

## Regularized Linear Models

CS6780 – Advanced Machine Learning  
Spring 2015

Thorsten Joachims  
Cornell University

Reading: Murphy 3.1-3.2  
Murphy 8.1-8.3, Murphy 7.5

## Discriminative ERM Learning

- Modeling Step:
  - Select classification rules  $H$  to consider (hypothesis space)
- Training Principle:
  - Given training sample  $(\vec{x}_1, y_1), \dots, (\vec{x}_n, y_n)$
  - Find  $h$  from  $H$  with lowest training error  
→ Empirical Risk Minimization
  - Argument: generalization error bounds → low training error leads to low prediction error, if overfitting is controlled.
- Examples: SVM, decision trees, Perceptron

## Bayes Decision Rule

- Assumption:
  - learning task  $P(X, Y) = P(Y|X)P(X)$  is known
- Question:
  - Given instance  $x$ , how should it be classified to minimize prediction error?
- Bayes Decision Rule (for zero/one loss):

$$h_{\text{bayes}}(\vec{x}) = \operatorname{argmax}_{y \in Y} [P(Y = y|X = \vec{x})]$$

## Generative vs. Conditional vs. ERM

- Empirical Risk Minimization
  - Find  $h = \operatorname{argmin}_{h \in H} \operatorname{Err}_S(h)$  s.t. overfitting control
  - Pro: directly estimate decision rule
  - Con: committed to loss,  $X, Y$
- Discriminative Conditional Model
  - Find  $P(Y|X)$ , then derive  $h(x)$  via Bayes rule
  - Pro: not committed to loss
  - Con: committed to  $X, Y$ ; conditional distributions more complex than decision rule
- Generative Model
  - Find  $P(X, Y)$ , then derive  $h(x)$  via Bayes rule
  - Pro: not committed to loss function,  $X$ , and  $Y$ ; often computationally easy
  - Con: Model dependencies in  $X$

## Logistic Regression

- Data:
  - $S = ((x_1, y_1) \dots (x_n, y_n))$ ,  $x \in \mathcal{R}^N$  and  $y \in \{-1, +1\}$
- Model:
  - $P(y|x, w) = \operatorname{Ber}(y|\operatorname{sigm}(w \cdot x))$
- Training objective:

$$\hat{w} = \operatorname{argmin}_w \sum_{i=1}^n \log(1 + \exp(-y_i w \cdot x_i))$$

- Algorithm:
  - Stochastic gradient descent, Newton, etc.

## Regularized Logistic Regression

- Data:
  - $S = ((x_1, y_1) \dots (x_n, y_n))$ ,  $x \in \mathcal{R}^N$  and  $y \in \{-1, +1\}$
- Model:
  - $P(y|x, w) = \operatorname{Ber}(y|\operatorname{sigm}(w \cdot x))$ ,  $P(w) = N(w|0, \Sigma)$
- Training objective:

$$\hat{w} = \operatorname{argmin}_w \frac{1}{2} w \cdot w + C \sum_{i=1}^n \log(1 + \exp(-y_i w \cdot x_i))$$

- Algorithm:
  - Stochastic gradient descent, Newton, etc.

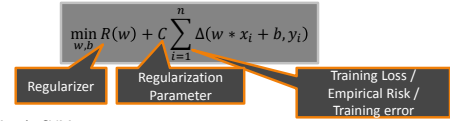
## Ridge Regression

- **Data:**
  - $S = ((x_1, y_1) \dots (x_n, y_n))$ ,  $x \in \mathfrak{R}^N$  and  $y \in \mathfrak{R}$
- **Model:**
  - $P(y|x, w) = N(y|w \cdot x, E)$ ,  $P(w) = N(w|0, \Sigma)$
- **Training objective:**

$$\hat{w} = \operatorname{argmin}_w \frac{1}{2} w \cdot w + C \sum_{i=1}^n (w \cdot x_i - y_i)^2$$

- **Algorithm:**
  - $\hat{w} = (\operatorname{diag}(C) + X^T X)^{-1} X^T y$

## Discriminative Training of Linear Rules



- |  |  |
|--|--|
| <ul style="list-style-type: none"> <li>• <b>Soft-Margin SVM</b> <ul style="list-style-type: none"> <li>– <math>R(w) = \frac{1}{2} w \cdot w</math></li> <li>– <math>\Delta(\bar{y}, y_i) = \max(0, 1 - y_i \bar{y})</math></li> </ul> </li> <li>• <b>Perceptron</b> <ul style="list-style-type: none"> <li>– <math>R(w) = 0</math></li> <li>– <math>\Delta(\bar{y}, y_i) = \max(0, -y_i \bar{y})</math></li> </ul> </li> <li>• <b>Linear Regression</b> <ul style="list-style-type: none"> <li>– <math>R(w) = 0</math></li> <li>– <math>\Delta(\bar{y}, y_i) = (y_i - \bar{y})^2</math></li> </ul> </li> </ul> | <ul style="list-style-type: none"> <li>• <b>Ridge Regression</b> <ul style="list-style-type: none"> <li>– <math>R(w) = \frac{1}{2} w \cdot w</math></li> <li>– <math>\Delta(\bar{y}, y_i) = (y_i - \bar{y})^2</math></li> </ul> </li> <li>• <b>Lasso</b> <ul style="list-style-type: none"> <li>– <math>R(w) = \frac{1}{2} \sum  w_i </math></li> <li>– <math>\Delta(\bar{y}, y_i) = (y_i - \bar{y})^2</math></li> </ul> </li> <li>• <b>Regularized Logistic Regression / Conditional Random Field</b> <ul style="list-style-type: none"> <li>– <math>R(w) = \frac{1}{2} w \cdot w</math></li> <li>– <math>\Delta(\bar{y}, y_i) = \log(1 + e^{-y_i \bar{y}})</math></li> </ul> </li> </ul> |
|--|--|