Deep Learning

AlejandroMllo

This document serves as a very brief summary of the topics covered in each chapter of the book Deep Learning [1].

Disclaimer: This document is completely extracted from [1], the author does not attribute any ownership over the material.

15 Representation Learning

- 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.
- Generally speaking, a good representation is one that makes a subsequent learning task easier.
- Representation learning provides one way to perform unsupervised and semi-supervised learning.
- Greedy layer-wise unsupervised pretraining: it relies on a single-layer representation learning algorithm, a single-layer autoencoder, a sparse coding model, or another model that learns latent representations. Each layer is pretrained using unsupervised learning, taking the output of the previous layer and producing as output a new representation of the data, whose distribution is hopefully simpler. See algorithm 15.1.
  - It optimizes each piece of the solution independently.
  - It is expected to be more effective when the initial representation is poor.
  - It is likely to be most useful when the function to be learned is extremely complicated.
- Unsupervised pretraining combines two different ideas:
  1. The choice of initial parameters for a deep NN can have a significant regularizing effect on the model.
  2. Learning about the input distribution can help with learning about the mappings from inputs to outputs.
- NN that receive unsupervised pretraining consistently halt in the same region of function space.
- Deep learning techniques based on supervised learning, regularized with dropout or batch normalization, are able to achieve human-level performance on many tasks, but only with extremely large labeled datasets.
- Transfer learning and domain adaptation refer to the situation where what has been learned in one setting is exploited to improve generalization in another setting.
- In transfer learning, the learner must perform two or more different tasks, but it is assumed that many of the factors that explain the variations in distribution P_1 are relevant to the variations that need to be captured for learning P_2.
- In domain adaptation, the task (and the optimal input-to-output mapping) remains the same between each setting, but the input distribution is slightly different.
- Concept drift: form of transfer learning due to gradual changes in the data distribution over time.
- Two extreme forms of transfer learning are:
  - One-shot learning: only one labeled example of the transfer task is given. The representation learns to cleanly separate the underlying classes during the first stage.
  - Zero-shot(data) learning: no labeled examples are given.
- Zero-data learning and zero-shot learning are only possible because additional information has been exploited during training.
- In a zero-data learning scenario, the model is trained to estimate the conditional distribution $p(\boldsymbol{y}|\boldsymbol{x}, T)$, where T is a description (in a way that allows some sort of generalization) of the task that the model is expected to perform.

- Multimodal learning: captures a representation in one modality, a representation in the other, and the relationship (in general a joint distribution) between pairs $(\boldsymbol{x}, \boldsymbol{y})$ consisting of one observation $\boldsymbol{x}$ in one modality and another observation $\boldsymbol{y}$ in the other modality.
- An emerging strategy for unsupervised learning is to modify the definition of which underlying causes are most salient.
- Generative adversarial networks: a generative model is trained to fool a feedforward classifier. The feedforward classifier attempts to recognize all samples from the generative model as being fake and all samples from the training set as being real. In this framework, any structured pattern that the feedforward network can recognize is highly salient.
- A benefit of learning the underlying causal factors, is that if the true generative process has $\mathbf{x}$ as an effect and $\mathbf{y}$ as a cause, then modeling $p(\mathbf{x} \mid \mathbf{y})$ is robust to changes in $p(\mathbf{y})$.
- Distributed representation of concepts: representations composed of many elements that can be set separately from each other. They can use n features with k values to describe k^n different concepts.
- An ideal representation is one that disentangles the underlying causal factors of variation that generated the data, especially those that are relevant to the specific application.
- An adequate regularizer might help the learning algorithm discover features that correspond to underlying factors.
- Some generic regularization strategies:
  - Smoothness: this is the assumption that $f(\boldsymbol{x} + \epsilon \boldsymbol{d}) \approx f(\boldsymbol{x})$ for unit $\boldsymbol{d}$ and small ϵ. This idea is insufficient to overcome the curse of dimensionality.
  - Linearity: the learning algorithm assumes that the relationships between some variables are linear.
  - Multiple explanatory factors: the learning algorithm assumes that the data is generated by multiple underlying explanatory factors, and that most tasks can be solved easily given the state of each of these factors.
  - Causal factors: the model is constructed in such a way that is treats the factors of variation described by the learned representation $\boldsymbol{h}$ as the causes of the observed data $\boldsymbol{x}$, and not vice versa.
  - Depth or a hierarchical organization of explanatory factors: the use of a deep architecture expresses the belief that the task should be accomplished via a multistep program, with each step referring back to the output of the processing accomplished via previous steps.
  - Shared factors across tasks: when many tasks corresponding to different y_i variables sharing the same input $\mathbf{x}$, or when each task is associated with a subset or a function $f^{(i)}(\mathbf{x})$ of a global input $\mathbf{x}$, the assumption is that each y_i is associated with a different subset from a common pool of relevant factors $\mathbf{h}$.
  - Manifolds: probability mass concentrates, and the regions in which it concentrates are locally connected and occupy a tiny volume.
  - Natural clustering: the learning algorithm assume that each connected manifold in the input space may be assigned to a single class.
  - Temporal and spatial coherence: the algorithm assumes that the most important explanatory factors change slowly over time, or at least that it is easier to predict the true underlying explanatory factors than to predict raw observations.
  - Sparsity: most features should presumably not be relevant to describing most inputs. It is therefore reasonable to impose a prior that any feature that can be interpreted as ”present” or ”absent” should be absent most of the time.
  - Simplicity of factor dependencies: in good high-level representations, the factors are related to each other through simple dependencies.

References

[1] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. `http://www.deeplearningbook.org`.