Machine Learning (part II)

Representation Learning

Angelo Ciaramella

Introduction

- information processing
  - tasks can be very easy or very difficult depending on how the information is represented
  - what makes one representation better than another?
  - feedforward networks trained by supervised learning as performing a kind of representation learning
    - Hidden layers
    - classes that were not linearly separable in the input features may become linearly separable in the last hidden layer

Unsupervised pretraining

- greedy layer-wise unsupervised pretraining
  - Representation learned for one task can sometimes be useful for another task
  - Each layer is pretrained using unsupervised learning (e.g., RBM)
  - Taking the output of the previous layer and producing as output a new representation of the data
  - optimizes each piece of the solution independently
  - layer-wise because these independent pieces are the layers of the network
  - supervised learning task
    - regularizer and a form of parameter initialization

Protocol

Algorithm 15.1 Greedy layer-wise unsupervised pretraining protocol.
Given the following: Unsupervised feature learning algorithm $\mathcal{L}$, which takes a training set of examples and returns an encoder or feature function f. The raw input data is $\boldsymbol{X}$, with one row per example and $f^{(1)}(\boldsymbol{X})$ is the output of the first stage encoder on $\boldsymbol{X}$ and the dataset used by the second level unsupervised feature learner. In the case where fine-tuning is performed, we use a learner $\mathcal{T}$ which takes an initial function f, input examples $\boldsymbol{X}$ (and in the supervised fine-tuning case, associated targets $\boldsymbol{Y}$), and returns a tuned function. The number of stages is m.

$f \leftarrow$ Identity function
$\tilde{\boldsymbol{X}} = \boldsymbol{X}$
for $k = 1, \ldots, m$ **do**
  $f^{(k)} = \mathcal{L}(\tilde{\boldsymbol{X}})$
  $f \leftarrow f^{(k)} \circ f$
  $\tilde{\boldsymbol{X}} \leftarrow f^{(k)}(\tilde{\boldsymbol{X}})$
end for
if *fine-tuning* ***then***
  $f \leftarrow \mathcal{T}(f, \boldsymbol{X}, \boldsymbol{Y})$
end if
Return f

Transfer learning

- Idea
  - what has been learned in one setting (i.e., distribution P_1) is exploited to improve generalization in another setting (say distribution P_2)
  - For example
    - we may learn about one set of visual categories
      - cats and dogs
    - learn about a different set of visual categories
      - ants and wasps

Domain adaption

- Idea
  - The task remains the same between each setting, but the input distribution is slightly different
  - Very successful for sentiment analysis
- Concept drift
  - form of transfer learning due to gradual changes in the data distribution over time

Extreme forms

- One- shot learning
  - one labeled example of the transfer task
- Zero-shot learning
  - no labeled examples are given at all for the zero-shot learning task

Extreme forms