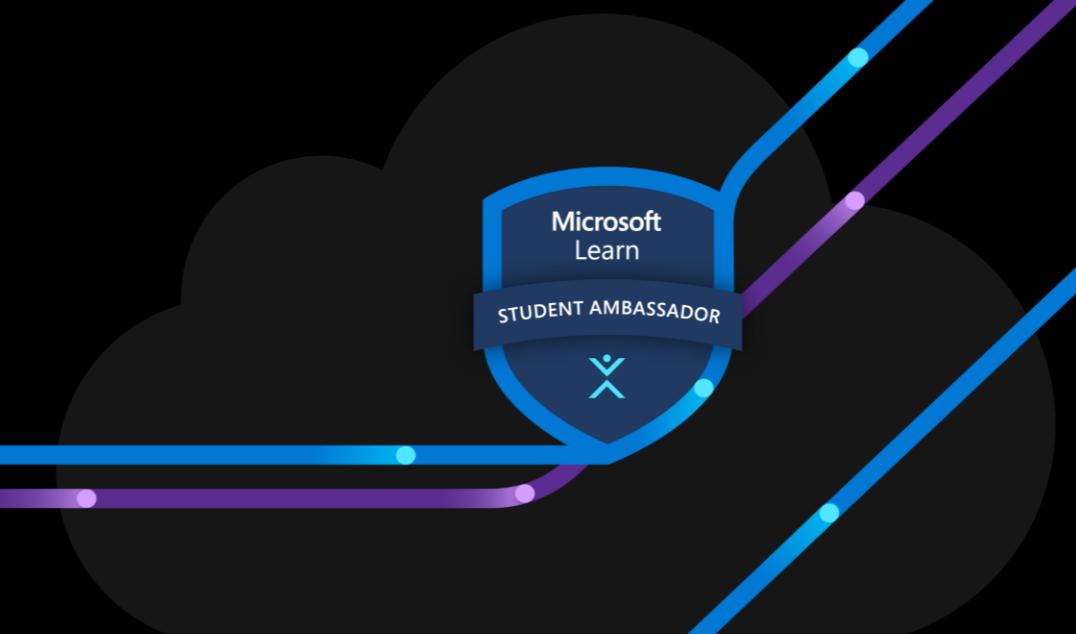


Microsoft Learn  
Student Ambassadors

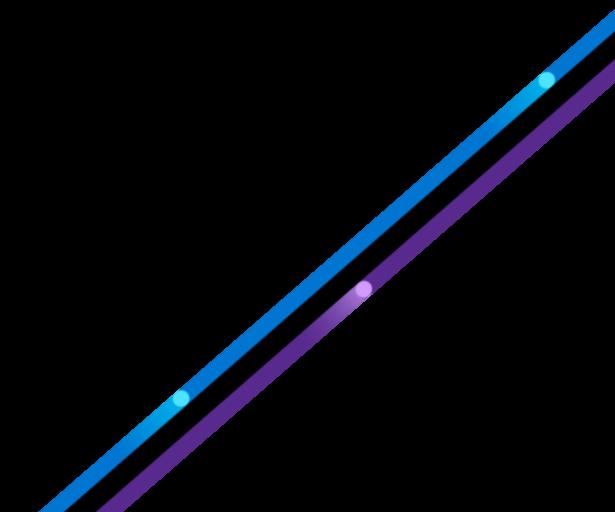
# Azure ML Jumpstart: Unveiling Machine Learning Magic

BY  
HASAN JEHANGIR  
BETA MLSA ,  
MACHINE LEARNING HEAD @TCPC  
PRESIDENT @ AICP UET CHAPTER.



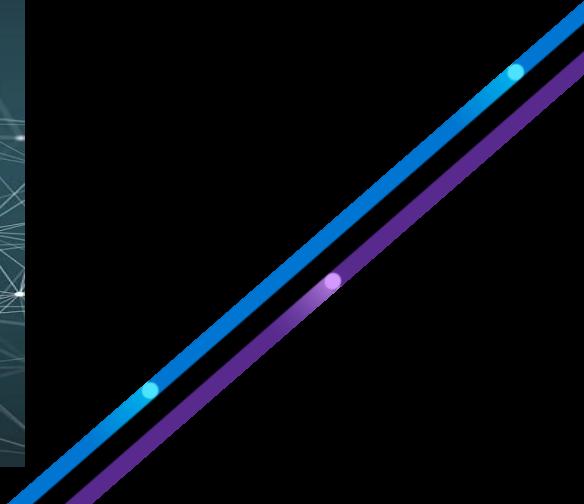
# Agenda:

- Introduction to ML
- ML Types Snapshot
- ML Lingo 101
- Microsoft's ML Playground
- Deploy & Apply
- Interactive Q&A Blitz



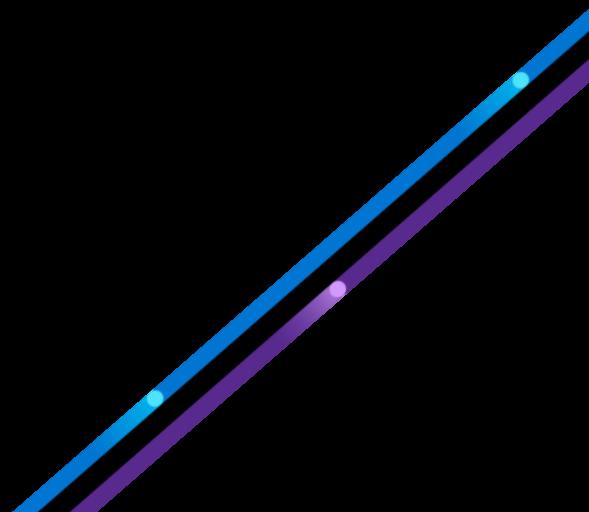
# Introduction to ML

- Machine Learning (ML) is a subset of artificial intelligence, teaching computers to learn patterns from data and make decisions without explicit programming.
- **Core Concept:** The ability of machines to learn from vast amounts of data using **ALGORITHMS**



# The Core Engine

- Algorithms
- Data
- Learning Process



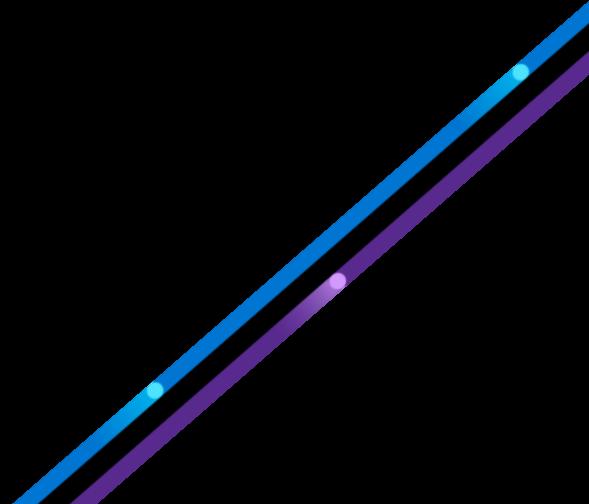
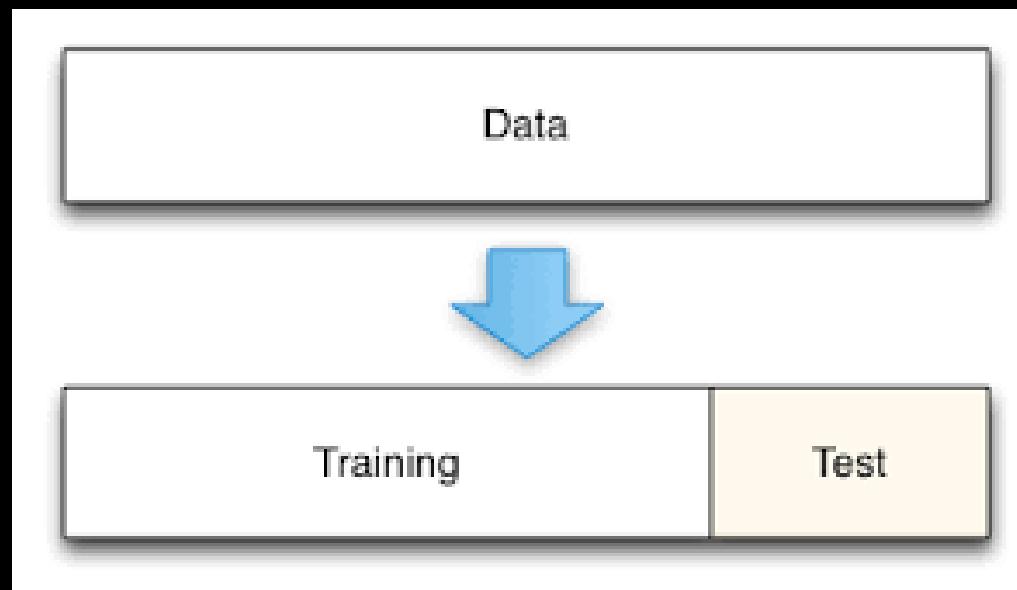
# Algorithms

- Algorithms are step-by-step procedures or sets of rules designed to perform specific tasks or solve particular problems
- They learn from data, uncovering patterns and relationships invisible to humans.



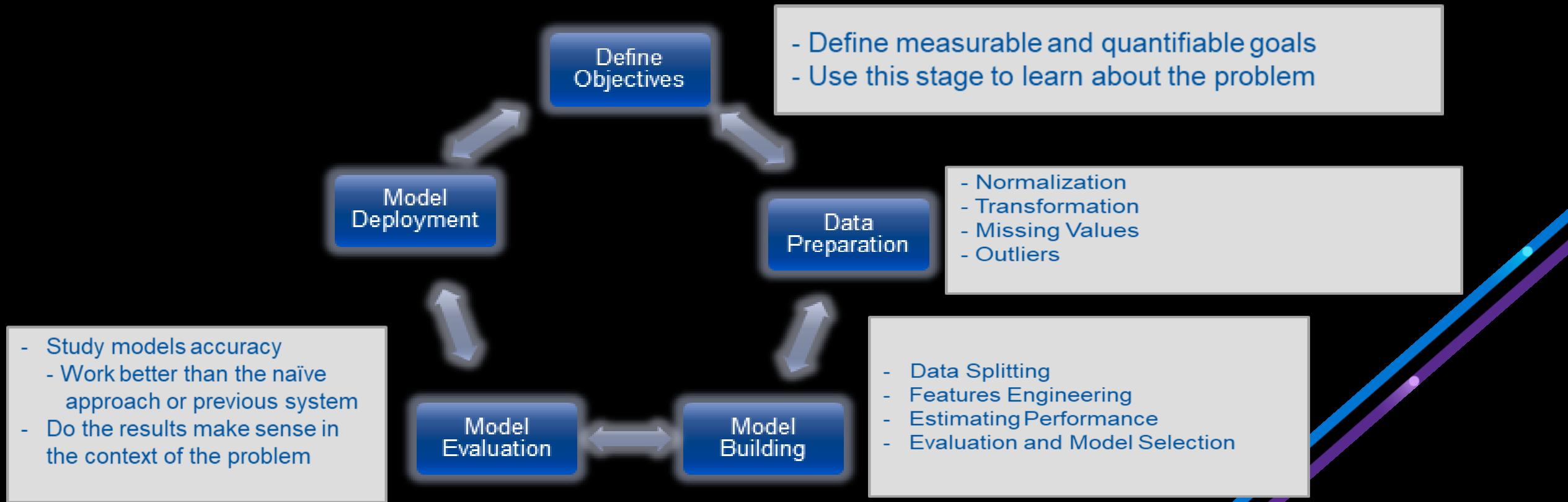
# DATA

- The fuel that powers ML.
- The more diverse and relevant the data, the better the model learns and performs.
- The data is divided into TWO parts ( Training , Test )



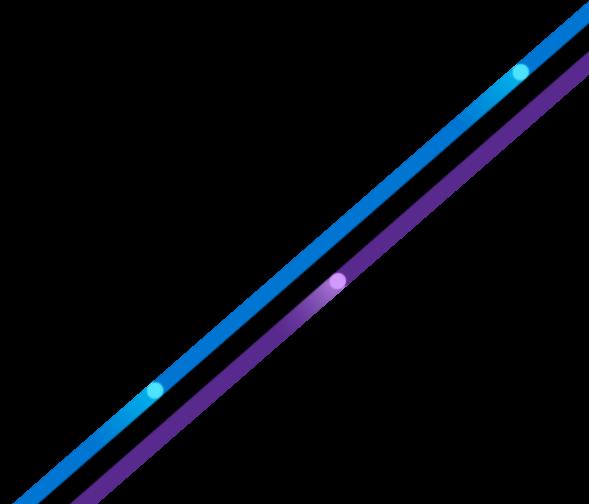
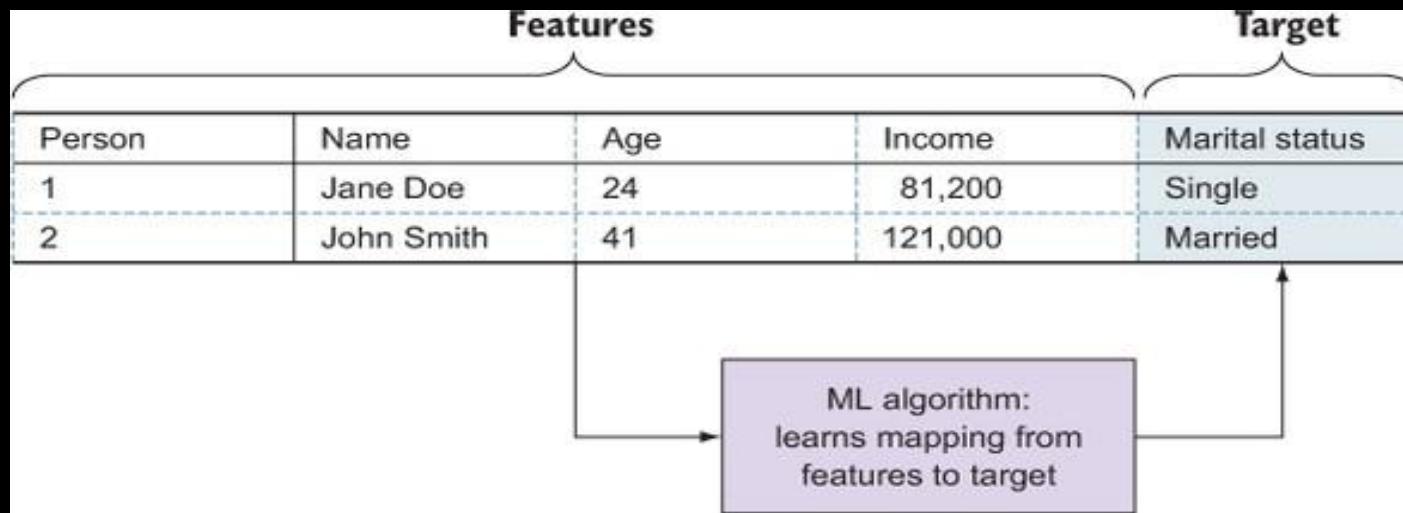
# Learning Process

- This is where the magic unfolds.
- Different algorithms utilize various techniques to learn from data and improve their predictions or insights.

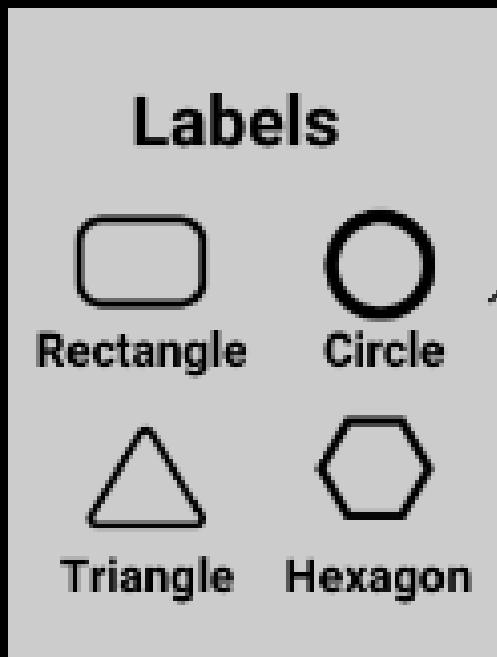


# Essential Terms in a Nutshell

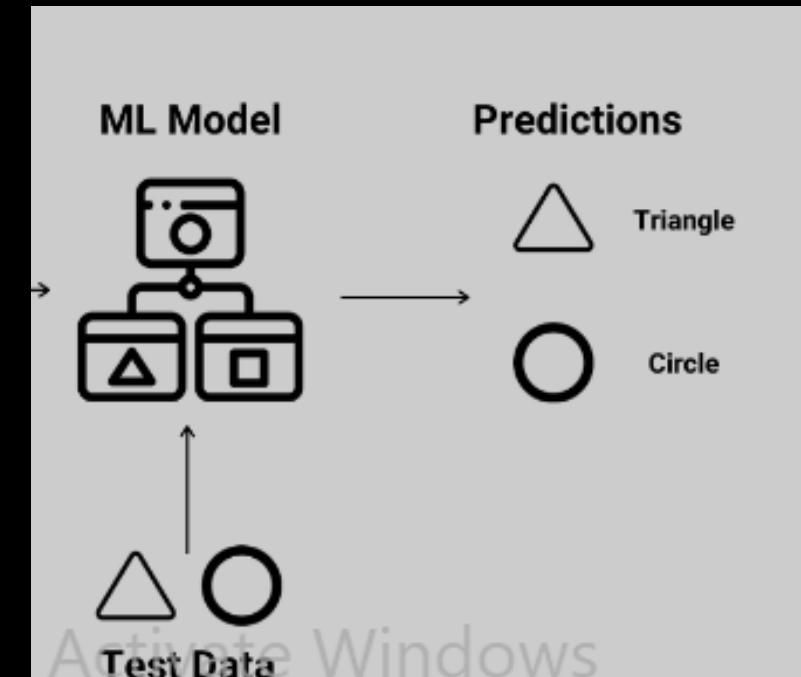
- **Feature:** An input variable used by a machine learning model to make predictions. Features can be numerical or categorical.



**Label:** The output or the target variable that the machine learning model aims to predict.



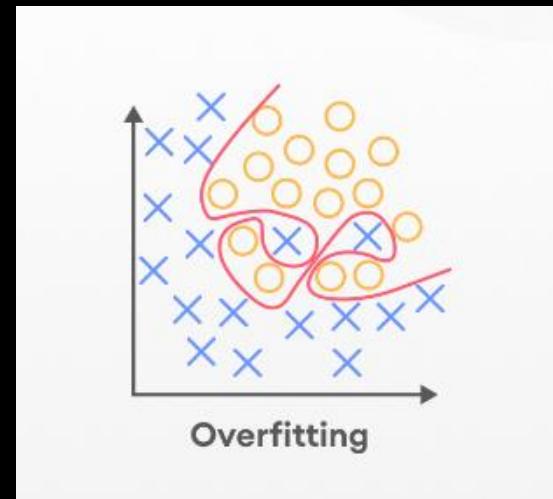
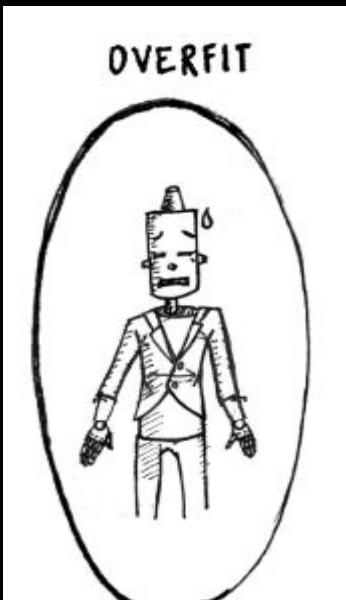
**Model:** The trained version of the algorithm, ready to make predictions



# OVERFITTING VS UNDERFITTING

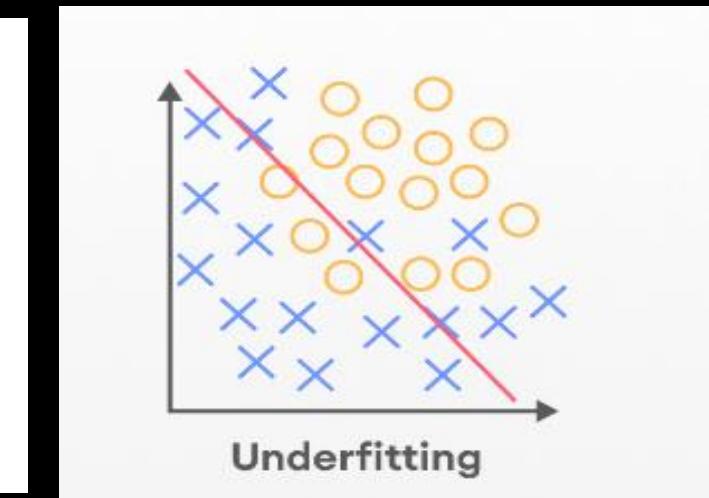
**Overfitting:** When your model memorizes the training data capturing noise and irrelevant patterns but fails on new data.

Complex models used for simple data.



**Underfitting:** when model is too simple and fails to capture the underlying patterns in the training data.

Simple models used for complex data.



**Bias:** Think of it as your model having a "favorite answer" due to unfairness in the data or its design.

**Accuracy:** How often your model makes the correct prediction, regardless of which type of error it makes. predicts both positive and negative instances

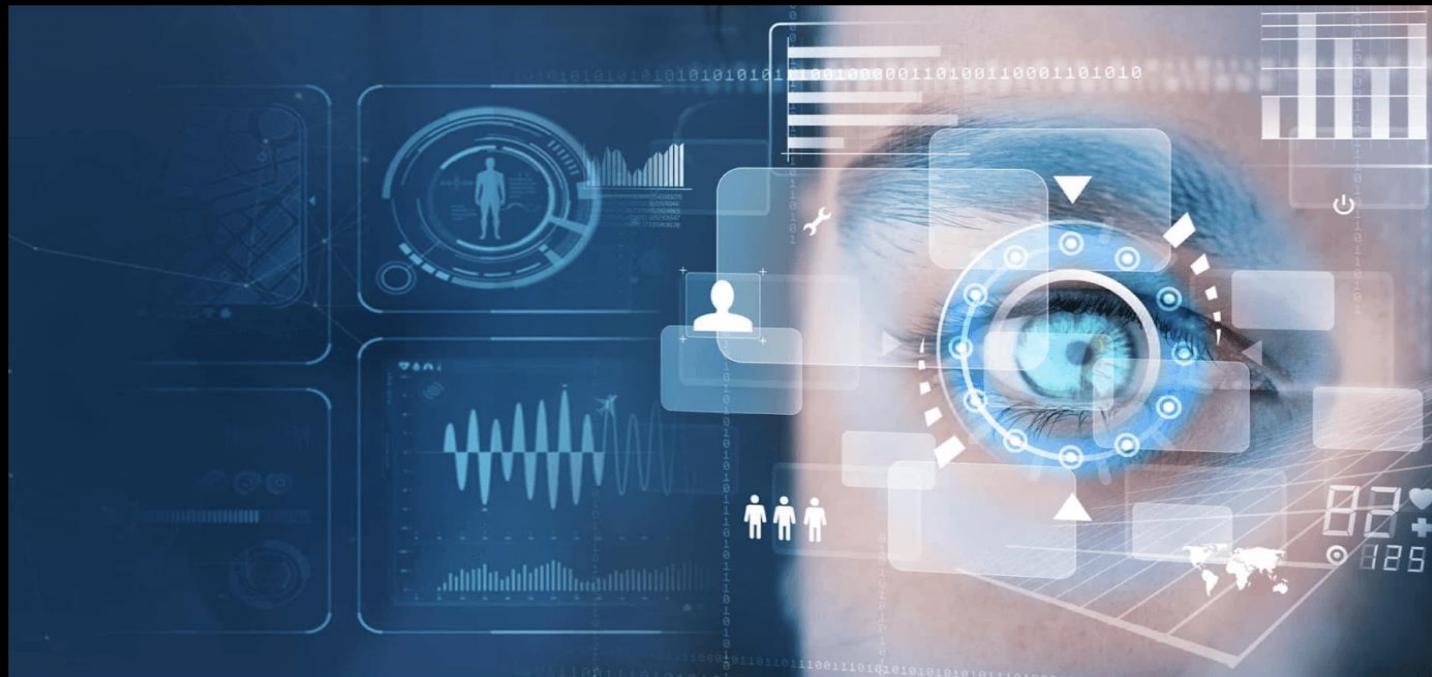
**Recall:** How good your model is at catching all the true positives (finding true cats and not missing any).

**Precision:** How accurate your model is when it says something is positive (only saying "cat" for actual cats)

# Unveiling the Tricks

- Pattern Recognition:

ML excels at finding complex patterns in data, similar to how we recognize faces or predict weather patterns. Imagine analyzing millions of images to identify cats - an impossible feat for humans but a breeze for ML! . ML has ability to go beyond simple tasks.



# ML Types Snapshot

1

- SUPERVISED LEARNING

2

- UNSUPERVISED LEARNING

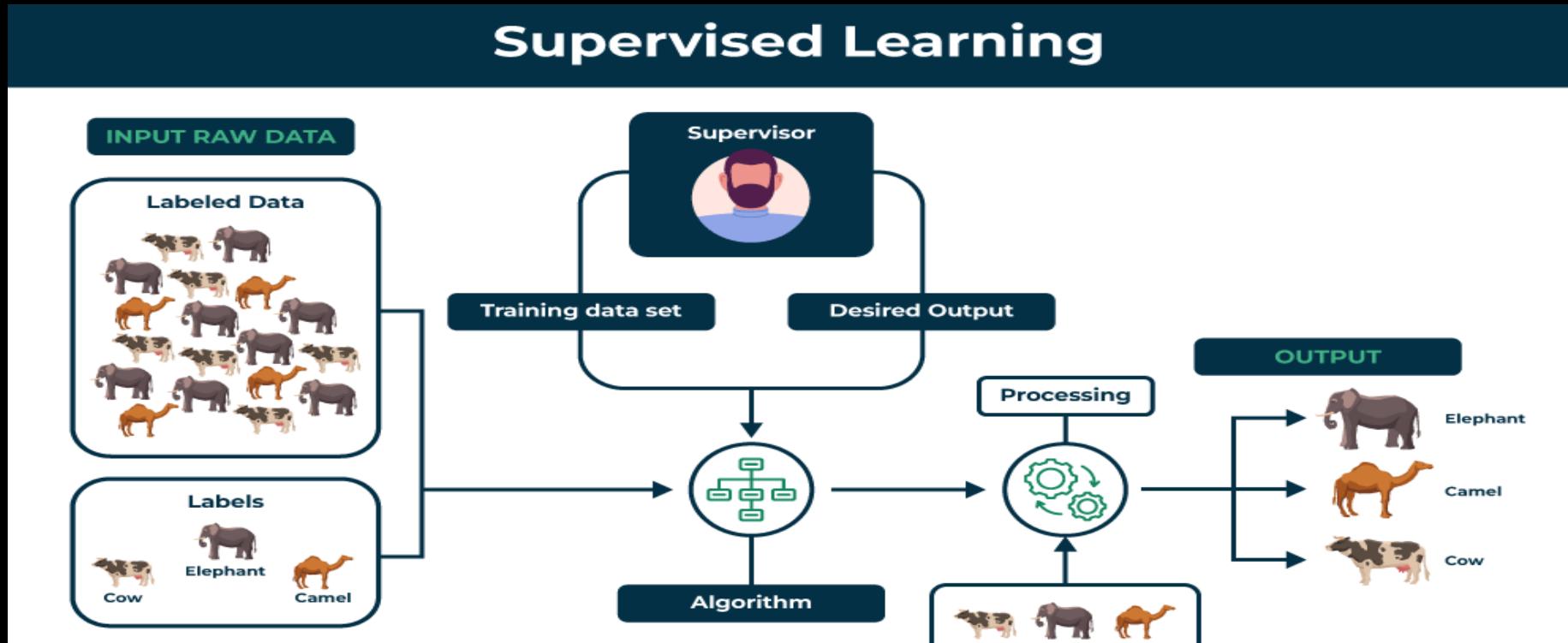
3

- RE-INFORCEMENT LEARNING



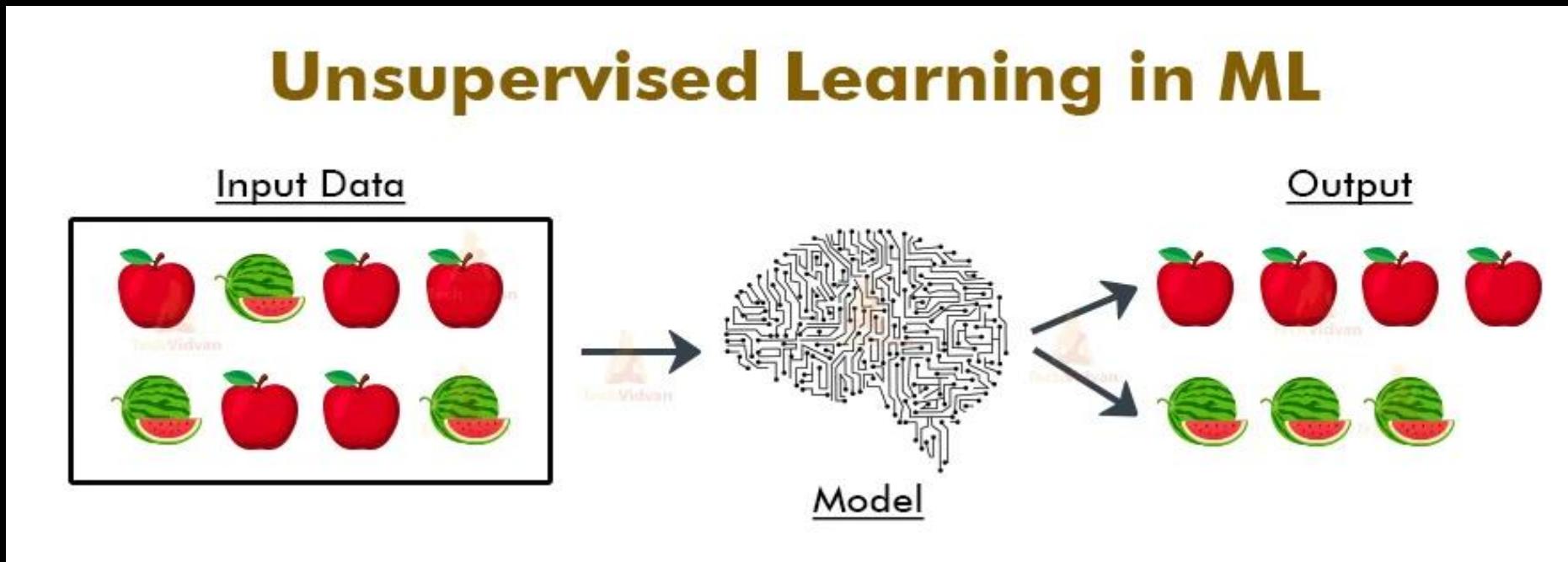
# Supervised Learning:

- **Imagine:** Having a teacher guide you.
- **Process:** Learns from labeled data (inputs with corresponding outputs).
- **Goal:** Predicts new, unseen outputs based on learned patterns.



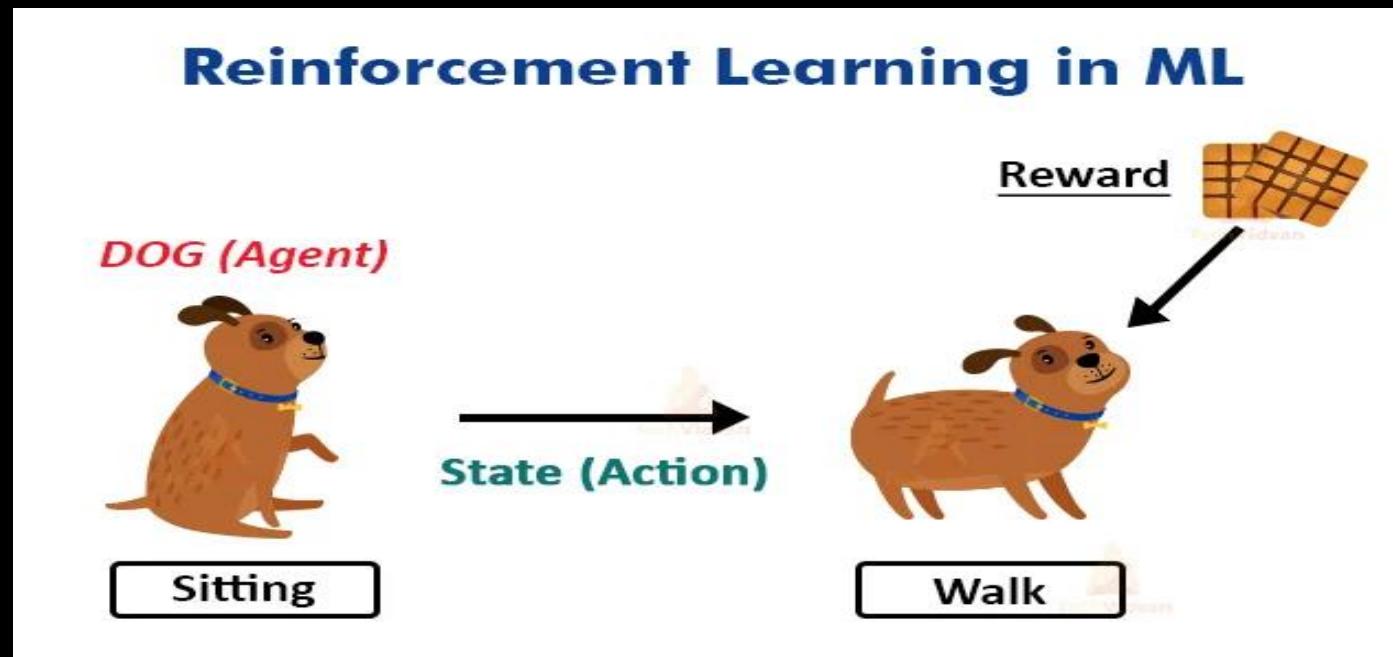
# Unsupervised Learning:

- **Imagine:** Exploring a new city on your own.
- **Process:** Discovers hidden patterns and structures in unlabeled data.
- **Goal:** Uncovers insights and trends without predetermined outputs.



# Reinforcement Learning:

- **Imagine:** Learning through trial and error in a game.
- **Process:** Learns through interactions with an environment, receiving rewards for desired actions.
- **Goal:** Maximizes long-term rewards by adapting its behavior.



# Algorithms

## Supervised

- Linear Regression
- Decision Trees
- Random Forest etc

## Unsupervised

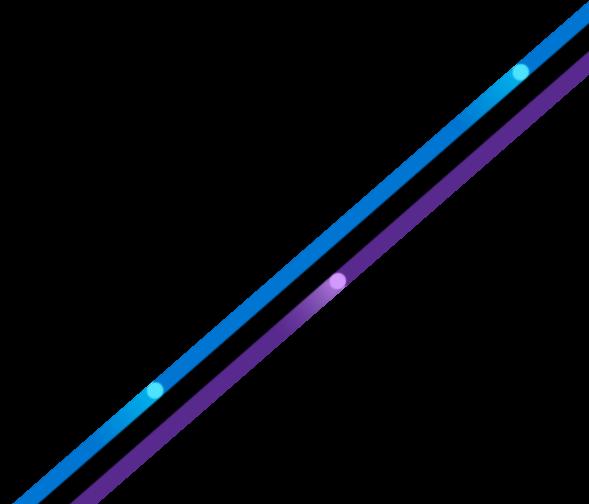
- K-Means Clustering
- Hierarchical Clustering
- Apriori (Association Rule Learning) etc

## Reinforcement

- Q-Learning
- Deep Q Networks (DQN)
- Policy Gradient Methods etc

# Steps of model developing

- Define the Problem
- Collect and Prepare Data
- Select a Model
- Split the Data
- Train the Model
- Validate and Tune
- Test the Model
- Deploy the Model



# Python code for model development

```
# Import necessary libraries
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score

# Load the Iris dataset
from sklearn.datasets import load_iris
iris = load_iris()
X, y = iris.data, iris.target

# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2)

# Choose a machine learning algorithm (Decision Tree in this case)
model = DecisionTreeClassifier()

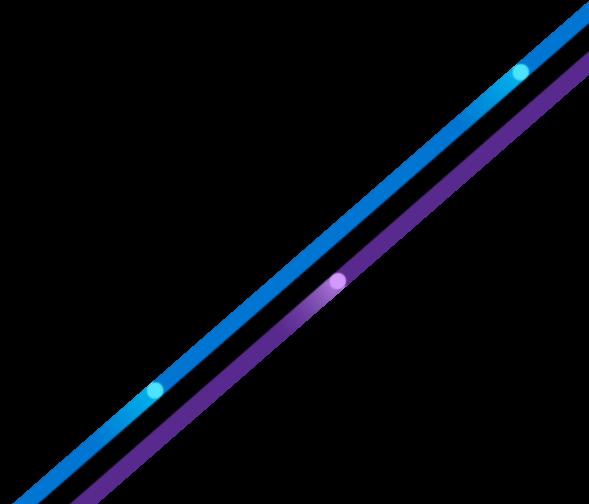
# Train the model on the training data
model.fit(X_train, y_train)

# Make predictions on the testing set
predictions = model.predict(X_test)

# Evaluate the model's accuracy
accuracy = accuracy_score(y_test, predictions)
print(f'Model Accuracy: {accuracy}')
```

# AI vs. ML

- **Misconception:** Artificial Intelligence and Machine Learning are the same.
- **Clarification:** AI is the broader concept, while ML is a subset of AI that focuses specifically on learning from data.
- We can get AI without using machine learning, but this would require building millions of lines of codes with complex rules and decision-trees.



# AI VS ML VS DL

**Artificial Intelligence**

**Machine Learning**

**Deep Learning**

The subset of machine learning composed of algorithms that permit software to train itself to perform tasks, like speech and image recognition, by exposing multilayered neural networks to vast amounts of data.

A subset of AI that includes abstruse statistical techniques that enable machines to improve at tasks with experience. The category includes deep learning

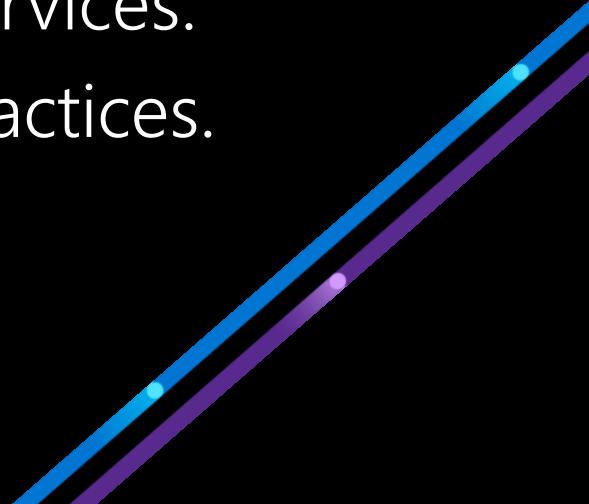
Any technique that enables computers to mimic human intelligence, using logic, if-then rules, decision trees, and machine learning (including deep learning)

# Azure Machine Learning:

- Cloud-based platform for building, training, deploying, and managing machine learning models at scale.

Exploring data and preparing it for modeling.

- Training and evaluating machine learning models.
- Registering and managing trained models.
- Deploying trained models for use by applications and services.
- Reviewing and applying responsible AI principles and practices.



# *Azure Machine Learning workspace*

- The primary resource required for Azure Machine Learning is an *Azure Machine Learning workspace*
- To create an Azure Machine Learning workspace, you can use the Azure portal



Machine Learning  
Workspace

Azure Machine Learning - Micro ... +

https://portal.azure.com/#create/Microsoft.MachineLearningServices

Microsoft Azure Search resources, services, and docs (G+/)

Home > Create a resource > Marketplace >

## Azure Machine Learning

Create a machine learning workspace

Resource details

Every workspace must be assigned to an Azure subscription, which is where billing happens. You use resource groups like folders to organize and manage resources, including the workspace you're about to create.

[Learn more about Azure resource groups](#)

Subscription \* (Visual Studio Enterprise)

Resource group \* (New) aml-resources  
[Create new](#)

Workspace details

Configure your basic workspace settings like its storage connection, authentication, container, and more. [Learn more](#)

Name \* aml-workspace

Region \* East US

Storage account \* (new) amlworkspace9230665149  
[Create new](#)

Key vault \* (new) amlworkspace3155955552  
[Create new](#)

Application insights \* (new) amlworkspace7380896529  
[Create new](#)

Container registry \* None

[Review + create](#)

< Previous

Next : Networking

# Azure Machine Learning studio

- *Azure Machine Learning studio* is a browser-based service that provides no-code and code-first solutions to visually create, train, and manage models through a web UI
- Azure Machine Learning studio allows the *Python SDK* to seamlessly integrate with the natively supported *Jupyter Notebooks* for collaborative notes and coding.
- Data within Azure Machine Learning studio is simple to manage with intuitive **data visualization** and **AI-assisted image** or text labeling features.

The screenshot below shows the **Metrics** page for a trained model in Azure Machine Learning studio, in which you can see the evaluation metrics for a trained multiclass classification model.

The screenshot displays the Azure Machine Learning Studio Metrics page for a completed job named "penguin-classifier". The top navigation bar shows the URL as https://ml.azure.com/runs/mslearn-automl\_0?wsid=/subscriptions/fd3e3786-c21... and the workspace as "Visual Studio Enterprise aml-workspace". The left sidebar menu is visible, showing sections like "All workspaces", "Home", "Model catalog (PREVIEW)", "Authoring" (Notebooks, Automated ML, Designer, Prompt flow PREVIEW), "Assets" (Data, Jobs, Components, Pipelines, Environments, Models, Endpoints), and "Manage" (Compute). The main content area is titled "penguin-classifier" and shows the "Metrics" tab selected. Below the tabs are buttons for Refresh, Cancel, Create custom chart, View as..., Current view: Local, and Edit view. A "Select metrics" button is also present. Four evaluation metrics are displayed in cards: accuracy (0.9710924), AUC\_weighted (0.9990379), average\_precision\_score (0.9979805), and f1\_score\_macro (0.9675935). Below these cards is a "confusion\_matrix" section with a "Raw" dropdown set to "Confusion Matrix". The confusion matrix table shows the following data:

		Predicted Label		
		0	1	2
True Label	0	147	1	4
	1	1	123	0
2	4	0	64	

# Create an Azure Machine Learning workspace

- To set up Azure Machine Learning service, **access Azure, sign in, create a resource group, and establish a workspace.** The workspace triggers the creation of essential resources like **storage, key vault, application insights**, and container registry for effective management and monitoring.
- You can create an Azure Machine Learning workspace in
  - user interface in the **Azure portal**
  - **Azure Resource Manager (ARM) template**
  - **Azure Command Line Interface (CLI)**
  - **Azure Machine Learning Python SDK.**

# Python SDK

following code uses the Python SDK to create a workspace named mlw-example

Python

```
from azure.ai.ml.entities import Workspace

workspace_name = "mlw-example"

ws_basic = Workspace(
    name=workspace_name,
    location="eastus",
    display_name="Basic workspace-example",
    description="This example shows how to create a basic workspace",
)
ml_client.workspaces.begin_create(ws_basic)
```

# Explore the workspace in the Azure portal

- Creating an Azure Machine Learning workspace will typically take between 5-10 minutes to complete.

mlw-dp100-labs ★ ...

Search < Download config.json Delete

Overview Activity log Access control (IAM) Tags Diagnose and solve problems Events

Resource group rg-dp100-labs Location East US Subscription Demo-Subscription Subscription ID

Studio web URL <https://ml.azure.com/>

Container Registry Key Vault Application Insights

MLflow tracking URI <azureml://eastus.api.azureml.ms/mlflow/v1.0/subscri...>

Storage

Networking Properties Locks

Alerts Metrics Diagnostic settings Logs

Tasks (preview) Export template

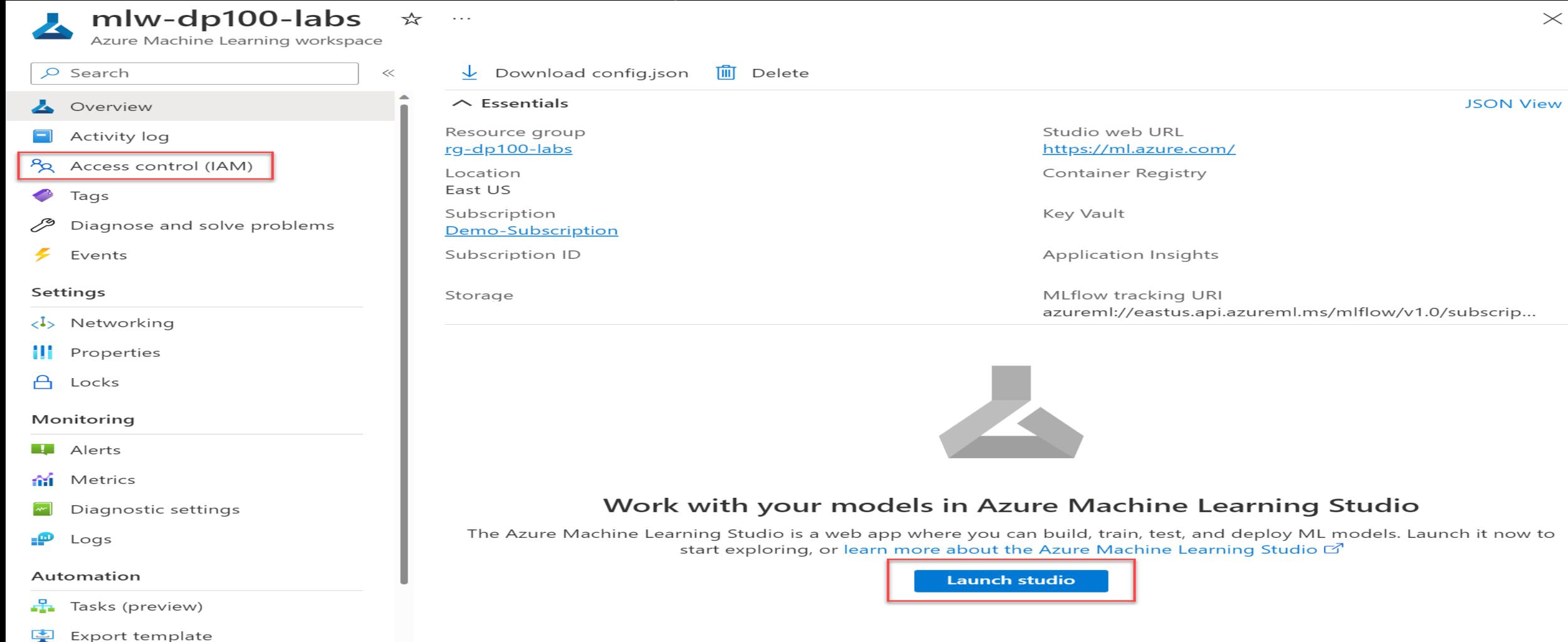
JSON View

Essentials

Work with your models in Azure Machine Learning Studio

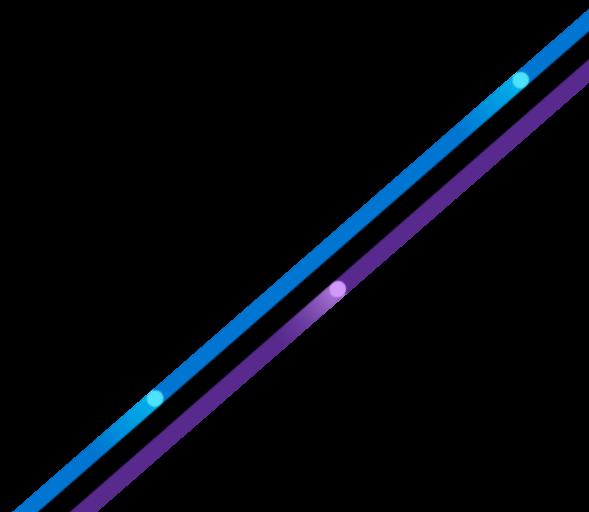
The Azure Machine Learning Studio is a web app where you can build, train, test, and deploy ML models. Launch it now to start exploring, or [learn more about the Azure Machine Learning Studio](#)

Launch studio



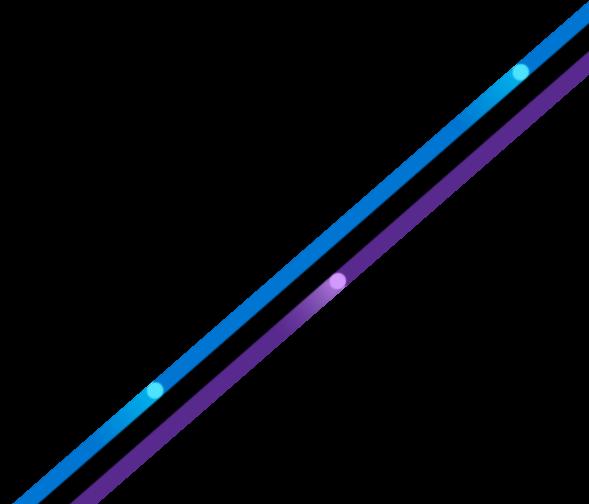
# Overview of AZURE ML WORKSPACE

- From the **Overview** page of the **Azure Machine Learning workspace** in the Azure portal, you can **launch** the Azure Machine Learning studio. The Azure Machine Learning studio is a web portal and provides an easy-to-use interface to create, manage, and use resources and assets in the workspace.
- From the Azure portal, you can also give others access to the Azure Machine Learning workspace, using the **Access control**.



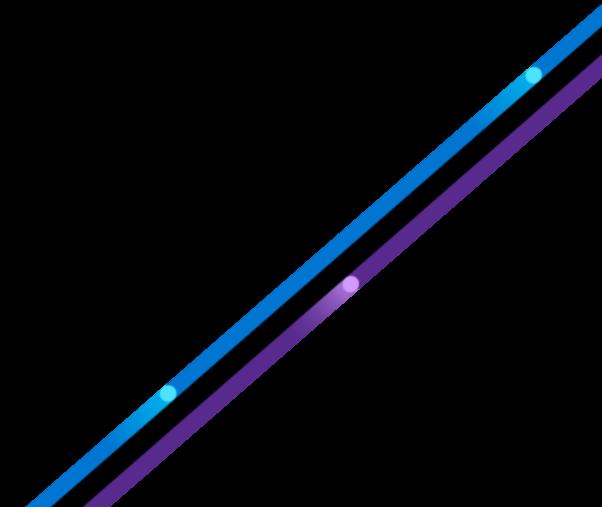
# Give access to the Azure Machine Learning workspace

- Control access to Azure Machine Learning using role-based access in the [Access control tab](#). Set permissions for specific tasks, like letting Azure administrators create resources or [data scientists run and register models](#).
- Use roles like [Owner](#), [Contributor](#), and [Reader](#), or specific roles like [AzureML Data Scientist](#) and [AzureML Compute Operator](#).



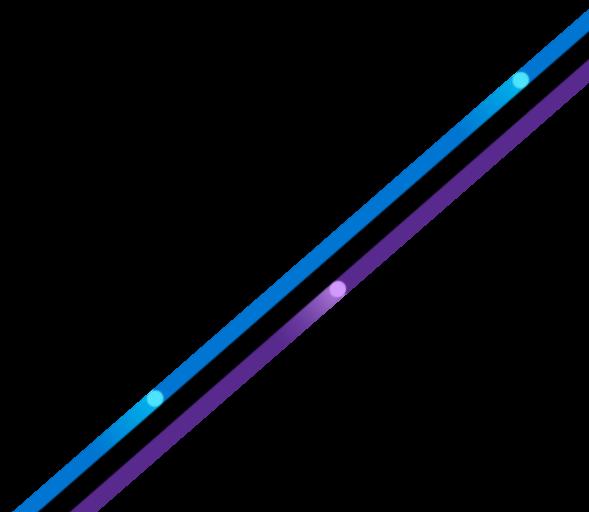
# Create and manage the workspace

- The **workspace** is the resource for Azure Machine Learning. Data scientists need access to the workspace to train and track models, and to deploy the models to endpoints.
- be careful with who has *full* access to the workspace.
- you can find all logs, metrics, outputs, models, and snapshots of your code in the workspace.



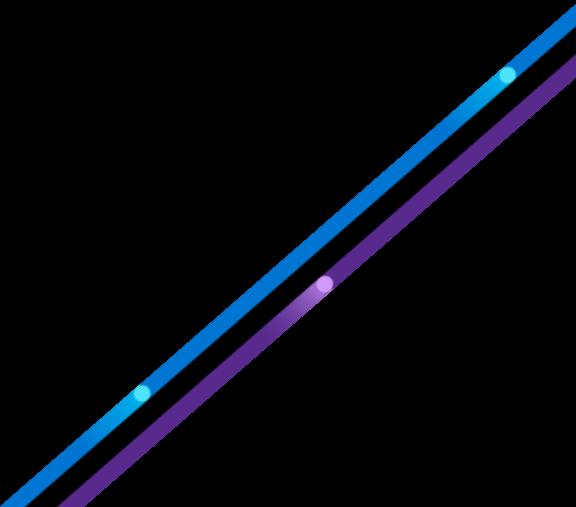
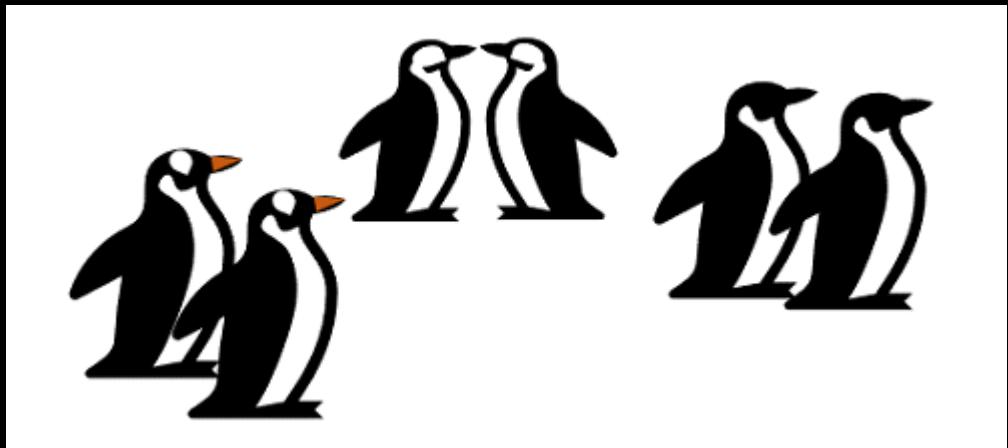
# clustering model azure machine learning

- You can use Microsoft Azure Machine Learning designer to create clustering models by using a drag and drop visual interface, without needing to write any code



# Clustering

- *Clustering* is a form of machine learning that is used to group similar items into clusters based on their features
- Clustering is an example of *unsupervised* machine learning, in which you train a model to separate items into clusters based purely on their characteristics, or *features*. There is no previously known cluster value (or *label*) from which to train the model.



# Steps

- ❖ To use Azure Machine Learning, you first create a *workspace* resource in your Azure subscription
- ❖ After you have created an Azure Machine Learning workspace, you can develop solutions with the Azure Machine Learning studio web portal
- ❖ To begin using the web portal, you need to assign the workspace you created in the Azure portal to Azure Machine Learning studio

≡  
← All workspaces

Home

Model catalog PREVIEW

#### Authoring

- Notebooks
- Automated ML
- Designer

#### Assets

- Data
- Jobs
- Components
- Pipelines
- Environments
- Models
- Endpoints

#### Manage

- Compute
- Linked Services
- Data Labeling



## Shortcuts ⋮



### Create notebook

Use notebooks for interactive cloud development.

[Create new notebook](#)



### Add compute

A designated resource for running your training script, notebook, or hosting your service deployment.

[Add compute](#)



### Connect data

Connect data from datastores, local files, public URLs, or Open Datasets assets.

[Add data](#)



### Train a model

Submit a command to your model using

[Create job](#)



## Recently viewed ⋮



### Resource type

### Name



### Status

### Quick actions

There are no recently viewed items to display

## Compute instances ⋮



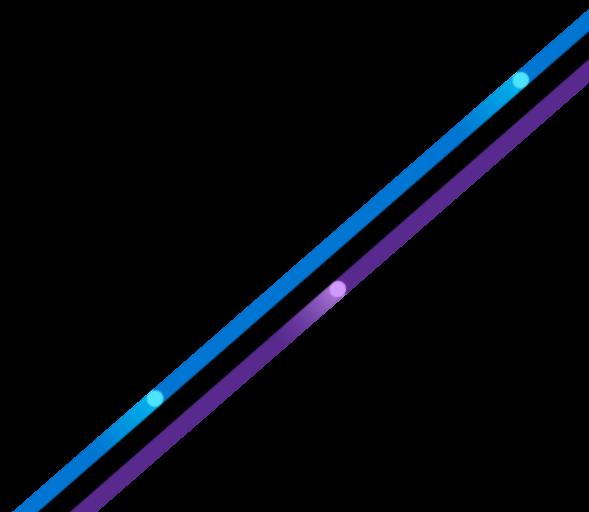
### Add compute instance

# Azure Machine Learning compute

- **Compute targets** are cloud-based resources on which you can run model training and data exploration processes.
- There are **four** kinds of compute resource you can create:
- **Compute Instances**: Development workstations that data scientists can use to work with data and models.
- **Compute Clusters**: Scalable clusters of virtual machines for on-demand processing of experiment code.
- **Kubernetes Clusters**: Deployment targets for predictive services that use your trained models. You can access previous versions of "inference clusters" here.
- **Attached Compute**: Links to existing Azure compute resources, such as Virtual Machines or Azure Databricks clusters.

# Azure Machine Learning designer

- In Azure Machine Learning studio, there are several ways to author clustering machine learning models.
- One way is to use a visual interface called *designer* that you can use to train, test, and deploy machine learning models
- The drag-and-drop interface makes use of clearly defined inputs and outputs that can be shared, reused,



Each *designer* project, known as a pipeline, has a left panel for navigation and a *canvas* on your right hand side. To use *designer*, identify the building blocks, or components, needed for your model, place and connect them on your canvas, and run a machine learning job.

The screenshot shows the Azure Machine Learning Designer interface. The left sidebar contains a navigation menu with the following sections and items:

- Author**:
  - New
  - Home
- Designer**:
  - Assets
    - Data
    - Jobs
    - Components
    - Pipelines
    - Environments
    - Models
    - Endpoints
  - Manage
    - Compute
    - Datastores
    - Linked Services
    - Data Labeling

The right side of the interface features a search bar at the top, followed by filter options for "Registry name : All", "Tags : All", and "Add filter". Below these are two tabs: "Data" and "Component", with "Component" currently selected. A list of "86 assets" is displayed, showing the following components:

- Add Columns** (Microsoft)  
Adds a set of columns from one dataset to another.  
[Learn More](https://aka.ms/aml/add-columns)  
azureml.Designer:true | azureml.prebuildC5/11/2022
- Add Rows** (Microsoft)  
Appends a set of rows from an input dataset to the end of another dataset.  
[Learn More](https://aka.ms/aml/add-rows)  
azureml.Designer:true | azureml.prebuildC5/11/2022
- Apply Image Transformation** (Microsoft)  
Applies a image transformation to a image directory.  
[Learn More](https://aka.ms/aml/apply-image-transformation)  
azureml.Designer:true | azureml.prebuildC5/11/2022
- Apply Math Operation** (Microsoft)  
Applies a mathematical operation to column values.  
[Learn More](https://aka.ms/aml/apply-math-operation)  
azureml.Designer:true | azureml.prebuildC5/11/2022

At the bottom right, there are "Navigator" and "Hand" icons.

# Pipelines

Pipelines let you organize, manage, and reuse complex machine learning workflows across projects and users.

A **pipeline** starts with the dataset from which you want to train the model.

Each time you run a pipeline, the configuration of the pipeline and its results are stored in your workspace as a **pipeline job**

The screenshot shows the Microsoft Pipelines interface. At the top left is the Microsoft logo with a back arrow. The top right features the word "Pipelines". Below the header are three navigation tabs: "Pipeline jobs" (underlined), "Pipeline endpoints", and "Pipeline drafts". A horizontal toolbar includes "New pipeline" (plus icon), "Refresh" (refresh icon), "Edit columns" (grid icon), and "Reset" (refresh icon). A search bar with a magnifying glass icon is positioned below the toolbar. The main area displays the message "Showing 1-8 of 8 jobs". A table header row includes "Display name" and "Experiment" (with a star icon). The left sidebar contains a vertical list of items: "New", "Home", "Author", "Notebooks", "Automated ML", "Designer", "Assets", "Data", "Jobs", "Components", "Pipelines" (which is highlighted with a red box), "Environments", "Models", and "Endpoints".

# Components

Azure Machine Learning **component** acts as a building block for Azure Machine Learning pipelines. In a pipeline project, you can access data assets and components from the left panel's **Asset Library** tab.

The screenshot shows the Azure Machine Learning Studio interface. On the left, a vertical navigation bar lists several options: New, Home, Author, Notebooks, Automated ML, Designer (which is highlighted with a gray background), Assets, Data, Jobs, Components (which is also highlighted with a gray background), and Pipelines. To the right of the navigation bar, there is a search bar with the placeholder "Search by name, tags and descrip...". Below the search bar are two filters: "Registry name : All" and "Tags : All". The main area is divided into two tabs: "Data" (which is active) and "Components" (which is also highlighted). Under the "Data" tab, there is a summary section showing "86" items, a refresh icon, and a plus sign. Below this, a list of components is displayed, each with a thumbnail, name, author, and a brief description. The first component listed is "Add Columns" by Microsoft, which adds a set of columns from one dataset to another. The second component listed is "Add Rows" by Microsoft, which appends a set of rows from an input dataset.

Thumbnail	Name	Author	Description
	Add Columns	Microsoft	Adds a set of columns from one dataset to another [Learn More](https://aka.ms/aml/add-column)
	Add Rows	Microsoft	Appends a set of rows from an input dataset

# Datasets

You can create data assets on the **Data** page from local files, a datastore, web files, and Open Datasets. These data assets will appear along with standard sample datasets in *designer's Asset Library*

The screenshot shows the Microsoft Data page interface. On the left, there is a navigation sidebar with the following items:

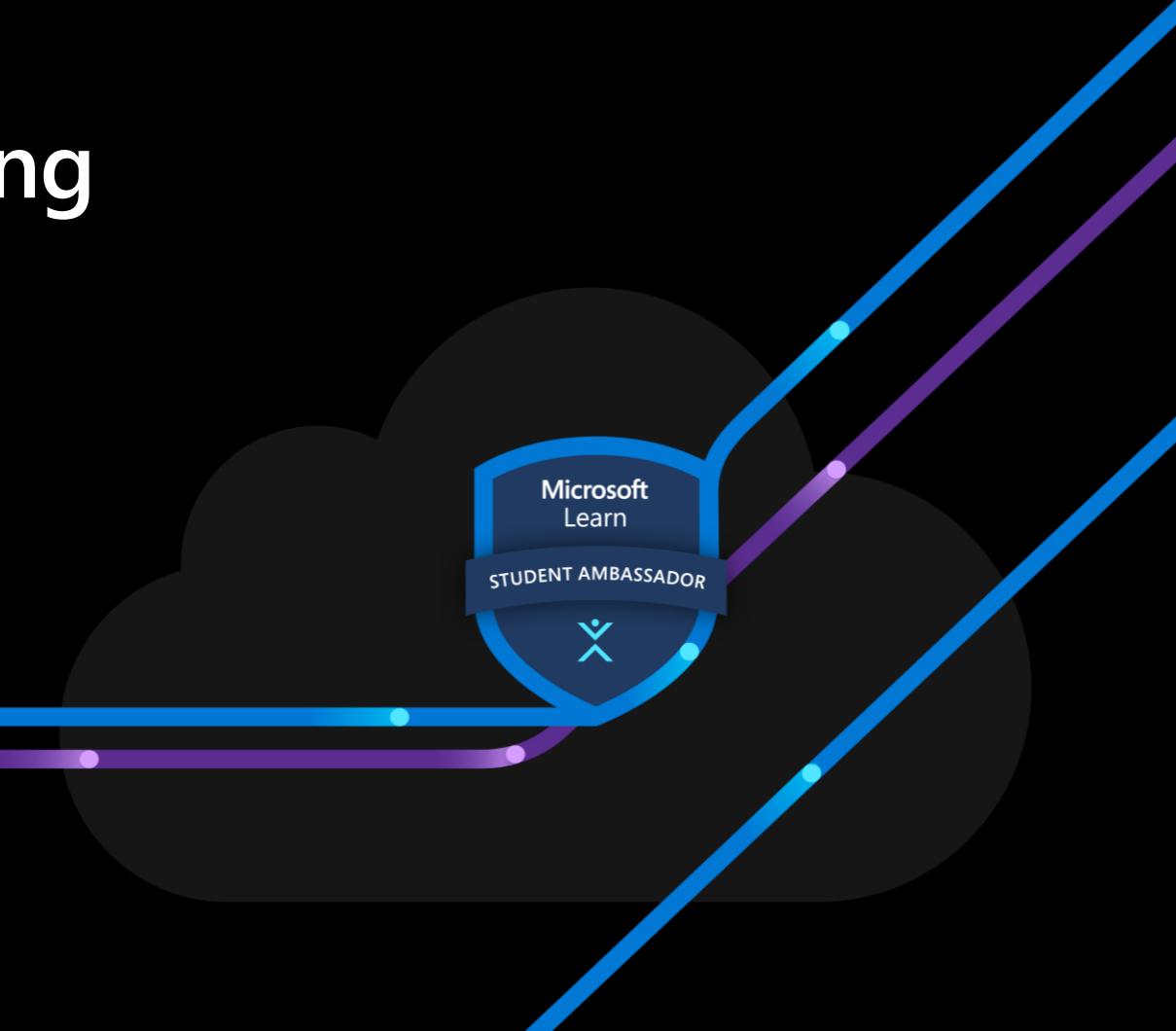
- Microsoft
- + New
- Home
- Author
- Notebooks
- Automated ML
- Designer
- Assets
- Data** (this item is highlighted with a red box)
- Jobs
- Components
- Pipelines

The main content area is titled "Data" and displays the "Registered data assets" section. It includes a toolbar with "Create", "Refresh", "Unregister", "Edit columns", and "Reset view" buttons. A dropdown menu is open under the "Create" button, listing four options:

- From local files
- From datastore
- From web files
- From Open Datasets

Below the toolbar, there are two columns: "Version" and "Data source".

# Understand steps for clustering



# steps

Prepare data:

Train model:

Evaluate performance:

Deploy a predictive service:



# Prepare data

- 1) To train a clustering model, you need a dataset that includes multiple observations of the items you want to cluster
- 2) Azure Machine Learning **designer** has several pre-built components that can be used to prepare data for training. These components enable you to clean data, normalize features, join tables, and more

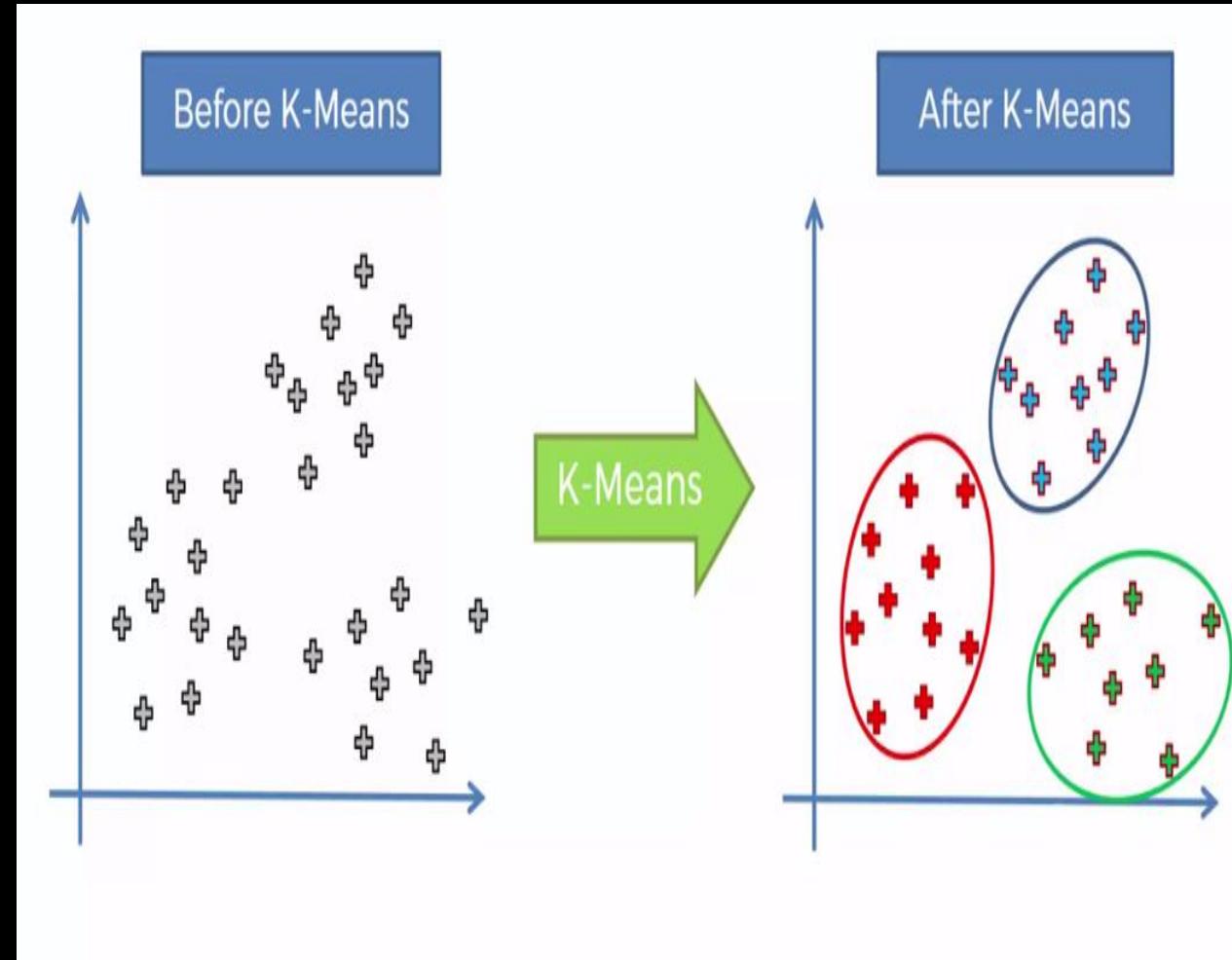
The screenshot shows the Azure Machine Learning Designer interface. On the left, a sidebar menu includes options like New, Home, Author, Notebooks, Automated ML, Designer (which is selected and highlighted in grey), Assets, Data, and Jobs. The main area displays a search bar with placeholder text 'Search by name, tags and description'. Below the search bar are filters for 'Registry name : All', 'Tags : All', and a 'Add filter' button. A navigation bar at the top right includes tabs for 'Data' (selected) and 'Component'. The main content area shows a list of assets with 86 assets found. One asset is highlighted: 'Clean Missing Data' by Microsoft, which specifies how to handle missing values in a dataset. The asset details show the identifier 'azureml.Designer:true' and the date '5/11/2022'.

# Train model

To train a clustering model, you need to apply a clustering algorithm to the data, using only the features that you have selected for clustering

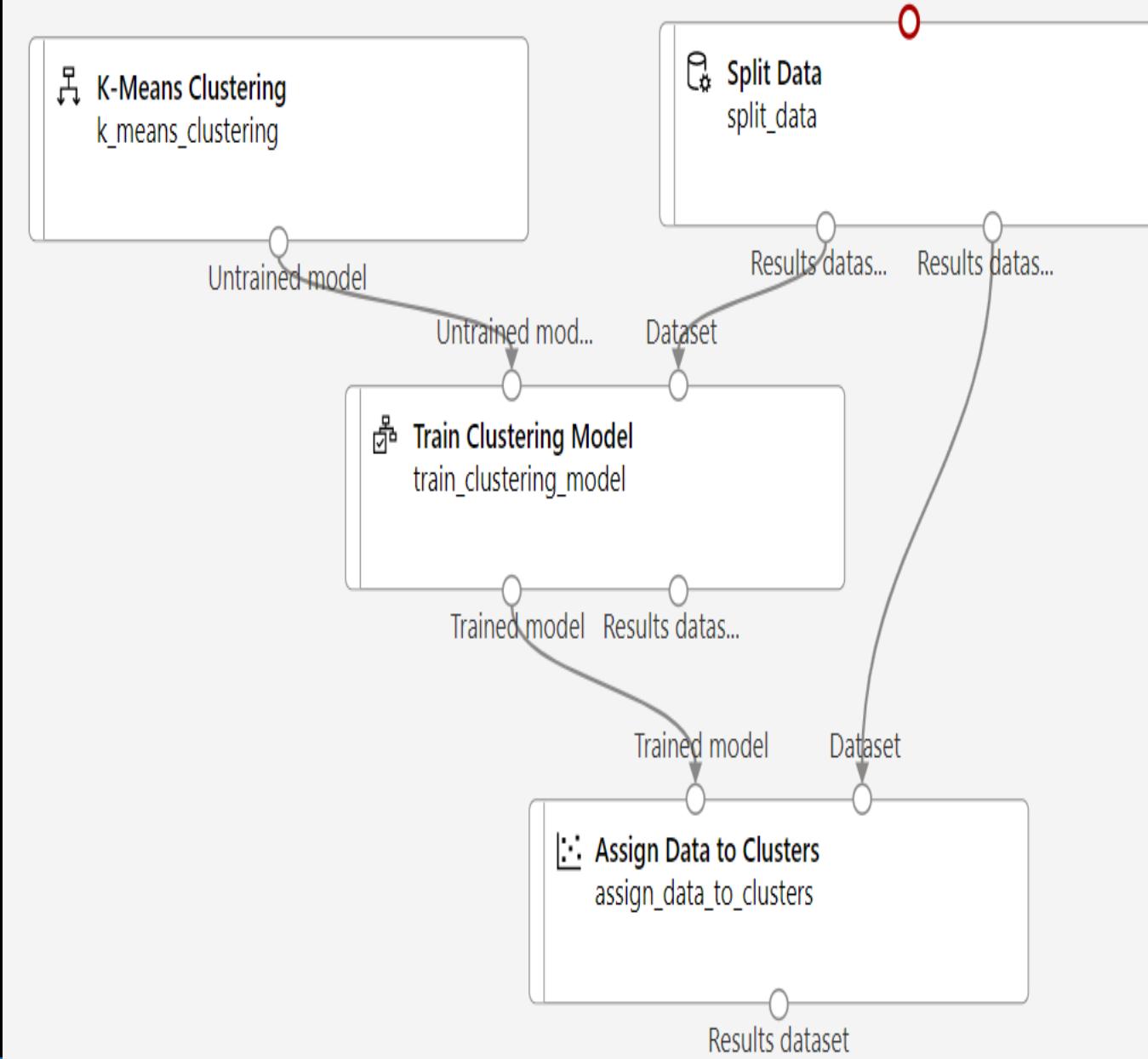
The **K-Means**

**Clustering** algorithm groups items into the number of clusters, or centroids, you specify - a value referred to as  $K$ .



# Designer's Assign Data

You will use ***designer's Assign Data*** to Clusters component to group the data into clusters. Once you connect all the components, you will want to run an experiment, which will use the data asset on the canvas to train a model.



# Evaluate performance

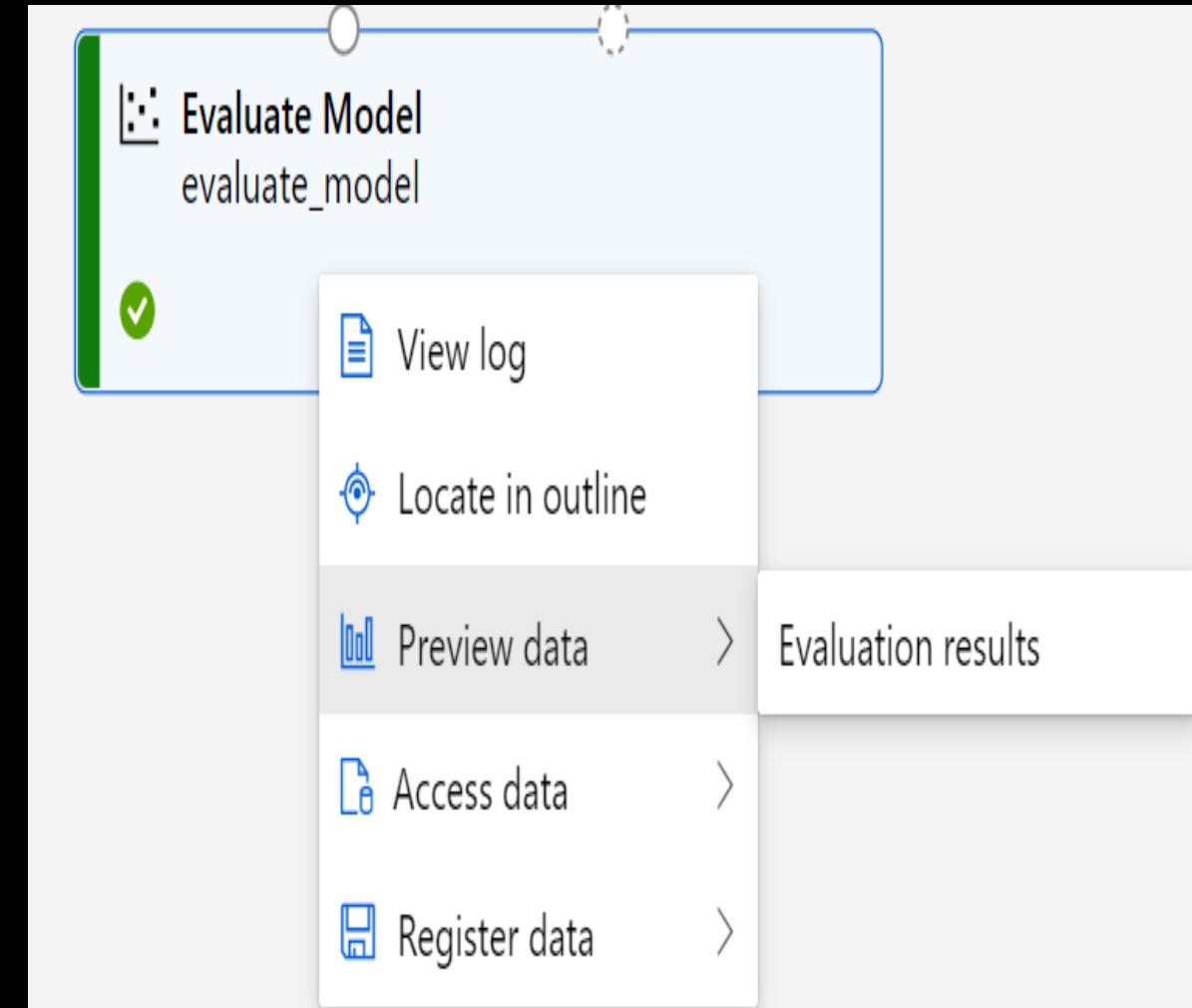
After training a model, it is important to evaluate its performance  
There are many performance metrics and methodologies for evaluating  
how well a model makes predictions



# AZURE EVALUATION

When the experiment run has finished, select **Job details**. Right click on the **Evaluate Model** module and select **Preview data**, then select **Evaluation results**.

These metrics can help data scientists assess how well the model separates the clusters.



# Deployment

In the **endpoints** page, you can view deployment details, test your pipeline service with sample data, and find credentials

It will take a while for your endpoint to be deployed.

The Deployment state on the **Details** tab will indicate *Healthy* when deployment is **successful**.

The screenshot shows the Azure Machine Learning studio interface. On the left is a navigation sidebar with icons for New, Home, Author, Notebooks, Automated ML, Designer, Assets, Data, Jobs, Components, Pipelines, Environments, Models, and Endpoints. The 'Endpoints' item is highlighted with a red box. At the top right, there are tabs: Details (which is underlined and highlighted with a red box), Test, Consume, and Deployment logs. The main content area is titled 'Attributes' and contains the following information:

Service ID	---
Description	--
Deployment state	Healthy <small>(i)</small>
Operation state	Succeeded
Compute type	
Created by	
Model ID	

# Application

## Healthcare:

*Example:* Predictive analytics for disease diagnosis, personalized treatment plans.

## Finance:

*Example:* Fraud detection, algorithmic trading, credit scoring.

## Retail:

*Example:* Customer segmentation, demand forecasting, personalized recommendations. etc



Q & A??

