Machine Learning
CS 6830

Lecture 07

Razvan C. Bunescu

School of Electrical Engineering and Computer Science

bunescu@ohio.edu

Probabilistic Generative Models: Binary Classification (K = 2)

- Model class-conditional $p(\mathbf{x} \mid C_1)$, $p(\mathbf{x} \mid C_2)$ as well as the priors $p(C_1)$, $p(C_2)$, then use Bayes's theorem to find $p(C_1 \mid \mathbf{x})$, $p(C_2 \mid \mathbf{x})$:

$$p(C_1 \mid \mathbf{x}) = \frac{p(\mathbf{x} \mid C_1) p(C_1)}{p(\mathbf{x} \mid C_1) p(C_1) + p(\mathbf{x} \mid C_2) p(C_2)}$$

$$= \sigma(a(\mathbf{x}))$$

logistic sigmoid

$$\text{where } \sigma(a) = \frac{1}{1 + \exp(-a)}$$

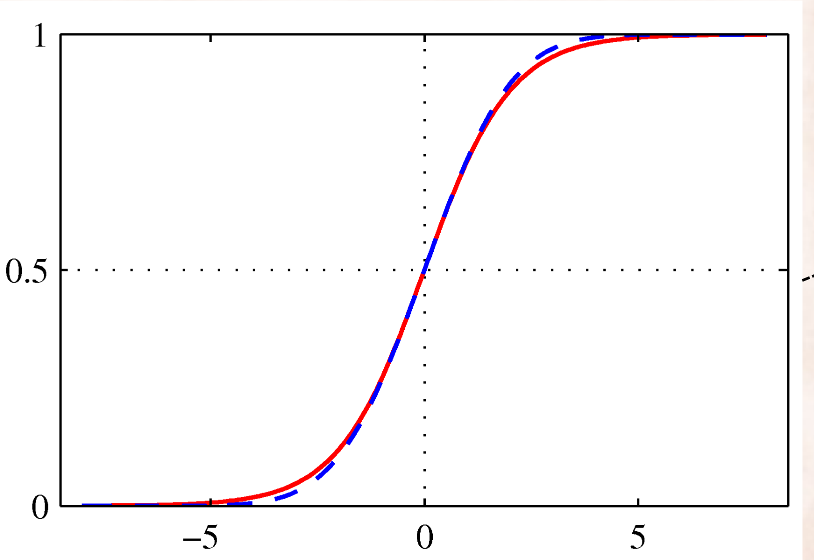
log odds

$$a(\mathbf{x}) = \ln \frac{p(\mathbf{x} \mid C_1) p(C_1)}{p(\mathbf{x} \mid C_2) p(C_2)} = \ln \frac{p(C_1 \mid \mathbf{x})}{p(C_2 \mid \mathbf{x})}$$

Probabilistic Generative Models: Binary Classification (K = 2)

- If $a(\mathbf{x})$ is a linear function of x $\Rightarrow p(C_1 \mid \mathbf{x})$ is a *generalized linear model*:

$$p(C_1 \mid \mathbf{x}) = \frac{1}{1+\exp(-a(\mathbf{x}))} = \sigma(a(\mathbf{x})) = \sigma(\boldsymbol{\lambda}^T \mathbf{x})$$



$\sigma(a)$ is a *squashing function*

Three Parametric Approaches to Classification

2) Probabilistic Discriminative Models: directly model the posterior class probabilities $p(C_k \mid \mathbf{x})$.
   - Inference and decision are separate.
   - Less data needed to estimate $p(C_k \mid \mathbf{x})$ than $p(\mathbf{x} \mid C_k)$.
   - Can accommodate many overlapping features.
     - Logistic Regression
     - Conditional Random Fields

Logistic Regression (K = 2)

- Directly model posterior class probabilities:

$$p(C_1 | \mathbf{x}) = \frac{1}{1 + \exp(-a(\mathbf{x}))} = \sigma(a(\mathbf{x})) = \sigma(\mathbf{w}^T \varphi(\mathbf{x}))$$

- Dataset $D = \{\langle \varphi(\mathbf{x}_n), t_n \rangle \mid t_n \in \{0,1\}, n \in 1 \ldots N\}$
- The likelihood function is:

$$p(\mathbf{t} | \mathbf{w}) = \prod_{n=1}^{N} y_n^{t_n} (1 - y_n)^{(1-t_n)}$$

$$y_n = p(C_1 | \mathbf{x}_n) \Leftrightarrow y_n = p(t_n = 1 | \mathbf{x}_n)$$

- ML solution is: $\mathbf{w}_{ML} = \arg\max_{\mathbf{w}} p(\mathbf{t} | \mathbf{w})$

Logistic Regression (K = 2)

- The negative log-likelihood error function is:

$$E(\mathbf{w}) = -\sum_{n=1}^{N} \left\{ t_n \ln y_n + (1 - t_n) \ln(1 - y_n) \right\}$$

- $\nabla E(\mathbf{w}) = 0 \Rightarrow$ ML solution is given by:

$$\nabla E(\mathbf{w}) = \sum_{n=1}^{N} (y_n - t_n) \varphi(\mathbf{x}_n)^T = 0$$

$\Rightarrow$ for every feature φ_i, the *expected value* on predicted D_+ should be the same as the *observed value* on D_+:

$$\sum_{n=1}^{N} \varphi_i(\mathbf{x}_n) p(t_n = 1 \mid \mathbf{x}_n) = \sum_{n=1}^{N} \varphi_i(\mathbf{x}_n) t_n = \sum_{n \in D_+} \phi_i(\mathbf{x}_n)$$

Logistic Regression vs. Linear Regression

- Logistic Regression solution:

$$\nabla E_D(\mathbf{w}) = \sum_{n=1}^{N} (t_n - y_n)\varphi(\mathbf{x}_n)^T = 0, \quad \text{where } \mathrm{y}_\mathrm{n} = \sigma(\mathbf{w}^T \varphi(\mathbf{x}_n))$$

- Linear Regression solution:

$$\nabla E_D(\mathbf{w}) = \sum_{n=1}^{N} (t_n - y_n)\varphi(\mathbf{x}_n)^T = 0, \quad \text{where } \mathrm{y}_\mathrm{n} = \mathbf{w}^T \varphi(\mathbf{x}_n)$$

- Like in linear regression, solution is prone to overfitting:
  - when data is linearly separable, ML solution is a hyperplane

    $\sigma(\mathbf{w}^T\mathbf{x}) = 0.5 \;\Leftrightarrow\; \mathbf{w}^T\mathbf{x} = 0$ and $\|\mathbf{w}\| = \infty$.

Regularized Logistic Regression

- Use a Gaussian prior over the parameters:

$$\mathbf{w} = [w_0, w_1, \ldots, w_M]^{\mathrm{T}}$$

$$p(\mathbf{w}) = N(\mathbf{0}, \alpha^{-1}\mathbf{I}) = \left(\frac{\alpha}{2\pi}\right)^{(M+1)/2} \exp\left\{-\frac{\alpha}{2}\mathbf{w}^T\mathbf{w}\right\}$$

- Bayes' Theorem:

$$p(\mathbf{w} \mid \mathbf{t}) = \frac{p(\mathbf{t} \mid \mathbf{w})\, p(\mathbf{w})}{p(\mathbf{t})} \propto p(\mathbf{t} \mid \mathbf{w})\, p(\mathbf{w})$$

- MAP solution:

$$\mathbf{w}_{MAP} = \arg\max_{\mathbf{w}} p(\mathbf{w} \mid \mathbf{t})$$

Regularized Logistic Regression

- MAP solution:

$$\mathbf{w}_{MAP} = \arg\max_{\mathbf{w}} p(\mathbf{w} \mid \mathbf{t}) \quad = \arg\max_{\mathbf{w}} p(\mathbf{t} \mid \mathbf{w}) p(\mathbf{w})$$

$$= \arg\min_{\mathbf{w}} - \ln p(\mathbf{t} \mid \mathbf{w}) p(\mathbf{w})$$

$$= \arg\min_{\mathbf{w}} - \ln p(\mathbf{t} \mid \mathbf{w}) - \ln p(\mathbf{w})$$

$$= \arg\min_{\mathbf{w}} E_D(\mathbf{w}) - \ln p(\mathbf{w})$$

$$= \arg\min_{\mathbf{w}} E_D(\mathbf{w}) + \frac{\alpha}{2} \mathbf{w}^T \mathbf{w} \quad \boxed{= \arg\min_{\mathbf{w}} E_D(\mathbf{w}) + E_{\mathbf{w}}(\mathbf{w})}$$

$$E_D(\mathbf{w}) = -\sum_{n=1}^{N} \left\{ t_n \ln y_n + (1 - t_n) \ln(1 - y_n) \right\} \dashrightarrow \boxed{\textit{data term}}$$

$$E_{\mathbf{w}}(\mathbf{w}) = \frac{\alpha}{2} \mathbf{w}^T \mathbf{w} \dashrightarrow \boxed{\textit{regularization term}}$$

Regularized Logistic Regression

- MAP solution:

$$\mathbf{w}_{MAP} = \arg\min_{\mathbf{w}} \boxed{E_D(\mathbf{w}) + E_{\mathbf{w}}(\mathbf{w})}$$

convex in $\mathbf{w}$

set $\nabla E_D(\mathbf{w}) + \nabla E_{\mathbf{w}}(\mathbf{w}) = 0$:

$$\Rightarrow \sum_{n=1}^{N} (y_n - t_n)\varphi(\mathbf{x}_n)^T + \alpha \mathbf{w}^T = 0, \text{ where } y_n = \sigma(\mathbf{w}^T \varphi(\mathbf{x}_n))$$

- Solve numerically:
  - Stochastic gradient descent [chapter 3.1.3].
  - Newton Raphson iterative optimization [chapter 4.3.3].

Multiclass Logistic Regression (K ≥ 2)

1) Train one weight vector per class [Chapter 4.3.4]:

$$p(C_k \mid \mathbf{x}) = \frac{\exp(\mathbf{w}_k^T \varphi(\mathbf{x}))}{\sum_j \exp(\mathbf{w}_j^T \varphi(\mathbf{x}))}$$

2) More general approach:

$$p(C_k \mid \mathbf{x}) = \frac{\exp(\mathbf{w}^T \varphi(\mathbf{x}, C_k))}{\sum_j \exp(\mathbf{w}^T \varphi(\mathbf{x}, C_j))}$$

- Inference:

$$C_* = \arg\max_{C_k} p(C_k \mid \mathbf{x})$$

Logistic Regression (K ≥ 2)

2) Inference in more general approach:

$$
\begin{aligned}
C_* &= \arg\max_{C_k} p(C_k \mid \mathbf{x}) \\
&= \arg\max_{C_k} \frac{\exp(\mathbf{w}^T \varphi(\mathbf{x}, C_k))}{\boxed{\sum_j \exp(\mathbf{w}^T \varphi(\mathbf{x}, C_j))}} \\
&= \arg\max_{C_k} \exp(\mathbf{w}^T \varphi(\mathbf{x}, C_k)) \\
&= \arg\max_{C_k} \mathbf{w}^T \varphi(\mathbf{x}, C_k)
\end{aligned}
$$

$Z(\mathbf{x})$ a normalization constant

- Training using:
  - Maximum Likelihood (ML)
  - Maximum A Posteriori (MAP) with a Gaussian prior on **w**.

Logistic Regression (K ≥ 2) with ML

- The negative log-likelihood error function is:

$$E_D(\mathbf{w}) = -\ln \prod_{n=1}^{N} p(t_n \mid \mathbf{x}_n) = \boxed{-\sum_{n=1}^{N} \ln \frac{\exp(\mathbf{w}^T \varphi(\mathbf{x}_n, t_n))}{Z(\mathbf{x}_n)}}$$

$$\mathbf{w}_{ML} = \arg\min_{\mathbf{w}} E_D(\mathbf{w})$$

convex in $\mathbf{w}$

- The gradient is (prove it):

$$\nabla E_D(\mathbf{w}) = \left[\frac{\partial E_D(\mathbf{w})}{\partial w_0}, \frac{\partial E_D(\mathbf{w})}{\partial w_1}, \ldots, \frac{\partial E_D(\mathbf{w})}{\partial w_M} \right]$$

$$\frac{\partial E_D(\mathbf{w})}{\partial w_i} = -\sum_{n=1}^{N} \varphi_i(\mathbf{x}_n, t_n) + \sum_{n=1}^{N} \sum_{k=1}^{K} p(C_k \mid \mathbf{x}_n) \varphi_i(\mathbf{x}_n, C_k)$$

Logistic Regression (K ≥ 2) with ML

- Set $\nabla E_D(\mathbf{w}) = 0 \Rightarrow$ ML solution satisfies:

$$\sum_{n=1}^{N} \varphi_i(\mathbf{x}_n, t_n) = \sum_{n=1}^{N} \sum_{k=1}^{K} p(C_k \mid \mathbf{x}_n) \varphi_i(\mathbf{x}_n, C_k)$$

$\Rightarrow$ for every feature φ_i, the *observed value* on D should be the same as the *expected value* on D!

- Solve numerically:
  - Stochastic gradient descent [chapter 3.1.3].
  - Newton Raphson iterative optimization (large Hessian!).
  - Limited memory Newton methods (e.g. L-BFGS).

The Maximum Entropy Principle

- Principle of Insufficient Reason
- Principle of Indifference
  - can be traced back to Pierre Laplace and Jacob Bernoulli.

➢ A. L. Berger, S. A. Della Pietra, and V. J. Della Pietra. 1996. A maximum entropy approach to natural language processing. Computational Linguistics, 22(1).
  - "*model all that is known and assume nothing about that which is unknown*".
  - "*given a collection of facts, choose a model consistent with all the facts, but otherwise as uniform as possible*".

Maximum Likelihood ⇔ Maximum Entropy

1) Maximize conditional likelihood:

$$\mathbf{w}_{ML} = \arg\max_{\mathbf{w}} p(\mathbf{t} \mid \mathbf{w})$$

$$p(\mathbf{t} \mid \mathbf{w}) = \prod_{n=1}^{N} p_{\mathbf{w}}(t_n \mid \mathbf{x}_n) = \prod_{n=1}^{N} \frac{\exp(\mathbf{w}^T \varphi(\mathbf{x}_n, t_n))}{Z(\mathbf{x}_n)}$$

2) Maximize conditional entropy:

$$p_{ME} = \arg\max_{p} \sum_{n=1}^{N} \sum_{k=1}^{K} - p(C_k \mid \mathbf{x}_n) \log p(C_k \mid \mathbf{x}_n)$$

subject to:

$$\sum_{n=1}^{N} \varphi(\mathbf{x}_n, t_n) = \sum_{n=1}^{N} \sum_{k=1}^{K} p(C_k \mid \mathbf{x}_n) \varphi(\mathbf{x}_n, C_k)$$

$\Rightarrow$ solution is: $p_{ME}(t_n \mid \mathbf{x}_n) = p_{\mathbf{w}_{ML}}(t_n \mid \mathbf{x}_n) = \dfrac{\exp(\mathbf{w}_{ML}^T \varphi(\mathbf{x}_n, t_n))}{Z(\mathbf{x}_n)}$