Introduction

Madiba Hudson-Quansah

Contents

Chapter 1

Introduction

1.1 Machine Learning

- Performing a Task
- With Experience
- Improving Performance

1.2 Artificial Intelligence (AI)

Definition 1.2.1: Artificial Intelligence

The science and engineering of making intelligent machines, especially intelligent computer programs.

1.3 Deep Learning vs Machine Learning

1.3.1 Machine Learning

- Subfield of AI focused on algorithms that learn from data.
- Works well with structured data.
- Simpler models.
- Requires manual feature extraction and selection.
- Involves predictive modelling, clustering, and classification.
- Feature extraction and application are done separately.

1.3.2 Deep Learning

- Subfield of ML using neural networks with many layers.
- Works well with large amounts of unstructured data.
- Complex models with multiple layers.
- Automatically extracts features from raw data.
- Involves image and speech recognition, natural language processing, and recommendation systems.
- Feature extraction and application are done together by the neural network.

1.4 Supervised Learning

Definition 1.4.1: Supervised Learning

A subfield of Machine Learning where labelled datasets are used to train algorithms that classify data or predict outcomes.

1.4.1 Terminology

Definition 1.4.2: Feature / Input Feature / Independent Variable / X

A feature is an individual measurable property or characteristic of a phenomenon being observed.

Definition 1.4.3: Label / Dependent Variable / Y

The output / target variable that we are trying to predict.

Definition 1.4.4: Classification

Involves predicting a categorical label.

Definition 1.4.5: Regression

Involves predicting a quantitative continuous label.

1.4.2 Supervised Learning Pipeline

1. Determine the type of training dataset.
2. Gather the labelled training data.
3. Split the training dataset into training dataset, test dataset.
4. Determine the most suitable algorithm for the model.
5. Execute the algorithm on the training dataset.
6. Evaluate the accuracy of the model by providing the test set.

Definition 1.4.6: Independent Identical Distribution (IID)

A set of random variables is independent and identically distributed if each random variable has the same probability distribution as the others and all are mutually independent.

1.4.3 Math

For a model:

$$h(x) = \theta_0 + \theta_1 x$$

Where $h(x)$ is he hypothesis, The θ are our parameters, and x is an input feature.

$$h(x) = \boldsymbol{\theta} \cdot \mathbf{x}$$

Where $x_0 = 1$, where the number of elements in the parameter vector $\boldsymbol{\theta}$ and input feature vector $\mathbf{x}$ is $n + 1$ or

$$h(x) = \sum_{i=0}^{n} \theta_i x_i$$

Where $x_0 = 1$, For multiple input features.

For the training set $\left(X^i, Y^i\right)$, represents the i-th input and the i-th label.

Definition 1.4.7: Gradient Descent

A first-order iterative optimization algorithm for finding the minimum of a function.

1.4.3.0.1 Batch Gradient Descent

θ is chosen such that $h(x) \approx y$ for a training example (x, y). This means minimizing some cost/loss function $L(\theta)$. I.e

$$
\begin{aligned}
h(x) &= \sum_{i=0}^{n} \theta_i x_i \\
h_\theta(x) &= h(x) \\
\text{Let } L(\theta) &= \frac{1}{m} \sum_{i=1}^{m} \left(h_\theta\left(x^i\right) - y^i\right)^2 \\
L(\theta) &= \frac{1}{m} \left(h_\theta(x) - \mathbf{y}\right)^T \left(h_\theta(x) - \mathbf{y}\right) \\
\text{Let } h_\theta(x) &= \mathbf{x}\boldsymbol{\theta} \\
L(\theta) &= \frac{1}{m} (\mathbf{x}\boldsymbol{\theta} - \mathbf{y})^T (\mathbf{x}\boldsymbol{\theta} - \mathbf{y}) \\
\text{Then } \nabla L(\theta) &= 0
\end{aligned}
$$

Or iteratively:

$$
\begin{aligned}
h_\theta(x) &= \sum_{i=0}^{m} \theta_i x_i \\
\theta_i &= \theta_i - \alpha \frac{\partial L(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \\
\text{Until } \frac{\partial L(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} &= 0
\end{aligned}
$$

Where α is the learning rate / step size.

The partial derivative $\frac{\partial L(\theta)}{\partial \theta}$ is found;

$$
\begin{aligned}
\frac{\partial L(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} &= \frac{\partial}{\partial \boldsymbol{\theta}} \left(\frac{1}{2m} \sum_{i=0}^{m} \left(h_\theta \left(x^i \right) - y^i \right)^2 \right) \\
&= \frac{\partial}{\partial \boldsymbol{\theta}} \left(\frac{1}{2m} \left(h_\theta \left(x^i \right) - y^i \right)^2 \right) \\
\\
&= 2 \times \frac{1}{2m} \times \frac{\partial}{\partial \theta} \left(h_\theta (x) - y \right) \left(h_\theta (x) - y \right) \\
&= \frac{1}{m} \times \frac{\partial}{\partial \theta} (\boldsymbol{\theta} \mathbf{x} - \mathbf{y}) (\boldsymbol{\theta} \mathbf{x} - \mathbf{y}) \\
& \text{As } \boldsymbol{\theta} \text{ is a vector of constants its partial derivative in each case is 1} \\
&= \frac{1}{m} (\mathbf{x}) (\boldsymbol{\theta} \mathbf{x} - \mathbf{y}) \\
&= \frac{1}{m} (\boldsymbol{\theta} \mathbf{x} - \mathbf{y}) \mathbf{x} \\
\frac{\partial L(\theta)}{\partial \theta} &= \frac{1}{m} (\boldsymbol{\theta} \mathbf{x} - \mathbf{y}) \mathbf{x}
\end{aligned}
$$

This method is called the batch gradient descent algorithm.

1.4.3.0.2 Stochastic Gradient Descent

Definition 1.4.8: Stochastic

Randomly determined; having a random probability distribution or pattern that may be analyzed statistically but may not be predicted precisely.

The Stochastic Gradient Descent algorithm is a variation of the gradient descent algorithm that updates the weights after each training example. So instead of the equation above, we have:

$$\theta_i = \theta_i - \alpha \left(h_\theta \left(x^i \right) - y^i \right) x^i$$

Where i is the i-th training example. In this method we calculate the gradient of the loss function at that specific parameter-training set and update the parameter accordingly.