$$\textbf{Entropy}(S) = -\sum_{i=1}^{n} p(i)\log_2 p(i)$$

where S is the dataset or sample, $p(i)$ is the proportion of elements in class i within S, and n is the number of classes.

$$\textbf{Gini Index} = 1 - \sum_{j} p_j^2$$

where p_j is the probability of a sample belonging to class j.

$$\textbf{Accuracy} = \frac{TP + TN}{P + N}$$

$$\textbf{Precision} = \frac{TP}{TP + FP} \quad \textbf{Recall} = \frac{TP}{TP + FN}$$

$$F_\beta = (1 + \beta^2) \cdot \frac{\text{Precision} \cdot \text{Recall}}{\beta^2 \cdot \text{Precision} + \text{Recall}}$$

$$t = \frac{\bar{d}}{s_d/\sqrt{n}}$$

where $\bar{d}$ is the mean difference between paired scores, s_d is the standard deviation of the differences between paired scores, and n is the number of paired scores.

Applications of Supervised Learning: Linear Regression, Logistic Regression, Object Detection

Applications of Unsupervised Learning: Clustering, Fraud Detection, Principal Component Analysis (Transforms the data into a set of linearly uncorrelated components that capture the most variance)

Methods of Reinforcement Learning:

1. **Q-Learning**: Updates a Q-table that maps state-action pairs to expected future rewards
2. **Deep Q-Networks (DQN)**: Uses a neural network to approximate the Q-function, allowing it to handle large state spaces

$$\text{Minkowski Distance}(x, X_i) = \left(\sum_{j=1}^{d} |x_j - X_{ij}|^p\right)^{\frac{1}{p}}$$

where $p = 2$ simplifies to Euclidean distance and $p = 1$ simplifies to Manhattan distance.

Gradient Descent Update Rule: $\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta)$
Normal Equation: $\theta = (X^T X)^{-1} X^T y$

Sigmoid Function: $\sigma(z) = \frac{1}{1+e^{-z}}$

$$\text{Cross Entropy Loss}(h_\theta(x), y) = -\log(1 - y + (2y - 1)h_\theta(x))$$

Logistic regression handles multi-class classification by using multiple one-vs-all (OvR) classifiers.

$$\text{Norm of a Vector: } \|\mathbf{x}\|_p = \left(\sum_{i=1}^{n} \|x_i\|^p\right)^{\frac{1}{p}}$$

$$\|\mathbf{x}\|_1 = \sum_{i=1}^{n} |x_i| \quad \|\mathbf{x}\|_2 = \sqrt{\sum_{i=1}^{n} x_i^2} = \sqrt{\mathbf{x}^T\mathbf{x}} \quad \|\mathbf{x}\|_\infty = \max |x_i|$$

$$\text{Orthogonal projection of } y \text{ onto } x\text{: } \text{proj}_x y = \frac{x \cdot y}{x \cdot x} x$$

$$\text{Bayes' Theorem: } P(X \mid Y) = \frac{P(Y \mid X)P(X)}{P(Y)}$$

$$\text{Covariance}(f(x), g(Y)) = \mathbb{E}\Big[(f(X) - \mathbb{E}[f(x)])(g(Y) - \mathbb{E}[g(Y)])\Big]$$

$$\text{Correlation: } \rho(X, Y) = \frac{\text{Cov}(X, Y)}{\sigma_x \cdot \sigma_y}$$

$$\text{L1 Regularization: Cost} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 + \lambda \sum_{i=1}^{m} |\theta_i|$$

$$\text{L2 Regularization: Cost} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 + \lambda \sum_{i=1}^{m} \theta_i^2$$

$$\text{Silhouette Score: } S(i) = \frac{b(i) - a(i)}{\max(a(i), b(i))}$$

where $a(i)$ is the average distance between that point and all other points in its cluster and $b(i)$ is the average distance between that point and all points in the neareast different cluster.