LESSON 2: INTRODUCTION TO MACHINE LEARNING

MODULE 1: Lesson Overview

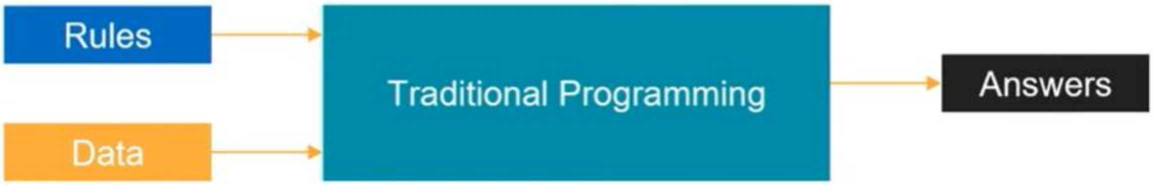
We'll cover the following modules in this lesson:

- What machine learning is and why it's so important in today's world
- The historical context of machine learning
- The data science process
- The types of data that machine learning deals with
- The two main perspectives in ML: the *statistical* perspective and the *computer science* perspective
- The essential tools needed for designing and training machine learning models
- The basics of Azure ML
- The distinction between models and algorithms
- The basics of a linear regression model
- The distinction between parametric vs. non-parametric functions
- The distinction between classical machine learning vs. deep learning
- The main approaches to machine learning
- The trade-offs that come up when making decisions about how to design and training machine learning models

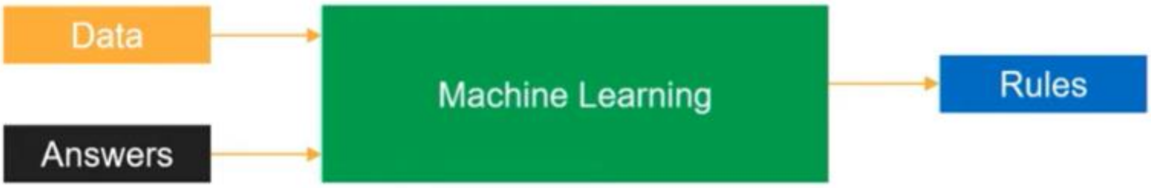
MODULE 2: What is Machine Learning?

- **Machine Learning**
  A data science technique used to extract patterns from data, allowing computers to identify related data, and forecast future outcomes, behaviours, and trends.

- **Traditional Programming Paradigm**



- **Machine Learning Paradigm**

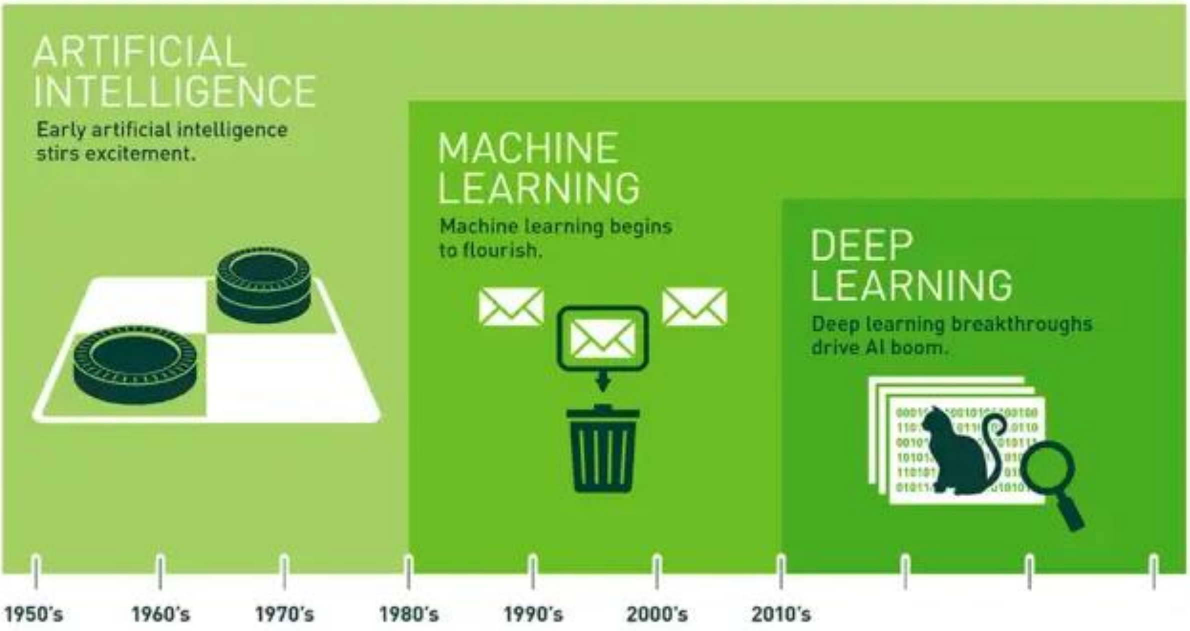


- Machine Learning uses historical data to generate rules that we have not thought of.
- Machine Learning is best suited for tasks like pattern recognition, anomaly detection, time series forecasting and recommendation systems.

MODULE 3: Applications of Machine Learning

- Machine Learning/ Deep Learning/ Reinforcement Learning
  - Natural Language Processing (NLP)
    - Text: summarization, topic detection, similarity, search
    - Speech: speech-to-text, text-to-speech, translation
  - Computer Vision (CV)
    - Self-driving cars
    - Image classification
    - Object detection
    - Object identification
    - LIDAR and Visible Spectrum
  - Analytics
    - Regression
    - Classification
    - Forecasting
    - Clustering
  - Decision Making
    - Sequence decision making problems
    - Recommenders
- Examples of Machine Learning
  - Automating the recognising the disease.
    - Google has trained a deep learning model to detect breast cancer
    - Stanford researchers have used deep learning models to diagnose skin cancer
  - Recommend next best actions for individual care plans using patient's digital health footprint.
    - EMRs (Electronic Medical Records) and EHRs (Electronic Health Records)
    - IBM Watson Oncology
  - Enabling real-time, personalized and interactive banking experience with chat bots. This allows resolving simple issues without the need of human intervention.
    - https://www.drift.com/learn/chatbot/ai-chatbots/
  - Identify next best action for the customer (ex: showing relevant deals).
    - Sentiment analysis
  - Capture, prioritise and route service requests to correct employee to improve response times (ex: feedback mails received from the customers can be forwarded to the concerned department by looking at the content of the mail.)
    - Introduction to Ticket Routing using AI https://monkeylearn.com/blog/ticket-routing/

MODULE 4: HISTORY OF MACHINE LEARNING


ARTIFICIAL INTELLIGENCE
Early artificial intelligence stirs excitement.
MACHINE LEARNING
Machine learning begins to flourish.
DEEP LEARNING
Deep learning breakthroughs drive AI boom.
1950's
1960's
1970's
1980's
1990's
2000's
2010's

Since an early flush of optimism in the 1950s, smaller subsets of artificial intelligence – first machine learning, then deep learning, a subset of machine learning – have created ever larger disruptions.

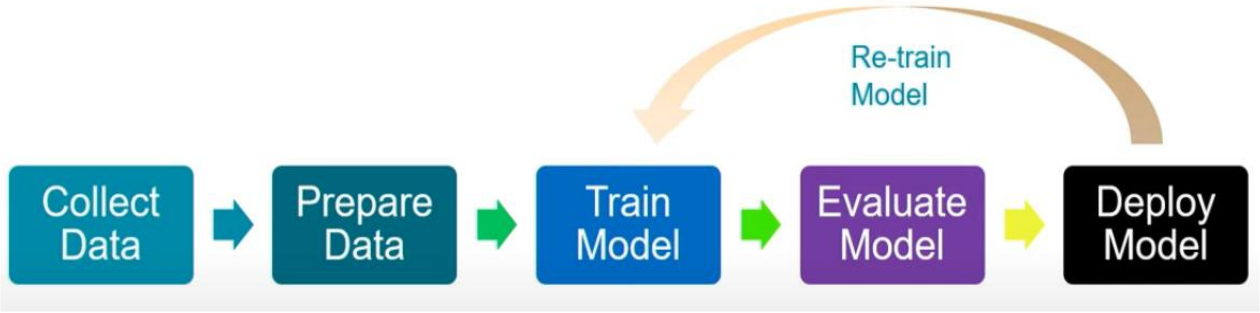
- **Artificial Intelligence**: A broad term that refers to computers thinking more like humans.
- **Machine Learning**: A subcategory of artificial intelligence that involves learning from data without being explicitly programmed.
- **Deep Learning**: A subcategory of machine learning that uses a layered neural-network architecture originally inspired by the human brain.

- Artificial Neural Network
  - A class of Machine Learning algorithms inspired by the functioning of brain.
  - Development was stagnant because of compute challenges.
  - Research & development was boosted with the emergence of GPU in 2000's and 2010's.
- Further readings:
  What's the Difference Between Artificial Intelligence, Machine Learning and Deep Learning? by Michael Copeland at NVIDIA

MODULE 5: THE DATA SCIENCE PROCESS

- Data is being generated at a very high rate on a very large scale. Most of the generated data remains unused.
- Big Data is a term which is used to define the data which cannot be processed locally using traditional methods.
- To process the large amount of data, new concepts like Cloud Computing, Distributed Processing have emerged.

- Today, companies are making every effort to gain insights from the data in order to improve their profitability.
- Big Data has the following 4 main characteristics:
  - Volume
  - Variety
  - Velocity
  - Veracity
- However, this huge amount of raw data cannot be directly used to derive insights or train ML models because of issues like missing values, noise in the data, unsupported format, etc.
- In order to derive any meaningful insights or feed this data to an ML model, this data first needs to be cleaned and processed.
- Today, the ability to combine large, disparate data sets into a format more appropriate for analysis is an increasingly crucial skill.
- The data science process typically starts with collecting and preparing the data before moving on to training, evaluating, and deploying a model.
- Below are the steps involved in a standard data science process:
  - **Collect Data**: This step involves collecting data from different sources like mobile devices, IoT devices, sensors, software, etc. A developer may have to write queries and code to extract data from databases and webpages.
  - **Prepare Data**: This step involves cleaning the data and converting it into a desired format. This step involves activities like handling missing values, noisy data, creating new features, etc. A developer may have to write code to remove noisy data, handle missing values and perform data visualization.
  - **Train Model**: This step involves deciding an algorithm, splitting our data into train, validation and test sets, and training a model. A developer may have to write code to create and train the model.
  - **Evaluate Model**: This step involves evaluation of the performance of our model using different metrics like accuracy, loss, speed, etc.
  - **Deploy Model**: Once you're satisfied with the performance of your model, you can deploy your model using different techniques to derive useful insights and outputs.
  - **Re-train Model**: This is an iterative step which involves training the model on fresh data at regular intervals to make sure the performance of your model is in sync with the changing data environment.



MODULE 6: COMMON DATA TYPES

- Numerical
- Time-Series (numeric data, but in specific order)
- Categorical (represents different categories in real life)
- Text
- Image

All data in machine learning eventually ends up being numerical, regardless of whether it is numerical in its original form, so it can be processed by machine learning algorithms.

MODULE 7: TABULAR DATA

- This is the most common type of data encountered in Machine Learning,
- In tabular data, typically each cell describes a single value, each row describes a single item, while each column describes different properties of the item.
- A **vector** is simply an array of numbers, such as (1, 2, 3)—or a nested array that contains other arrays of numbers, such as (1, 2, (1, 2, 3))
- Khan Academy: Introduction to Linear Algebra
  https://www.khanacademy.org/math/linear-algebra
- Linear Algebra Refresher Course
  https://www.udacity.com/course/linear-algebra-refresher-course--ud953
- All non-numerical data types (such as images, text, and categories) must eventually be represented as numbers.

MODULE 8: SCALING DATA

- Scaling the data means transforming it in way that it fits within some range or scale, like 0-100 or 0-1.
- Methods of scaling data:
  - Standardization
    - Scales the data to have Mean = 0 and Variance = 1.
    - Scaling is done using the formula: **(x-Mean)/Variance**
  - Normalization
    - Scales the data in the range 0-1.
    - Scaling is done using the formula: $\mathbf{(x - x_{min}) / (x_{max} - x_{min})}$

MODULE 9: ENCODING CATEGORICAL DATA

- Machine Learning algorithms required data in the numerical format. Hence, it becomes important to convert our data in the required format.
- Ex: Converting categorical data (male, female, others) into numerical data.
- There are two common approaches for encoding categorical data:
  - Ordinal encoding
  - One hot encoding

  Let's take a look at them one by one.
- **Ordinal Encoding**
  - Converts the categories into integer code ranging from 0 to (number of categories – 1).
  - Ex:

Colour	**Encoding**
Green	0
Blue	1
Red	2

  - This method has one major drawback that it assumes that there is a particular order in the categories, like the colour Green is more important than Blue and Blue is more important than Red, or vice-versa. This may or may not be the case in reality.
  - In order to overcome this drawback, lets take a look at One hot Encoding.
- **One hot Encoding**
  - A new column is added for each distinct category in the data.
  - If a row belongs to a particular category, the value of column corresponding to that category will me marked as 1, while all other columns corresponding to all other categories will be marked as 0.
  - Ex:

Name	**Gender**
Rayan	M
Jessy	F
Liz	F

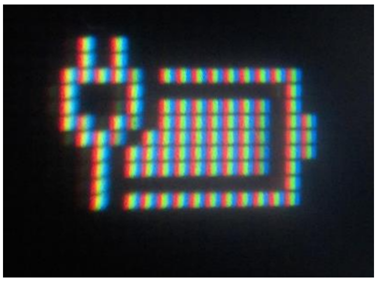
  - The above column can be converted into the following format after applying one hot encoding:

Name	**Male**	**Female**	**Others**
Rayan	1	0	0
Jessy	0	1	0
Liz	0	1	0

  - This approach gets rid of the drawback created by the ordinal encoding.
  - However, this approach gives rise to another problem of having large number of columns in case you have more categories.

MODULE 10: IMAGE DATA

- Zooming in on the below image, you'll find that this image is made of small square titles called a "Pixel".



- Digitally, images are represented in form of pixels. A pixel is a smallest unit of an image.
- Images are described in terms of total number of pixels i.e.,

  Height x Width x Number of channels
- In machine learning, square images are most commonly used.
- The colour of each pixel can be represented in different format:
  - **Greyscale**: Each pixel is represented by a single value ranging between 0-255. Here the number 0 represents black and 255 represents white colour.
  - **Coloured**: Each pixel is represented by a vector of 3 number, where each number ranges between 0-255.
- The number of channels required to represent a colour is called **colour depth** or simply, **depth**.
  - In case of a **greyscale** image, the **colour depth** is **1**.
  - While in the case of an RGB image, the **colour depth** is **3.**
- We can fully encode an image numerically by using a vector with three dimensions. The size of the vector required for any given image would be the **height * width * depth** of that image.
- We may want to perform other processing operations on an image after encoding it:
  - Normalization: Subtracting the mean pixel value in a channel from each pixel value in that channel.
  - Rotation
  - Cropping
  - Resizing
  - Denoising
  - Centring

MODULE 11: TEXT DATA (DAY 1/50)

- Text is another form of data that is non-numerical initially and must be processed before feeding it to the machine learning algorithms.
- **Normalization**
  - Normalization means converting a piece of text into a canonical/official form.
  - It is often seen that many different words used in text mean the same thing:
    - Ex: the verb **to be** may show up as **am, is, are**.
  - Also, many words have 2 different spellings.
    - Ex: **behavior** and **behaviour**
  - Thus, it becomes necessary to perform normalization to resolve all the above-mentioned inconsistencies.
- **Lemmatization**
  - **Lemma** is the dictionary form of a word.
  - Lemmatization is a form of Normalization which involves reducing multiple inflections to the dictionary form of the word.
  - This can be understood with the following example:

Original Word	**Lemmatized Word**
am	be
is	be
are	be

- **Removing Stop words**
  - Stop words are high-frequency words which add little meaning during analysis.
  - For example, after removing the stop words, the phrase **how to reach the Mount Everest** is reduced to **reach Mount Everest**, which still conveys pretty much the same meaning.
- **Tokenization**
  - Tokenization is a very common practice in text processing where we split each string into smaller parts, called **tokens**.
  - The below examples demonstrate tokenization:

Original String	**Tokenized text**
I like mangoes	[I, like, mangoes]
The train left the station	[The, train, left, the, station]

- **Vectorization**
  - After the normalization of text, next we convert it into a numerical vector, and the process is called vectorization.
  - There are many different methods of vectorization, but the 2 most common ones are as follows:
    - Term-Frequency Inverse Document Frequency (TF-IDF)
      https://en.wikipedia.org/wiki/Tf-idf
    - Word Embedding
      Word2Vec: https://en.wikipedia.org/wiki/Word2vec
      Global Vectors (GloVe): https://nlp.stanford.edu/pubs/glove.pdf

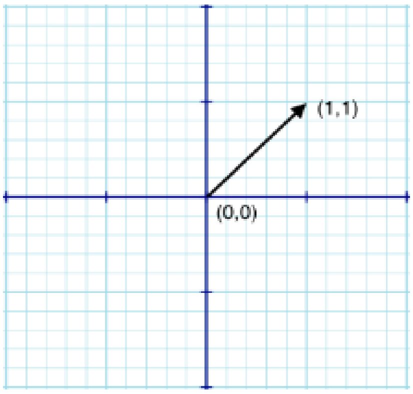
- **TF-IDF**
  - This approach gives lesser importance to words (like **is, the, am, will**) which are most common in the document and represent little information.
  - Words which contain more information and appear less frequently are given more importance using this approach.
  - This approach can be understood with the example below:

python	**is**	**a**	**general**	**purpose**	**programming**	**language**
0.32	0.0	0.0	0.10	0.13	0.45	0.25

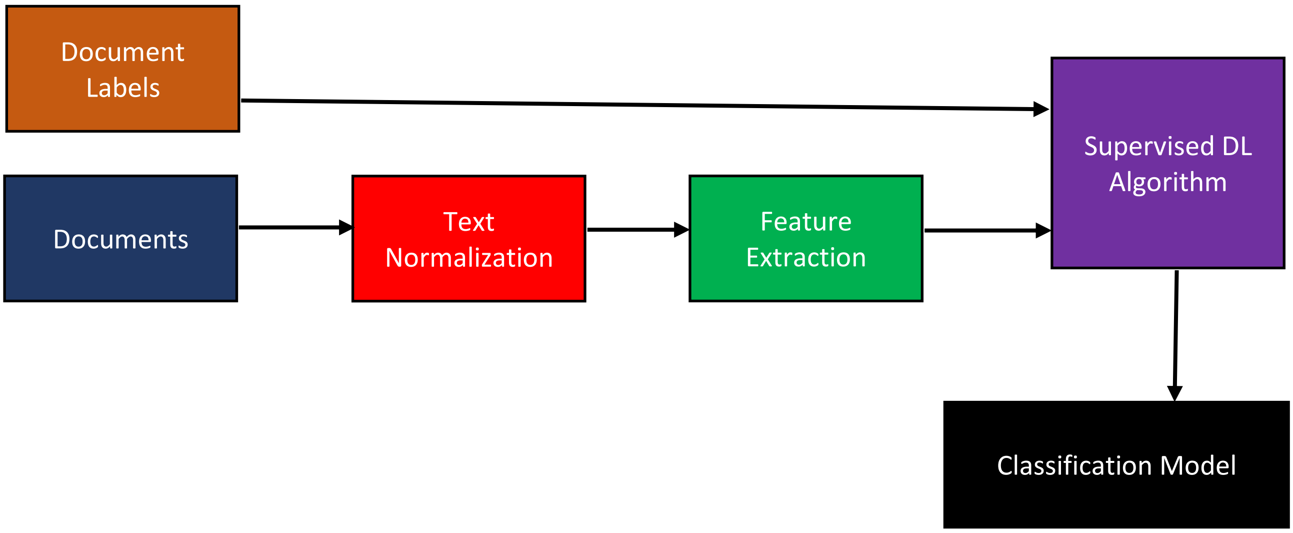
  - The above table contains the phrase **python is a general-purpose programming**
  - **language** split up into tokens and given weightage depending upon their importance in the document.
  - Taking a chunk of above text would result in the following table:

	python	general	purpose	programming	language
[python, general]	0.32	0.10	0.0.	0.0	0.0
[purpose, programming]	0.0	0.0	0.13	0.45	0.0
Language	0.0	0.0	0.0	0.0	0.25

  - It can be noticed that the words **'is'** and **'a'** are not a part of the above table since they do not contain any important information.
- **Feature Extraction**
  - The text in the previous example can be represented in form of a single vector containing 5 elements (since there are 5 total words after removing stop words).
    [python, general] = [0.32, 0.10, 0, 0, 0]
    [purpose, programming] = [0.0, 0.0, 0.13, 0.45, 0.0]
    [language] = [0.0, 0.0, 0.0, 0.0, 0.25]
  - Vectors of length **n** can be represented in **n-dimensional** space.
  - For example, a vector (1,1) can be viewed as a line starting from (0,0) and going till (1,1).



- How close two vectors are can be calculated using vector distance.
- When two vectors are close to each other i.e., they have a small vector distance, it can be understood that those two vectors either have similar meaning or are closely related to each other.
- Below is the end-to-end pipeline for classification model using text data.



MODULE 12: TWO PERSPECTIVES ON ML

- Machine Learning can be described from two different perspectives.
  - Computer Science
  - Statistical
- From the **Computer Science** perspective, we may state that we're using input features to create a program that can generate the desired output.
- From a **Statistical perspective**, we may state that we're trying to find a function which that can generate the values of the dependent variables given the values of the independent variables.
- We can map the two perspectives in the following manner:

Computer Science	**Statistics**
Program	Function
Input	Independent variables
Output	Dependent variables

MODULE 13: THE COMPUTER SCIENCE PERSPECTIVE

- Data can be present in form of rows and columns in a spreadsheet, as displayed below.

Name	**Gender**	**Height**	**Weight**
Tom	M	175	87
Vic	F	169	65
Laura	F	180	76

- From the Computer Science perspective, each row can be considered as an **entity** or an **observation about an entity**.

Tom	M	175	87

- Each column in the spreadsheet can be then considered as an **attribute** or a **feature** of the aforementioned entity.

Gender	**Height**
M	175
F	169
F	180

- A row may also be called an **instance.**
- **INPUT VECTOR**: A group of input variables
- In the computer science terms, we can understand machine learning as:

Output = Program (Input Features)

MODULE 14: THE STATISTICAL PERSPECTIVE

- In the Statistical perspective, the machine learning algorithm is trying to find a hypothetical function, f, such that:

Output Variables = f (Input Variables)

- The input variables are also called the **independent variables** while the output variables are also known as the **dependent variables**. Thus, the above equation can be rewritten as:

Dependent Variables = f (Independent Variables)

- Often the output variable is represented with the alphabet **Y** and the input variable is represented with the alphabet X. So, the above equation is commonly represented with the shorthand as:

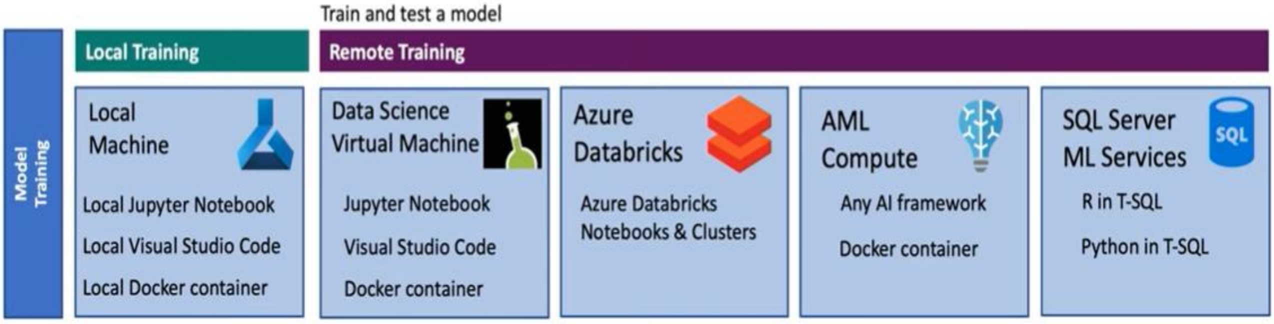
Y = f (X)

MODULE 15: THE TOOLS FOR MACHINE LEARNING

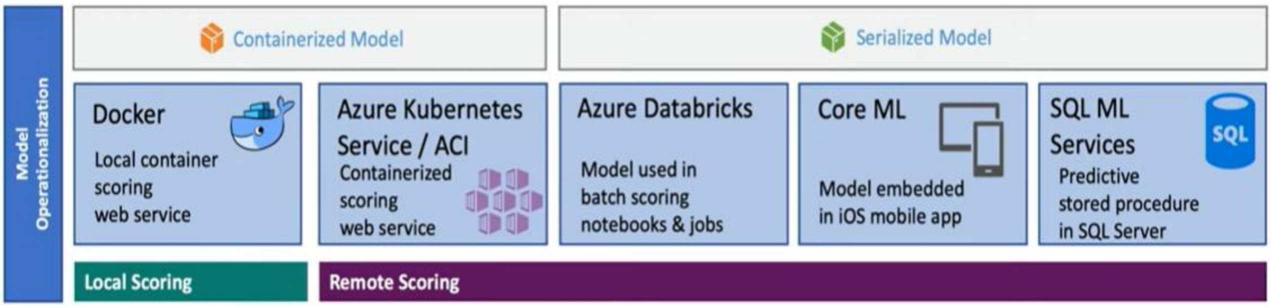
- Let’s take a look at some of the most popular libraries and tools which form an integral part of any Machine Learning Ecosystem.

Tool/Library Category	Tool/Library Name	Description
Libraries	Scikit-Learn	Classical ML library
	• TensorFlow • Keras • PyTorch	Deep Learning libraries
Development Environments	• Jupyter Notebooks • Azure Notebooks • Azure Databricks • Visual Studio • Visual Studio Code	Provide interface to build, train and test your model by writing code.
Cloud Services	• Microsoft Azure Machine Learning • AWS • GCP	Service providers which allow you to develop and deploy your ML models on Cloud.

- Microsoft Azure provides environment for both development and deployment (operationalization) of your model.
- Below are the different tools/options provided for training your ML model.



- After training your model, you can also deploy/operationalize your model using different tools/options provided by Microsoft Azure.


Model Operationalization
Containerized Model
Serialized Model
Docker
Local container scoring web service
Azure Kubernetes Service / ACI
Containerized scoring web service
Azure Databricks
Model used in batch scoring notebooks & jobs
Core ML
Model embedded in iOS mobile app
SQL ML Services
SQL
Predictive stored procedure in SQL Server
Local Scoring
Remote Scoring