Logistic Regression

Reading:

X. Z. Fern’s Notes on Logistic Regression

(linked from class website)
Discriminative vs. Generative Models

• Discriminative: NN, SVM, Logistic Regression

• Generative: Naïve Bayes, Markov Chains, etc.

• E.g.: http://projects.haykranen.nl/markov/demo/

• E.g., how could you use a Naïve Bayes model to learn to generate spam-like messages?

• How could you use one to fool spam detectors?
Learning Probabilistic Models

Goal is to learn a function mapping an instance
\( \mathbf{x} = (x_1, \ldots, x_n) \) to a probability distribution over classes \( y \)

I.e., learn \( P(y \mid \mathbf{x}) \)
**Softmax**

Computes $P(y \mid x)$ by turning output values into probability distribution

$$P(y \mid x) = y_{sm}(o_i) = \frac{e^{o_i}}{\sum_{k=1}^{K} e^{o_k}}$$

$y_{sm} = .501 \quad y_{sm} = .499$

**Bayesian learning**

Computes $P(y \mid x)$ by learning $P(y)$ and $P(x \mid y)$ from data, and computing

$$P(y \mid x) \approx \frac{P(x \mid y)P(y)}{P(x)}$$

**Logistic Regression**

Learns to directly map inputs to probabilities:

$$x \longrightarrow P(class \mid x)$$
Probability of passing exam versus hours of studying

https://en.wikipedia.org/wiki/Logistic_regression
Logistic Regression (binary case)

\[ y = b_0 + b_1 x \] \hspace{1cm} \text{Linear Model}

\[ p = \frac{1}{1 + e^{-(b_0 + b_1 x)}} \] \hspace{1cm} \text{Logistic Model}
For ease of notation, let $\mathbf{x} = (x_0, x_1, \ldots, x_n)$, where $x_0 = 1$.

Let $\mathbf{w} = (w_0, w_1, \ldots, w_n)$, where $w_0$ is the bias weight.

Class $y \in \{0, 1\}$
Logistic Regression

Define:

\[
P(y = 1 \mid x) = \sigma(w \cdot x) = \frac{1}{1 + e^{-w \cdot x}}
\]

\[
P(y = 0 \mid x) = 1 - \sigma(w \cdot x) = \frac{e^{-w \cdot x}}{1 + e^{-w \cdot x}}
\]

Note: \( P(y = 0 \mid x) + P(y = 1 \mid x) = 1 \)

To classify a new \( x \), assign class that maximizes \( P(y = y_k \mid x) \).

Class is 1 if \( 1 > e^{-w \cdot x} \)

\[\Rightarrow \log 1 > \log e^{-w \cdot x}\]

\[\Rightarrow 0 > -w \cdot x\]

\[\text{class}_{LR}(x) = \begin{cases} 1 \text{ if } w \cdot x > 0 \\ 0 \text{ otherwise} \end{cases}\]
Logistic Regression: Learning Weights

Goal is to learn weights $w$.

Let $(x^j, y^j)$ be the $j$th training example and its label.

We want:

$$w^* = \arg\max_w \prod_j P(y^j | x^j, w), \text{ for training examples } j = 1, \ldots, M$$

This is equivalent to:

$$w^* = \arg\max_w \sum_j \ln P(y^j | x^j, w)$$

This is called "log of conditional likelihood"
We can write the log conditional likelihood this way:

$$l(w) = \sum_j \ln P(y^j | x^j, w)$$

$$= \sum_j y^j \ln P(y^j = 1 | x^j, w) + (1 - y^j) \ln P(y^j = 0 | x^j, w)$$

since $y^j$ is either 0 or 1

$$= \sum_j y^j \ln \sigma(w \cdot x) + (1 - y^j) \ln (1 - \sigma(w \cdot x))$$

This is what we want to maximize.
Use gradient ascent to maximize $l(w)$.

This is called “Maximum likelihood estimation” (or MLE).

Recall: \[ \frac{d\sigma(z)}{dz} = \sigma(z)(1 - \sigma(z)) \]

We have:

\[ l(w) = \sum_j y^j \ln \sigma(w \cdot x) + (1 - y^j) \ln (1 - \sigma(w \cdot x)) \]

Let’s find the gradient with respect to $w_i$:

\[ \frac{\partial l(w)}{\partial w_i} = \frac{\partial}{\partial w_i} \left( \sum_j y^j \ln \sigma(w \cdot x) + (1 - y^j) \ln (1 - \sigma(w \cdot x)) \right) \]
Using chain rule and algebra

\[
\frac{\partial l(w)}{\partial w_i} = \frac{\partial}{\partial w_i} \left( \sum_j y^j \ln \sigma(w \cdot x) + (1 - y^j) \ln (1 - \sigma(w \cdot x)) \right)
\]

\[
= \sum_j y^j \frac{1}{\sigma(w \cdot x)} (\sigma(w \cdot x)(1 - \sigma(w \cdot x)x_i) + (1 - y^j) \frac{1}{1 - \sigma(w \cdot x)} (-\sigma(w \cdot x)(1 - \sigma(w \cdot x)x_i)))
\]

\[
= \sum_j y^j (1 - \sigma(w \cdot x)) x_i - (1 - y^j) \sigma(w \cdot x) x_i
\]

\[
= \sum_j x_i \left( y^j - y^i \sigma(w \cdot x) - \sigma(w \cdot x) + y^j \sigma(w \cdot x) \right)
\]

\[
= \sum_j \left( y^j - \sigma(w \cdot x) \right) x_i
\]
Stochastic Gradient Ascent for Logistic Regression

- Start with small random initial weights, both positive and negative:

  \[ w = (w_0, w_1, \ldots, w_n) \]

Repeat until convergence, or for some max number of epochs

For each training example \((x^j, y^j)\):

\[ \Delta w_i = (y^j - \sigma(w \cdot x^j))x_i^j \]

\[ w_i \leftarrow w_i + \eta \Delta w_i \]

Note again that \(w\) includes the bias weight \(w_0\), and \(x\) includes the bias term \(x_0 = 1\).
Homework 4, Part 2