‘cs229’—Notes

Malcolm

Started 1st October 2024

Chapter 1

Supervised learning

Given a dataset of n *training examples* $\{(x^{(i)}, y^{(i)}); i = 1\}, \ldots, n\}$—a *training set*—where $\boldsymbol{x}$ represents the *features* and $\boldsymbol{y}$ the "output" or *target* variable we are trying to predict. If not already obvious, we denote the vector space of $\boldsymbol{x}$ as $\mathcal{X}$ and that of the outputs $\boldsymbol{y}$ as $\mathcal{Y}$.

Our goal is, given a training set, to learn a function $h : \mathcal{X} \mapsto \mathcal{Y}$ so that $h(x)$ is a "good" predictor for the corresponding y. This function h is called a *hypothesis*.

When trying to predict a continuous target variable, we call this a *regression* problem; whereas when y can take on only a small number of discrete values we call that a *classification* problem.

1.1 Linear Regression

Say we decide to approximate y as a linear function of x:

$$h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \ldots$$

Where $\boldsymbol{\theta}$ represents the *parameters/weights* (parametrising the space of linear functions mapping from $\mathcal{X}$ to $\mathcal{Y}$). We can simplify our notation as such: (by convention letting $x_0 = 1$, aptly named the *intercept* term)

$$h(x) = \sum_{i=0}^{d} \theta_i x_i = \boldsymbol{\theta}^T \boldsymbol{x}$$

(next page)

Loss Function
Say we have