WEEK13

## **反傳遞演算法**

反向傳播是「誤差反向傳播」的簡稱，是一種與最優化方法（如梯度下降法）結 合使用的，用來訓練人工神經網絡的常見方法。該方法對網絡中所有權重計算損 失函數的梯度。這個梯度會反饋給最優化方法，用來更新權值以最小化損失函 數。

反向傳播要求有對每個輸入值想得到的已知輸出，來計算損失函數梯度。因此， 它通常被認為是一種監督式學習方法，雖然它也用在一些無監督網絡（如自動編 碼器）中。它是多層前饋網絡的Delta規則的推廣，可以用鏈式法則對每層疊 代 計算梯度。反向傳播要求人工神經元（或「節點」）的激勵函數可微。

# PyTorch

**PyTorch是一個**[**開源**](https://zh.wikipedia.org/wiki/%E5%BC%80%E6%BA%90)**的**[**Python**](https://zh.wikipedia.org/wiki/Python)[**機器學習**](https://zh.wikipedia.org/wiki/%E6%9C%BA%E5%99%A8%E5%AD%A6%E4%B9%A0)[**庫**](https://zh.wikipedia.org/wiki/%E5%BA%93)**，基於**[**Torch**](https://zh.wikipedia.org/w/index.php?title=Torch_(%E6%9C%BA%E5%99%A8%E5%AD%A6%E4%B9%A0)&action=edit&redlink=1)[**[3]**](https://zh.wikipedia.org/wiki/PyTorch#cite_note-3)[**[4]**](https://zh.wikipedia.org/wiki/PyTorch#cite_note-4)[**[5]**](https://zh.wikipedia.org/wiki/PyTorch#cite_note-5)**，底層由C++實現，應用於**[**人**](https://zh.wikipedia.org/wiki/%E4%BA%BA%E5%B7%A5%E6%99%BA%E8%83%BD) **工智慧領域，如**[**自然語言處理**](https://zh.wikipedia.org/wiki/%E8%87%AA%E7%84%B6%E8%AF%AD%E8%A8%80%E5%A4%84%E7%90%86)**。它最初由**[**Facebook**](https://zh.wikipedia.org/wiki/Facebook)**的人工智慧研究團隊開發，**[**[7]**](https://zh.wikipedia.org/wiki/PyTorch#cite_note-7)[**[8]**](https://zh.wikipedia.org/wiki/PyTorch#cite_note-8)[**[9]**](https://zh.wikipedia.org/wiki/PyTorch#cite_note-9)**並且** **被用於**[**Uber**](https://zh.wikipedia.org/wiki/%E5%84%AA%E6%AD%A5)**的**[**概率編程**](https://zh.wikipedia.org/wiki/%E6%A6%82%E7%8E%87%E7%BC%96%E7%A8%8B)**軟體Pyro。**

**PyTorch主要有兩大特徵：**

* **類似於**[**NumPy**](https://zh.wikipedia.org/wiki/NumPy)**的**[**張量**](https://zh.wikipedia.org/wiki/%E5%BC%A0%E9%87%8F)**計算，可使用**[**GPU**](https://zh.wikipedia.org/wiki/%E5%9C%96%E5%BD%A2%E8%99%95%E7%90%86%E5%99%A8)**加速；**
* **基於帶**[**自動微分**](https://zh.wikipedia.org/wiki/%E8%87%AA%E5%8A%A8%E5%BE%AE%E5%88%86)**系統**[**[12]**](https://zh.wikipedia.org/wiki/PyTorch#cite_note-12)[**[13]**](https://zh.wikipedia.org/wiki/PyTorch#cite_note-13)**的深度**[**神經網路**](https://zh.wikipedia.org/wiki/%E7%A5%9E%E7%BB%8F%E7%BD%91%E7%BB%9C)**。**

**PyTorch包括torch.nn、torch.optim等子模組。**

參考資料

<https://zh.wikipedia.org/wiki/%E5%8F%8D%E5%90%91%E4%BC%A0%E6%92%AD%E7%AE%97%E6%B3%95>

<https://zh.wikipedia.org/wiki/PyTorch>