Threshold Machines

Machine Learning

Introduction I

For the class of *numeric* representations, machine learning is viewed as:

"searching" a space of *functions* . . .

represented as mathematical models (linear equations, neural nets, . . .).

Introduction II

Some methods:

- linear regression: the process of computing an expression that predicts a numeric quantity
- perceptron: a biologically-inspired linear prediction method
  - an "artificial neuron"
- logistic regression: learning a probability model using a non-linear transformation applied to the data
- multi-layer neural networks: learning non-linear predictors via hidden nodes between input and output (cascaded logistic regression)
- regression trees: tree where each leaf predicts a numeric quantity. The internal nodes are usually tests that decide how the tree is traversed (if $Mainmemory > 512$ then go to the right subtree, otherwise go to the left subtree)

Introduction III

- prediction in a leaf is the average value of (training) instances that reach the leaf
- internal nodes test discrete **or** continuous attributes

- model trees: regression tree with linear or non-linear models at the leaf nodes

We will look at the simplest model for numerica prediction: a *regression equation*
The outcome will be a linear sum of feature values with appropriate weights.

Regression
The process of determining the weights for the regression equation.

The Classification Problem

- Given a set of data points x_i, y_i $(i = 1 \dots n)$, where $y_i \in \mathcal{L}$ (a finite set of *class labels*), what is the relationship between x and y?
- Regression models are not immediately applicable, since they require $yin\Re$. Instead, we will look at answering this question in stages
  1. Construct "linear threshold machines", which use some linear function to separate classes
  2. Construct "support vector machines", which apply a non-linear transformation of the data, and then use linear models for separating classes
  3. Construct a functional approximation for $P(Y|X)$ and use probability-based discrimination, as was done for Bayes-optimal classification
  4. Construct an approximation for $P(X, Y)$ and use that to compute $P(Y|X)$ (and then use probability-based discrimination)
  5. Construct a model for estimating $P(X, Y)$ and use that to compute $P(Y|X)$

Discriminant Functions

- For $\mathcal{L} = \{0, 1\}$, a discriminant function $g(X)$ can be used to construct a classifier:

$$
\begin{aligned}
//h(X) &= 1 \qquad \text{if } g(X) > 0 \\
&= 0 \qquad \text{otherwise}
\end{aligned}
$$

- The Bayes-optimal classifier uses the discriminant function $P(X|Y=1)P(Y=1) - P(X|Y=0)P(Y=0)$
- If we do not have the probabilities, we can either (a) avoid them; (b) approximate them; or (c) estimate them
- We will first look at discriminant functions that do not use probabilities

Linear Discriminant Functions

- A simple discriminant function that has the structure of a linear model and parameters $W = (w_0, w_1, \ldots, w_d)$ is:

$$g(W, X) = \sum_{i=0}^{d} w_i x_i = W^T \cdot X$$

  Here, we are assuming the data are $d + 1$-dimensional vectors, of the kind $(1, x_1, \ldots, x_d)$. That is, the discriminant function is really:

$$g(W, X) = w_0 + \sum i = 1^d w_i x_i$$

- The *linear* part refers to being linear in the w_i (not the x_i). So, in fact the summation could be over any function of X

A Probabilistic Discriminant Function

- Previously, we looked at the Bayes Classifier (turned around here, from before):

$$\begin{aligned} h_B(X) &= 1 \quad \text{if } P(Y=1|X) > P(Y=0|X) \\ &= 0 \quad \text{otherwise} \end{aligned}$$

  That is, the class with the maximum posterior probability is selected.
- This is an example of a *probabilistic threshold machine* since the decision to classify an instance is made based on whether $P(Y=1|X) - P(Y=0|X) > 0$
- We know we cannot really use the Bayes Classifier, since we do not know the underlying probabilities to obtain $P(Y|X)$. But we can try to estimate it

A Regression Model for $P(Y|X$ I

- If we are concerned with the problem of conditional class probability estimation ($P(Y|\mathrm{x})$), then there is a well-known technique that assumes that the probability can be estimated using a specific non-linear function of x
- Suppose data points are d-dimensional vectors of the kind $\mathrm{x} = [x_1, x_2, \ldots, x_d]^T$ where $x_i \in \Re$. We wish to obtain an estimate of the conditional probabilities of the values of a *dependent* or *outcome* or *response* random variable Y (for simplicity, let us assume this is a random variable that takes values from a discrete set (say: $\{0, 1\}$)

A Regression Model for $P(Y|X$ II

- Now, we can try to estimate this probability using linear regression:

  $$P(Y = 1|\mathrm{x}) \;=\; f(\mathrm{x}) \;=\; \mathrm{w}^T\mathrm{x}$$

  where w is a d-dimensional weight vector, which was obtained using least-squares on some data points. (We can introduce a constant by having a $d+1$-dimension vector, with $\mathrm{x} = (1, x_1, \ldots, x_d)$ and $\mathrm{w} = (w_0, w_1, \ldots, w_d)$
- But we cannot do this since usually $\mathrm{w}^T\mathrm{x}$ will not be restricted to $[0, 1]$. So, let us hack it.

A Regression Model for $P(Y|X$ III

- Let us use a function $g(\mathrm{w}^T\mathrm{x})$ (correctly, $g(\mathrm{w}, \mathrm{x})$) which has the following properties instead:

$$
\begin{aligned}
g(\mathrm{w}^T\mathrm{x}) &= 0 \;\; \text{if } \mathrm{w}^T\mathrm{x} = -\infty \\
&= 1 \;\; \text{if } \mathrm{w}^T\mathrm{x} = \infty \\
&= p \in (0,1) \;\; \text{otherwise}
\end{aligned}
$$

- There is one well-known functions g that can be used to implement this trick. This is the *sigmoid* function

$$
\sigma(x) = \frac{1}{1 + e^{-x}}
$$

A Regression Model for $P(Y|X$ IV

- We will model the conditional probability of $P(Y|X)$ using this function. Specifically:

$$P(Y = 1|\mathrm{x}) \;=\; \frac{1}{1 + e^{-\mathrm{w}^T\mathrm{x}}}$$

This is the same as:

$$\ln \frac{P(Y = 1|\mathrm{x})}{1 - P(Y = 1|\mathrm{x})} \;=\; \mathrm{w}^T\mathrm{x}$$

(Can you show this?)

- The quantity on the l.h.s. is called the *logit* and all we are doing is a linear model for the logit.

A Regression Model for $P(Y|X$ V

- So, we are really using linear regression, which means we can use the same procedures as before to find the structure and parameters (analytically using partial derivatives; numerically using gradient descent; search-based determination of structure; or cost minimisation using regularisation)
- This entire procedure is called *logistic regression*: linear modelling of the logit, with structure and parameter estimation

The Probabilistic Model for Logistic Regression

- But why should we use the logistic function σ at all? Is it just a mathematical trick? If the data satisfy a specific assumption, then this is exactly the right function to use
- The *exponential family* is a class of probability distributions with the following form:

$$f(\mathrm{x}|\theta, \phi) \;=\; h(\mathrm{x}, \phi) e^{\frac{\theta^T \mathrm{x} - A(\theta)}{a(\phi)}}$$

  where θ is a *location* parameter, and ϕ is a *scale* parameter. A number of well-known distributions are from this class (Normal, Binomial, Dirichlet, *etc*.)
- RESULT: (without proof) The sigmoid function is the correct function to use when all the $P(\mathrm{x}|Y)$ are from the same exponential family and the same scale factor ϕ