Representation Learning

강남웅

Content

- Representation Learning
- Unsupervised Pretraining
  - Greedy Layer-Wise Unsupervised Pretraining
- Transfer Learning & Domain Adaptation
- Disentangling of Causal Factors
- Distributed Representation
- Clues of Underlying Causes

Before start…

- Which representation would be better to use/read?
  - 210 * 6 vs CCX * VI
  - 210 vs CCX
  - 6 vs VI
  - former expression (Arabic numeral representation) would be much better than latter expression (Roman numeral representation)

Representation Learning

- Representation
  - how the information/data is expressed

- Good representation
  - Representation which makes learning task easier
  - Depends on the task
    - Classification
      - given data, linearly separable representation would be one of good representation
    - Probabilistic model
      - latent vectors are independent

Representation Learning

- Main Hypothesis
  - Unlabeled data can be used to learn a good representation

Unsupervised Pretraining

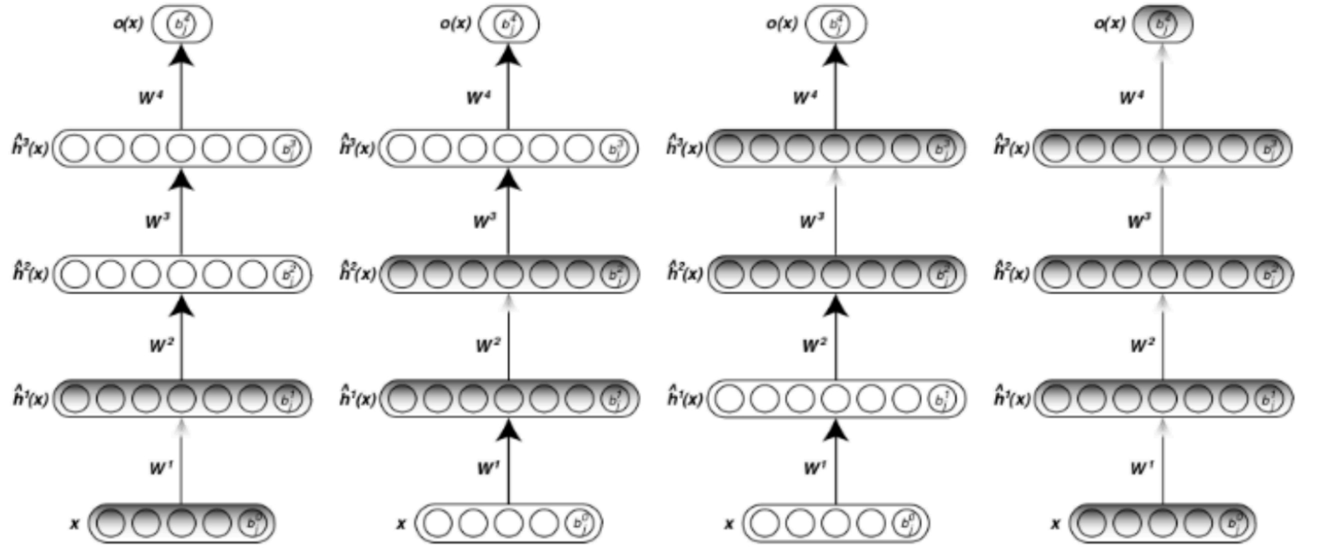
- Motivation
  - Given Neural Network, we initialize parameters randomly
  - Before training, would it be more efficient if we initialize parameters with given data?

Unsupervised Pretraining

- As representation learning
  - Representation learned for **one task** can sometimes be useful for **another task**
  - **one task** : pretraining weight matrix ← unsupervised learning
  - **another task** : training neural network ← supervised learning

Greedy Layer-Wise Unsupervised Pretraining

- Algorithm
  - Given network with at least one hidden layer, pre-training each layer using output of previous layer by fixing other parameters



(a) First hidden layer pre-training (b) Second hidden layer pre-training (c) Third hidden layer pre-training (d) Fine-tuning of whole network

Greedy Layer-Wise Unsupervised Pretraining

- **Greedy**
  → Optimize each piece of the solution independently
- **Layer-Wise**
  → independent pieces are the layer of the network
- **Unsupervised**
  → trained with an unsupervised representation learning algorithm
- **Pretraining**
  → step before a joint training algorithm is applied to fine-tune

Greedy Layer-Wise Unsupervised Pretraining

- Effect
  - Regularizer (In supervised learning context)
  - A form a parameter initialization

Unsupervised pretraining

- Good or Bad?
  - Substantial improvements in test error for classification tasks
  - On many other tasks, however, unsupervised pretraining either does not confer a benefit or even causes noticeable harm
  - Pretraining was slightly harmful, but for many tasks was significantly helpful

  → unsupervised pretraining is *sometimes* helpful

Unsupervised pretraining

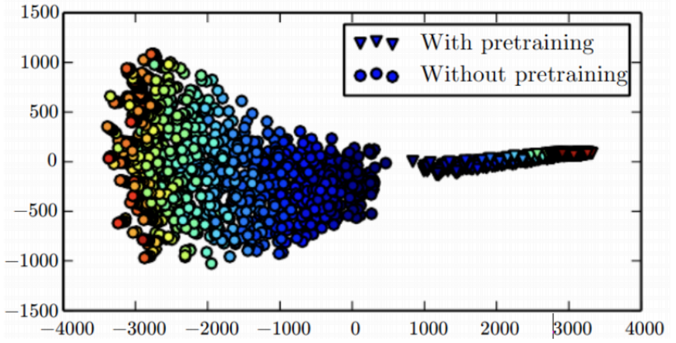
- Basic concept
  - Choice of initial parameters can have a significant regularizing effect on the model
    - Initialize model in a location where that would otherwise inaccessible
      - where the cost function varies much
      - areas where the Hessian matrix is so poorly conditioned that gradient descent methods must use very small steps
  - Learning about the input distribution can help with learning about the mapping from input to output
    - If we think input as poor representation, unsupervised pretraining might give better representation

Unsupervised pretraining

- Advantage
  - (As a regularizer) Target function is extremely complicated
    - Other regularizer (weight decay …) reinforces to view target function to be simple

  - Improvement of training error and test error
    - Since it could lead model to inaccessible region, it sometimes improve errors

  - Consistently halt in the same region of function space
    - Reduces variance of estimation process
    - Without pretraining, it consistently halt in different region



Unsupervised pretraining

- Disadvantage
  - Does not offer clear way to adjust strength of regularization
    - If we use unsupervised and supervised learning simultaneously, we can determine how strongly unsupervised objective will regularize supervised model with one parameter
  - Two separate training phases
    - Separate phase → Long delay between updates of second phase

Unsupervised pretraining

- Conclusion
  - **When to use**
    - **variation of test/train error is large**
  - **Largely Abandoned**
    - **other techniques outperforms (ex. batch normalization, …)**

Transfer Learning

- Concept
  - Using learned representation in one setting(task) will improve generalization in another setting
    - Input is same but target may be different

  - Factors explaining in one setting are relevant to factors of another setting

- Example
  - Model trained for classification of cats and dogs
    → Model for classification of ants and wasps

One-shot Learning

- Concept
  - During transfer learning stage, only **one** labeled example is given
  - Inferring data that cluster around the same point has same label

- Example
  - Objective : Face recognition of each member in company
  - Datasets : one photo for each member
  - Method : Using Transfer Learning → Human face recognition model

Zero-shot Learning

- Concept
  - During transfer learning stage, **no** labeled example is given
  - Inferring data that cluster around the same point has same label

- Example
  - Objective : Finding zebra in given picture
  - Datasets : some pictures with zebra and some are not
  - Method : Using Transfer Learning → Horse recognition model

Domain Adaptation

- Concept
  - Using learned representation in one setting(task) will improve generalization in another setting
  - Input distribution is slightly different
- Example
  - Analyzing customer reviews of books
    → Analyzing review comments on videos

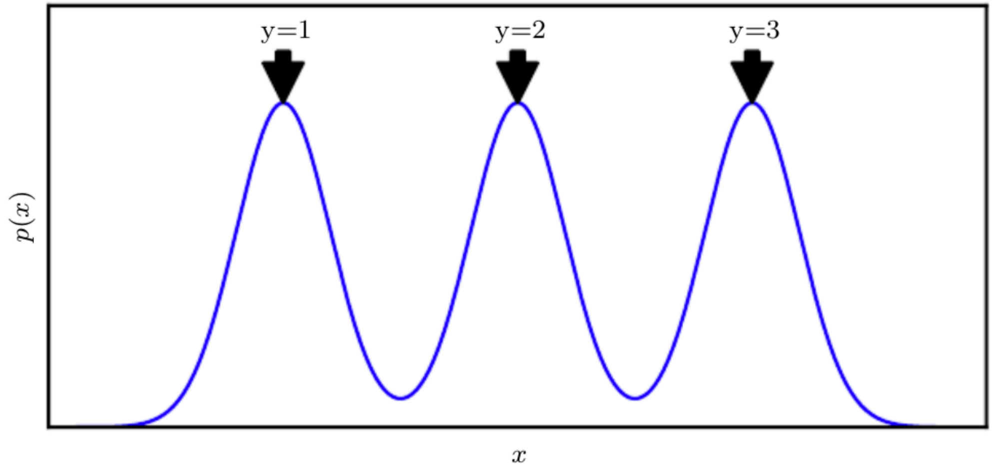
Disentangling of Causal Factors

- Concept
  - Given Data x with probability p(x), we want to find good representation of x
  - We want to distinguish each data by underlying causes
    - Underlying Causal Factors h
    - Among underlying factors, important factor y
    - If y is important factor, computing $p(y|x)$ would be good representation

Disentangling of Causal Factors

- Example Situation that fails to learn factors
  - Our objective is to learn salient factor by $f(x) = E[y|x]$
  - For given data x, $p(x)$ is uniformly distributed
  - Training x gives no information about $p(y|x)$

- Example Situation that succeed to learn factors
  - x arises from a mixture by value of y



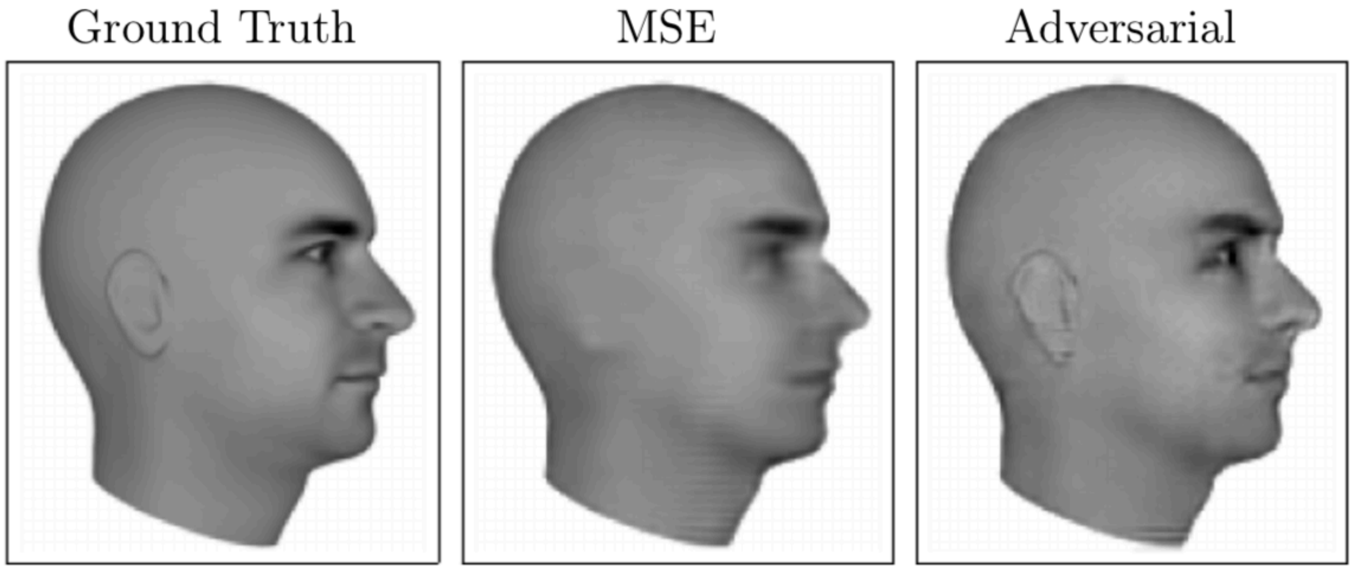
Disentangling of Causal Factors

- In practice, dimension of underlying causal factors is large

- Possible solution
  - Brute force solution
    - learn representation that captures all the reasonably salient generative factors
    - disentangles them from each other
  - Using supervised learning simultaneously
  - Use larger representation (larger dimension of latent space)

Disentangling of Causal Factors

- Salience (salient factor y)
  - Many possible definition/explanation for salience of a factor
  - Example of Learning to generate human head image
    - View of MSE
      - Salience would be caused by extreme difference in brightness
    - View of GAN
      - Salience would be highly recognizable pattern → Ear, Eye …



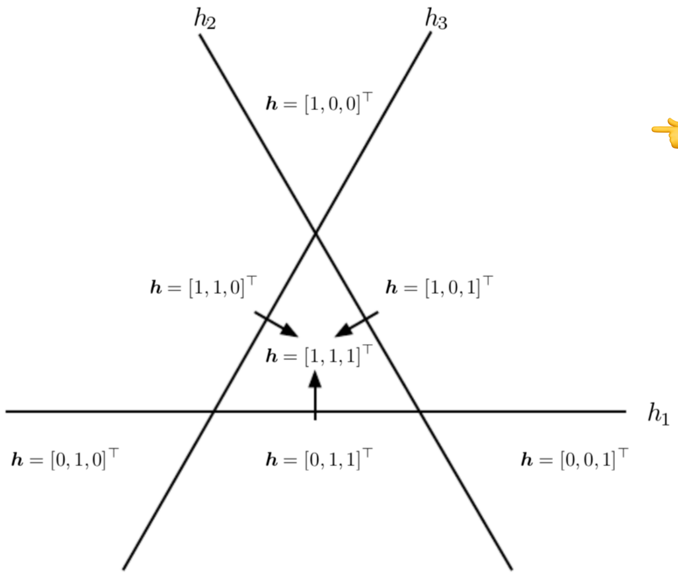
Distributed Representation

- Distributed representations
  - representations composed of many elements that can be set **separately** from each other

- Symbolic representation
  - the input is associated with **a single** symbol or a category
  - also known as one-hot representation
  - Ex. k-means clustering, decision tree, …
    - Based on smoothness assumption
      - $\text{if } \mathrm{u} \approx v \; then \; f(u) \approx f(v)$
      - $\rightarrow \; f(x+\epsilon) \approx f(x)$

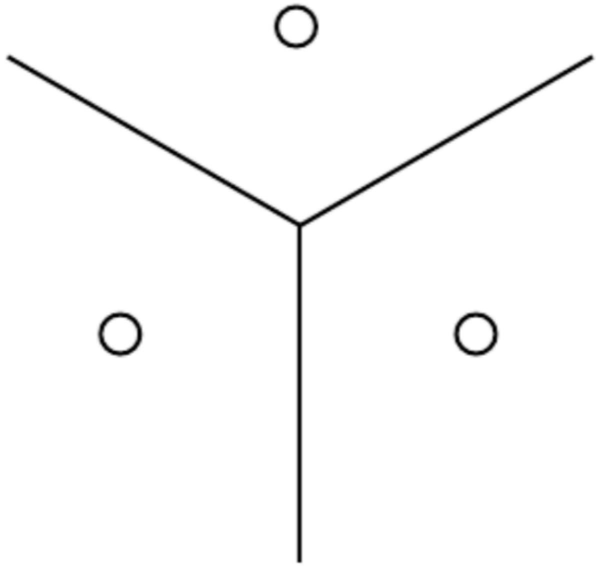
Distributed Representation

- Advantage
  - With only $O(nd)$ parameters, we can specify $O(n^d)$ regions(features)
    - $n \text{ hyperplanes in } R^d$
      - distinguish up to $\sum_{j=0}^{d} \binom{n}{j} = O(n^d)$ regions
      - if symbolic representation, $O(n^d)$ parameters are required



👈 Distributed Representation

Symbolic Representation 👉
(1 symbol for 1 region)



Distributed Representation

- Why distributed representation generalize well?
  - Able to distinctly encode enormous regions with relatively small amount of parameters
  - However, capacity of representation is limited
- It is experimentally validated

Distributed Representation

- Example : Distributed representation of generative model
  - Each picture is represented by
  - Male / Female

    Glasses

  - Man with glasses – man without glasses
    + women = women with glasses
    (vector calcuation)

Clues to Discover Underlying Causes

- Natural clustering
  - Assume that each connected manifold input space may be assigned to a single class
  - disconnected manifolds, but the class remains constant within each one of these

- Simplicity of Factor Dependencies
  - In good high-level representations, the factors are related to each other through simple dependencies
  - linear dependencies or those captured by a shallow autoencoder are also reasonable assumptions