Representation Learning

Topics in Representation Learning

- Overview

1. Greedy Layer-Wise Unsupervised Pretraining
2. Transfer Learning and Domain Adaptation
3. Semi-supervised Disentangling of Causal Factors
4. Distributed Representation
5. Exponential Gains from depth
6. Providing Clues to Discover Underlying Causes

Overview of Representation Learning

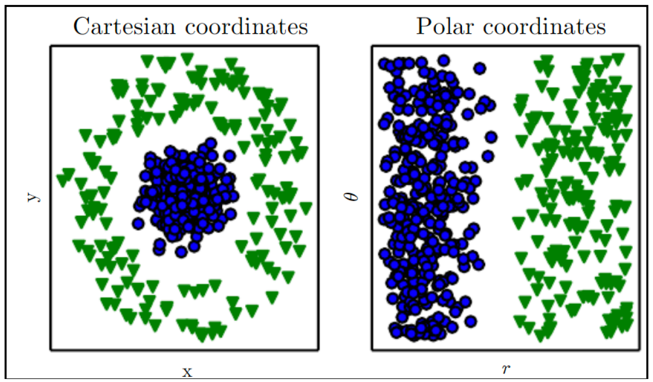
1. We first discuss what it means to learn representations
   - How notion of representation is useful in deep architecture design
2. How learning algorithms share statistical strength across different tasks
   - Including using information from unsupervised tasks to perform supervised tasks
     - Shared representations are useful to handle multiple modalities or domains
       - Or to transfer learned knowledge to tasks for which few or no examples are given but a task representation exists

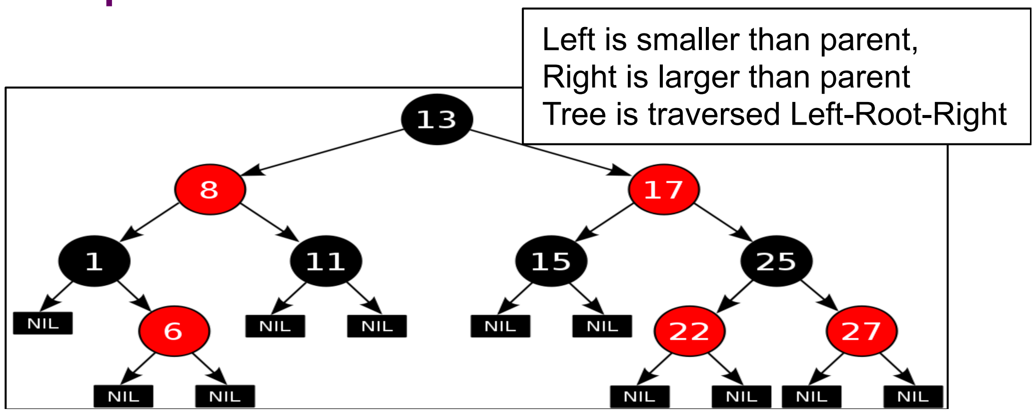
Overview of Representation Learning (Contd.)

3. Reasons for success of representation learning
   - *Starting with*: Theoretical advantages of distributed representations and deep representations
   - *Ending with*: More general idea of underlying assumptions about the data generation process
     - In particular about underlying causes of the observed data

Importance of Representation

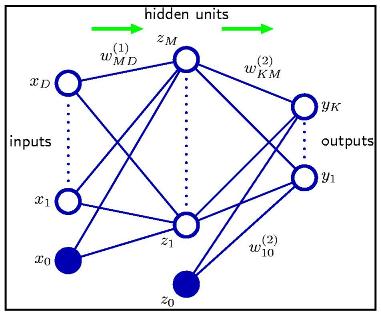
- Information Processing Tasks
  - Easy/difficult: depends on representation
    - Applicable to daily life, CS in general or ML
  - Arabic or Roman numeral
    - Task of $210/6$ versus CCX/VI using long division
      - Most modern people convert from Roman to Arabic
  - Can quantify run times for representations
    - Inserting a no. in a sorted list is $O(n)$
    - But only $O(\log\, n\,)$ if list represented as a red-black tree

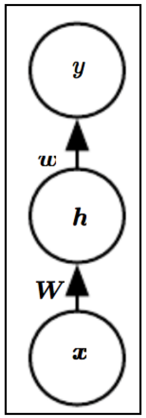




What is a good representation for ML?

- Answer:
  - Representation makes subsequent learning easier
- Choice usually depends on subsequent learning task
- A feed-forward network $f(\boldsymbol{x}) = f^{(3)}[f^{(2)}[f^{(1)}(\boldsymbol{x})]]$ trained by supervised learning performs representation learning





Supervised Feedforward Learning

- Last layer of network is a linear classifier such as softmax regression classifier
  - Rest of network learns representation for classifier
  - Every hidden layer makes the classification easier
    - Ex: classes not linearly separable in input features may become linearly separable in the last layer
  - Last layer could also be another model:
    - Such as a nearest neighbor classifier

How do we specify representation?

1. Supervised learning of feed-forward networks:
   – No imposition of any conditions on learned features
2. Other representation learning algorithms do so
   - Ex: In density estimation, encourage h_i to be independent
3. Unsupervised deep learning algorithms
   – have a main training objective, but like supervised learning they learn a representation as a side effect

- Regardless of how representation was obtained, it can be used for another task
  – Multiple tasks (supervised/unsupervised) can share representation

Trade-off in representation

- Representation learning involves a trade-off between:
  1. Preserving as much information about the input as possible
  2. Attaining nice properties (such as independence)

Semi-supervised Learning

- Representation learning is a way of performing unsupervised and semi-supervised learning
  - Often we have very little labeled data and very large amounts of unlabeled data
    - Training on labeled data results in severe overfitting
    - Semi-supervised learning offers a solution

Human/Animal Learning

- Humans learn with few labeled samples
- We do not yet know how this is possible
- Many factors could explain this:
  - The brain may use very large ensembles of classifiers or Bayesian inference techniques
- One hypothesis is that the brain leverages unsupervised or semi-supervised learning
  - There are many ways to leverage unlabeled data
  - We focus on hypothesis that unlabeled data can be used to learn a good representation

Unsupervised Learning and Deep Learning

- Unsupervised learning revived deep neural networks
  - Enabling training a deep supervised network without specializations such as convolution or recurrence
- Canonical example of a representation learned for one task can be useful for another task
  - First task: unsupervised learning (trying to capture the shape of a distribution)
  - Other task: supervised learning with the same input domain