CMPUT 367

Roderick Lan

Contents

1 Lecture 4 - Jan 23

1.1 Review from prev lec

Classification:

Logistic regression

$$\mathbf{x}^{(m)} \in \mathbb{R}^d$$
$$t^{(m)} \in \{0, 1\}$$

Scoring Function: (not in input space)

$$z^{(m)} = \mathbf{w}^\top \mathbf{x}^{(m)} + b$$

Linear/affine function used because it is simple; have to specify human/function preference.

Can also build nonlinear models
Design non linear features (ie. $x_2 = x^2$) $\rightarrow$ model no longer linear in terms of raw output, but function still linear
Kernel method = uses nonlinear inner product
Stacking multiple (learnable) linear classifiers (basis of NN)

Complex models can reduce to simpler models
Linear models are the basis for all ML

Linear scoring function $\rightarrow$ sigmoid function ('squish' it to have range (0,1))

Definition 1.1.1: Sigmoid Function

$$y^{(m)} = \sigma(z^{(m)}) = \frac{1}{1 + e^{-z^{(m)}}} = \frac{1}{1 + e^{-\mathbf{w}^\top \mathbf{x}^{(m)} + b}}$$

"goodness" function; no negative $\rightarrow$ "badness" function

$\Pr[t^{(m)} = 1 | x^{(m)}] = y^{(m)}$
$\Pr[t^{(m)} = 0 | x^{(m)}] = 1 - y^{(m)}$
$\mathcal{D}_{\text{train}} = \{(x^{(m)}, t^{(m)})_{m=1}^M$

Likelihood:

$$p(\mathcal{D}_{\text{train}}; w, b) = \prod_{m=1}^{M} y^{(m)t^{(m)}} (1 - y^{(m)})^{1 - t^{(m)}}$$

Loss: (CE loss)

$$J(w, b) = \frac{1}{M} \sum_{m=1}^{M} \left[-t^{(m)} \ln y^{(m)} - (1 - t^{(m)}) \ln(1 - y^{(m)}) \right]$$

1.2 Multiclass Classification

(softmax)
$\mathbf{x}^{(m)} \in \mathbb{R}^d$
$t^{(m)} \in \{1, 2, \ldots, k\}$
Need multiple scoring functions for multiple classes;
Convenient to have scoring function for each class
$z_1^{(m)} = \mathbf{w}_1^\top \mathbf{x}^{(m)} + b_1, \cdots, \mathbf{w}_k^\top \mathbf{x}^{(m)} + b_k$
k scoring functions for convenience, but one is redundant

Definition 1.2.1: Categorical Distribution

$$\text{Cat}(\pi_1, \cdots, \pi_k) \quad \pi_k \geq 0 \ \forall k$$

Probability of $k = \pi_k$

$$\sum_k \pi_k = 1$$

use exp because it is montonically increasing and unbounded; maps $\mathbb{R} \to \mathbb{R}_{++}$

$$T = \sum_{k=1}^{k} \exp(\mathbf{w}_k^\top \mathbf{x}^{(m)} + b_k) \text{ for normalizing}$$

$$y^{(1)} = \frac{\exp(\mathbf{w}_1^\top \mathbf{x}^{(m)} + b_1)}{T}, \cdots, \frac{\exp(\mathbf{w}_k^\top \mathbf{x}^{(m)} + b_k)}{T} = y^{(k)}$$

Range of function should be $(0, +\infty)$

> "Why does it have to be positively unbounded, what would happen if bounded?" Can't tell the difference b/w two similar samples

$$\Pr[t^{(m)} = k | x^{(m)}] = y_k^{(m)}$$

$\mathcal{D}_{\text{train}} = \{(x^{(m)}, t^{(m)})\}_{m=1}^M$
Likelihood:

$$p(\mathcal{D}_{\text{train}}; \{w_k, b_k\}_{k=1}^K) = \prod_{m=1}^{M} {y_1^{(m)}}^{\mathbb{I}\{t^{(m)}=1\}} {y_2^{(m)}}^{\mathbb{I}\{t^{(m)}=2\}} \cdots {y_k^{(m)}}^{\mathbb{I}\{t^{(m)}=k\}}$$

$$t_k^{(m)} = \mathbb{I}\{t^{(m)} = k\}$$

$$p(\mathcal{D}_{\text{train}}; \{w_k, b_k\}_{k=1}^K) = \prod_{m=1}^{M} {y_1^{(m)}}^{t_1^{(m)}} {y_2^{(m)}}^{t_2^{(m)}} \cdots {y_k^{(m)}}^{t_k^{(m)}}$$

Log-Likelihood:

$$\begin{aligned}\ln p(\mathcal{D}_{\text{train}}) &= \sum_{m=1}^{M} \ln {y_1^{(m)}}^{t_1^{(m)}} \cdots {y_k^{(m)}}^{t_k^{(m)}} \\ &= \sum_{m=1}^{M} \left[t_1^{(m)} \ln y_1^{(m)} + \cdots + t_k^{(m)} \ln y_k^{(m)} \right] \\ &= \sum_{m=1}^{M} \sum_{k=1}^{K} t_k^{(m)} \ln y_k^{(m)}\end{aligned}$$

Loss:

$$\begin{aligned}J &= -\sum_{m=1}^{M} \sum_{k=1}^{K} t_k^{(m)} \ln y_k^{(m)} \\ &= -\sum_{m=1}^{M} \ln y_{t^{(m)}}^{(m)} \qquad\qquad t^{(m)} \text{ serves as index}\end{aligned}$$

$t^{(m)} \in \{1, \ldots, k\}$ index representation

$$\mathbf{t}^{(m)} = \begin{bmatrix} t_1^{(m)} \\ \vdots \\ t_k^{(m)} \end{bmatrix} = \begin{bmatrix} \mathbb{I}\{t^{(m)} = 1\} \\ \vdots \\ \mathbb{I}\{t^{(m)} = k\} \end{bmatrix} \qquad \text{one-hot representation}$$

$\hat{t}_* = \arg\max_k = y_k^*$
$y_k^* = \Pr[t^* = k | x^*]$
[Max A Posteriori Inference]

Questions:

For softmax, what if $k = 2$?
Optimization for log. regression and softmax?