

Autoencoders take this idea further and aim to learn the structure of the manifold.

To understand how autoencoders do this, we must present some important characteristics of manifolds.

An important characterization of a manifold is the set of its **tangent planes**. At a point  $\mathbf{x}$  on a  $d$ -dimensional manifold, the tangent plane is given by  $d$  basis vectors that span the local directions of variation allowed on the manifold. As illustrated in figure 14.6, these local directions specify how one can change  $\mathbf{x}$  infinitesimally while staying on the manifold.

All autoencoder training procedures involve a compromise between two forces:

1. Learning a representation  $\mathbf{h}$  of a training example  $\mathbf{x}$  such that  $\mathbf{x}$  can be approximately recovered from  $\mathbf{h}$  through a decoder. The fact that  $\mathbf{x}$  is drawn from the training data is crucial, because it means the autoencoder need not successfully reconstruct inputs that are not probable under the data generating distribution.
2. Satisfying the constraint or regularization penalty. This can be an architectural constraint that limits the capacity of the autoencoder, or it can be a regularization term added to the reconstruction cost. These techniques generally prefer solutions that are less sensitive to the input.

Clearly, neither force alone would be useful—copying the input to the output is not useful on its own, nor is ignoring the input. Instead, the two forces together are useful because they force the hidden representation to capture information about the structure of the data generating distribution. The important principle is that the autoencoder can afford to represent *only the variations that are needed to reconstruct training examples*. If the data generating distribution concentrates near a low-dimensional manifold, this yields representations that implicitly capture a local coordinate system for this manifold: only the variations tangent to the manifold around  $\mathbf{x}$  need to correspond to changes in  $\mathbf{h} = f(\mathbf{x})$ . Hence the encoder learns a mapping from the input space  $\mathbf{x}$  to a representation space, a mapping that is only sensitive to changes along the manifold directions, but that is insensitive to changes orthogonal to the manifold.

A one-dimensional example is illustrated in figure 14.7, showing that, by making the reconstruction function insensitive to perturbations of the input around the data points, we cause the autoencoder to recover the manifold structure.

To understand why autoencoders are useful for manifold learning, it is instructive to compare them to other approaches. What is most commonly learned to characterize a manifold is a **representation** of the data points on (or near)