitions data by iteratively moving cluster centers and re-assigning data points based on which center is closest to the point	Uses spatial density to cluster—finds cluster points within at most epsilon distance of another point in the cluster
Pros	<ul style="list-style-type: none">• Fast, simple, and easy to understand• Scalable to large datasets	<ul style="list-style-type: none">• Can identify clusters of any arbitrary shapes• Automatically determines the number of clusters• Robust to outliers
Cons	<ul style="list-style-type: none">• Assumes spherical clusters• Requires specifying the number of clusters in advance• Sensitive to outliers	<ul style="list-style-type: none">• Less efficient for very large datasets• Requires specifying minimum points and maximum neighborhood distance

Use cases:

- Market segmentation
- Anomaly detection
- *Data Imputation*

Machine Learning Model Development Workflow

Systematic process of developing, training, evaluating, and deploying machine learning models



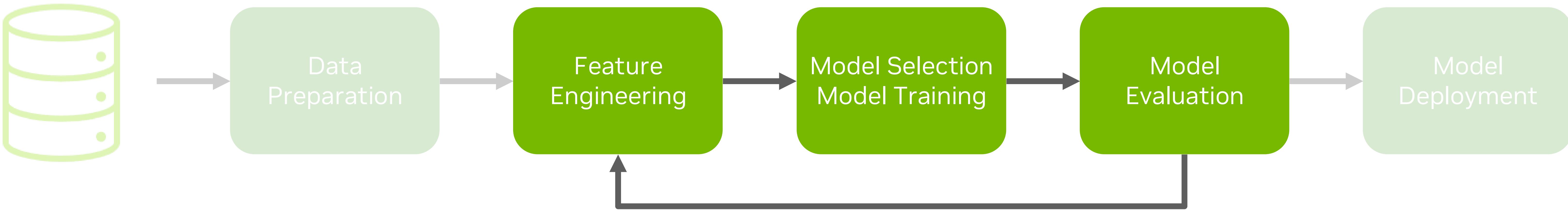
- **Model evaluation** – assessing the quality and performance of a trained machine learning model using various metrics
 - Balance accuracy and inference speed
 - Maximize throughput given an application's accuracy requirement and latency budget

Evaluation metrics:

- **Accuracy**
 - Underfit vs overfit
 - Cross validation is a resampling method to protect against overfitting
- **Latency and throughput**
 - Evaluate how well the inference performance scales with increasing data volumes or concurrent requests
 - Ensure that optimizations for speed don't significantly degrade the model's accuracy or performance on its intended task

Machine Learning Model Development Workflow

Systematic process of developing, training, evaluating, and deploying machine learning models



- **Model tuning**

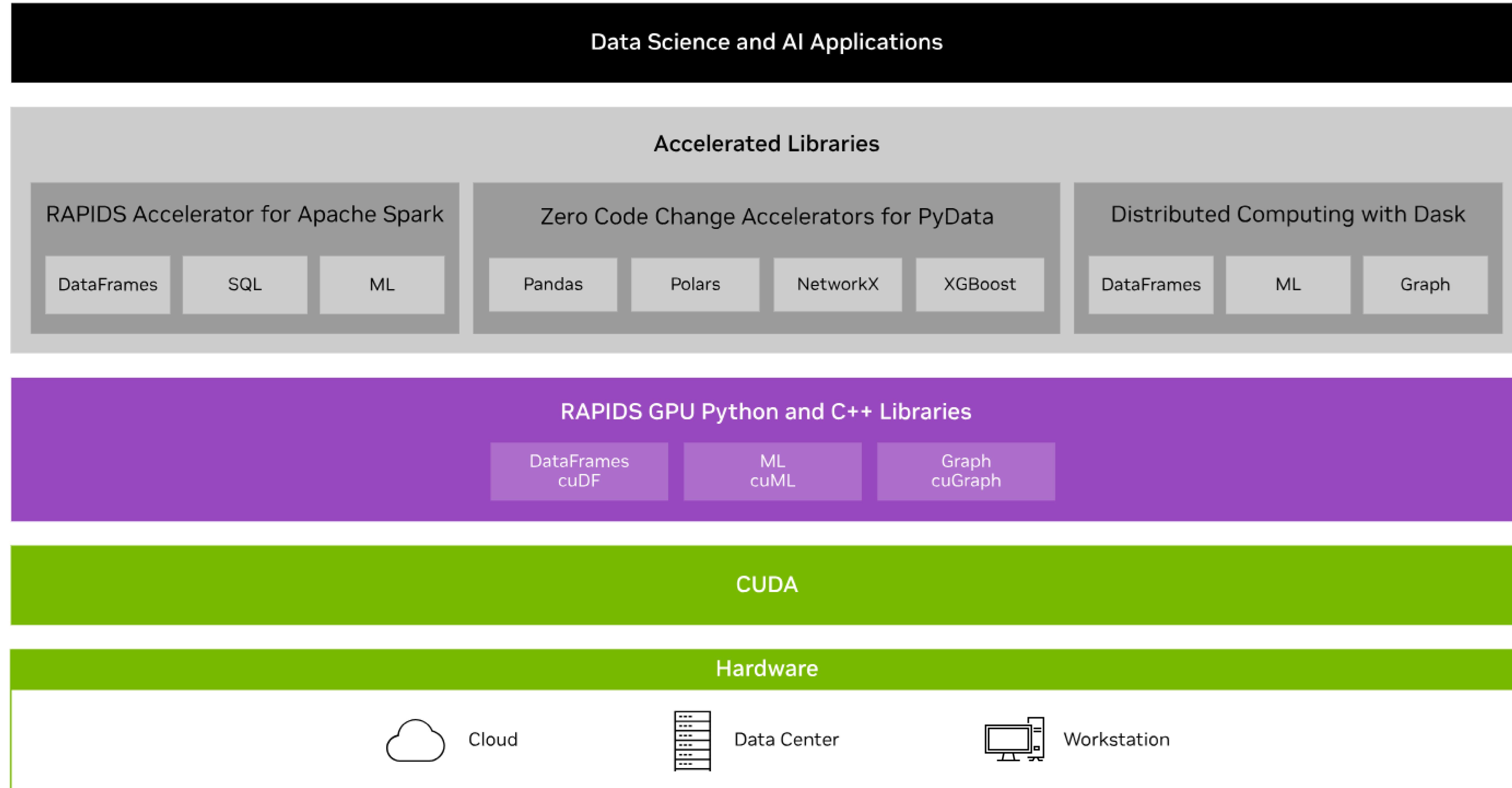
- Feature selection finds the optimal subset of features
- Hyperparameter tuning using techniques such as **grid search**

Use cuML to accelerate machine learning model training

- Low data processing cycle time translates to higher productivity, more experimentation, and improved model performance

RAPIDS Transforms Data Science

An optimized hardware-to-software stack for the entire data science pipeline



XGBoost + RAPIDS

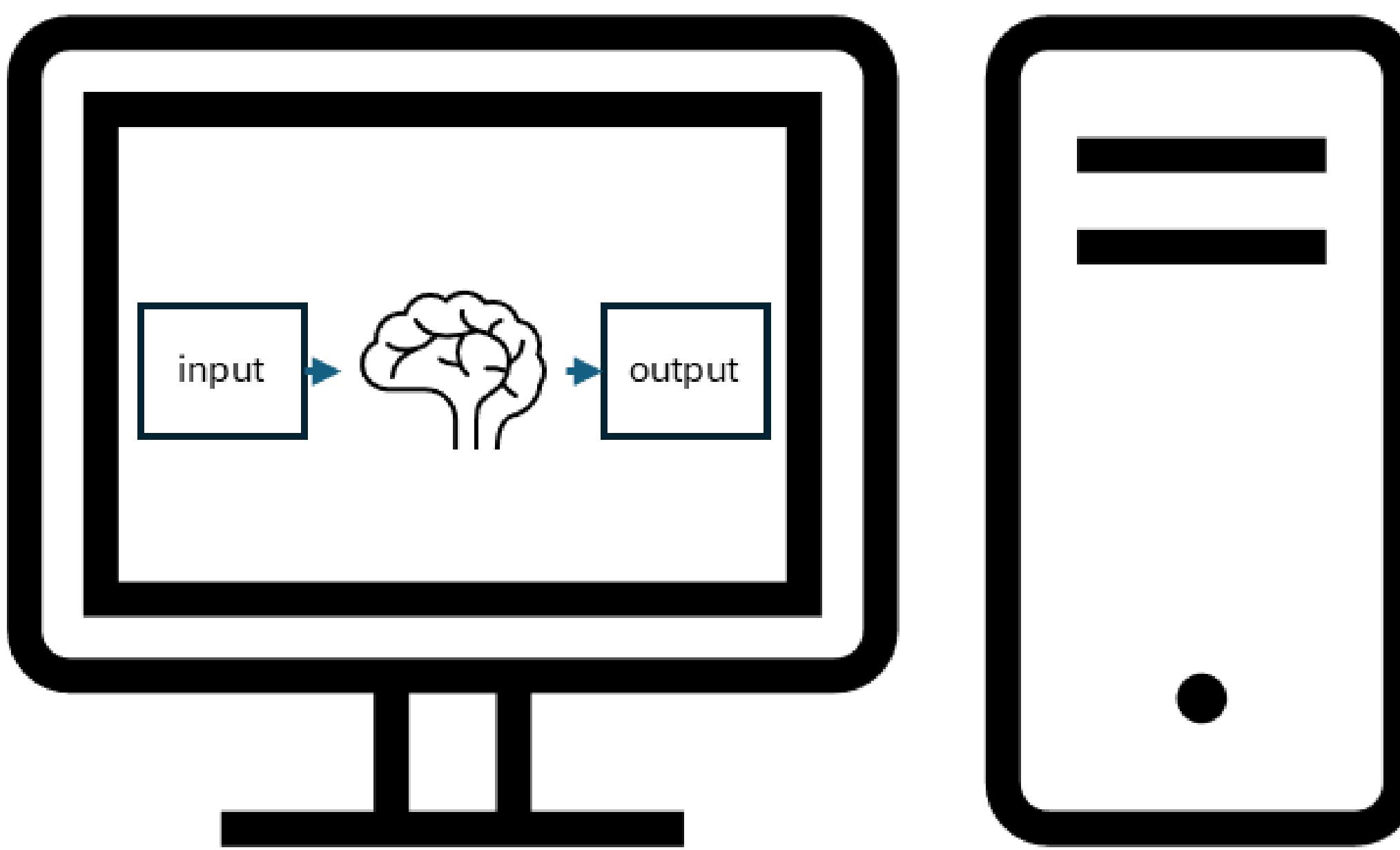
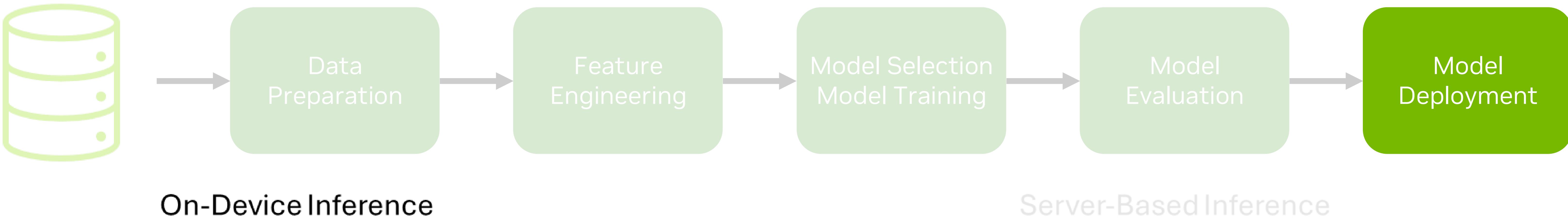
Optimized gradient boosting machine learning algorithm

- Extreme Gradient Boosting, is a scalable, distributed gradient-boosted decision tree machine learning algorithm
- Gold standard in single model performance for classification and regression
 - Delivers excellent accuracy and efficiency
 - Handles large datasets effectively through multi-threading
 - Handles imbalanced datasets and outliers
 - Uses regularization to prevent overfitting and improve model generalization
- RAPIDS was built from the start to integrate with this amazing algorithm, including seamless GPU-acceleration such as loading data from cuDF DataFrames and cuPy arrays - this significantly speeds up model training and improves accuracy for better predictions



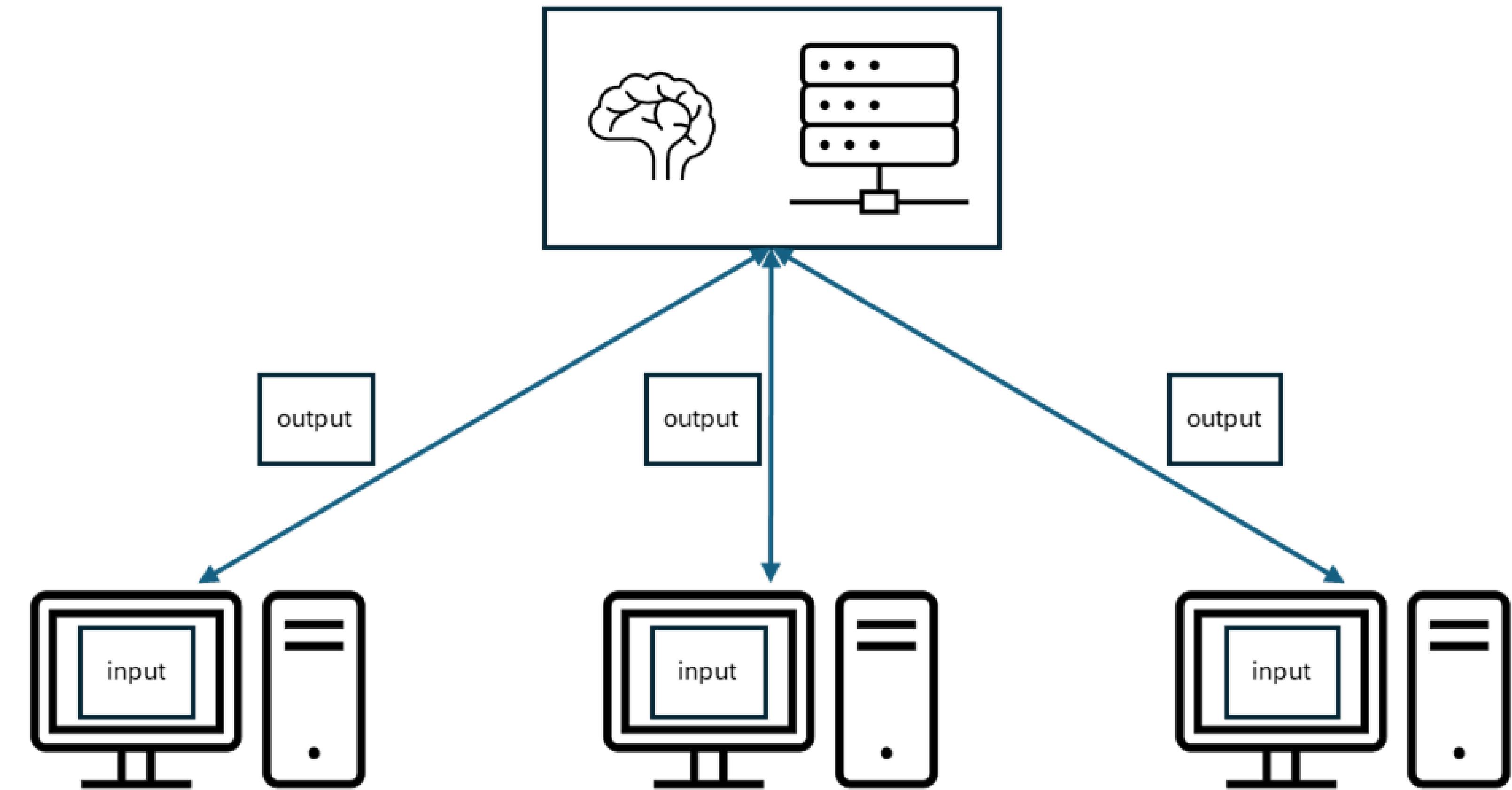
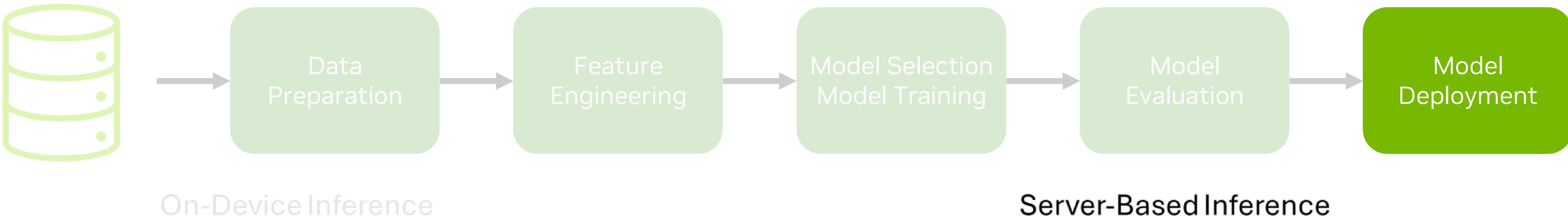
Machine Learning Model Development Workflow

Systematic process of developing, training, evaluating, and deploying machine learning models



Machine Learning Model Development Workflow

Systematic process of developing, training, evaluating, and deploying machine learning models



Triton Inference Server

Open-source inference serving software that is designed for deploying AI models at scale

Key Features:

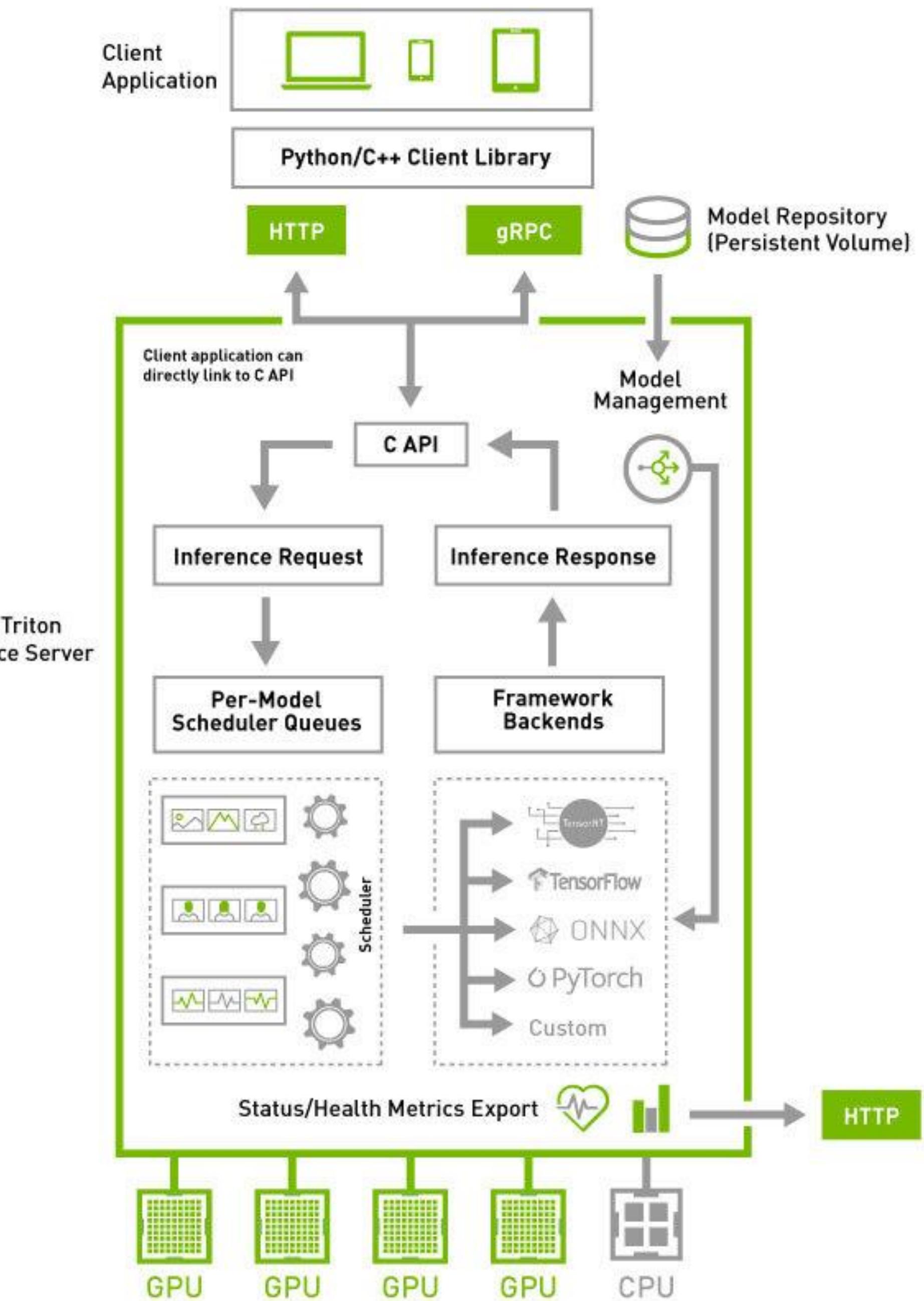
- Multi-framework support (TensorFlow, PyTorch, ONNX, etc.)
- Concurrent model execution
- Dynamic batching for improved performance
- Model versioning and ensembling

Benefits for Data Scientists:

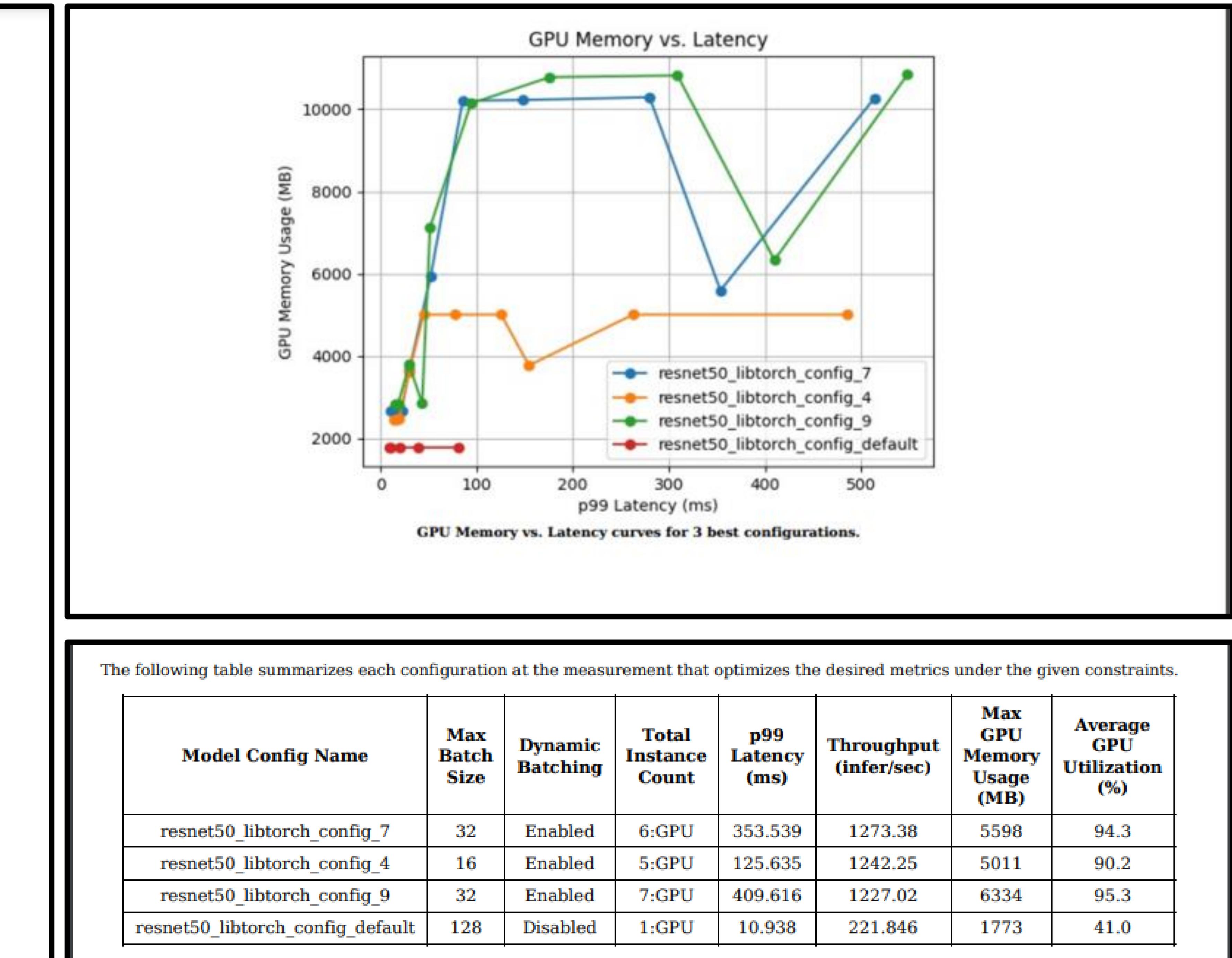
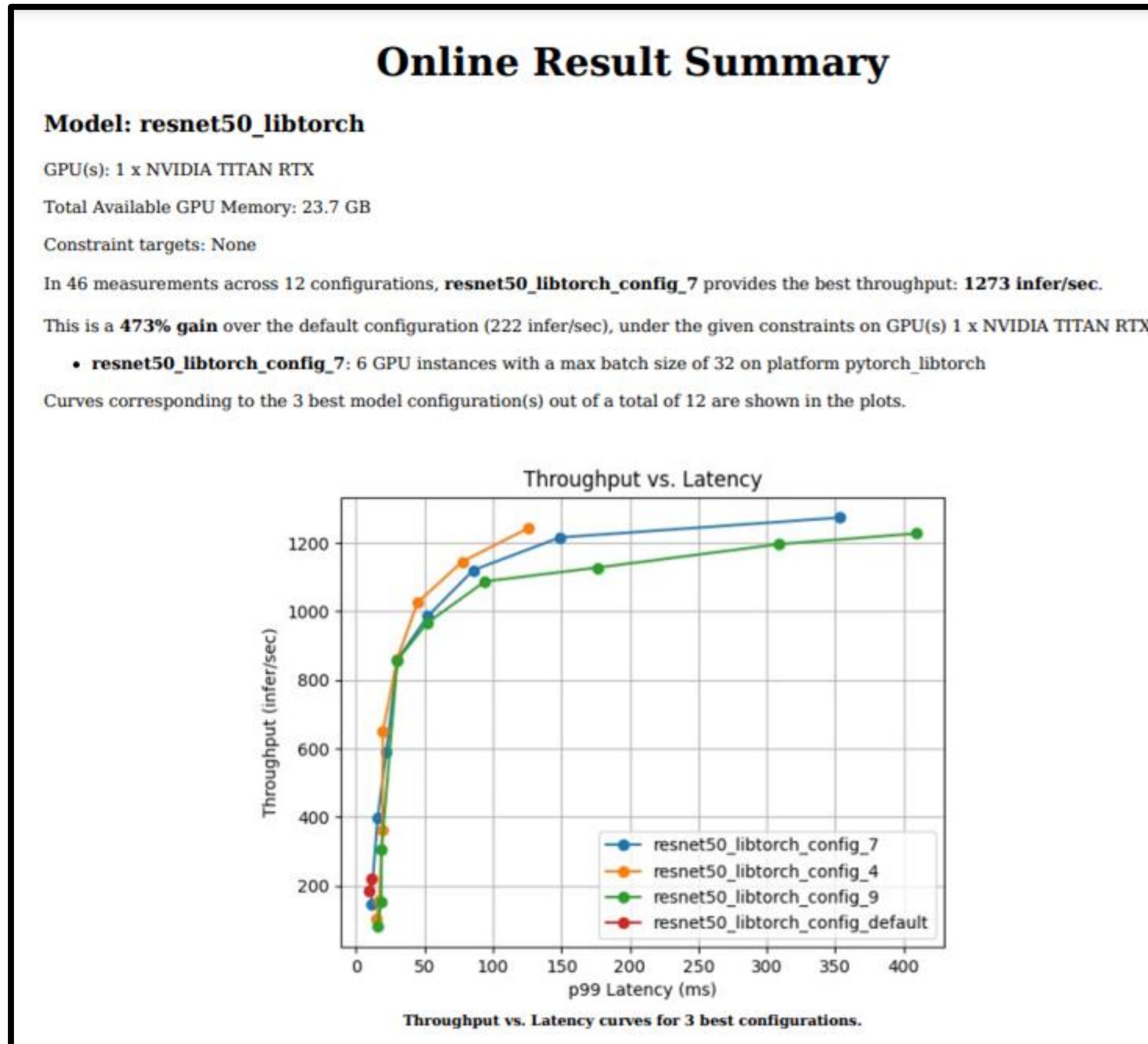
- Simplifies model deployment in production environments
- Enables efficient scaling of machine learning models
- Provides a consistent interface across different ML frameworks

Optimizing Performance:

- Performance analyzer
- Model analyzer

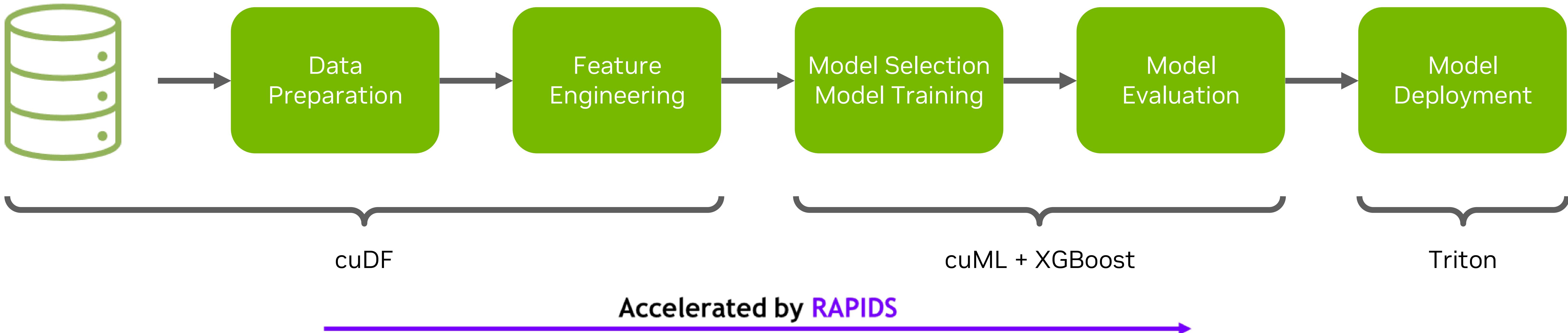


Example of Model Analyzer Report



Machine Learning Model Development Workflow

Systematic process of developing, training, evaluating, and deploying machine learning models

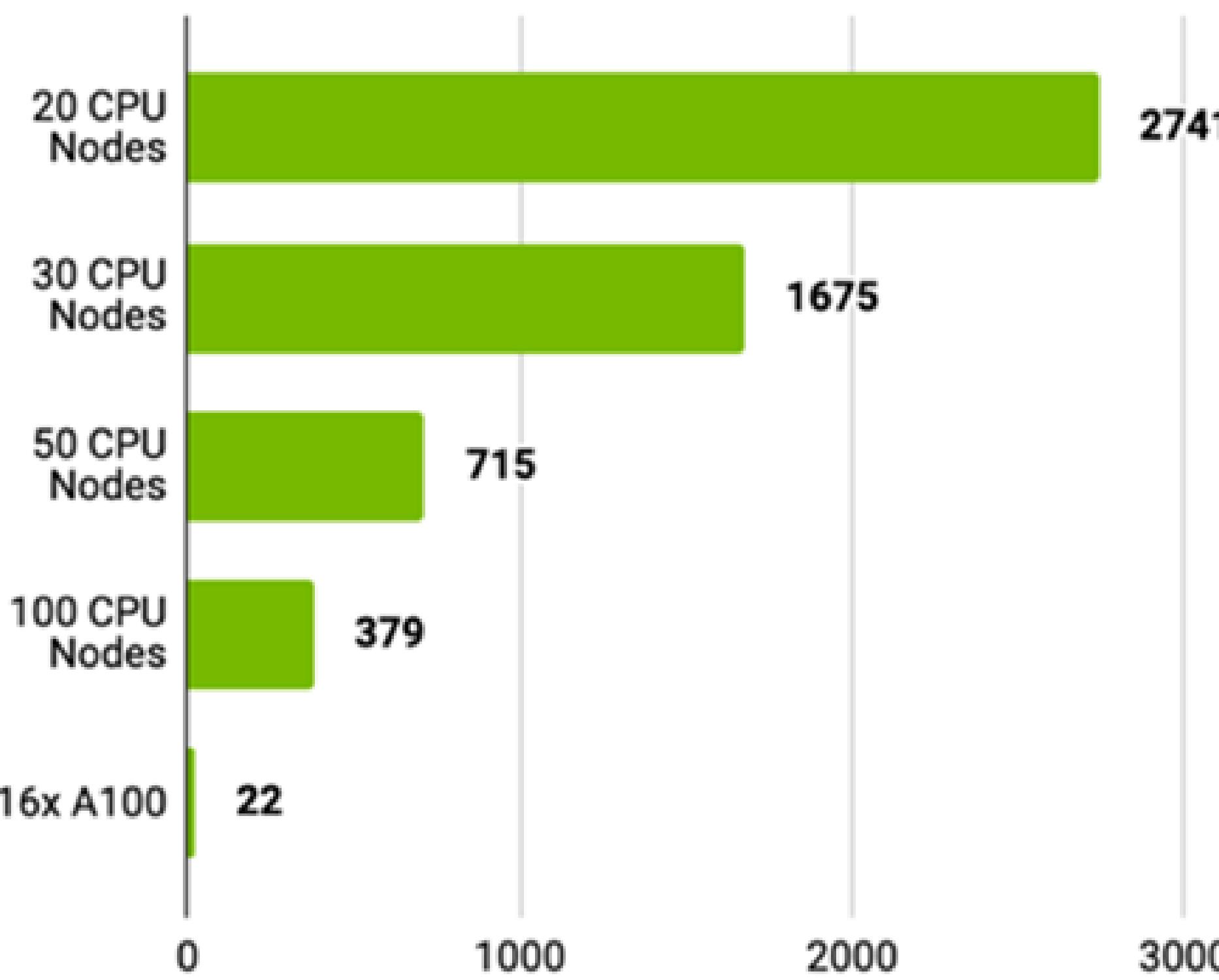


- Efficient data processing on large datasets leads to better results
 - Facilitates higher data quality and quantity that directly impact model performance
 - Increases productivity
 - Enables more experimentation for model tuning
 - Fosters innovation and progress
 - Reduces time to insight and total cost of ownership
- Seamlessly integrate with existing workflows through plugins
- Optimal model deployment that balances performance and resource allocation

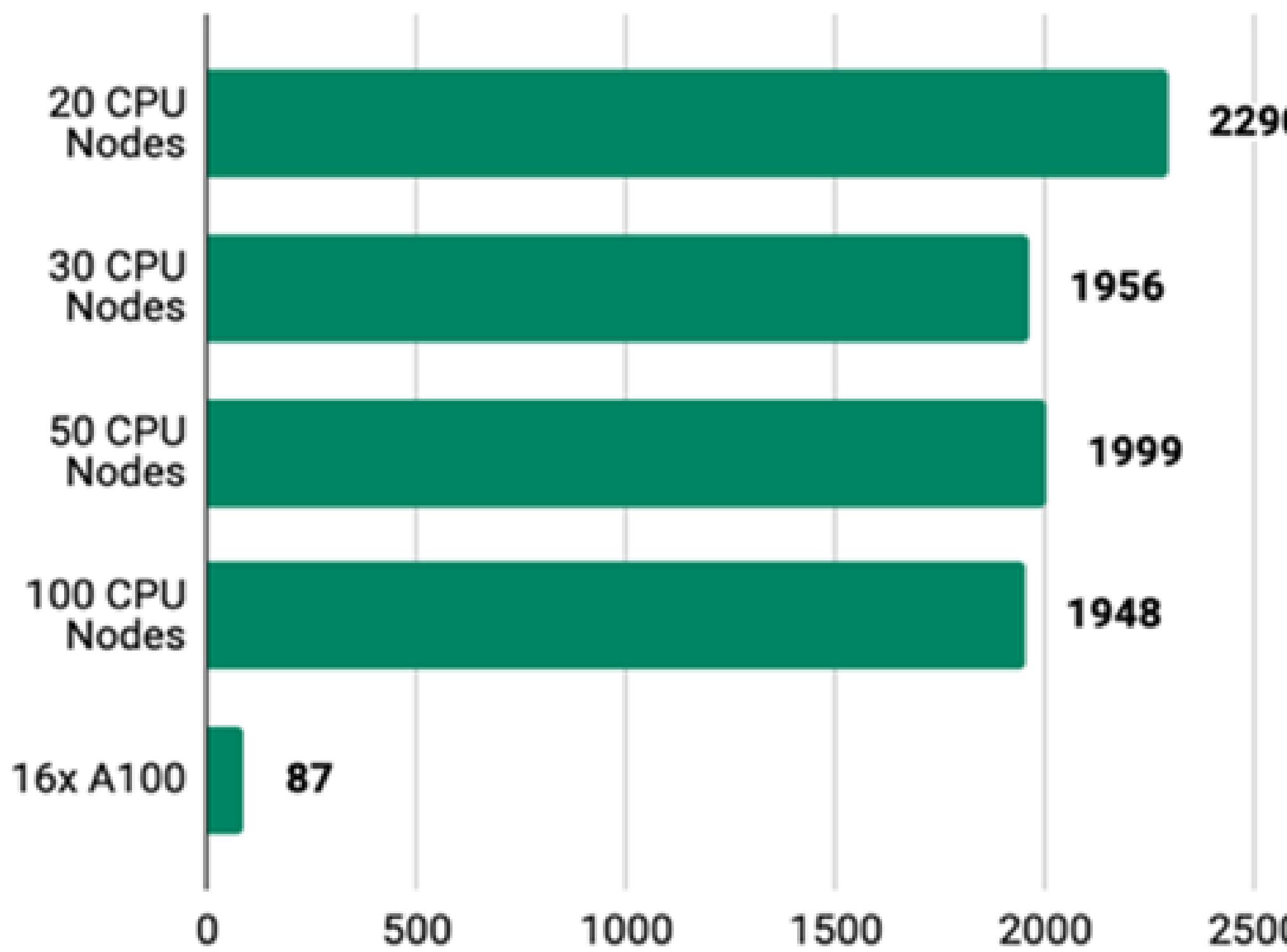
Proven Faster for Data Science

Realize efficiency and scalability with multi-GPU for massive datasets

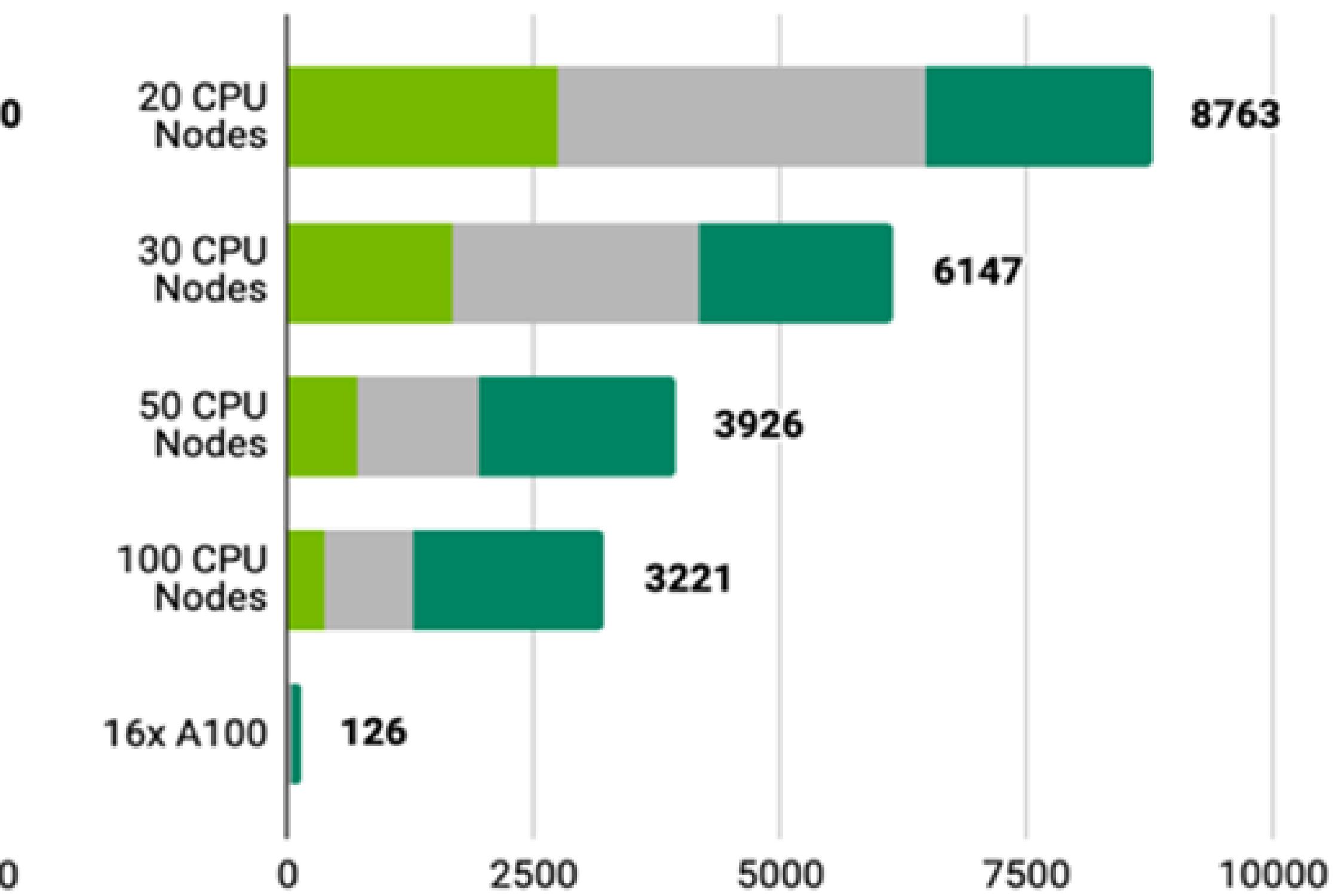
cuIO/cuDF - Load and Data Preparation



XGBoost Machine Learning



End-to-End Model Training Pipeline



Benchmark

200GB CSV dataset; Data preparation includes joins, variable transformations.

CPU Cluster Configuration

CPU nodes (61 GiB of memory, 8 vCPUs, 64-bit platform), Apache Spark

DGX Cluster Configuration

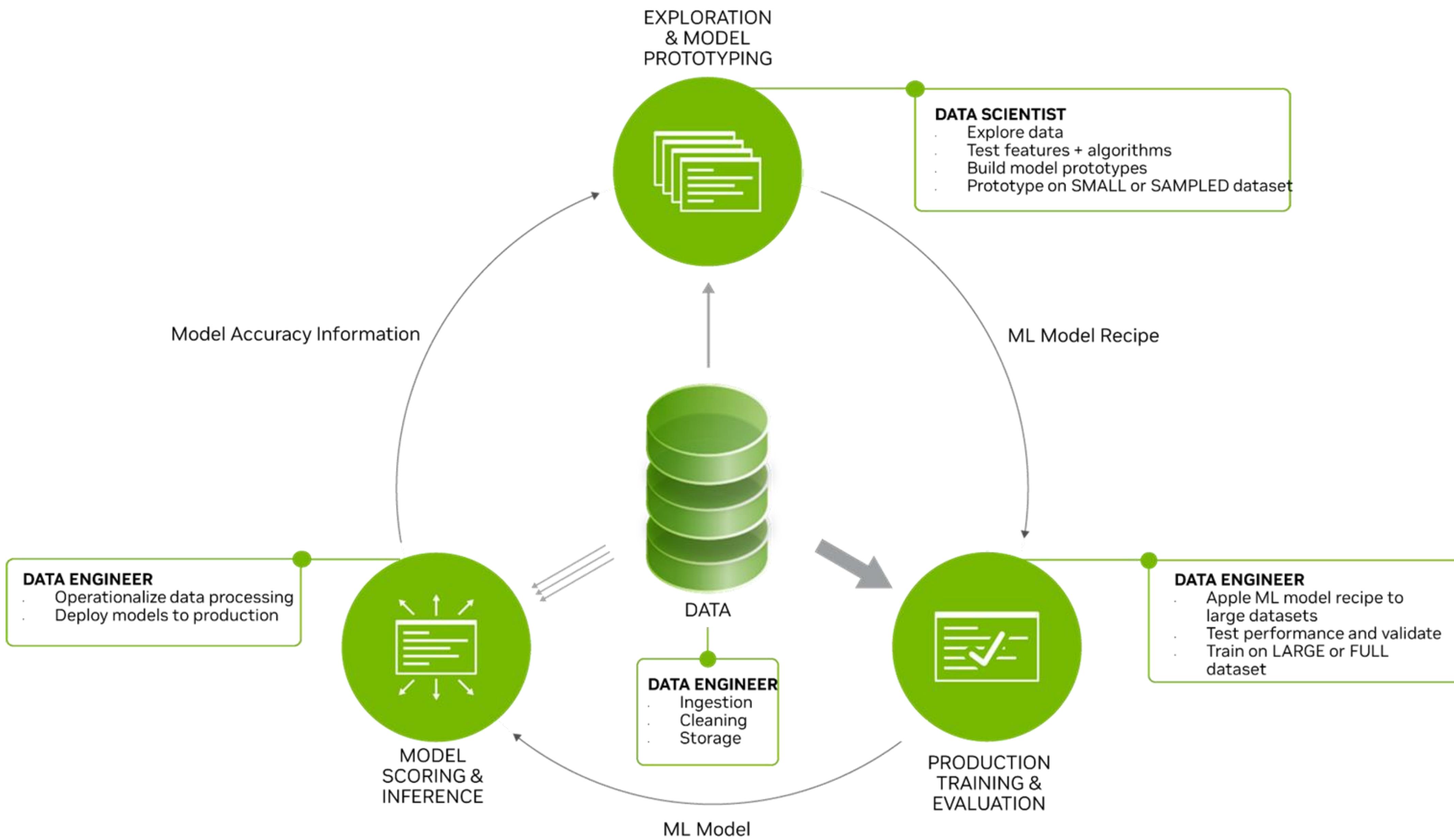
5x DGX-1 on InfiniBand network

Time in seconds — Shorter is better

■ cuIO / cuDF (Load and Data Preparation) ■ Data Conversion ■ XGBoost

Data Flywheel

Improve performance through the cyclical nature of data-driven growth and innovation



Hands-On Lab