Introduction to Machine Learning

Logistic Regression

Varun Chandola
Computer Science & Engineering
State University of New York at Buffalo
Buffalo, NY, USA
chandola@buffalo.edu
Generative vs. Discriminative Classifiers

Logistic Regression

Logistic Regression - Training
- Using Gradient Descent for Learning Weights
- Using Newton’s Method
- Regularization with Logistic Regression
- Handling Multiple Classes
- Bayesian Logistic Regression
- Laplace Approximation
- Posterior of $\mathbf{w}$ for Logistic Regression
- Approximating the Posterior
- Getting Prediction on Unseen Examples
Generative vs. Discriminative Classifiers

▶ Probabilistic classification task:

\[ p(Y = \text{benign}|X = x), p(Y = \text{malicious}|X = x) \]

▶ How do you estimate \( p(y|x) \)?

\[
p(y|x) = \frac{p(y, x)}{p(x)} = \frac{p(x|y)p(y)}{p(x)}
\]

▶ Two step approach - Estimate generative model and then posterior for \( y \) (Naïve Bayes)

▶ Solving a more general problem \([2, 1]\)

▶ Why not directly model \( p(y|x) \)? - Discriminative approach
Which is Better?

- Number of training examples needed to learn a PAC-learnable classifier $\propto VC$-dimension of the hypothesis space
- VC-dimension of a probabilistic classifier $\propto$ Number of parameters [2] (or a small polynomial in the number of parameters)
- Number of parameters for $p(y, x) >$ Number of parameters for $p(y|x)$

Discriminative classifiers need lesser training examples to for PAC learning than generative classifiers
Logistic Regression

- $y \mid x$ is a *Bernoulli* distribution with parameter $\theta = sigmoid(w^T x)$
- When a new input $x^*$ arrives, we toss a coin which has $sigmoid(w^T x^*)$ as the probability of heads
- If outcome is heads, the predicted class is 1 else 0
- Learns a linear boundary

Learning Task for Logistic Regression

Given training examples $\langle x_i, y_i \rangle_{i=1}^{D}$, learn $w$
Logistic Regression - Recap

Bayesian Interpretation

- Directly model $p(y|x)$ ($y \in \{0, 1\}$)
- $p(y|x) \sim Bernoulli(\theta = \text{sigmoid}(w^\top x))$

Geometric Interpretation

- Use regression to predict discrete values
- *Squash* output to [0, 1] using sigmoid function
- Output less than 0.5 is one class and greater than 0.5 is the other
MLE Approach

Assume that $y \in \{0, 1\}$

What is the likelihood for a Bernoulli sample?

- If $y_i = 1$, $p(y_i) = \theta_i = \frac{1}{1 + \exp(-w^\top x_i)}$
- If $y_i = 0$, $p(y_i) = 1 - \theta_i = \frac{1}{1 + \exp(w^\top x_i)}$
- In general, $p(y_i) = \theta_i^{y_i}(1 - \theta_i)^{1-y_i}$

Log-likelihood

$$LL(w) = \sum_{i=1}^{N} y_i \log \theta_i + (1 - y_i) \log (1 - \theta_i)$$

No closed form solution for maximizing log-likelihood
Using Gradient Descent for Learning Weights

- Compute gradient of LL with respect to $w$
- A convex function of $w$ with a unique global maximum

$$\frac{d}{dw} LL(w) = \sum_{i=1}^{N} (y_i - \theta_i)x_i$$

- Update rule:

$$w_{k+1} = w_k + \eta \frac{d}{dw_k} LL(w_k)$$
Using Newton’s Method

- Setting $\eta$ is sometimes tricky
- Too large – incorrect results
- Too small – slow convergence
- Another way to speed up convergence:

Newton’s Method

$$w_{k+1} = w_k + \eta H_k^{-1} \frac{d}{dw_k} LL(w_k)$$
What is the Hessian?

- Hessian or $H$ is the second order derivative of the objective function.
- Newton’s method belong to the family of second order optimization algorithms.
- For logistic regression, the Hessian is:

$$H = - \sum_i \theta_i (1 - \theta_i) x_i x_i^T$$
Overfitting is an issue, especially with large number of features.

Add a Gaussian prior $\sim \mathcal{N}(0, \tau^2)$ (Or a regularization penalty).

Easy to incorporate in the gradient descent based approach.

\[
LL'(\mathbf{w}) = LL(\mathbf{w}) - \frac{1}{2} \lambda \mathbf{w}^\top \mathbf{w}
\]

\[
\frac{d}{d\mathbf{w}} LL'(\mathbf{w}) = \frac{d}{d\mathbf{w}} LL(\mathbf{w}) - \lambda \mathbf{w}
\]

\[
H' = H - \lambda I
\]

where $I$ is the identity matrix.
Handling Multiple Classes

- One vs. Rest and One vs. Other
- \( p(y|x) \sim Multinoulli(\theta) \)
- Multinoulli parameter vector \( \theta \) is defined as:
  \[
  \theta_j = \frac{\exp(w_j^T x)}{\sum_{k=1}^C \exp(w_k^T x)}
  \]
- Multiclass logistic regression has \( C \) weight vectors to learn
How to get the posterior for $w$?

Not easy - Why?

Laplace Approximation

We do not know what the true posterior distribution for $w$ is.

Is there a close-enough (approximate) Gaussian distribution?
Laplace Approximation

Problem Statement
How to approximate a posterior with a Gaussian distribution?

- When is this needed?
  - When direct computation of posterior is not possible.
  - No conjugate prior 😊
Assume that the posterior is:
\[ p(w|D) = \frac{1}{Z} e^{-E(w)} \]

\( E(w) \) is the energy function, which is equivalent to the negative log of the unnormalized log posterior.

Let \( w_{\text{MAP}} \) be the mode or expected value of the posterior distribution of \( w \).

Taylor series expansion of \( E(w) \) around the mode:
\[ E(w) \approx E(w_{\text{MAP}}) + (w - w^*)^\top E'(w) + (w - w_{\text{MAP}})^\top E''(w) (w - w_{\text{MAP}}) + \ldots \]

where \( E'(w) \) is the gradient and \( E''(w) \) is the Hessian (second derivative).
Assume that posterior is:

\[ p(w|D) = \frac{1}{Z} e^{-E(w)} \]

\( E(w) \) is **energy function** ≡ negative log of unnormalized log posterior
Assume that posterior is:

\[ p(w|D) = \frac{1}{Z} e^{-E(w)} \]

\( E(w) \) is **energy function** \( \equiv \) negative log of unnormalized log posterior

Let \( w_{MAP} \) be the *mode* or *expected value* of the posterior distribution of \( w \)
Laplace Approximation using Taylor Series Expansion

Assume that posterior is:

\[ p(w|D) = \frac{1}{Z} e^{-E(w)} \]

\( E(w) \) is energy function \( \equiv \) negative log of unnormalized log posterior

Let \( w_{MAP} \) be the mode or expected value of the posterior distribution of \( w \)

Taylor series expansion of \( E(w) \) around the mode

\[ E(w) = E(w_{MAP}) + (w - w^*)^T E'(w) + (w - w^*)^T E''(w)(w - w^*) + \ldots \]

\( E'(w) = \nabla - \) first derivative of \( E(w) \) (gradient) and \( E''(w) = H \) is the second derivative (Hessian)
Since $w_{MAP}$ is the mode, the first derivative or gradient is zero

$$E(w) \approx E(w_{MAP}) + (w - w^*)^\top H(w - w^*)$$
Since $w_{MAP}$ is the mode, the first derivative or gradient is zero

$$E(w) \approx E(w_{MAP}) + (w - w^*)^\top H(w - w^*)$$

Posterior $p(w|D)$ may be written as:

$$p(w|D) \approx \frac{1}{Z} e^{-E(w_{MAP})} \exp \left[ -\frac{1}{2} (w - w^*)^\top H(w - w^*) \right]$$

$$= \mathcal{N}(w_{MAP}, H^{-1})$$

$w_{MAP}$ is the mode obtained by maximizing the posterior using gradient ascent
Posterior of $\mathbf{w}$ for Logistic Regression

- Prior:
  \[
p(\mathbf{w}) = \mathcal{N}(\mathbf{0}, \tau^2 \mathbf{I})
  \]

- Likelihood of data
  \[
p(D|\mathbf{w}) = \prod_{i=1}^{N} \theta_i^{y_i} (1 - \theta_i)^{1-y_i}
  \]
  where $\theta_i = \frac{1}{1 + e^{-\mathbf{w}^\top \mathbf{x}_i}}$

- Posterior:
  \[
p(\mathbf{w}|D) = \frac{\mathcal{N}(\mathbf{0}, \tau^2 \mathbf{I}) \prod_{i=1}^{N} \theta_i^{y_i} (1 - \theta_i)^{1-y_i}}{\int p(D|\mathbf{w}) d\mathbf{w}}
  \]
Approximating the Posterior - Laplace Approximation

- Approximate posterior distribution

\[ p(w|D) = \mathcal{N}(w_{MAP}, H^{-1}) \]

- \( H \) is the Hessian of the negative log-posterior w.r.t. \( w \)
Getting Prediction

\[ p(y|x) = \int p(y|x, w)p(w|D)dw \]

1. Use a point estimate of \( w \) (MLE or MAP)
2. Analytical Result
3. Monte Carlo Approximation
   - Numerical integration
   - Sample finite “versions” of \( w \) using \( p(w|D) \)
     \[ p(w|D) = \mathcal{N}(w_{MAP}, H^{-1}) \]
   - Compute \( p(y|x) \) using the samples and add
A. Y. Ng and M. I. Jordan. 
On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes. 

V. Vapnik. 
*Statistical learning theory*. 