Chapter 9

Representations for machine learning

At first, it might seem that the applicability of linear regression and classification to real-life problems is greatly limited. After all, it is not clear whether it is realistic (most of the time) to assume that the target variable is a linear combination of features. Fortunately, the applicability of linear regression is broader than originally thought. The main idea is to apply a non-linear transformation to the data matrix $\mathbf{x}$ prior to the fitting step, which then enables a non-linear fit. Obtaining such a useful feature representation is a central problem in machine learning.

We will first examine fixed representations for linear regression: polynomial curve fitting and radial basis function (RBF) networks. Then, we will discuss learning representations.

9.1 Radial basis function networks and kernel representations

The idea of radial basis function (RBF) networks is a natural generalization of the polynomial curve fitting and approaches from the previous Section. Given data set $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$, we start by picking p points to serve as the "centers" in the input space $\mathcal{X}$. We denote those centers as $\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_p$. Usually, these can be selected from $\mathcal{D}$ or computed using some clustering technique (e.g. the EM algorithm, K-means).

When the clusters are determined using a Gaussian mixture model, the basis functions can be selected as

$$\phi_j(\mathbf{x}) \quad = \quad e^{-\frac{1}{2}(\mathbf{x}-\mathbf{c}_j)^T \Sigma_j^{-1} (\mathbf{x}-\mathbf{c}_j)},$$

where the cluster centers and the covariance matrix are found during clustering. When K-means or other clustering is used, we can use

$$\phi_j(\mathbf{x}) = e^{-\frac{\left\|\mathbf{x}-\mathbf{c}_j\right\|^2}{2\sigma_j^2}},$$

where σ_j's can be separately optimized; e.g. using a validation set. In the context of multidimensional transformations from $\mathbf{x}$ to $\mathbf{\Phi}$, the basis functions can also be referred to as *kernel functions*, i.e. $\phi_j(\mathbf{x}) = k_j(\mathbf{x}, \mathbf{c}_j)$. Matrix

$$\mathbf{\Phi} = \begin{bmatrix} \phi_0(\mathbf{x}_1) & \phi_1(\mathbf{x}_1) & \cdots & \phi_p(\mathbf{x}_1) \\ \phi_0(\mathbf{x}_2) & \phi_1(\mathbf{x}_2) & & \\ \vdots & & \ddots & \\ \phi_0(\mathbf{x}_n) & & & \phi_p(\mathbf{x}_n) \end{bmatrix}$$

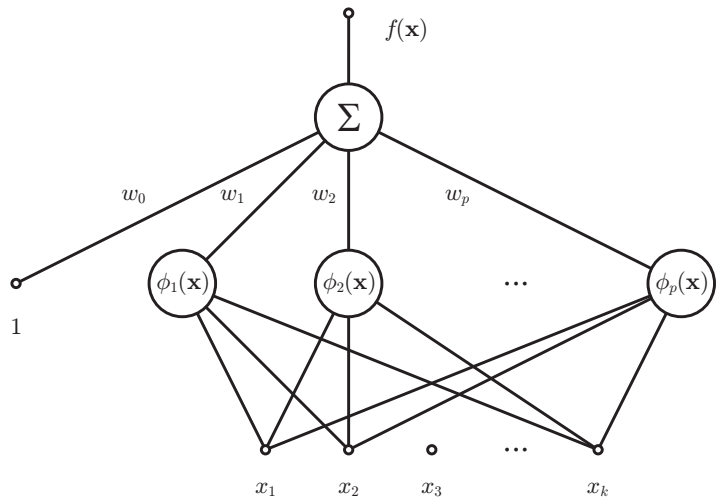

Figure 9.1: Radial basis function network.

is now used as a new data matrix. For a given input $\mathbf{x}$, the prediction of the target y will be calculated as

$$\begin{aligned} f(\mathbf{x}) &= w_0 + \sum_{j=1}^{p} w_j \phi_j(\mathbf{x}) \\ &= \sum_{j=0}^{p} w_j \phi_j(\mathbf{x}) \end{aligned}$$

where $\phi_0(\mathbf{x}) = 1$ and $\mathbf{w}$ is to be found. It can be proved that with a sufficiently large number of radial basis functions we can accurately approximate any function. As seen in Figure 9.1, we can think of RBFs as neural networks.

RBF networks and kernel representations are highly related. The main distinction is that kernel representations use any kernel function for the similarity measure $k(\mathbf{x}, \mathbf{c}_j) = \phi_j(\mathbf{x})$, where radial basis functions are one example of a kernel. In addition, if an RBF kernel is chosen, for kernel representations typical the centers are selected from the training dataset. For RBF networks, the selection of the centers is left generally as an important step, where they can be selected from the training set but can also be selected in other ways.

9.2 Learning representations

There are many approaches to learning representations. Two dominant approaches are (semi-supervised) matrix factorization techniques and neural networks. Neural networks build on the generalized linear models we have discussed, stacking multiple generalized linear models together. Matrix factorization techniques (e.g., dimensionality reduction, sparse coding) typically factorize the input data into a dictionary and a new representation (a basis). We will first discuss neural networks, and then discuss the many unsupervised and semisupervised learning techniques that are encompassed by matrix factorizations.