

BATCH LEARNING AND ON-LINE LEARNING

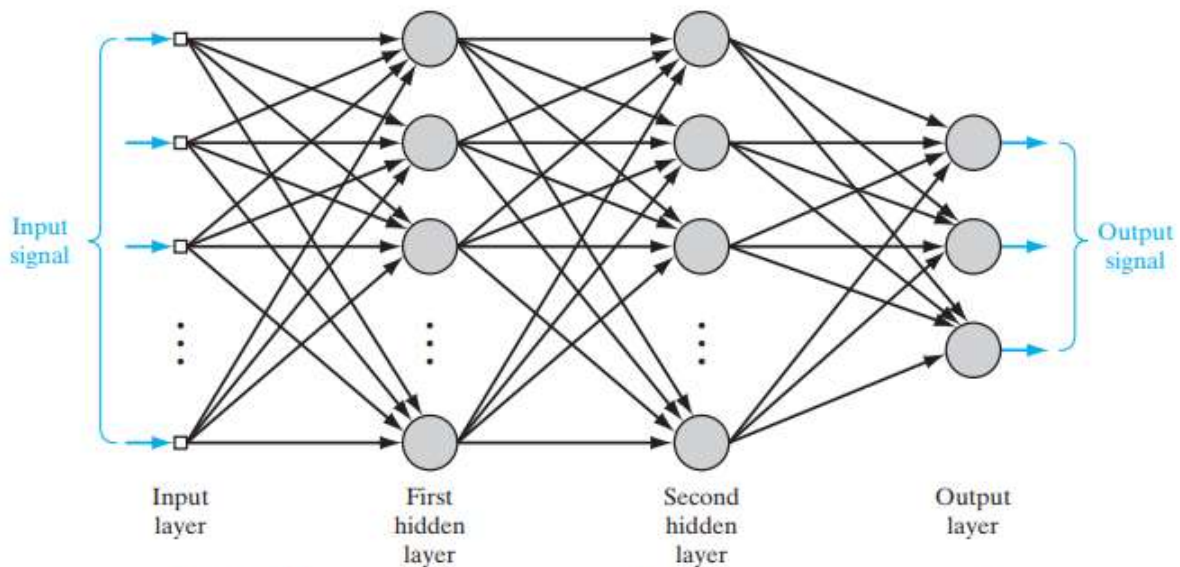


FIGURE 4.1 Architectural graph of a multilayer perceptron with two hidden layers.

Xét một multilayer perceptron như hình trên

Cho bộ dữ liệu training thể hiện như sau (training sample):

$$J = \{(x(n), d(n))\}_{n=1}^N$$

Giả sử:

$y_j(n)$: function signal at the output of neuron j tại output layer với input $x(n)$

$d_j(n)$ là phần tử thứ j desired output vector $d(n)$.

Ta có

- error signal:

$$e_j(n) = d_j(n) - y_j(n)$$

- instantaneous error energy – lượng lỗi tức thời

$$\mathcal{E}_j(n) = \frac{1}{2} e_j^2(n)$$

- Suy ra: total instantaneous error energy of whole network

$$\begin{aligned}\mathcal{E}(n) &= \sum_{j \in C} \mathcal{E}_j(n) \\ &= \frac{1}{2} \sum_{j \in C} e_j^2(n)\end{aligned}$$

Với C : tất cả các neuron tại output layer

- Lại có Training Sample có N ví dụ (số lượng dữ liệu), vậy trung bình lượng lỗi tính được là
(the error energy averaged over the training sample or the empirical risk)

$$\begin{aligned}\mathcal{E}_{av}(N) &= \frac{1}{N} \sum_{n=1}^N \mathcal{E}(n) \\ &= \frac{1}{2N} \sum_{n=1}^N \sum_{j \in C} e_j^2(n)\end{aligned}$$

Tùy thuộc vào cách supervised learning mà chỉ ra quá trình học của multilayer perceptron thực sự được thực hiện

Có 2 method :

1. Batch Learning

Việc điều chỉnh các trọng số w trong multilayer perceptron được thực hiện khi tất cả các example (1 dữ liệu training) trong training samples tạo thành 1 epoch.

Nói cách khác, cost function cho batch learning chính là the average error energy .

$$\begin{aligned}\mathcal{E}_{av}(N) &= \frac{1}{N} \sum_{n=1}^N \mathcal{E}(n) \\ &= \frac{1}{2N} \sum_{n=1}^N \sum_{j \in C} e_j^2(n)\end{aligned}$$

Điều chỉnh các trọng số w dựa trên cơ sở epoch-by-epoch.

Tương ứng, learning curve được vẽ bởi E_{av} so với số epochs.

Trong đó, sau mỗi epoch, bộ training sample phải được trộn ngẫu nhiên.

Với phương pháp tối ưu Gradient Descent sử dụng trong việc thực hiện training, các ưu điểm của batch learning là:

- Ước tính chính xác gradient vector (tức là đạo hàm của cost function E_{av} đối với trọng lượng véc tơ w), do đó đảm bảo, trong điều kiện đơn giản, sự hội tụ của các dốc nhất đến mức local minimum.

- Thực hiện song song quá trình học tập.

Nhược điểm: Khó khăn về mặt yêu cầu lưu trữ.

Batch learning rất phù hợp để giải quyết các vấn đề **hồi quy phi tuyến (nonlinear regression problems)**.

2. On-line Learning

Việc điều chỉnh các trọng số w trong multilayer perceptron được thực hiện trên cơ sở example-by example- example.

Do đó, cost function chính là total instantaneous error energy $e(n)$.

$$\begin{aligned}\mathcal{E}(n) &= \sum_{j \in C} \mathcal{E}_j(n) \\ &= \frac{1}{2} \sum_{j \in C} e_j^2(n)\end{aligned}$$

Xét một epoch của N training sample được sắp xếp theo thứ tự

$$\{x(1), d(1)\}, \{x(2), d(2)\}, \dots, \{x(N), d(N)\}$$

Trong cùng 1 epoch, cặp example $\{x(1), d(1)\}$ được present đầu tiên, và điều chỉnh trọng số w được thực hiện bằng phương pháp tối ưu gradient descent. Tiếp theo cặp example $\{x(2), d(2)\}$ trải qua các việc tương tự cho đến cặp thứ N .

Thủ tục trên chống lại việc học song song của on-line learning.

Learning curve thu được bằng cách vẽ giá trị cuối cùng $e(N)$ so với số lượng epoch được sử dụng trong training.

Cũng như batch learning, sau mỗi epoch, training sample phải được xáo trộn ngẫu nhiên.

Ưu điểm:

- on-line learning làm cho việc tìm kiếm (thực hiện) ngẫu nhiên trong không gian multidimensional weight trong tự nhiên. Sự ngẫu nhiên này có tác dụng mong muốn làm cho nó ít có khả năng cho quá trình học tập bị mắc kẹt ở mức local minimum.
- Trong thực tế, on-line learning đòi hỏi ít lưu trữ hơn nhiều so với batch learning.

- khả năng theo dõi những thay đổi nhỏ trong dữ liệu đào tạo, đặc biệt khi môi trường chịu trách nhiệm tạo dữ liệu là không tĩnh.

Rất phổ biến cho giải quyết các vấn đề **phân loại**, do:

- On-line learning is simple to implement
- Cung cấp các giải pháp hiệu quả để phân loại mô hình quy mô lớn và khó khăn các vấn đề