

Chapter 13

Linear Factor Models

Many of the research frontiers in deep learning involve building a probabilistic model of the input, $p_{\text{model}}(\mathbf{x})$. Such a model can, in principle, use probabilistic inference to predict any of the variables in its environment given any of the other variables. Many of these models also have latent variables \mathbf{h} , with $p_{\text{model}}(\mathbf{x}) = \mathbb{E}_{\mathbf{h}} p_{\text{model}}(\mathbf{x} | \mathbf{h})$. These latent variables provide another means of representing the data. Distributed representations based on latent variables can obtain all of the advantages of representation learning that we have seen with deep feedforward and recurrent networks.

In this chapter, we describe some of the simplest probabilistic models with latent variables: linear factor models. These models are sometimes used as building blocks of mixture models (Hinton *et al.*, 1995a; Ghahramani and Hinton, 1996; Roweis *et al.*, 2002) or larger, deep probabilistic models (Tang *et al.*, 2012). They also show many of the basic approaches necessary to build generative models that the more advanced deep models will extend further.

A linear factor model is defined by the use of a stochastic, linear decoder function that generates \mathbf{x} by adding noise to a linear transformation of \mathbf{h} .

These models are interesting because they allow us to discover explanatory factors that have a simple joint distribution. The simplicity of using a linear decoder made these models some of the first latent variable models to be extensively studied.

A linear factor model describes the data generation process as follows. First, we sample the explanatory factors \mathbf{h} from a distribution

$$\mathbf{h} \sim p(\mathbf{h}), \tag{13.1}$$

where $p(\mathbf{h})$ is a factorial distribution, with $p(\mathbf{h}) = \prod_i p(h_i)$, so that it is easy to