Deep-Learning-Algorithmen können auf unüberwachte Lernaufgaben angewendet werden. Dies ist ein wichtiger Vorteil, da nicht beschriftete Daten in größerer Zahl vorliegen als beschriftete Daten.

Unüberwachte Lernalgorithmen nehmen einen Datensatz, der nur Eingaben enthält, und suchen in den Daten nach einer Struktur, z. B. der Gruppierung oder dem Clustering von Datenpunkten. Die Algorithmen lernen daher aus Testdaten, die nicht gekennzeichnet, klassifiziert oder kategorisiert wurden. Anstatt auf Feedback zu reagieren, identifizieren unüberwachte Lernalgorithmen Gemeinsamkeiten in den Daten und reagieren auf der Grundlage des Vorhandenseins oder Fehlens solcher Gemeinsamkeiten in jedem neuen Datensatz.

Beim unüberwachten Lernen werden die Eingabedaten zusammen mit der Kostenfunktion gegeben, einer Funktion der Daten und der Ausgabe des Netzes. Die Kostenfunktion ist abhängig von der Aufgabe und allen a priori Annahmen (den impliziten Eigenschaften des Modells, seinen Parametern und den beobachteten Variablen). Als triviales Beispiel betrachten wir das Modell, bei dem a eine Konstante und die Kosten sind. Die Minimierung dieser Kosten führt zu einem Wert von a, der gleich dem Mittelwert der Daten ist. Die Kostenfunktion kann noch viel komplizierter sein. Ihre Form hängt von der Anwendung ab.

Bei einem unüberwachten Lernen Ansatz ist ein Trainingsdatensatz nur mit Eingabewerten gegeben. Bei dieser Form des maschinellen Lernens versucht das Netzwerke, Strukturen in den gegebenen Eingabedaten zu erkennen. Die Netzwerke lernen daher aus Testdaten, die in keiner Weise kategorisiert wurden. Anstatt auf Feedback zu reagieren, identifizieren unüberwachte Lernverfahren Gemeinsamkeiten in den Daten und reagieren auf der Grundlage des Vorhandenseins oder Fehlens solcher Gemeinsamkeiten in jedem neuen Datensatz. Er gibt verschiedene Methoden des unüberwachten Lernens. Auch hier gibt es die Möglichkeit eine Kostenfunktion C zu definieren, welche den Eingabewert und den vom Netzwerk generierten Output übergeben bekommt. Die Kostenfunktion ist abhängig von der Aufgabe und allen a priori Annahmen für die zu approximierende Funktion. Die Kostenfunktion kann aber auch viel komplizierter sein, da ihre Form von der jeweiligen Problemstellung abhängt. Hier kann man dann wieder den Ansatz aus Gleichung 3 übernehmen und den mittelwert aller Kotenfunktionen ausgewertet für alle trainingsdaten und die dazu vom Netzwerk generierten Ausgabewerte versuchen bezüglich der Gewichte und Biase zu minimieren. Der Ansatz des unüberwachten Lernen bietet den Vorteil, dass bei vielen Approximationsprobleme nur Eingabewerte zur Verfügung stehen.

Zu diesem Zweck wird eine Kostenfunktion 𝐶 der Eingabedaten 𝑥𝑖 und der Ausgabe des Netzes 𝐹(𝑥𝑖,𝑊,𝑏) bestimmt, die von der Aufgabe der zu approximierenden Funktion, wie z. B. der Eingabedomäne, und etwaigen a priori Annahmen, wie den impliziten Eigenschaften oder Merkmalen der zu approximierenden Funktion und ihrer Argumente, abhängig ist.

Man versucht auch, den Mittelwert über alle auf 𝑥𝑖 und 𝐹(𝑥𝑖,𝑊,𝑏) angewendeten Kostenfunktionen unter Berücksichtigung der Gewichte 𝑊 ={𝑊𝑙}𝑙=1,...,𝐿 und der Verzerrungen zu minimieren, was zu dem folgenden Optimierungsproblem führt: