Machine Learning - MT 2017
11. Classification: Logistic Regression

Varun Kanade

University of Oxford
November 1, 2017

Logistic Regression

Logistic Regression is actually a classification method

In its simplest form it is a binary (two classes) classification method

- Today's Lecture: We'll denote these by 0 and 1
- Next Week: Sometimes it's more convenient to call them -1 and $+1$
- Ultimately, the choice is just for mathematical convenience

It is a discriminative method. We only model:

$$p(y \mid \mathbf{w}, \mathbf{x})$$

Logistic Regression (LR)

- LR builds up on a linear model, composed with a sigmoid function

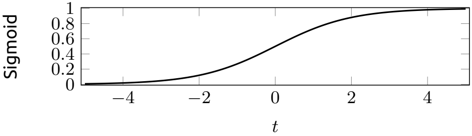
$$p(y \mid \mathbf{w}, \mathbf{x}) = \mathrm{Bernoulli}(\mathrm{sigmoid}(\mathbf{w} \cdot \mathbf{x}))$$

- $Z \sim \mathrm{Bernoulli}(\theta)$

$$Z = \begin{cases} 1 & \text{with probability } \theta \\ 0 & \text{with probability } 1 - \theta \end{cases}$$

- Recall that the sigmoid function is defined by:

$$\mathrm{sigmoid}(t) = \frac{1}{1 + e^{-t}}$$



- As we did in the case of linear models, we assume $x_0 = 1$ for all datapoints, so we do not need to handle the bias term w_0 separately

Prediction Using Logistic Regression

Suppose we have estimated the model parameters $\mathbf{w} \in \mathbb{R}^D$

For a new datapoint $\mathbf{x}_{\text{new}}$, the model gives us the probability

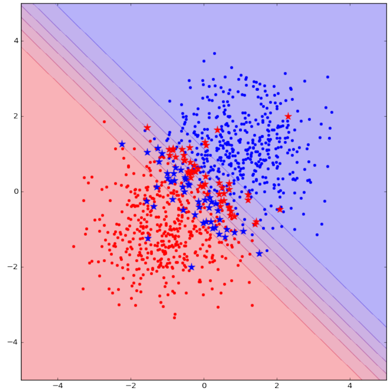
$$p(y_{\text{new}} = 1 \mid \mathbf{x}_{\text{new}}, \mathbf{w}) = \text{sigmoid}(\mathbf{w} \cdot \mathbf{x}_{\text{new}}) = \frac{1}{1 + \exp(-\mathbf{x}_{\text{new}} \cdot \mathbf{w})}$$

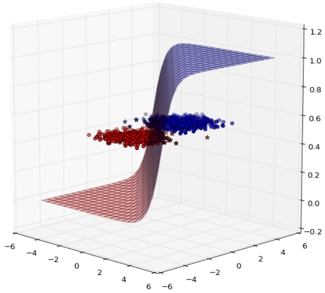
In order to make a prediction we can simply use a threshold at $\frac{1}{2}$

$$\widehat{y}_{\text{new}} = \mathbb{I}(\text{sigmoid}(\mathbf{w} \cdot \mathbf{x}_{\text{new}})) \geq \frac{1}{2}) = \mathbb{I}(\mathbf{w} \cdot \mathbf{x}_{\text{new}} \geq 0)$$

Class boundary is linear (separating hyperplane)

Prediction Using Logistic Regression





Likelihood of Logistic Regression

Data $\mathcal{D} = \langle(\mathbf{x}_i, y_i)\rangle_{i=1}^{N}$, where $\mathbf{x}_i \in \mathbb{R}^D$ and $y_i \in \{0, 1\}$

Let us denote the sigmoid function by σ

We can write the likelihood for of observing the data given model parameters $\mathbf{w}$ as:

$$p(\mathbf{y} \mid \mathbf{X}, \mathbf{w}) = \prod_{i=1}^{N} \sigma(\mathbf{w}^{\mathsf{T}}\mathbf{x}_i)^{y_i} \cdot (1 - \sigma(\mathbf{w}^{\mathsf{T}}\mathbf{x}_i))^{1-y_i}$$

Let us denote $\mu_i = \sigma(\mathbf{w}^{\mathsf{T}}\mathbf{x}_i)$

We can write the negative log-likelihood as:

$$\mathrm{NLL}(\mathbf{y} \mid \mathbf{X}, \mathbf{w}) = -\sum_{i=1}^{N}(y_i \log \mu_i + (1 - y_i) \log(1 - \mu_i))$$

Likelihood of Logistic Regression

Recall that $\mu_i = \sigma(\mathbf{w}^\mathsf{T}\mathbf{x}_i)$ and the negative log-likelihood is

$$\mathrm{NLL}(\mathbf{y} \mid \mathbf{X}, \mathbf{w}) = -\sum_{i=1}^{N}(y_i \log \mu_i + (1 - y_i)\log(1 - \mu_i))$$

Let us focus on a single datapoint, the contribution to the negative log-likelihood is

$$\mathrm{NLL}(y_i \mid \mathbf{x}_i, \mathbf{w}) = -(y_i \log \mu_i + (1 - y_i)\log(1 - \mu_i))$$

This is called the cross-entropy between y_i and μ_i

If $y_i = 1$, then as

- As $\mu_i \to 1$, $\mathrm{NLL}(y_i \mid \mathbf{x}_i, \mathbf{w}) \to 0$
- As $\mu_i \to 0$, $\mathrm{NLL}(y_i \mid \mathbf{x}_i, \mathbf{w}) \to \infty$

Maximum Likelihood Estimate for LR

Recall that $\mu_i = \sigma(\mathbf{w}^\mathsf{T}\mathbf{x}_i)$ and the negative log-likelihood is

$$\mathrm{NLL}(\mathbf{y} \mid \mathbf{X}, \mathbf{w}) = -\sum_{i=1}^{N}(y_i \log \mu_i + (1 - y_i)\log(1 - \mu_i))$$

We can take the gradient with respect to $\mathbf{w}$

$$\nabla_{\mathbf{w}}\mathrm{NLL}(\mathbf{y} \mid \mathbf{X}, \mathbf{w}) = \sum_{i=1}^{N}\mathbf{x}_i(\mu_i - y_i) = \mathbf{X}^\mathsf{T}(\boldsymbol{\mu} - \mathbf{y})$$

And the Hessian is given by,

$$\mathbf{H} = \mathbf{X}^\mathsf{T}\mathbf{S}\mathbf{X}$$

$\mathbf{S}$ is a diagonal matrix where $S_{ii} = \mu_i(1 - \mu_i)$

Iteratively Re-Weighted Least Squares (IRLS)

Depending on the dimension, we can apply Newton's method to estimate $\mathbf{w}$

Let $\mathbf{w}_t$ be the parameters after t Newton steps.

The gradient and Hessian are given by:

$$\begin{aligned} \mathbf{g}_t &= \mathbf{X}^\top(\boldsymbol{\mu}_t - \mathbf{y}) = -\mathbf{X}^\top(\mathbf{y} - \boldsymbol{\mu}_t) \\ \mathbf{H}_t &= \mathbf{X}^\top \mathbf{S}_t \mathbf{X} \end{aligned}$$

The Newton Update Rule is:

$$\begin{aligned} \mathbf{w}_{t+1} &= \mathbf{w}_t - \mathbf{H}_t^{-1}\mathbf{g}_t \\ &= \mathbf{w}_t + (\mathbf{X}^\top \mathbf{S}_t \mathbf{X})^{-1}\mathbf{X}^\top(\mathbf{y} - \boldsymbol{\mu}_t) \\ &= (\mathbf{X}^\top \mathbf{S}_t \mathbf{X})^{-1}\mathbf{X}^\top \mathbf{S}_t(\mathbf{X}\mathbf{w}_t + \mathbf{S}_t^{-1}(\mathbf{y} - \boldsymbol{\mu}_t)) \\ &= (\mathbf{X}^\top \mathbf{S}_t \mathbf{X})^{-1}\mathbf{X}^\top \mathbf{S}_t \mathbf{z}_t \end{aligned}$$

Where $\mathbf{z}_t = \mathbf{X}\mathbf{w}_t + \mathbf{S}_t^{-1}(\mathbf{y} - \boldsymbol{\mu}_t)$. Then $\mathbf{w}_{t+1}$ is a solution of the following:

Weighted Least Squares Problem

$$\text{minimise} \quad \sum_{i=1}^{N} S_{t,ii}(z_{t,i} - \mathbf{w}^\top \mathbf{x}_i)^2$$

Multiclass Logistic Regression

Multiclass logistic regression is also a discriminative classifier

Let the inputs be $\mathbf{x} \in \mathbb{R}^D$ and $y \in \{1, \ldots, C\}$

There are parameters $\mathbf{w}_c \in \mathbb{R}^D$ for every class $c = 1, \ldots, C$

We'll put this together in a matrix form $\mathbf{W}$ that is $D \times C$

The multiclass logistic model is given by:

$$p(y = c \mid \mathbf{x}, \mathbf{W}) = \frac{\exp(\mathbf{w}_c^\mathsf{T} \mathbf{x})}{\sum_{c'=1}^{C} \exp(\mathbf{w}_{c'}^\mathsf{T} \mathbf{x})}$$

Multiclass Logistic Regression

The multiclass logistic model is given by:

$$p(y = c \mid \mathbf{x}, \mathbf{W}) = \frac{\exp(\mathbf{w}_c^\mathsf{T} \mathbf{x})}{\sum_{c'=1}^{C} \exp(\mathbf{w}_{c'}^\mathsf{T} \mathbf{x})}$$

Recall the softmax function

Softmax

Softmax maps a set of numbers to a probability distribution with mode at the maximum

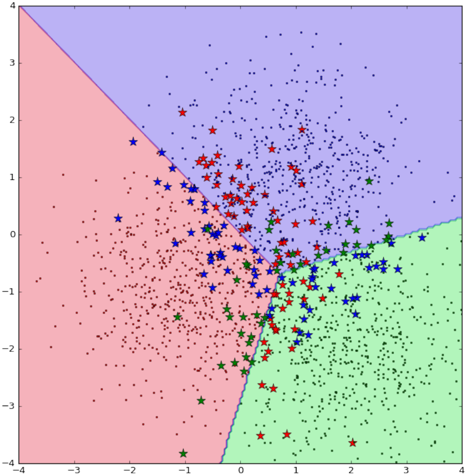
$$\operatorname{softmax}\left([a_1, \ldots, a_C]^\mathsf{T}\right) = \left[\frac{e^{a_1}}{Z}, \ldots, \frac{e^{a_C}}{Z}\right]^\mathsf{T}$$

where $Z = \sum_{c=1}^{C} e^{a_c}$.

The multiclass logistic model is simply:

$$p(y \mid \mathbf{x}, \mathbf{W}) = \operatorname{softmax}\left(\left[\mathbf{w}_1^\mathsf{T} \mathbf{x}, \ldots, \mathbf{w}_C^\mathsf{T} \mathbf{x}\right]^\mathsf{T}\right)$$

Multiclass Logistic Regression



Summary: Logistic Regression

- Logistic Regression is a (binary) classification method
- It is a discriminative model
- Extension to multiclass by replacing sigmoid by softmax
- Can derive Maximum Likelihood Estimates using Convex Optimization
- See Chap 8.3 in Murphy (for multiclass), but we'll revisit as a form of a neural network

Next Week

- Suppor Vector Machines
- Kernel Methods
- Revise Linear Programming and Convex Optimisation