Logistic Regression

Rishabh Iyer

University of Texas at Dallas

based on the slides of Nick Rouzzi and Vibhav Gogate

Last Time

- Supervised learning via naive Bayes
  - Use MLE to estimate a distribution $p(x, y) = p(y)p(x|y)$
  - Classify by looking at the conditional distribution, $p(y|x)$
- Today: logistic regression

Discriminative Functions
[Perceptron, SVMs]

1. Define a hypothesis function $h(x, \theta)$
2. Define a loss function $L(h(x, \theta), y)$
3. Solving a learning problem

$$\sum_{i=1}^{M} L(h(x_i, \theta), y_i)$$

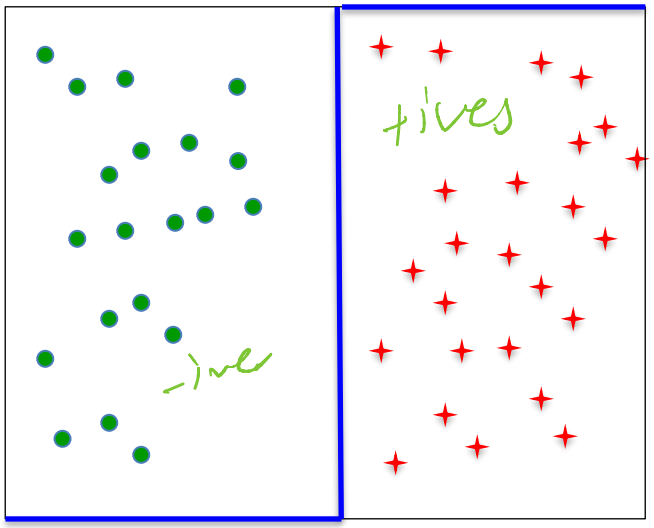
Probabilistic Learning
[Naive Bayes, Logistic Reg]

1. Define a hypothesis $h(x, \theta)$
2. Define a corresponding probability distribution
   $p(x, y, \theta)$, $p(y | x, \theta)$
   ↑ NB ↑ LR.
3. Solving a learning problem
   - Maximize MLE, MAP

Logistic Regression

- Learn $p(Y|X)$ directly from the data
  - Assume a particular functional form, e.g., a linear classifier $p(Y = 1|x) = 1$ on one side and 0 on the other
  - Not differentiable…
    - Makes it difficult to learn
    - Can't handle noisy labels

$p(Y = 1|x) = 0$



$p(Y = 1|x) = 1$

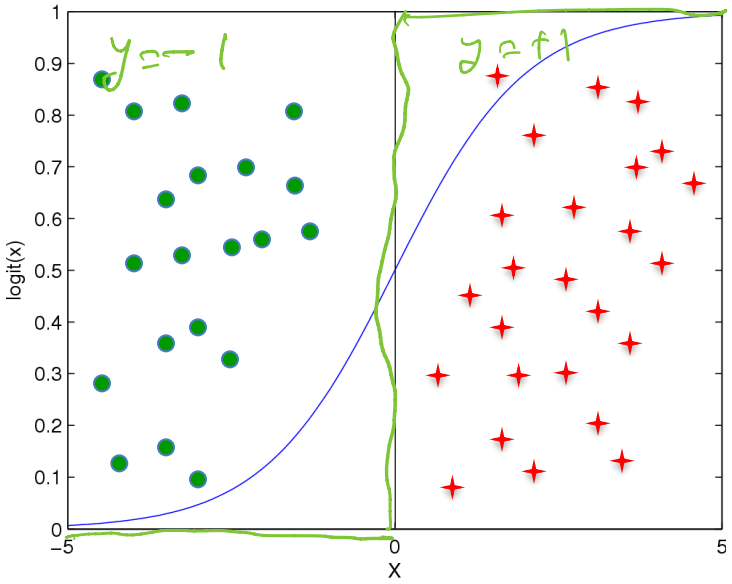
Logistic Regression

$$p(y|x) = \frac{1}{1+\exp(-y.(w^T x+b))}$$

- Learn $p(y|x)$ directly from the data
  - Assume a particular functional form

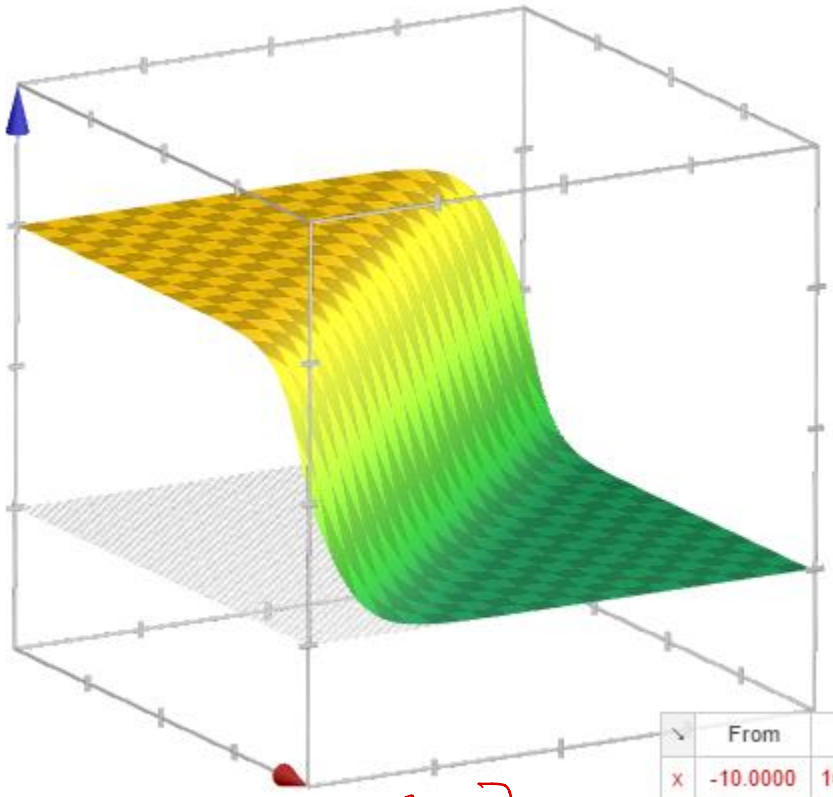
$$p(Y=-1|x) = \frac{1}{1+\exp(w^T x+b)}$$

$$p(Y=1|x) = \frac{\exp(w^T x+b)}{1+\exp(w^T x+b)}$$



$$p(Y=-1|x) + p(Y=+1|x) = 1$$

Logistic Function in m Dimensions



	From	To
x	-10.0000	10.0000
y	-10.0000	10.0000
z	-0.499119	1.49970

$$p(Y = -1|x) = \frac{1}{1 + \exp(w^T x + b)}$$

Can be applied to discrete and continuous features

$x \in [0, T]$

NB → Discr → [0, 0.1, 0.2, ...] (10 bins), 1, 2 ...

$x_f = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \end{bmatrix}$

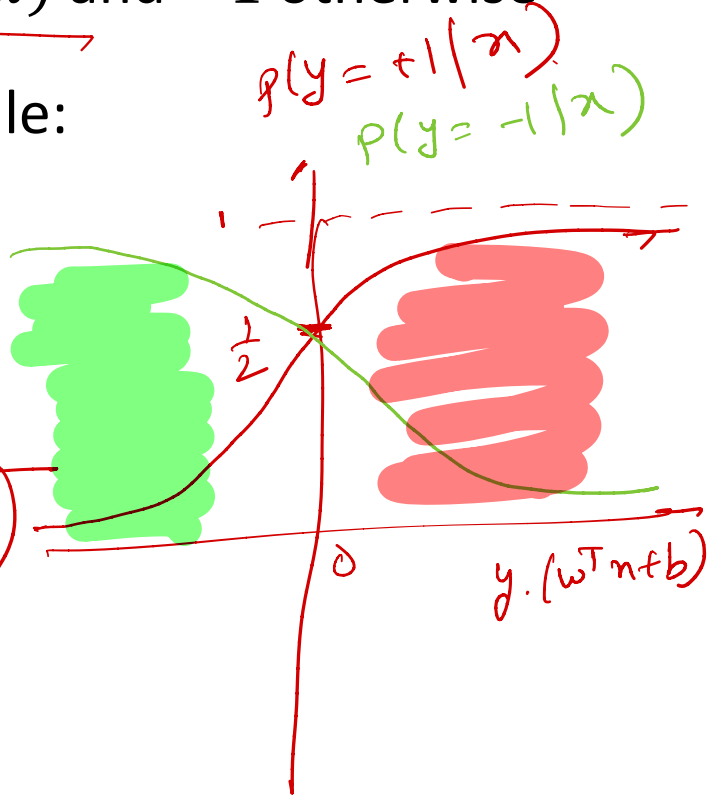
Sunny, Rain.

$f = [\text{Rain}, \text{Sunny}]$

One Hot encoding.

Functional Form: Two classes

- Given some w and b, we can classify a new point x by assigning the label 1 if $p(Y = 1|x) > p(Y = -1|x)$ and -1 otherwise
  - This leads to a linear classification rule:
    - Classify as a 1 if $w^T x + b > 0$
    - Classify as a -1 if $w^T x + b < 0$



$$p(y = +1|x) = \frac{1}{1 + \exp(-(w^T x + b))}$$

$$p(y = -1|x) = \frac{1}{1 + \exp(w^T x + b)}$$

Learning the Weights

Learning = $\{(x_1, y_1), \ldots, (x_M, y_M)\}$ MCLE

- To learn the weights, we maximize the conditional likelihood

$$p(y_D \mid x_D, \theta) = (w^*, b^*) = \arg\max_{w,b} \prod_{i=1}^{M} p(y^{(i)} \mid x^{(i)}, w, b)$$

w, b

- This is the not the same strategy that we used in the case of naive Bayes
  - For naive Bayes, we maximized the log-likelihood

Generative vs. Discriminative Classifiers

Generative classifier:
(e.g., Naïve Bayes)

- Assume some **functional form** for $p(x|y), p(y)$
- Estimate parameters of $p_\theta(x|y)$, $p_\theta(y)$ directly from training data
- Use Bayes rule to calculate $p(y|x) = \frac{p(x,y)}{p(x)} = \frac{p(x|y)\,p(y)}{p(x)}$
- This is a **generative model**
  - **Indirect** computation of $p(Y|X)$ through Bayes rule
  - As a result, **can also generate a sample of the data**, $p(x) = \sum_y p(y)p(x|y)$

Discriminative classifiers:
(e.g., Logistic Regression)

- Assume some **functional form for** $p(y|x)$
- Estimate parameters of $p(y|x)$ directly from training data
- This is a discriminative model
  - Directly learn $p(y|x)$
  - But **cannot obtain a sample of the data** as $p(x)$ is not available
  - Useful for discriminating labels

Learning the Weights

$$
\begin{aligned}
\ell(w,b) &= \ln \prod_{i=1}^{N} p(y^{(i)}|x^{(i)},w,b) \\
&= \sum_{i=1}^{N} \ln p(y^{(i)}|x^{(i)},w,b) \\
&= \sum_{i=1}^{N} \frac{y^{(i)}+1}{2} \ln p(Y=1|x^{(i)},w,b) + \left(1-\frac{y^{(i)}+1}{2}\right) \ln p(Y=-1|x^{(i)},w,b) \\
&= \sum_{i=1}^{N} \frac{y^{(i)}+1}{2} \ln \frac{p\left(Y=1\middle|x^{(i)},w,b\right)}{p\left(Y=-1\middle|x^{(i)},w,b\right)} + \ln p(Y=-1|x^{(i)},w,b) \\
&= \sum_{i=1}^{N} \frac{y^{(i)}+1}{2} \left(w^T x^{(i)}+b\right) - \ln\left(1+\exp\left(w^T x^{(i)}+b\right)\right)
\end{aligned}
$$

$$D = \{(x_1, y_1), \ldots, (x_M, y_M)\}$$

$$p(y_D \mid x_D, \theta) = \prod_{i=1}^{M} p(y_i \mid x_i, \theta) \quad (w, b)$$

$$= \prod_{i=1}^{m} \frac{1}{1 + \exp\left[-y_i \left[w^T x_i + b\right]\right)}$$

Log: $$\log p(y_D \mid x_D, \theta) = \sum_{i=1}^{m} \log \left[\frac{1}{1 + \exp\left(-y_i (w^T x_i + b)\right)}\right]$$

MLE: $$\max_{\theta} \log p(y_D \mid x_D, b)$$

$$= \max_{w,b} - \sum_{i=1}^{M} \log \left[1 + \exp\left[-y_i (w^T x_i + b)\right)\right]$$

$$\Rightarrow \min_{w,b} \sum_{i=1}^{m} \log \left[1 + \exp\left(-y_i (w^T x_i + b)\right)\right]$$

Learning the Weights

$$
\begin{aligned}
\ell(w,b) &= \ln \prod_{i=1}^{N} p(y^{(i)}|x^{(i)},w,b) \\
&= \sum_{i=1}^{N} \ln p(y^{(i)}|x^{(i)},w,b) \\
&= \sum_{i=1}^{N} \frac{y^{(i)}+1}{2} \ln p(Y=1|x^{(i)},w,b) + \left(1-\frac{y^{(i)}+1}{2}\right) \ln p(Y=-1|x^{(i)},w,b) \\
&= \sum_{i=1}^{N} \frac{y^{(i)}+1}{2} \ln \frac{p\left(Y=1\middle|x^{(i)},w,b\right)}{p\left(Y=-1\middle|x^{(i)},w,b\right)} + \ln p(Y=-1|x^{(i)},w,b) \\
&= \sum_{i=1}^{N} \frac{y^{(i)}+1}{2} \left(w^T x^{(i)}+b\right) - \ln\left(1+\exp\left(w^T x^{(i)}+b\right)\right)
\end{aligned}
$$

This is concave in w and b: take derivatives and solve!

Optimization Problem

$$\min_{w,b} \sum_{i=1}^{M} \log\left[1 + \exp\left(-y_i\left(w^T x_i + b\right)\right)\right] = L(w,b)$$

Convex - Minimization

Gradient Descent

$$\nabla_w L(w,b) = \sum_{i=1}^{M} \frac{-\exp\left(-y_i(w^T x_i + b)\right)}{1+\exp\left(-y_i(w^T x_i + b)\right)} y_i x_i$$

$$\nabla_b L(w,b) = \sum_{i=1}^{M} \frac{-\exp\left(-y_i(w^T x_i + b)\right)}{1+\exp\left(-y_i(w^T x_i + b)\right)} \cdot y_i$$

Learning the Weights

$$
\begin{aligned}
\ell(w,b) &= \ln \prod_{i=1}^{N} p(y^{(i)}|x^{(i)},w,b) \\
&= \sum_{i=1}^{N} \ln p(y^{(i)}|x^{(i)},w,b) \\
&= \sum_{i=1}^{N} \frac{y^{(i)}+1}{2} \ln p(Y=1|x^{(i)},w,b) + \left(1-\frac{y^{(i)}+1}{2}\right) \ln p(Y=-1|x^{(i)},w,b) \\
&= \sum_{i=1}^{N} \frac{y^{(i)}+1}{2} \ln \frac{p\left(Y=1\middle|x^{(i)},w,b\right)}{p\left(Y=-1\middle|x^{(i)},w,b\right)} + \ln p(Y=-1|x^{(i)},w,b) \\
&= \sum_{i=1}^{N} \frac{y^{(i)}+1}{2} \left(w^T x^{(i)}+b\right) - \ln\left(1+\exp\left(w^T x^{(i)}+b\right)\right)
\end{aligned}
$$

No closed form solution ☹

Learning the Weights

- Can apply gradient ascent to maximize the conditional likelihood

$$\frac{\partial \ell}{\partial b} = \sum_{i=1}^{N} \left[\frac{y^{(i)} + 1}{2} - p(Y = 1 | x^{(i)}, w, b) \right]$$

$$\frac{\partial \ell}{\partial w_j} = \sum_{i=1}^{N} x_j^{(i)} \left[\frac{y^{(i)} + 1}{2} - p(Y = 1 | x^{(i)}, w, b) \right]$$

Gradient - Descent

w^0 = Arbitrary, b^0 = Arbitrary, α = Learning Rate

for $t = 0 : T$

$$w^{t+1} = w^t - \alpha D_w L(w,b)\Big|_{w=w^t, b=b^t}$$

$$b^{t+1} = b^t - \alpha D_b L(w,b)\Big|_{b=b^t, w=w^t}$$

if $\left[\left|L(w^{t+1}, b^{t+1}) - L(w^t, b^t)\right| < \varepsilon\right]$

end.

end for.

Priors

- Can define priors on the weights to prevent overfitting
  - Normal distribution, zero mean, identity covariance

$$p(w) = \prod_j \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{w_j^2}{2\sigma^2}\right)$$

  - "Pushes" parameters towards zero
- Regularization
  - Helps avoid very large weights and overfitting

Priors as Regularization

- The log-MAP objective with this Gaussian prior is then

$$\max_{w,b} \left[\ln \prod_{i=1}^{N} p\left(y^{(i)} \middle| x^{(i)}, w, b\right) p(w) p(b) \right] = \left[\sum_{i}^{N} \ln p\left(y^{(i)} \middle| x^{(i)}, w, b\right) \right] - \frac{\lambda}{2} \|w\|_2^2 - \frac{\lambda}{2} b^2$$

Cond. Lik

- Quadratic penalty: drives weights towards zero
- Adds a negative linear term to the gradients
- Different priors can produce different kinds of regularization

$$\min_{w,b} \; L(w,b) + \underbrace{\frac{\lambda}{2}\|w\|^2 + \frac{\lambda}{2} b^2}_{\text{Reg.}}$$

Priors as Regularization

- The log-MAP objective with this Gaussian prior is then

$$\ln \prod_{i=1}^{N} p\left(y^{(i)} \middle| x^{(i)}, w, b\right) p(w) p(b) = \left[\sum_{i}^{N} \ln p\left(y^{(i)} \middle| x^{(i)}, w, b\right) \right] - \frac{\lambda}{2} \|w\|_2^2 - \frac{\lambda}{2} b^2$$

  - Quadratic penalty: drives weights towards zero
  - Adds a negative linear term to the gradients
  - Different priors can produce different kinds of regularization

Somtimes called an ℓ_2 regularizer

Loss-Function Perspective.

Perceptron: $L_i^P(w,b) = \max\left(0, -y_i(w^T x_i + b)\right)$

SVM: $L_i^{SVM}(w,b) = \max\left(0, 1 - y_i(w^T x_i + b)\right)$

LogReg: $L^{LR}(w,b) = \log\left[1 + \exp\left(-y_i(w^T x_i + b)\right)\right]$

Classfn: $y_i(w^T x_i + b) > 0$ [Good] $\Rightarrow$ Loss should be low.

$y_i(w^T x_i + b) < 0$ [Bad] $\Rightarrow$ Loss should be high.

$z = y(w^T x + b)$

Perc: $\max(0, -z)$ ← high if $z \ll 0$

SVM: $\max(0, 1-z)$ ← "

LogReg: $\log(1 + \exp(-z))$

Regularization

$$\min_{w,b} \sum_{i=1}^{n} L(x_i, y_i, \underbrace{w, b}_{\theta})$$

L2 - Reg.

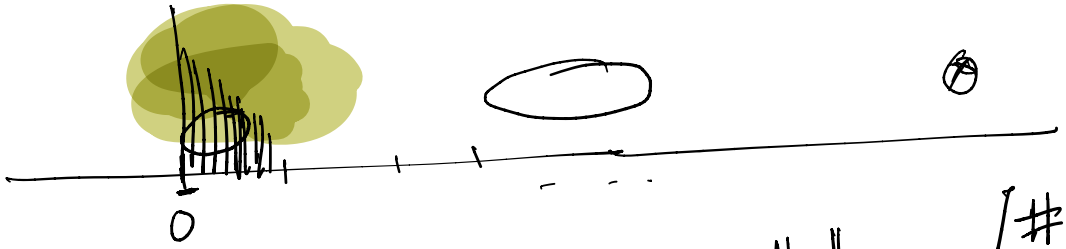
$$\min_{w,b} \sum_{i=1}^{n} L(x_i, y_i, w, b) + \frac{\lambda}{2}\|w\|^2 + \frac{\lambda}{2}b^2$$

L1 - Reg.

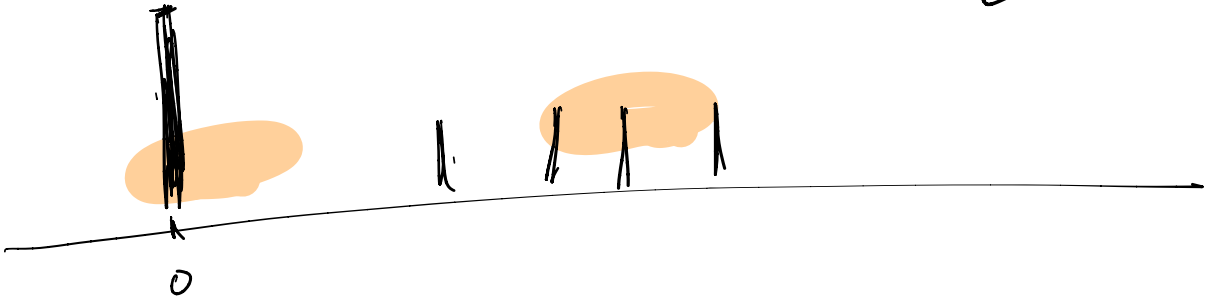
$$\min_{w,b} \sum_{i=1}^{n} L(x_i, y_i, w, b) + \lambda\|w\|_1 + \lambda|b|$$

$$\|w\|_1 = \sum_{i=1}^{d} |w_i|$$

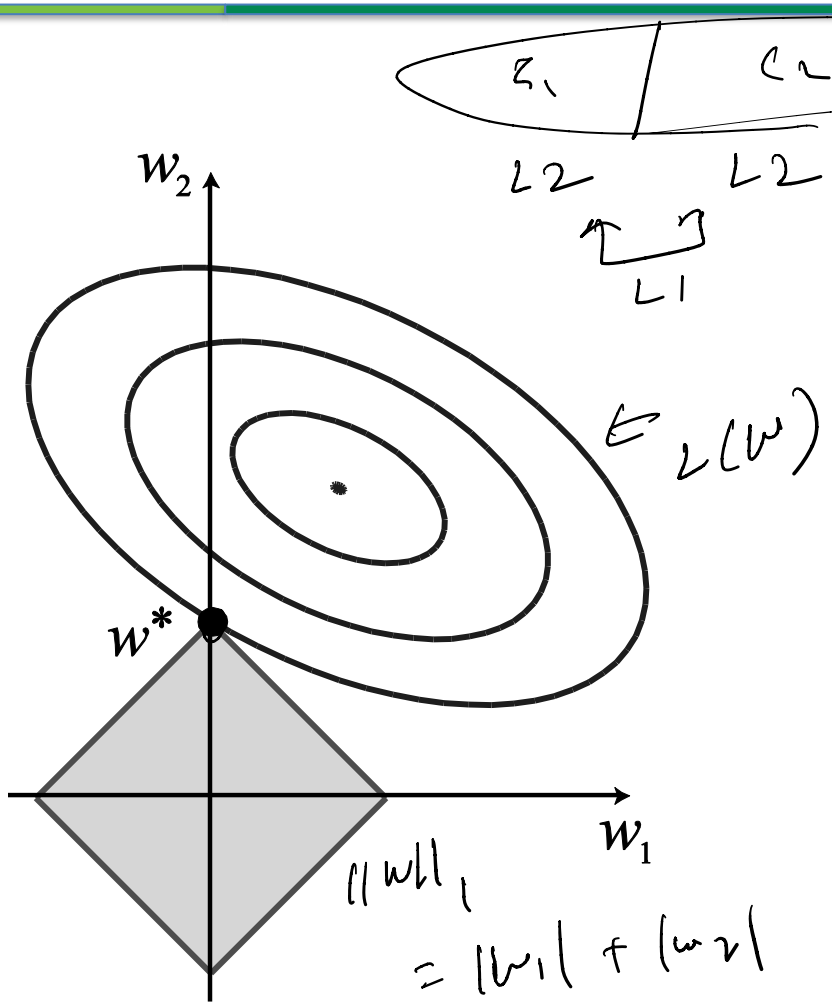
$$\|w\|^2 = \sum_{i=1}^{d} [w_i]^2 \quad \Big| \quad \|w\|_1 = \sum_{i=1}^{d} |w_i|$$



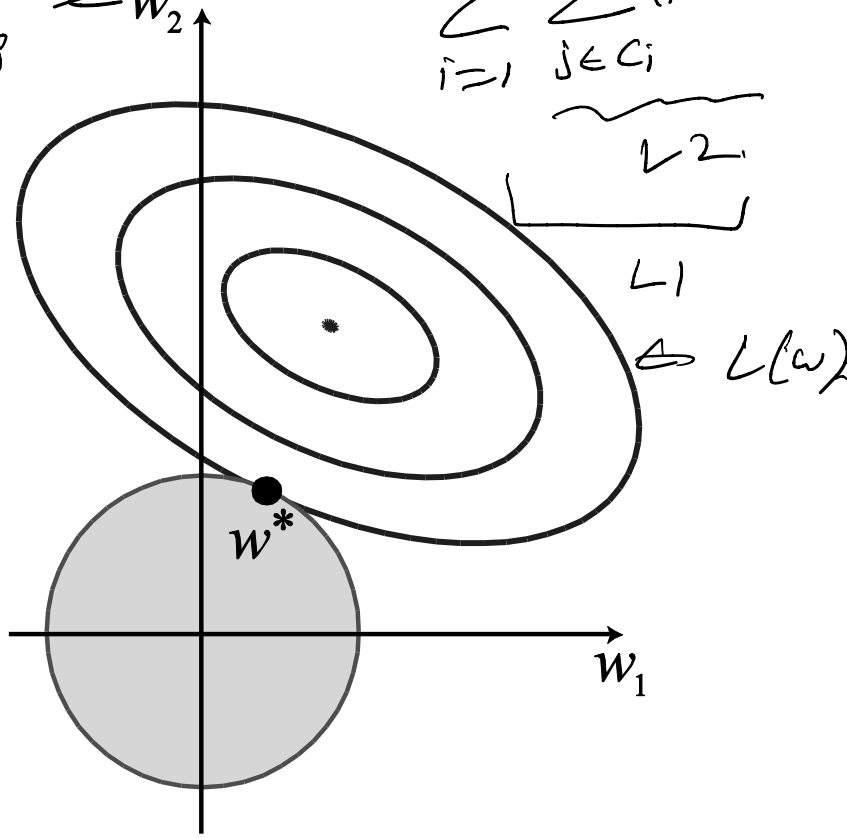
$$\|w\|_0 = \left[\begin{array}{c} \#\text{Non-zero values in} \\ w \end{array} \right]$$



Regularization



ℓ_1



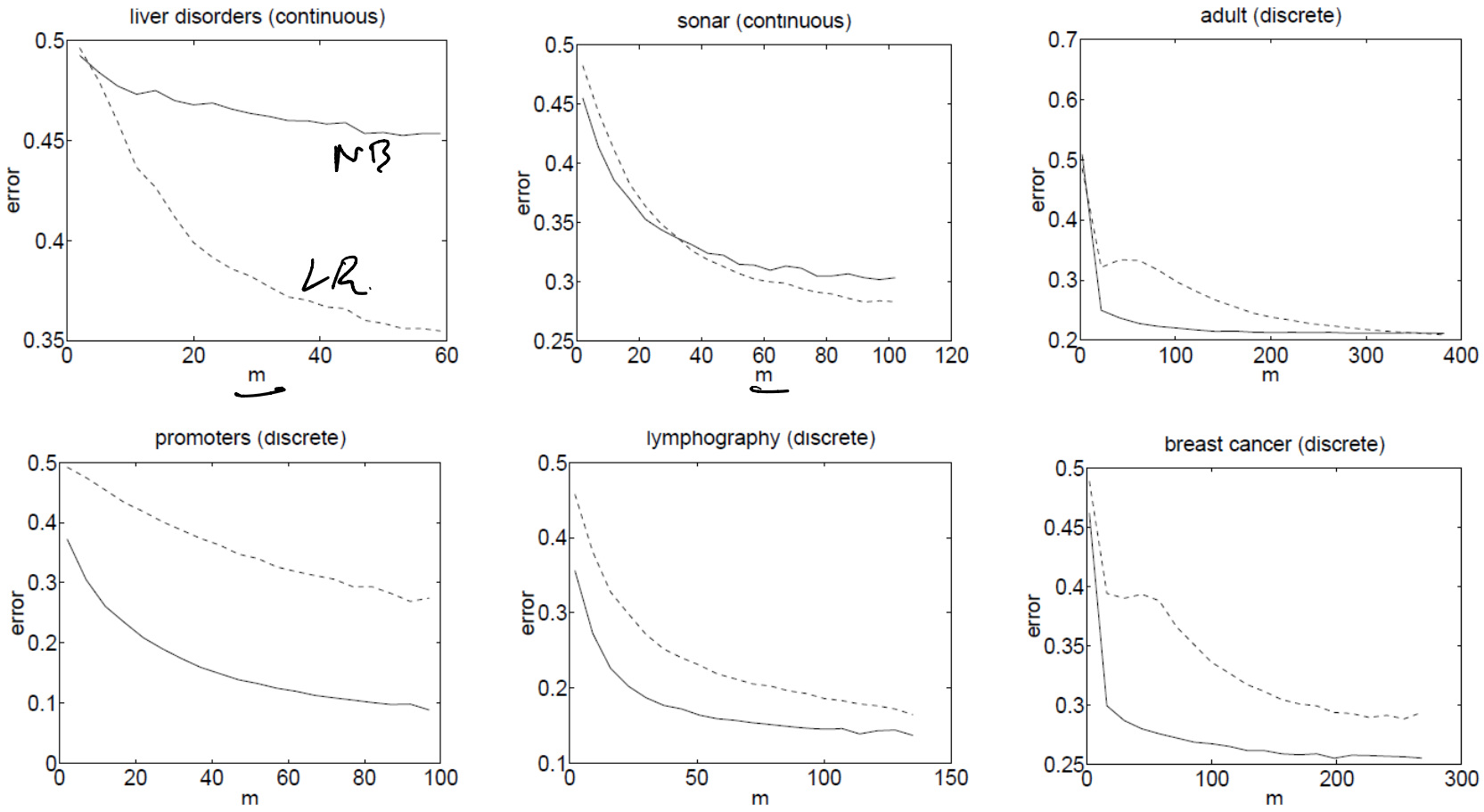
ℓ_2

Naïve Bayes vs. Logistic Regression

- Non-asymptotic analysis (for Gaussian NB)
  - Convergence rate of parameter estimates as size of training data tends to infinity (n = # of attributes in X)
    - Naïve Bayes needs $O(\log n)$ samples
      - NB converges quickly to its (perhaps less helpful) asymptotic estimates
    - Logistic Regression needs $O(n)$ samples
      - LR converges more slowly but makes no independence assumptions (typically less biased)

[Ng & Jordan, 2002]

NB vs. LR (on UCI datasets)



Naïve bayes

Logistic Regression

Sample size m

[Ng & Jordan, 2002]

LR in General

- Suppose that $y \in \{1, \ldots, R\}$, i.e., that there are R different class labels
- Can define a collection of weights and biases as follows
  - Choose a vector of biases and a matrix of weights such that for $y \neq R$

$$p(Y = k|x) = \frac{\exp(b_k + \sum_i w_{ki} x_i)}{1 + \sum_{j<R} \exp\left(b_j + \sum_i w_{ji} x_i\right)}$$

and

$$p(Y = R|x) = \frac{1}{1 + \sum_{j<R} \exp\left(b_j + \sum_i w_{ji} x_i\right)}$$

$$L(x, y, \theta) = \frac{1}{1 + \exp[-y[w^T x + b]]}$$

obs prob

webpage → click. } classfn

features → 0 1

webpage → CTR. } Regression

feats → 0.1

$w^T x + b$ → $\frac{1}{1 + \exp[-(w^T x + b)]}$