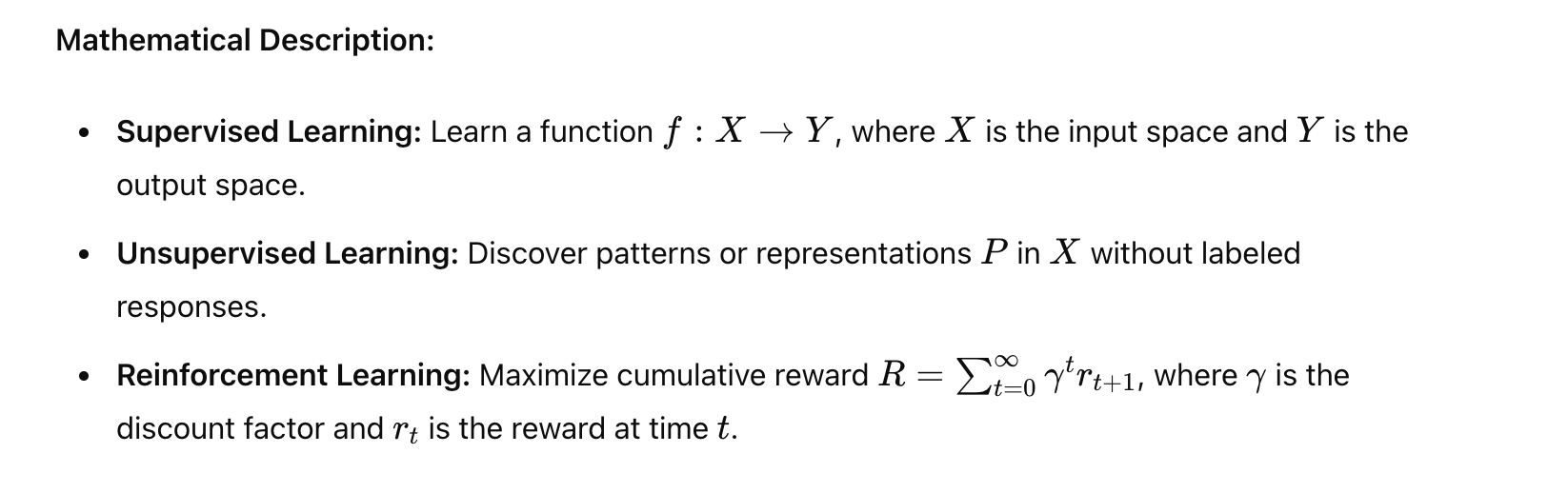
**1. ML Landscape, ML Categories**

**Theory:** Machine Learning (ML) is a subfield of artificial intelligence (AI) focused on developing algorithms that allow computers to learn from and make predictions based on data. It bridges the gap between raw data and insightful information by enabling systems to improve their performance on tasks through experience. ML can be broadly categorized into three types:

* **Supervised Learning:** In this approach, the algorithm is trained on a labeled dataset, meaning that each training example is paired with an output label. The goal is to learn a mapping from inputs to outputs. It is commonly used for classification and regression tasks.
* **Unsupervised Learning:** Here, the algorithm is given data without explicit labels. The goal is to infer the natural structure present within a set of data points. Techniques include clustering, dimensionality reduction, and anomaly detection.
* **Reinforcement Learning:** This method involves an agent that learns to make decisions by performing actions in an environment to maximize some notion of cumulative reward. It is widely used in robotics, game playing, and navigation tasks.

**Mathematical Description:**

* **Supervised Learning:** Learn a function f:X→Yf: X \rightarrow Yf:X→Y, where XXX is the input space and YYY is the output space.
* **Unsupervised Learning:** Discover patterns or representations PPP in XXX without labeled responses.
* **Reinforcement Learning:** Maximize cumulative reward R=∑t=0∞γtrt+1R = \sum\_{t=0}^\infty \gamma^t r\_{t+1}R=∑t=0∞​γtrt+1​, where γ\gammaγ is the discount factor and rtr\_trt​ is the reward at time ttt.



**Examples:**

* **Supervised Learning:** Predicting house prices using features like size and location (regression), classifying emails as spam or not spam (classification).
* **Unsupervised Learning:** Grouping customers into segments based on purchasing behavior (clustering), reducing the dimensionality of image data (PCA).
* **Reinforcement Learning:** Training a robot to navigate a maze by rewarding it for reaching the end, developing game-playing AI like AlphaGo.

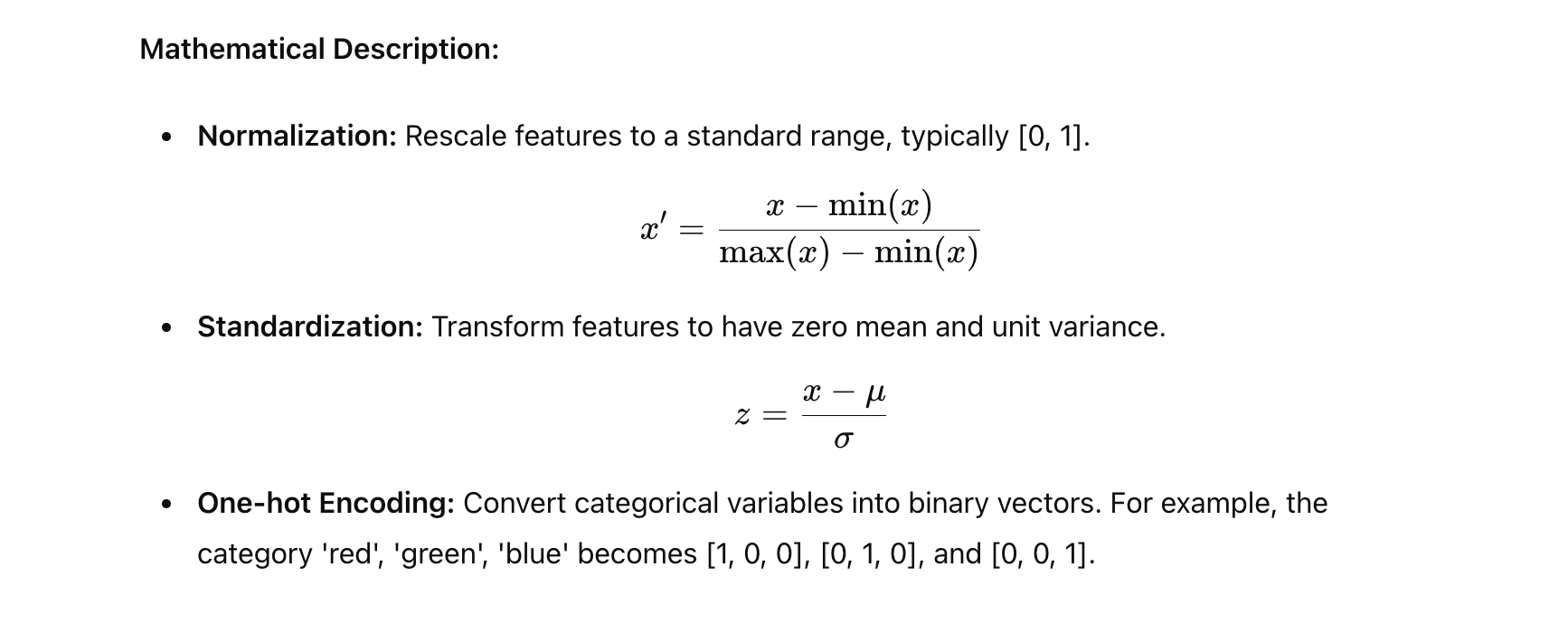
**2. Working with Data**

**Theory:** The process of working with data involves several crucial steps to ensure the data is suitable for model training and analysis. These steps include:

* **Data Collection:** Gathering data from various sources such as databases, web scraping, sensors, or user-generated content.
* **Data Cleaning:** Handling missing values, removing duplicates, and correcting inaccuracies to improve data quality.
* **Data Transformation:** Converting data into a suitable format for analysis. This can involve normalization, scaling, and encoding categorical variables.
* **Exploratory Data Analysis (EDA):** Understanding the data by summarizing its main characteristics using statistical graphics and visualization techniques.

**Mathematical Description:**

* **Normalization:** Rescale features to a standard range, typically [0, 1]. x′=x−min⁡(x)max⁡(x)−min⁡(x)x' = \frac{x - \min(x)}{\max(x) - \min(x)}x′=max(x)−min(x)x−min(x)​
* **Standardization:** Transform features to have zero mean and unit variance. z=x−μσz = \frac{x - \mu}{\sigma}z=σx−μ​
* **One-hot Encoding:** Convert categorical variables into binary vectors. For example, the category 'red', 'green', 'blue' becomes [1, 0, 0], [0, 1, 0], and [0, 0, 1].



**Examples:**

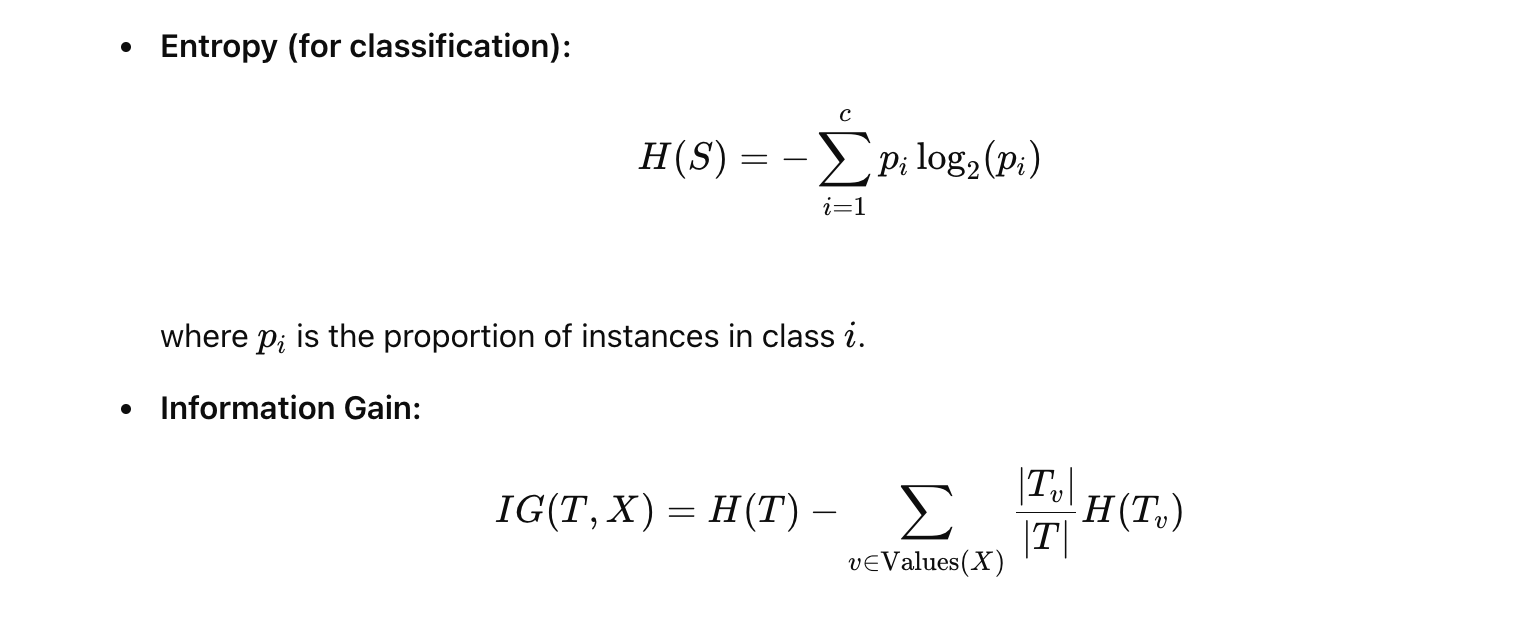
* **Data Cleaning:** Filling missing values in a dataset using mean or median imputation, removing rows with duplicate entries.
* **Normalization:** Scaling the features of a dataset containing heights (in cm) and weights (in kg) to ensure each feature contributes equally to the model.
* **One-hot Encoding:** Converting the 'color' feature with values 'red', 'green', 'blue' into three binary features.

**3. Supervised Learning, Decision Trees**

**Theory:** Supervised learning is a type of machine learning where the model is trained on a labeled dataset, meaning each training example is paired with a known output. Decision Trees are a type of supervised learning algorithm that can be used for both classification and regression tasks. They work by recursively splitting the data into subsets based on the value of input features, forming a tree structure where each node represents a decision based on a feature, and each leaf node represents a final prediction.

**Mathematical Description:**

* **Entropy (for classification):** H(S)=−∑i=1cpilog⁡2(pi)H(S) = - \sum\_{i=1}^c p\_i \log\_2(p\_i)H(S)=−i=1∑c​pi​log2​(pi​) where pip\_ipi​ is the proportion of instances in class iii.
* **Information Gain:** IG(T,X)=H(T)−∑v∈Values(X)∣Tv∣∣T∣H(Tv)IG(T, X) = H(T) - \sum\_{v \in \text{Values}(X)} \frac{|T\_v|}{|T|} H(T\_v)IG(T,X)=H(T)−v∈Values(X)∑​∣T∣∣Tv​∣​H(Tv​) where TTT is the set of training examples, XXX is a feature, and TvT\_vTv​ is the subset of TTT where XXX has value vvv.
* **Regression Trees:** Split data based on minimizing the variance in the target variable within each subset.



**Examples:**

* **Classification Tree:** Classifying whether a loan application is approved or denied based on features like credit score, income, and employment history.
* **Regression Tree:** Predicting the price of a car based on its age, mileage, and brand.

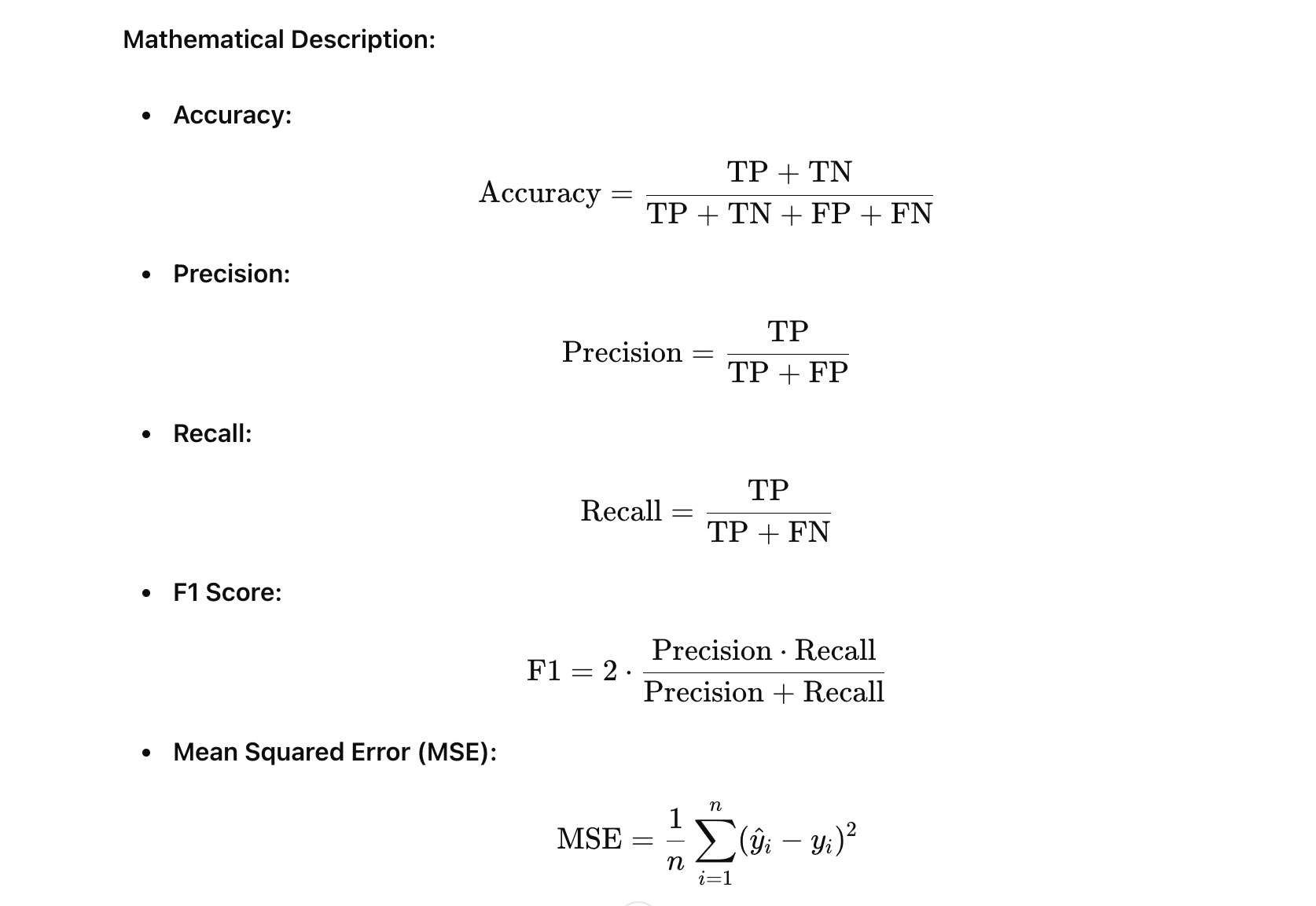
**4. ML Model Evaluation**

**Theory:** Evaluating ML models is crucial to understand their performance and generalization capability. Several metrics are used to evaluate models depending on the task (classification or regression):

* **Accuracy:** The proportion of correctly classified instances in the dataset.
* **Precision and Recall:** Precision measures the accuracy of positive predictions, while recall measures the coverage of actual positives. These are particularly useful in imbalanced datasets.
* **F1 Score:** The harmonic mean of precision and recall, providing a single metric that balances both.
* **Confusion Matrix:** A table used to describe the performance of a classification model by comparing actual vs. predicted values.
* **Mean Squared Error (MSE):** The average of the squared differences between predicted and actual values in regression tasks.
* **Cross-Validation:** A technique to assess the generalization ability of a model by dividing the dataset into multiple folds and training/testing the model on different folds.

**Mathematical Description:**

* **Accuracy:** Accuracy=TP + TNTP + TN + FP + FN\text{Accuracy} = \frac{\text{TP + TN}}{\text{TP + TN + FP + FN}}Accuracy=TP + TN + FP + FNTP + TN​
* **Precision:** Precision=TPTP + FP\text{Precision} = \frac{\text{TP}}{\text{TP + FP}}Precision=TP + FPTP​
* **Recall:** Recall=TPTP + FN\text{Recall} = \frac{\text{TP}}{\text{TP + FN}}Recall=TP + FNTP​
* **F1 Score:** F1=2⋅Precision⋅RecallPrecision + Recall\text{F1} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision + Recall}}F1=2⋅Precision + RecallPrecision⋅Recall​
* **Mean Squared Error (MSE):** MSE=1n∑i=1n(y^i−yi)2\text{MSE} = \frac{1}{n} \sum\_{i=1}^n (\hat{y}\_i - y\_i)^2MSE=n1​i=1∑n​(y^​i​−yi​)2



**Examples:**

* **Classification Metrics:** Evaluating a spam classifier's performance using precision, recall, and F1 score.
* **Cross-Validation:** Using 5-fold cross-validation to evaluate a model's generalization performance, ensuring the model is not overfitting.

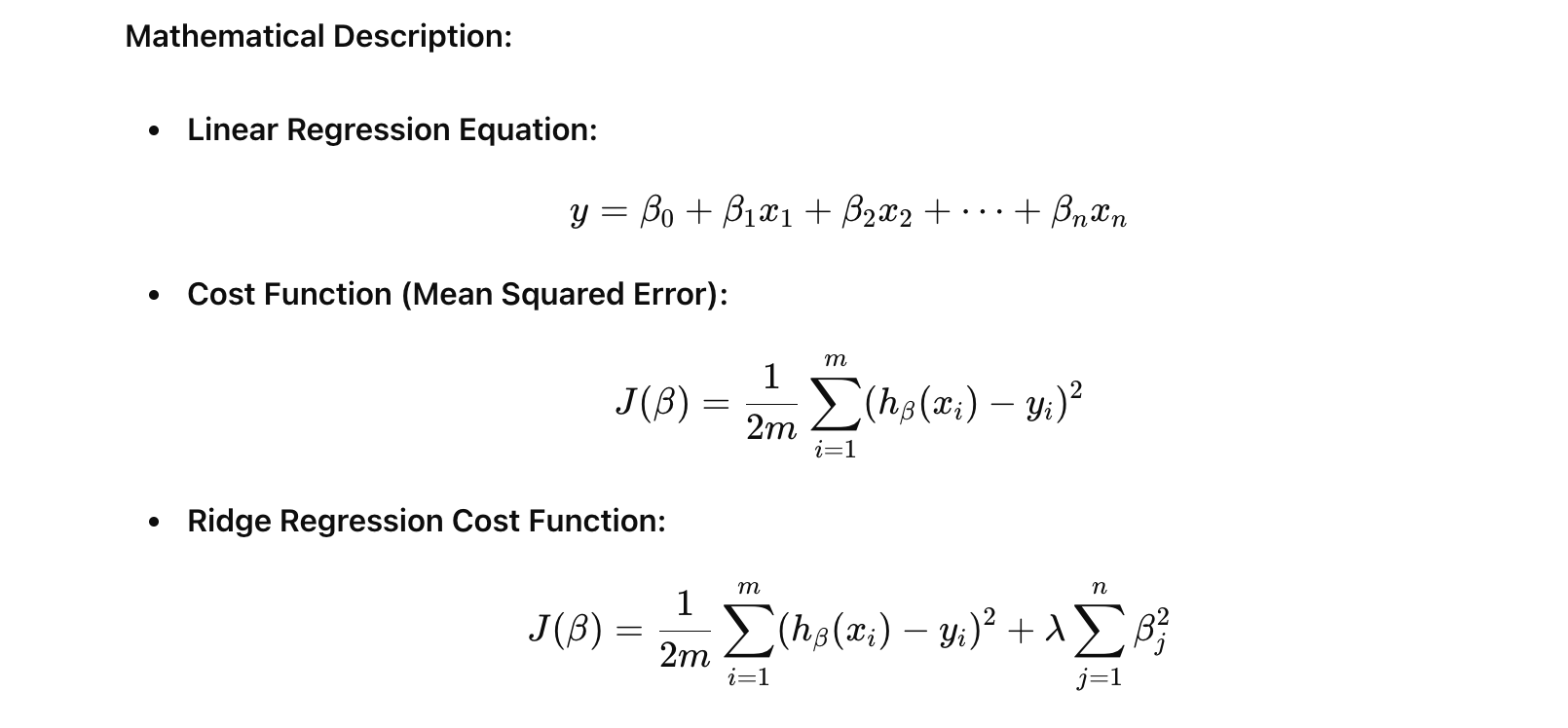
**5. Regression Models**

**Theory:** Regression models are used to predict a continuous output variable based on one or more input features. Common types of regression models include:

* **Linear Regression:** Models the relationship between a dependent variable and one or more independent variables by fitting a linear equation to the observed data.
* **Polynomial Regression:** A form of linear regression where the relationship between the independent variable and the dependent variable is modeled as an nth degree polynomial.
* **Ridge Regression:** A type of linear regression that includes an L2 penalty to prevent overfitting by adding a regularization term to the loss function.

**Mathematical Description:**

* **Linear Regression Equation:** y=β0+β1x1+β2x2+⋯+βnxny = \beta\_0 + \beta\_1 x\_1 + \beta\_2 x\_2 + \cdots + \beta\_n x\_ny=β0​+β1​x1​+β2​x2​+⋯+βn​xn​
* **Cost Function (Mean Squared Error):** J(β)=12m∑i=1m(hβ(xi)−yi)2J(\beta) = \frac{1}{2m} \sum\_{i=1}^m (h\_\beta(x\_i) - y\_i)^2J(β)=2m1​i=1∑m​(hβ​(xi​)−yi​)2
* **Ridge Regression Cost Function:** J(β)=12m∑i=1m(hβ(xi)−yi)2+λ∑j=1nβj2J(\beta) = \frac{1}{2m} \sum\_{i=1}^m (h\_\beta(x\_i) - y\_i)^2 + \lambda \sum\_{j=1}^n \beta\_j^2J(β)=2m1​i=1∑m​(hβ​(xi​)−yi​)2+λj=1∑n​βj2​



**Examples:**

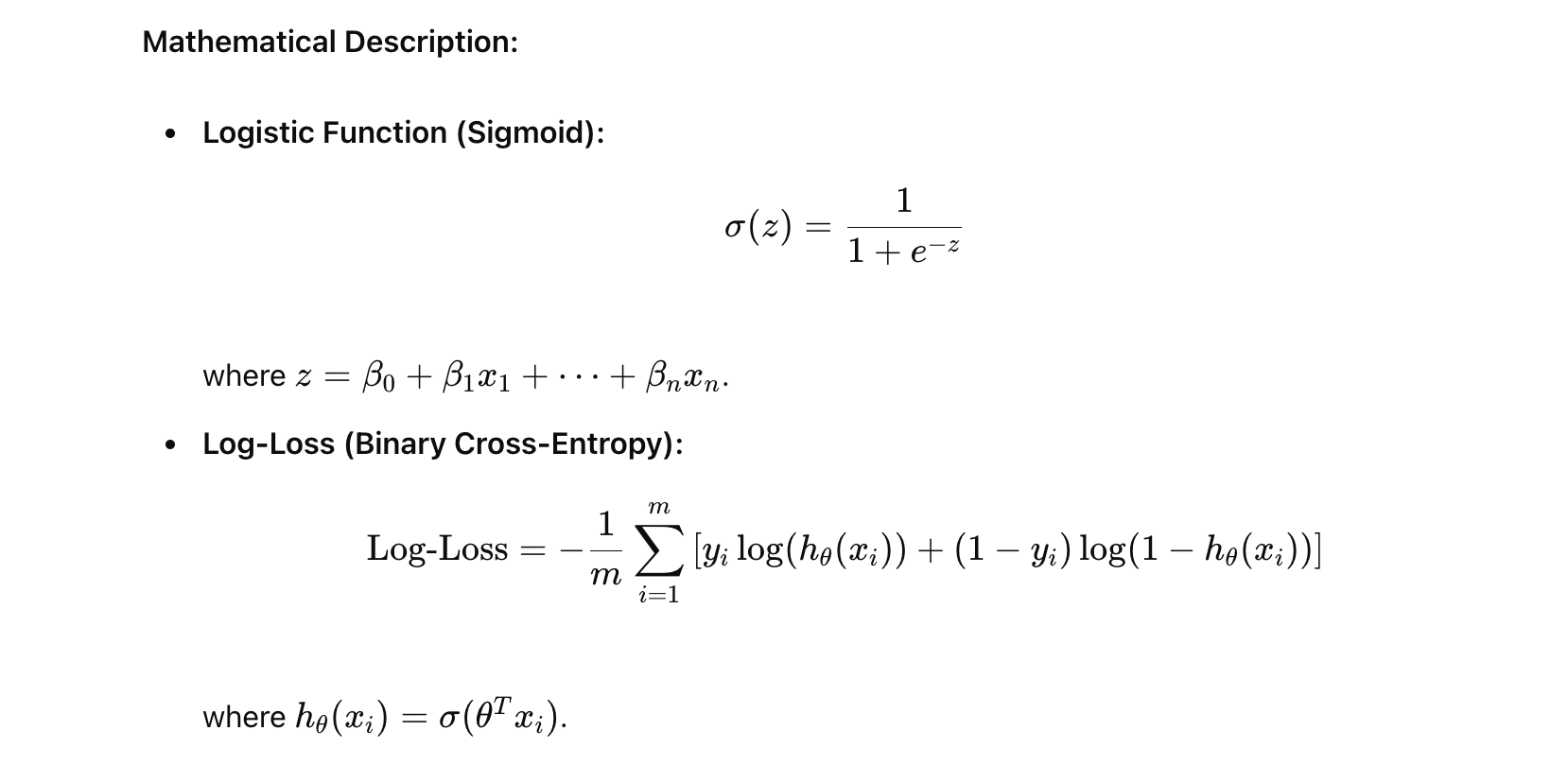
* **Linear Regression:** Predicting house prices based on features like size, number of bedrooms, and location.
* **Polynomial Regression:** Modeling the relationship between the amount of fertilizer used and crop yield.
* **Ridge Regression:** Predicting sales based on advertising spend in different media channels, where multicollinearity might be an issue.

**6. Logistic Regression**

**Theory:** Logistic regression is used for binary classification problems. Unlike linear regression, which predicts a continuous outcome, logistic regression predicts the probability that a given input point belongs to a particular class. The model uses the logistic function to map predicted values to probabilities.

**Mathematical Description:**

* **Logistic Function (Sigmoid):** σ(z)=11+e−z\sigma(z) = \frac{1}{1 + e^{-z}}σ(z)=1+e−z1​ where z=β0+β1x1+⋯+βnxnz = \beta\_0 + \beta\_1 x\_1 + \cdots + \beta\_n x\_nz=β0​+β1​x1​+⋯+βn​xn​.
* **Log-Loss (Binary Cross-Entropy):** Log-Loss=−1m∑i=1m[yilog⁡(hθ(xi))+(1−yi)log⁡(1−hθ(xi))]\text{Log-Loss} = -\frac{1}{m} \sum\_{i=1}^m \left[ y\_i \log(h\_\theta(x\_i)) + (1 - y\_i) \log(1 - h\_\theta(x\_i)) \right]Log-Loss=−m1​i=1∑m​[yi​log(hθ​(xi​))+(1−yi​)log(1−hθ​(xi​))] where hθ(xi)=σ(θTxi)h\_\theta(x\_i) = \sigma(\theta^T x\_i)hθ​(xi​)=σ(θTxi​).



**Examples:**

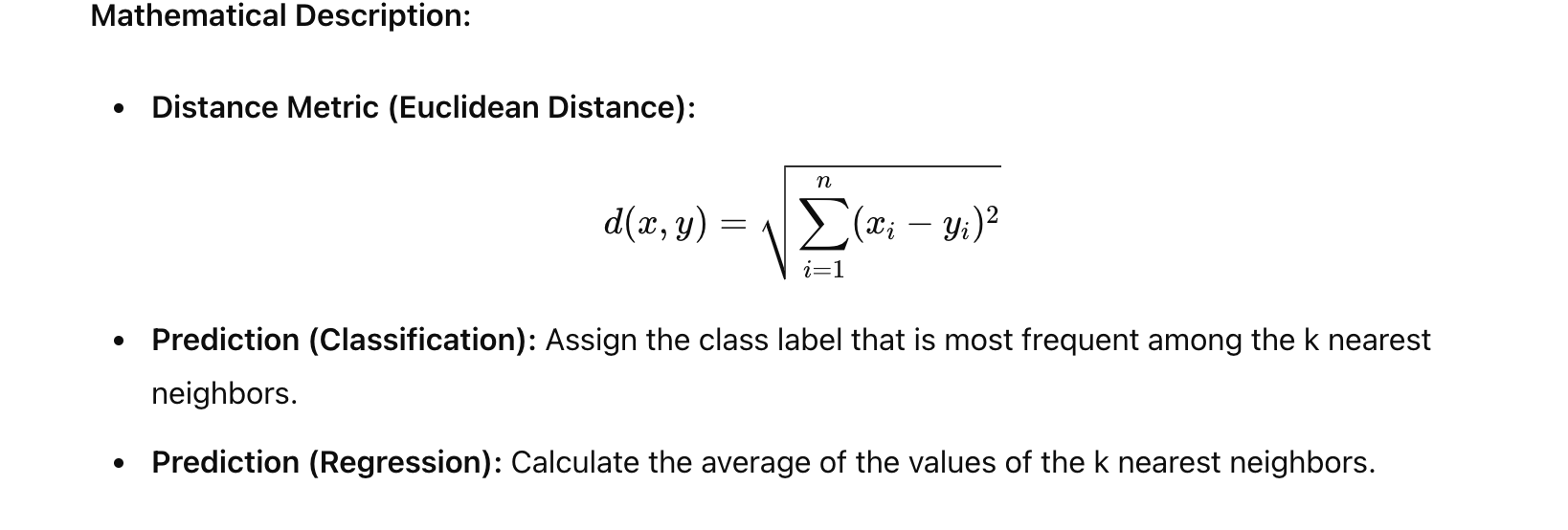
* **Spam Classification:** Predicting whether an email is spam or not based on features like word frequency and email metadata.
* **Credit Default Prediction:** Estimating the likelihood of a customer defaulting on a loan based on their financial history and credit score.

**7. K-Nearest Neighbors (KNN)**

**Theory:** K-Nearest Neighbors (KNN) is a simple, non-parametric, instance-based learning algorithm used for classification and regression. For classification, a new data point is assigned to the class most common among its k nearest neighbors. For regression, the predicted value is the average of the values of its k nearest neighbors.

**Mathematical Description:**

* **Distance Metric (Euclidean Distance):** d(x,y)=∑i=1n(xi−yi)2d(x, y) = \sqrt{\sum\_{i=1}^n (x\_i - y\_i)^2}d(x,y)=i=1∑n​(xi​−yi​)2​
* **Prediction (Classification):** Assign the class label that is most frequent among the k nearest neighbors.
* **Prediction (Regression):** Calculate the average of the values of the k nearest neighbors.



**Examples:**

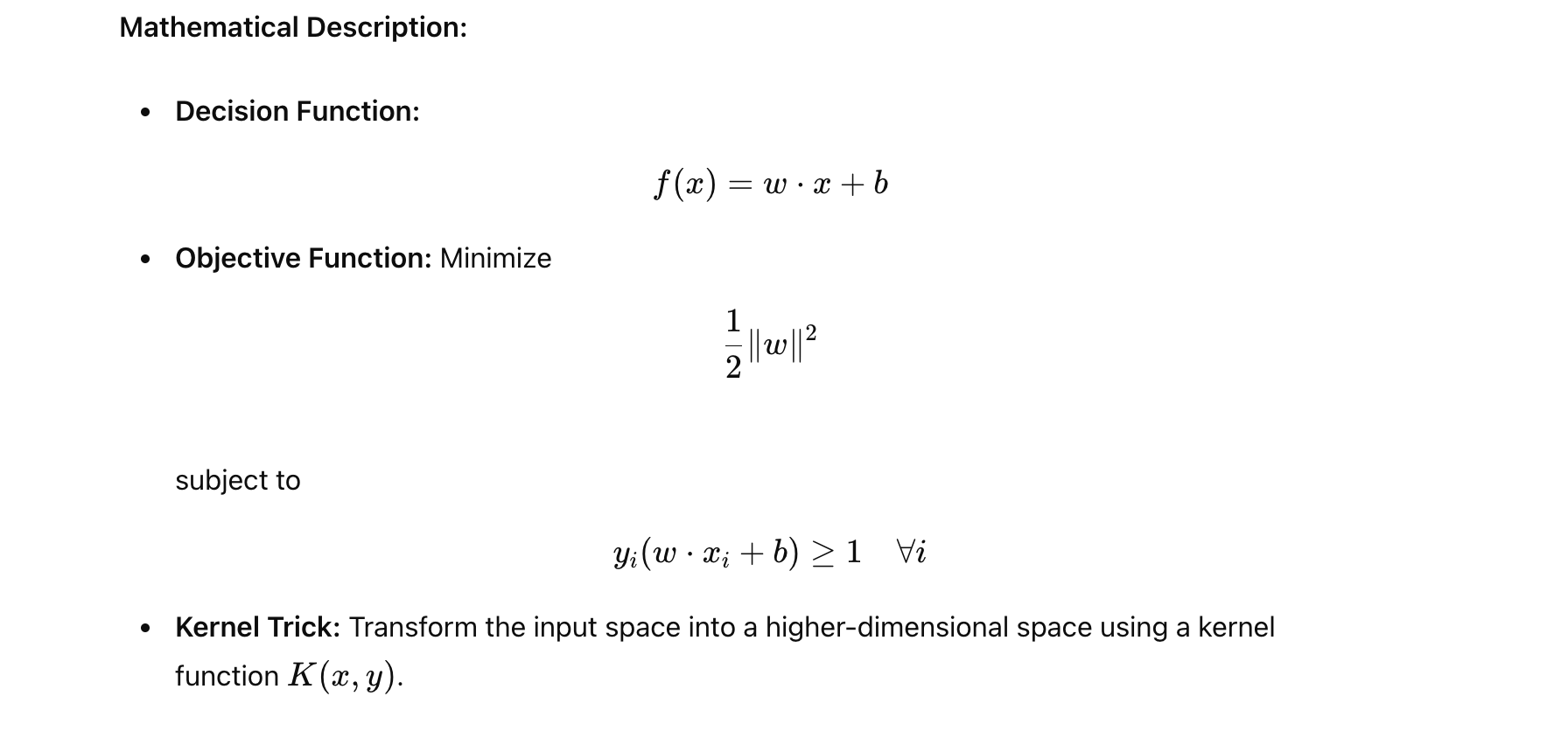
* **Classification:** Identifying the species of an iris flower based on measurements of its petals and sepals using KNN.
* **Regression:** Predicting the price of a house based on the prices of the k most similar houses in terms of features like size and location.

**8. Support Vector Machines (SVM)**

**Theory:** Support Vector Machines (SVM) are powerful, versatile algorithms used for classification and regression. The primary objective of an SVM is to find the optimal hyperplane that maximizes the margin between different classes in the feature space. SVMs can be extended to handle non-linear classification using the kernel trick.

**Mathematical Description:**

* **Decision Function:** f(x)=w⋅x+bf(x) = w \cdot x + bf(x)=w⋅x+b
* **Objective Function:** Minimize 12∥w∥2\frac{1}{2} \|w\|^221​∥w∥2 subject to yi(w⋅xi+b)≥1∀iy\_i (w \cdot x\_i + b) \geq 1 \quad \forall iyi​(w⋅xi​+b)≥1∀i
* **Kernel Trick:** Transform the input space into a higher-dimensional space using a kernel function K(x,y)K(x, y)K(x,y).



**Examples:**

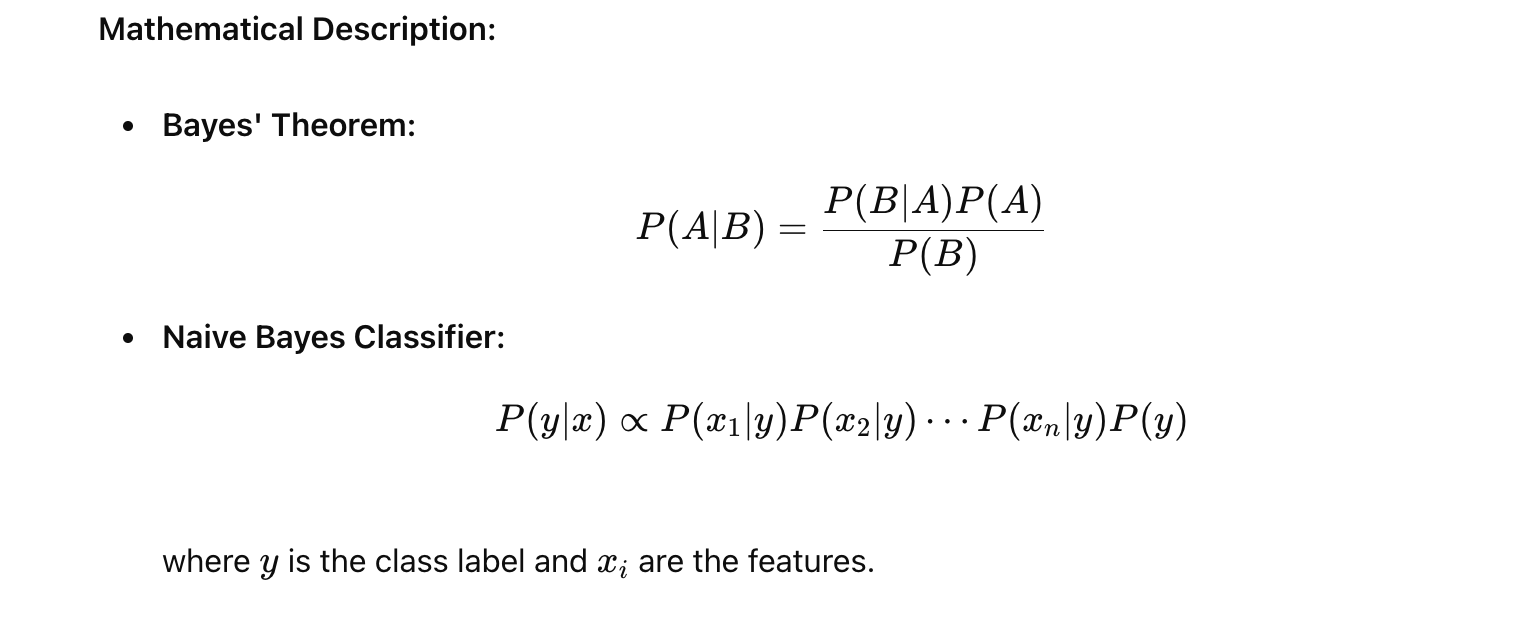
* **Classification:** Classifying images of handwritten digits from the MNIST dataset using a linear or RBF kernel SVM.
* **Regression (SVR):** Predicting housing prices while allowing for some error margin, using Support Vector Regression.

**9. Naive Bayes**

**Theory:** Naive Bayes is a probabilistic classifier based on Bayes' Theorem, assuming that features are conditionally independent given the class label. Despite its simplicity, it performs surprisingly well on many tasks and is particularly effective for large datasets.

**Mathematical Description:**

* **Bayes' Theorem:** P(A∣B)=P(B∣A)P(A)P(B)P(A|B) = \frac{P(B|A) P(A)}{P(B)}P(A∣B)=P(B)P(B∣A)P(A)​
* **Naive Bayes Classifier:** P(y∣x)∝P(x1∣y)P(x2∣y)⋯P(xn∣y)P(y)P(y|x) \propto P(x\_1|y) P(x\_2|y) \cdots P(x\_n|y) P(y)P(y∣x)∝P(x1​∣y)P(x2​∣y)⋯P(xn​∣y)P(y) where yyy is the class label and xix\_ixi​ are the features.



**Examples:**

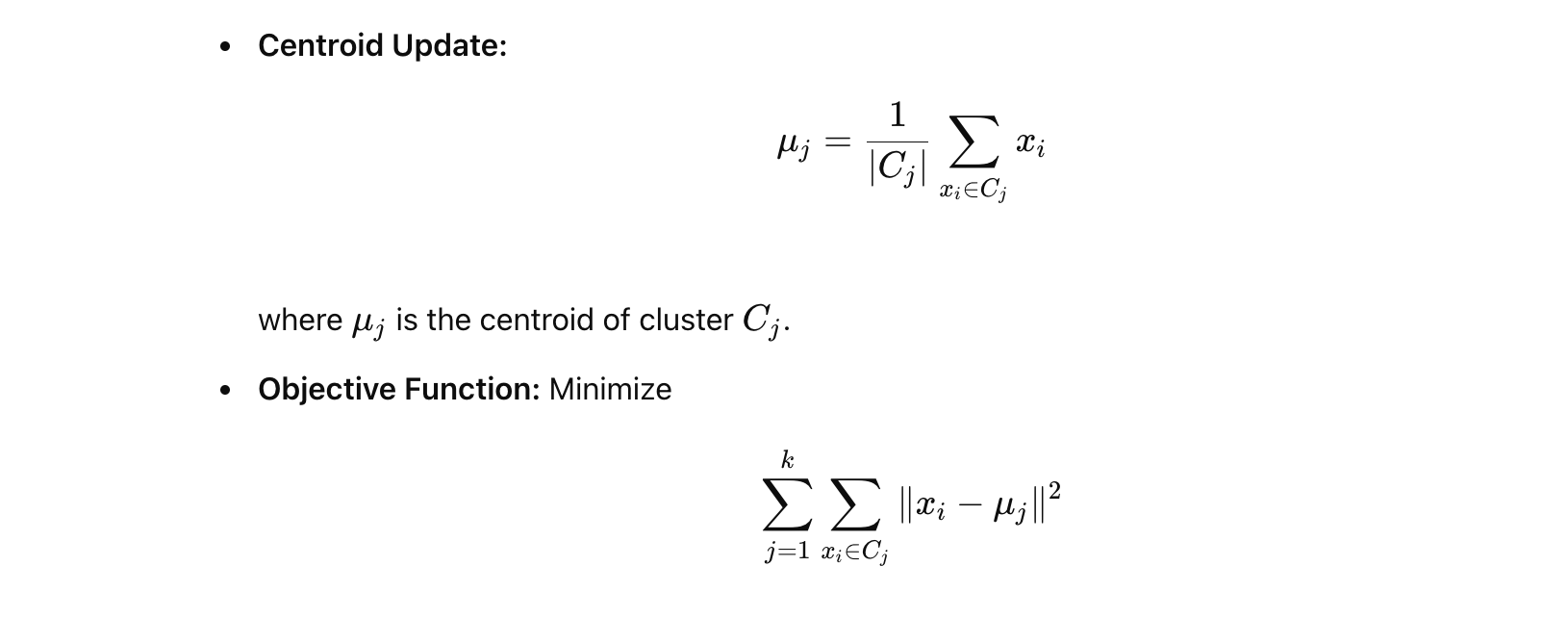
* **Spam Detection:** Classifying emails as spam or not spam based on word frequency.
* **Sentiment Analysis:** Determining the sentiment of a text review (positive or negative) based on the presence of certain words.

**10. Unsupervised Learning, K-Means**

**Theory:** Unsupervised learning involves training algorithms on data without labeled responses to find hidden patterns or intrinsic structures. K-Means is a popular clustering algorithm that partitions data into k distinct clusters based on feature similarity. It iteratively assigns data points to clusters and updates cluster centroids until convergence.

**Mathematical Description:**

* **Centroid Update:** μj=1∣Cj∣∑xi∈Cjxi\mu\_j = \frac{1}{|C\_j|} \sum\_{x\_i \in C\_j} x\_iμj​=∣Cj​∣1​xi​∈Cj​∑​xi​ where μj\mu\_jμj​ is the centroid of cluster CjC\_jCj​.
* **Objective Function:** Minimize ∑j=1k∑xi∈Cj∥xi−μj∥2\sum\_{j=1}^k \sum\_{x\_i \in C\_j} \|x\_i - \mu\_j\|^2j=1∑k​xi​∈Cj​∑​∥xi​−μj​∥2



**Examples:**

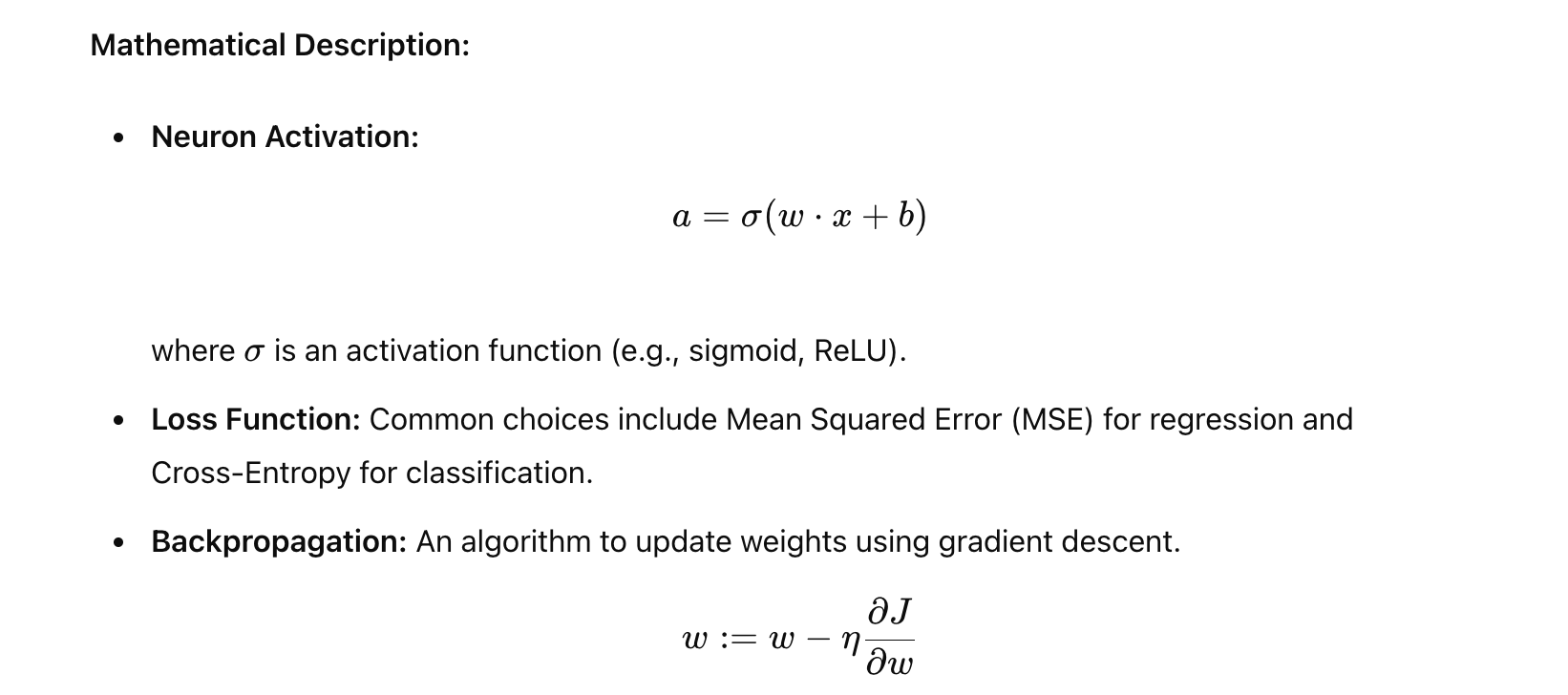
* **Customer Segmentation:** Grouping customers into clusters based on purchasing behavior for targeted marketing.
* **Image Compression:** Reducing the number of colors in an image by clustering similar colors together.

**11. Neural Networks**

**Theory:** Neural Networks (NNs) are computational models inspired by the human brain, consisting of layers of interconnected neurons. They are capable of learning complex patterns and representations from data. Deep learning, a subset of ML, involves neural networks with many layers (deep networks), which have achieved state-of-the-art performance in various domains.

**Mathematical Description:**

* **Neuron Activation:** a=σ(w⋅x+b)a = \sigma(w \cdot x + b)a=σ(w⋅x+b) where σ\sigmaσ is an activation function (e.g., sigmoid, ReLU).
* **Loss Function:** Common choices include Mean Squared Error (MSE) for regression and Cross-Entropy for classification.
* **Backpropagation:** An algorithm to update weights using gradient descent. w:=w−η∂J∂ww := w - \eta \frac{\partial J}{\partial w}w:=w−η∂w∂J​



**Examples:**

* **Image Classification:** Classifying images of handwritten digits (MNIST) using a Convolutional Neural Network (CNN).
* **Speech Recognition:** Transcribing spoken words to text using a Recurrent Neural Network (RNN) or Transformer.