What is ML?

What are the tools?

Hardware: CPU vs GPU
Operating System: Windows, Linux, MAC
Software Tools: R, Matlab (commercial), Python (Scikit-Learn, TensorFlow, PyTorch), Julia
Interface: Jupyter Lab (Supports R, Python, Julia)

Machine Learning Workflow

Data → Features → Model Building Algorithms → Evaluation → Model

Data

Raw Data Generation
Data Management

Features

Selecting Relevant Features
Feature Normalization
Feature Transformation
Feature Creation
Preprocessing
Contextualization

Model Building Algorithms

Supervised
Unsupervised
Reinforced

Evaluation

Train and Tune
Accuracy
Confusion Matrix
Log Loss
Area under ROC Curve

Model

Prediction
Production Data

Three Types of Data

1. Numeric: pressure, temp, volume, weight
2. Categorical: machine class, status
3. Ordinal: first/second/low/medium

Data Preprocessing

1. **Conversion of data:** Convert data into numeric features.
2. **Ignoring the missing values:** Remove the row or column of data.
3. **Filling the missing values:** Fill the missing data, mean, median or frequency.
4. **Machine learning:** Predict data at the empty position by using the existing data.
5. **Outliers detection:** There are some error data that might be present
6. **Significant features**: Which component has the most contribution to failure
7. Derived feature from time series

Data Split

1. Training Data
2. Validation Data
3. Testing Data

Accuracy = (True P +True N) / (Total)

Prediction:

What failures or events or future demand are we interested in?
What is the required accuracy?
What is the frequency?

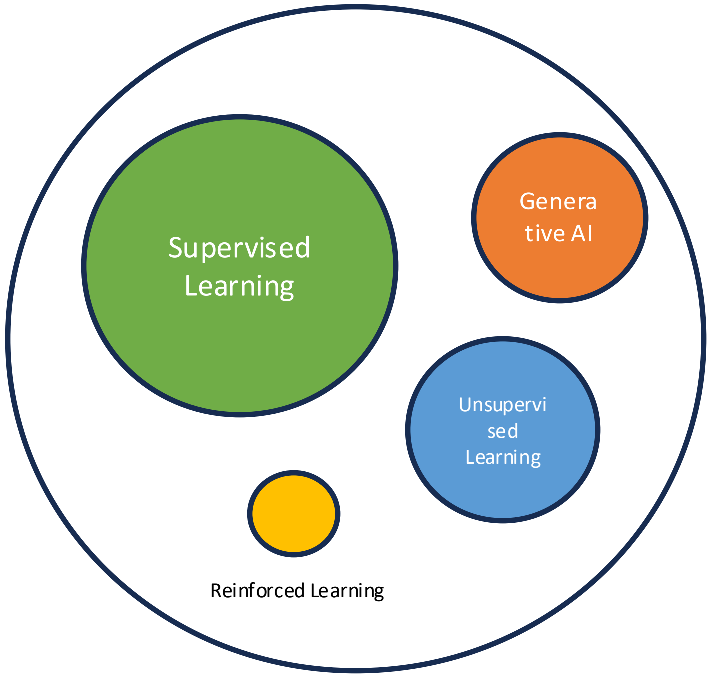
Machine Learning Workflow

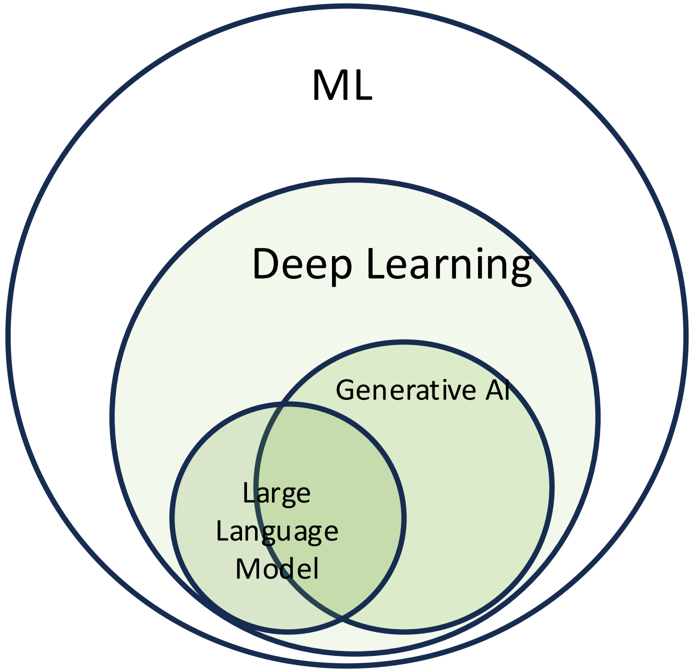
1. **Raw Data Collection**: Gather data from various sources.
2. **Data Management**: Store, organize, and secure data.
3. **Data Preprocessing**: Clean, transform, and scale data for modeling.
4. **Feature Selection**: Choose relevant features for model training.
5. **Model Selection**: Choose between supervised, unsupervised, or reinforcement learning.
6. **Model Training**: Train the model using the data.
7. **Model Evaluation**: Evaluate the model using appropriate metrics.
8. **Model Optimization**: Tune hyperparameters and improve the model.
9. **Model Deployment**: Deploy the model into production.
10. **Monitoring & Maintenance**: Continuously monitor and retrain the model as needed.

ML Terminology

- Label: Label is the property we are trying to predict. In a linear regression, we use ‘y’ to denote label
- Features: Input variables, in linear regression we use ‘x’ to denote features. x_1, x_2, x_3, …. $X_{n.}$
- Regression vs Classification: Regression model is used to predict continuous values (Eg:- house prices) whereas classification model predicts discrete values (cat or dog). Logistic regression model uses sigmoid function to predict a value between 0 and 1.
- Training: Determining the optimum values for the bias and weights to predict the result.
- Loss: It is a number indicating how bad the prediction was.
- Accuracy: Fraction of predictions that were right in a classification model
- Clustering: Grouping of related examples during unsupervised learning (Eg:- K-means)
- Hyperparameter: Values that are tweaked during the training of a model (Eg:- Learning rate)
- Supervised Learning: Training a model with input data with its corresponding labels
- Unsupervised Learning: Training a model to find patterns in data set that are unlabeled.
- Validation: A process used in machine learning to evaluate the quality of the trained model.
- Gradient descent Algorithm: Calculate the gradient of the loss curve at the starting point. Gradient is the derivative (slope) of the curve. Takes the direction of negative gradient to reduce the loss as quick as possible.
- Learning rate: step size in gradient descent to multiply the gradient with a scalar to determine the next point.
- Training set: A subset to train the model
- Test Set: A subset to test the model.

Machine Learning Types





Machine Learning Fundamentals

1) Cost Function

Cost Function is a measure of how much the predicted and actual value differs. The goal of Machine learning is to minimize the cost function.

2) Linear Regression Model

Single feature

$$y' = w_1\, x + b$$

Multiple features

$$y' = b + w_1 x_1 + w_2 x_2 + w_3 x_3 + \cdots$$

y' is the predicted result, x is the feature value, w_1 is the weight for feature x and b is the bias.

3) Squared Loss (L_2) loss

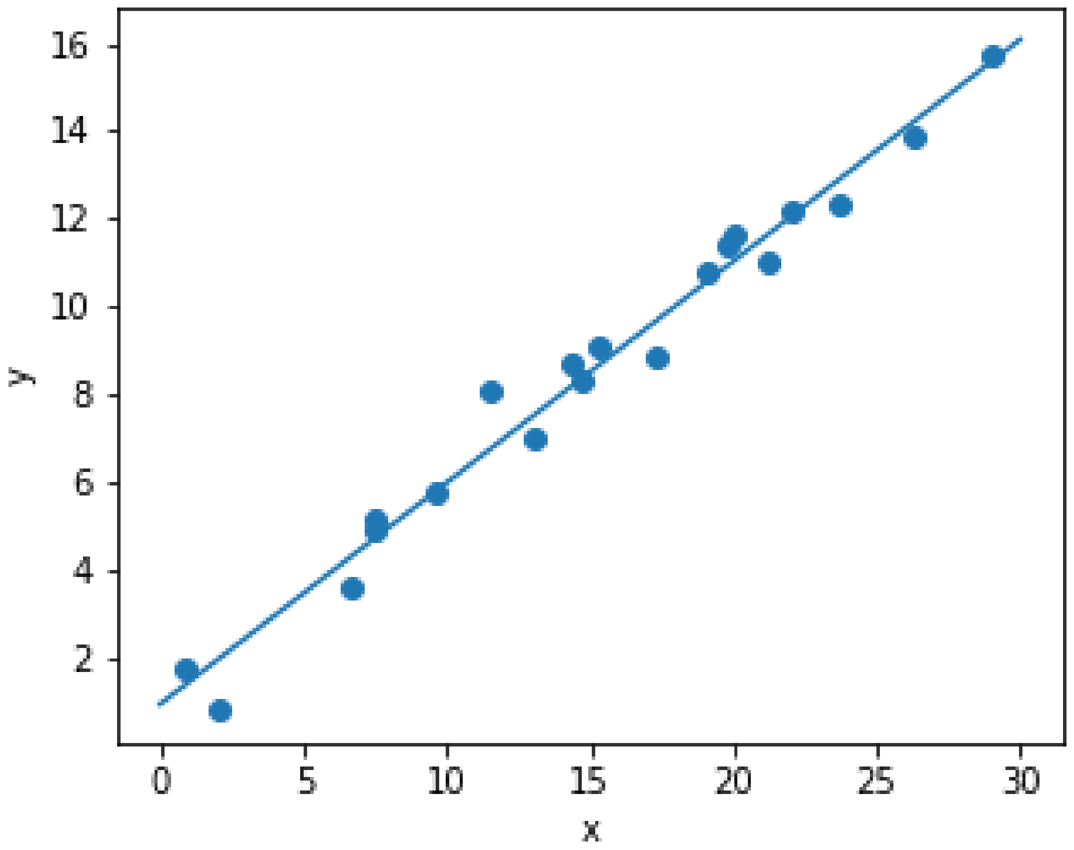
- the square of the difference between the label and the prediction
- (observation - prediction(**x**))^2
- $(y - y')^2$

4)Mean Square Error

$$MSE = \frac{1}{N} \sum_{(x,y)\in D} (y - y'(x))^2$$

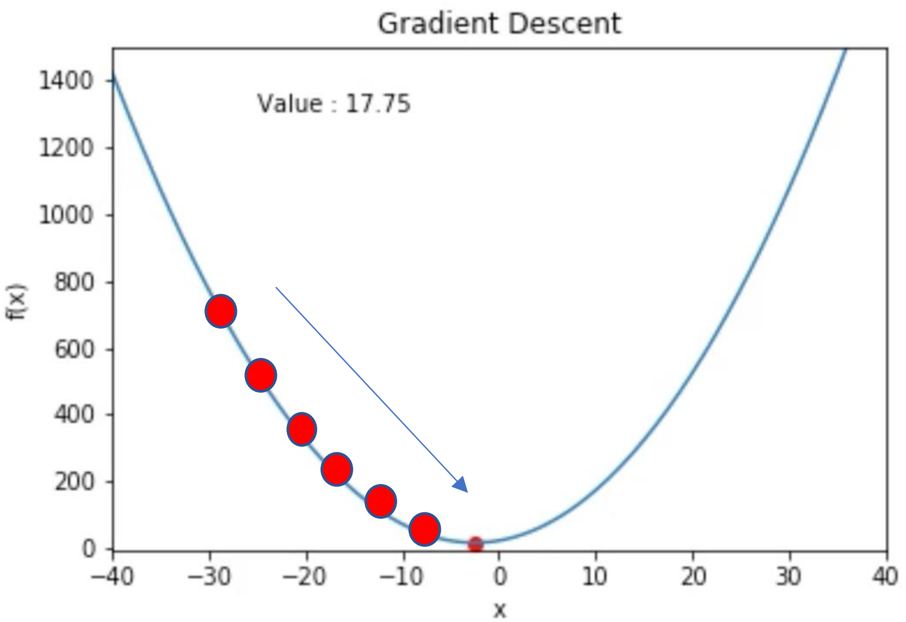
- x is set of features that are available in the dataset to make prediction
- y is the actual label,
- y'(x) is a function of weights and bias that gives the predicted value
- D is the dataset,
- N is the number of examples in the data set

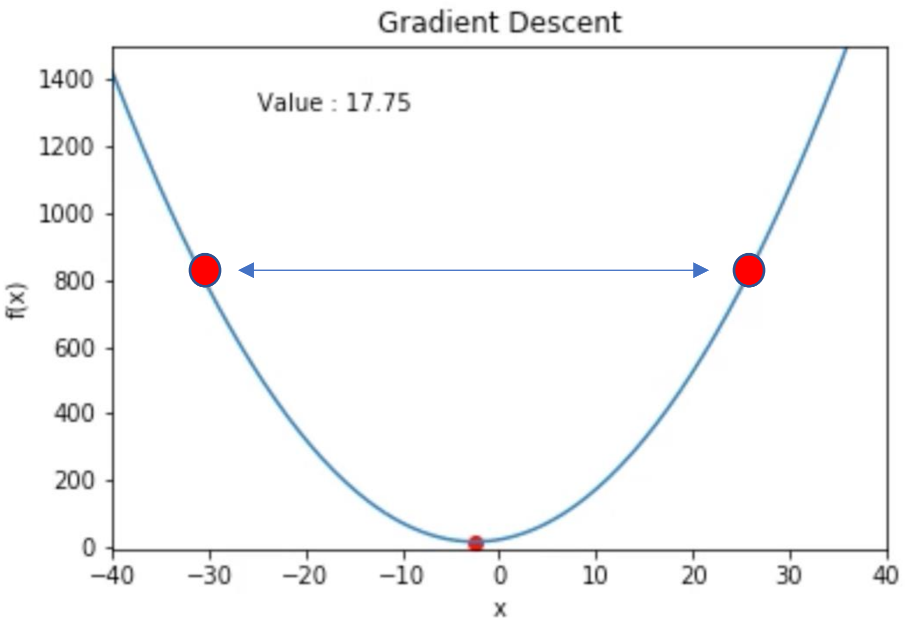
Linear Regression



Gradient Descent

(Hyerparameters – Learning rate, Epochs, Batch size)





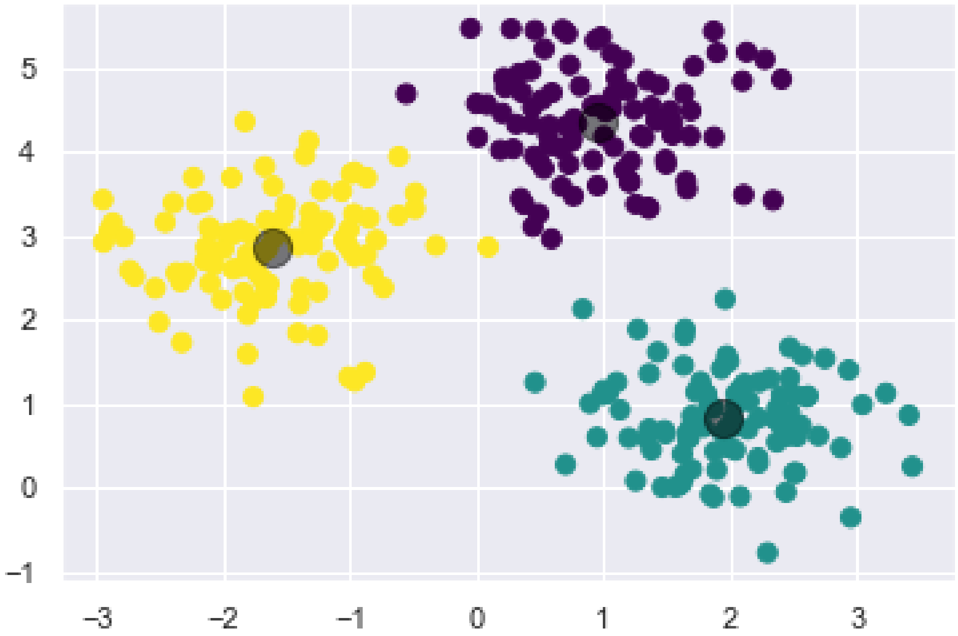
Gradient descent is a mathematical technique that iteratively finds the weights and bias that produce the model with the lowest loss

1. Calculate the loss with the current weight and bias.
2. Determine the direction to move the weights and bias that reduce loss.
3. Alter the weight and bias values a small amount in the direction that reduces loss.
4. Return to step one and repeat the process until the model can't reduce the loss any further.

K-means Clustering

- The cluster center is the arithmetic mean of all the points belonging to the cluster.
- Each point is closer to its own cluster center than to other cluster centers.
- Euclidean distance between two points a and b with k dimension is calculated as follows

$$\sqrt{\sum_{j=1}^{k}(a_j - b_j)^2}$$



Single Instruction Multiple Threads (SIMT) Algorithm, GPUs and Deep Learning
Data Parallel

Matrix Vector Multiplications (multiply and add operation)

$$\begin{pmatrix} \boxed{w_{11}, w_{12}, w_{13}} \\ w_{21}, w_{22}, w_{23} \\ w_{31}, w_{32}, w_{33} \end{pmatrix} \boxed{\begin{matrix} x_1 \\ x_2 \\ x_3 \end{matrix}} = \begin{pmatrix} \boxed{w_{11} * x_1 + w_{12} * x_2 + w_{13} * x_3} \\ w_{21} * x_1 + w_{22} * x_2 + w_{23} * x_3 \\ w_{31} * x_1 + w_{32} * x_2 + w_{33} * x_3 \end{pmatrix}$$

Weights Inputs

GPUs Vs CPUs

Tens of thousands of concurrent threads vs Complex Operation

Linearity vs Nonlinearity

Linear Function:

- Imagine a bunch of uncooked spaghetti.
- The strands are like a straight line on graph, similar to a linear function.

Non-linear Function:

- Now, picture cooked spaghetti.
- The strands are curved, bent, and intertwined, it no longer follows a straight line and represents a non-linear function.

Cooking process is like applying an activation function in a neural network, transforming the "uncooked" (linear) input into a "cooked" (non-linear) output. Sigmoid and ReLU f(x) = max(0, x) functions are two examples for activation function