Paper summary: Representation learning: A review and new perspectives

May 28, 2022

1 What makes aa representation good

1. *smoothness*: f s.t. $x \approx y$ implies $f(x) \approx f(y)$
2. *multiple explanatory factors* a.k.a. disentangling features
3. *semi-supervised learning*: for input Z and target Y, learning $P(X)$ helps learning $P(Y|X)$ because features of X help explain Y
4. *shared factors across tasks*: like previous point, but also works for different Ys
5. *manifolds*: probability mass concentrates in regions with much smaller dimensionality than data itself
6. *natural clustering*: different values of categorical variables are associated with separate manifolds.
7. *temporal and spatial coherence*: consecutive or spatially nearby observations thend to be associated with the same value of relevant categorical concepts or result in small surface move on the surface of the manifold
8. *sparsity*: could mean often many features are 0. could also be that the features are insensitive to small changes in x
9. *simplicity of factor dependecies*: ideally factors are related to each other linearly, or otherwise simply

2 Probabilistic models

From the probabilistic modeling perspective, feature learning can be interpreted as an attempt to recover a parsimonious set of latent random variables that describe a distribution over the observed data. $p(x, h)$ is the probabilistic model over the joint space of latent variables h and observed data x. Feature values are then the result of an inference process to determine the probability distribution of the latent variables given the data, i.e. $p(h|x)$, a.k.a posterior probability. Learning is the finding the parameters that (locally) maximize the regularized likelihood of the training data.

2.1 Directed graphical models

Directed latent factor models separately parametrize $p(x|h)$ and the prior $p(h)$ to construct $p(x,h) = p(x|h)p(h)$. They can explain away: a priori independent causes of an event can become nonindependent given the observation of the event. Can conceive them as cause models, where h activations cause the observed x, making h nonindependent. This makes recovering the posterior $p(h|x)$ intractable.

NOTE There's a lot more to this. You are well advised to learn it. However, it is not of immediate value so it will be postponed to a later time.

3 Directly learning a parametric map from input to representation

The posterior distribution becomes complicated quickly. Thus approximate inference becomes necessary, which is not ideal. Also, depending on the problem, one needs to derive feature vectors from the distribution. If we want deterministic feature values in the end, we might as well go ahead and use a nonprobabilistic feature learning paradigm.

3.1 Autoencoders

$$h^{(t)} = f_\theta(x^{(t)}) \tag{1}$$

There's also the reconstruction $r = g_\theta(h)$, used for the reconstruction error $L(x,r)$ over the training examples. Autoencoder training boils down to finding θ which minimizes:

$$\mathcal{J}_{AE}(\theta) = \sum_t L(x^{(t)}, g_\theta(f_\theta(x^{(t)}))) \tag{2}$$

One can tie the weights between the encoder and the decoder (i.e. make the same ones, just reversed).

3.2 Regularized Autoencoders

AE with regularization, i.e. a sparsity penalty (or something).

3.3 Denoising Autoencoders (DAE)

Train the AE to denoise: input is original data example + noise, loss is calculated against the original data example without the added noise.

3.4 Contractive AE (CAE)

Add an analytic *contractive penalty* to AE. This is the Frobenius norm of the encoder's Jacobian and results in penalizing the *sensitivity* of learned features to infinitesimal input variations. Let $J(x) = \frac{\partial f_\theta}{\partial x}(x)$ be the Jacobian of the encoder at x. Then the training objective is:

$$\mathcal{J}_{CAE} = \sum_t L(x^{(t)}, g_\theta(f_{\theta(x^{(t)})}) + \lambda ||J(x^{(t)})||_F^2) \tag{3}$$

If you don't want just robustness to infinitesimal input variations, add a term which encourages $J(x)$ and $J(x + \epsilon)$ to be close:

$$\mathcal{J}_{CAE+H} = \sum_t L(x^{(t)}, g_\theta(f_{\theta(x^{(t)})}) + \lambda ||J(x^{(t)})||_F^2) + \gamma \mathbb{E}\left[||J(x) - J(x+\epsilon)||_F^2\right] \tag{4}$$

where $\epsilon \sim \mathcal{N}(0, \sigma I)$ and γ is a hyperparameter.

4 MORE

There's more stuff here and it all seems very interesting. Pls give it a read once there's time.