



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



$$\underset{y}{\operatorname{argmax}} \quad p(y|x)$$

Discriminative Functions [Perception, SVMs]

① Define a hypothesis function $h(x, \theta)$

② Define a loss function

$$L(h(x_i, \theta), y_i)$$

③ Solving a learning

problem

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

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

$$h(x, \theta)$$

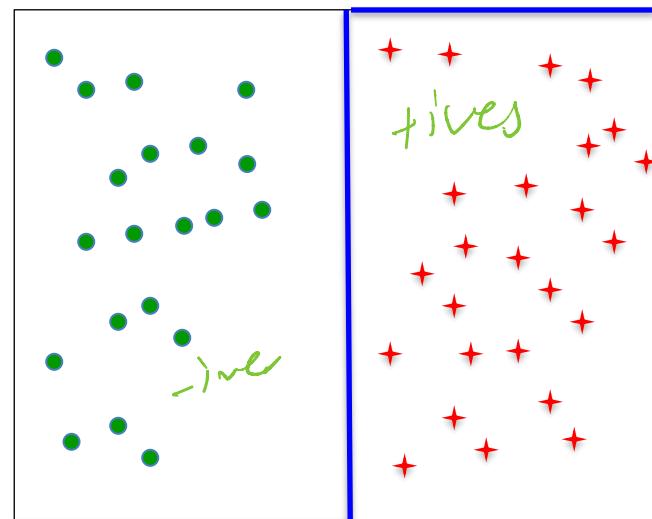
- Assume a particular functional form, e.g., a linear classifier
 $p(Y = 1|x) = 1$ on one side and 0 on the other

- Not differentiable...

- Makes it difficult to learn
- Can't handle noisy labels

$$\underline{p(Y=1|x)} = \underline{1}_{h(x,\theta) > 0}$$

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



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

Logistic Regression

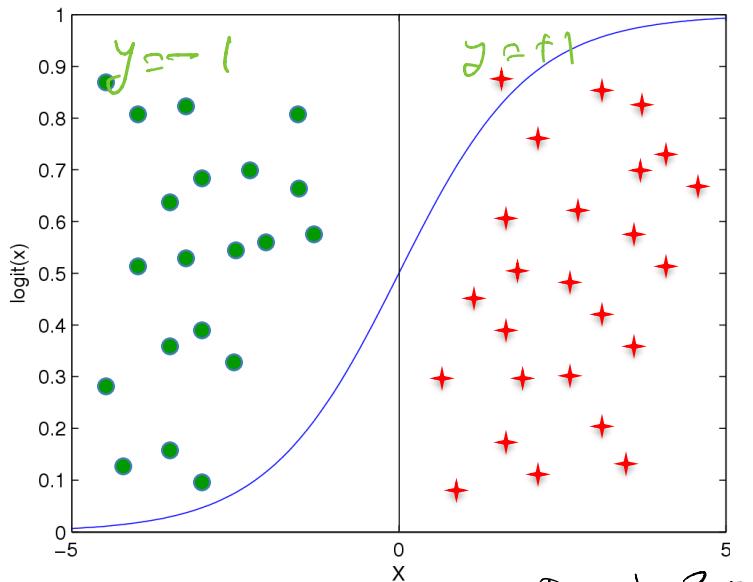
- Learn $p(y|x)$ directly from the data
 - Assume a particular functional form

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

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

$$P(Y=1|x) = \frac{\exp(h(x, \theta))}{1 + \exp(h(x, \theta))}$$

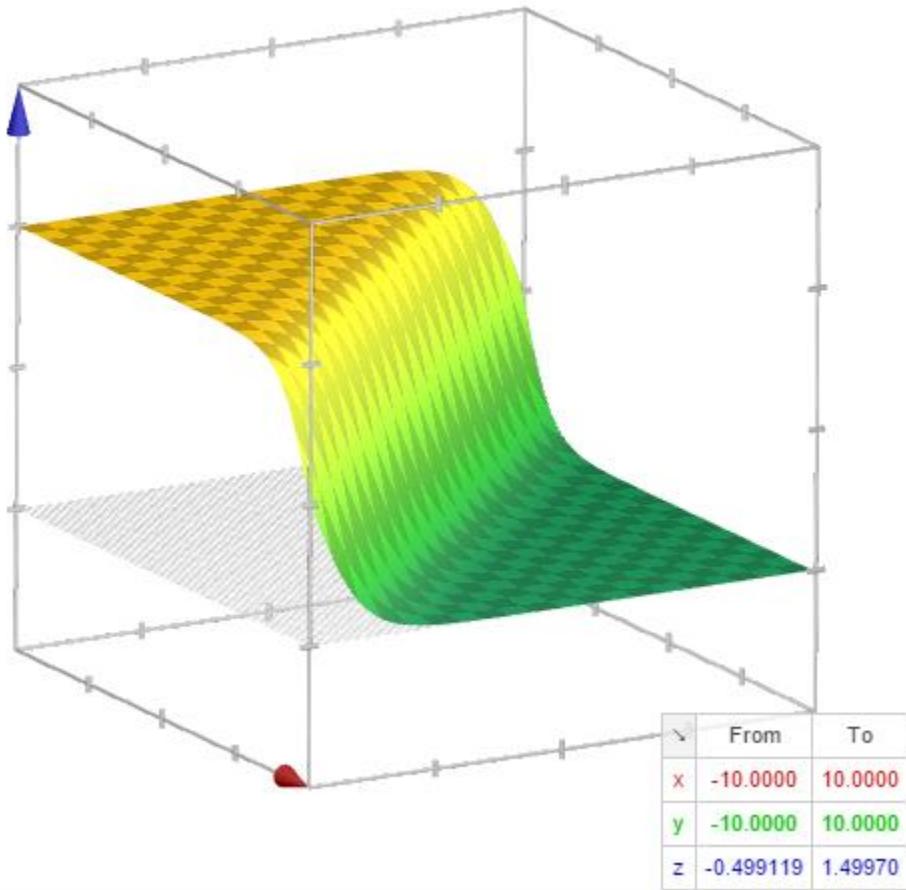
$$p(y|x) = \frac{1}{1 + \exp(-y \cdot (w^T x + b))}$$



$$w^T x + b \leq 0$$

$$w^T x + b > 0$$

Logistic Function in m Dimensions



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

Can be applied to
discrete and
continuous features

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 $w^T x + b < 0$

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

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

Q: What are functional conditions for which
 $p(Y=+|x) \geq p(Y=-|x)$ are good
fn's?

① Valid prob: $p(Y=+|x) + p(Y=-|x) = 1$

② $0 \leq p(Y|x) \leq 1$

③ Fn's are cont / diff
 $p(Y=+|x) \geq p(Y=-|x)$
Classfn Rule: $\Rightarrow Y=+$

Learning the Weights

- To learn the weights, we maximize the **conditional likelihood**

$$(w^*, b^*) = \arg \max_{w,b} \prod_{i=1}^N p(y^{(i)} | x^{(i)}, w, b)$$

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

$$\underset{w, b}{\operatorname{argmax}} P(y^{(1)}, \dots, y^{(m)} | x^{(1)}, \dots, x^{(m)}, w, b)$$

$$= \prod_{i=1}^m P(y^{(i)} | x^{(i)}, w, b).$$

Generative vs. Discriminative Classifiers



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

- Assume some **functional form** for $p(x|y), p(y)$
- Estimate parameters of $p(x|y)$, $p(y)$ directly from training data
- Use Bayes rule to calculate $p(y|x)$
- 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

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

Learning the Weights

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

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

Learning the Weights

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

No closed form solution 😞

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

$$p(y_D | x_D, \theta) = \prod_{i=1}^M p(y_i | x_i, \theta) \xrightarrow{(\omega, b)}$$

$$= \prod_{i=1}^M \frac{1}{1 + \exp(-y_i (\omega^T x_i + b))}$$

$$\text{Log : } \log P(y_D | x_D, \theta) = \sum_{i=1}^M \log \left[\frac{1}{1 + \exp(-y_i (\omega^T x_i + b))} \right]$$

$$(\text{MLE : } \max_{\theta} \log P(y_D | x_D, \theta))$$

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

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

Optimization Problem.

$$\min_{w, b}$$

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

Convex - Minimization.

gradient
descent

$$\begin{aligned} \nabla_w L(w, b) &= \sum_{i=1}^M \frac{-\exp(-y_i (w^T n_i + b))}{1 + \exp(-y_i (w^T n_i + b))} y_i n_i \\ \nabla_b L(w, b) &= \sum_{i=1}^M \frac{-\exp(-y_i (w^T n_i + b))}{1 + \exp(-y_i (w^T n_i + b))} y_i \end{aligned}$$

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 - Ascent

$w^0 = \text{Arbitrary}, b^0 = \text{Arbitrary}, \alpha = \text{Learning Rate}$

for $t=0:T$

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

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

$$\text{if } \left| LL(w^{t+1}, b^{t+1}) - LL(w^t, b^t) \right| < \epsilon \text{ then }$$

end.

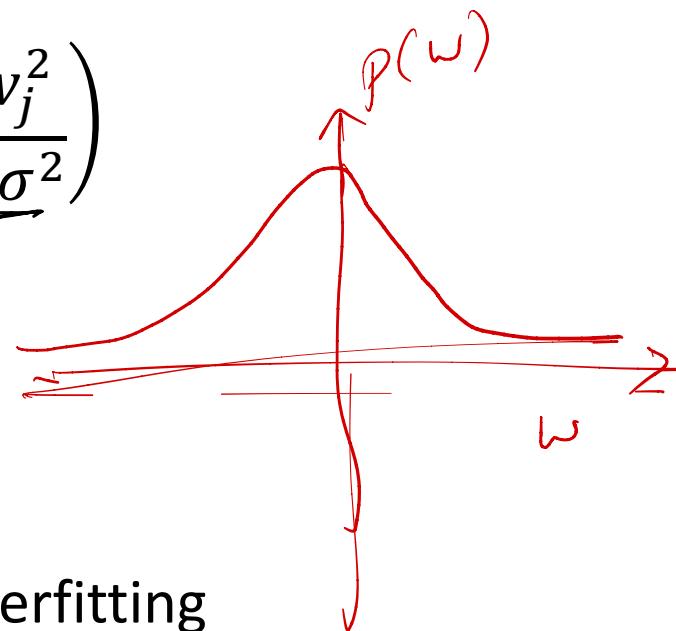
end for.

Priors

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

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

- “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

$$\underset{w, b}{\text{max}} \left[\ln \prod_{i=1}^N p(y^{(i)} | x^{(i)}, w, b) p(w)p(b) \right] = \underbrace{\left[\sum_i^N \ln p(y^{(i)} | x^{(i)}, w, b) \right]}_{\text{Cond. Lik}} - \frac{\lambda}{2} \|w\|_2^2 - \frac{\gamma}{2} b^2$$

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

$$\underset{w, b}{\min} \quad L(w, b) + \underbrace{\frac{\gamma}{2} \|w\|^2}_{\text{Reg.}} + \underbrace{\frac{\gamma}{2} b^2}_{\text{Reg.}}$$

↑
Logistic loss

Priors as Regularization

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

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

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

$\lambda = \frac{1}{2} G^2$ [Gauss Prior]

Sometimes called an ℓ_2 regularizer

Loss-Function Perspective.

Perceptron:

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

SVM :

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

Log Reg :

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

Classification :

$y_i(w^T x_i + b) > 0$ [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: $\max(0, -z) \leftarrow$ high if $z \ll 0$

SVM: $\max(0, 1 - z) \leftarrow$ "

Log Reg: $\log(1 + \exp(-z))$

Regularization

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

L2 - Reg.

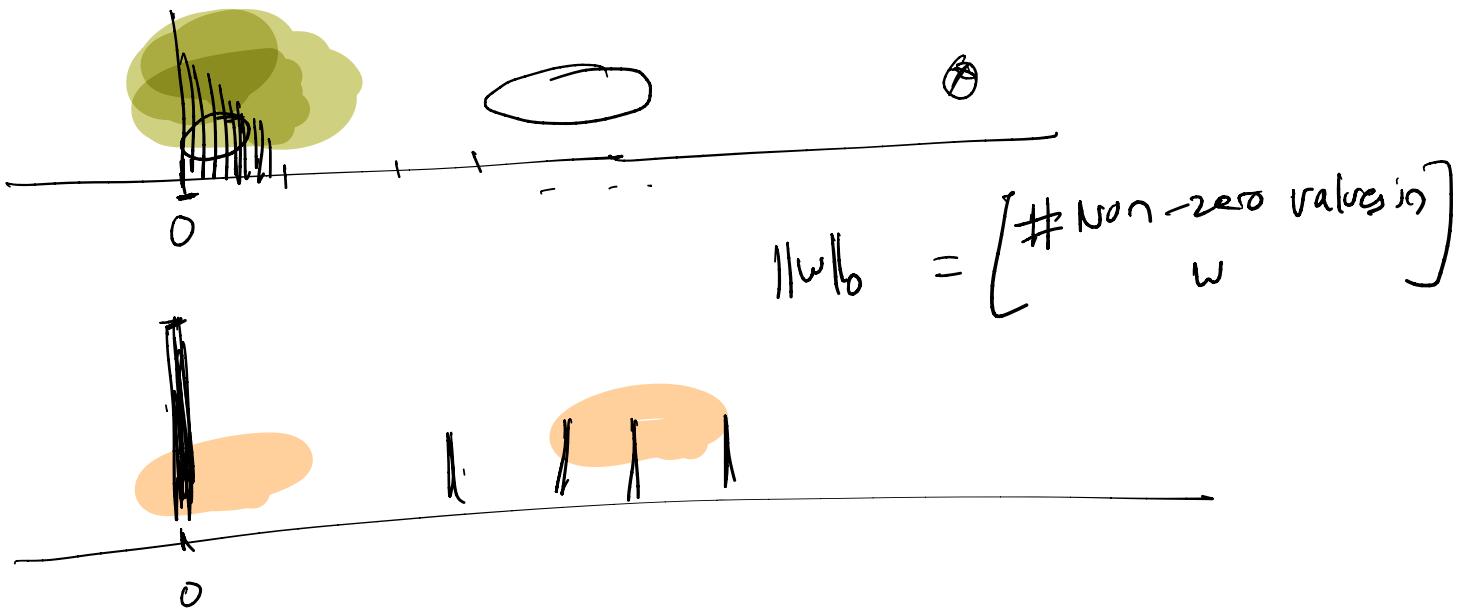
$$\min_{w, b} \sum_{i=1}^n L(x_i, y_i, w, b) + \frac{\lambda}{2} \|w\|^2 + \frac{\gamma}{2} b^2$$

L1 - R.g.

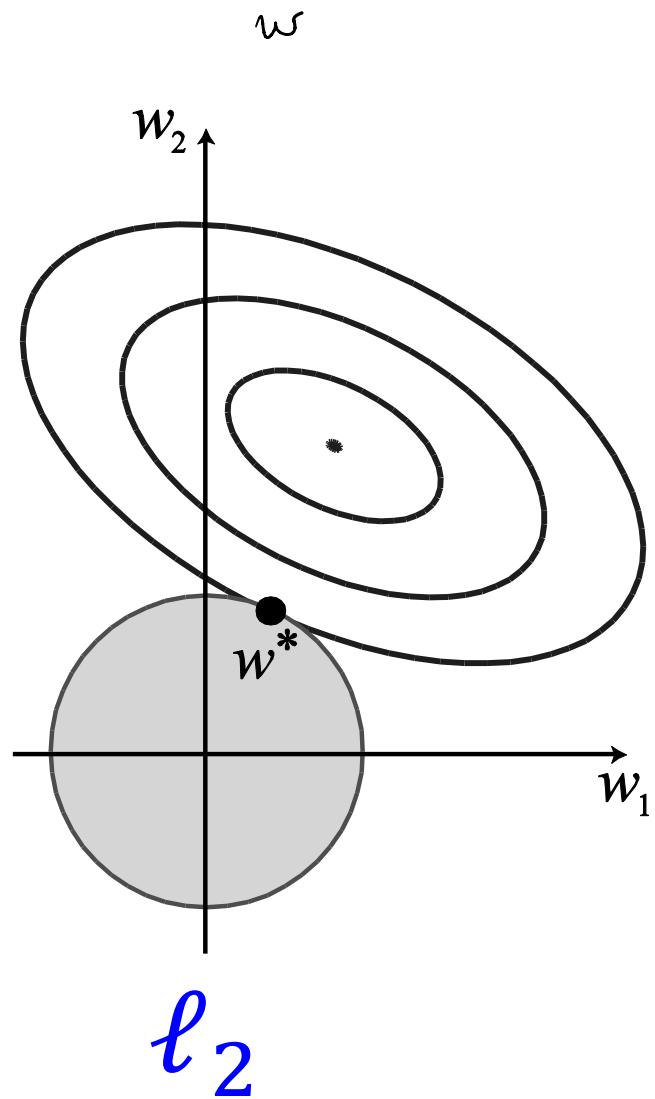
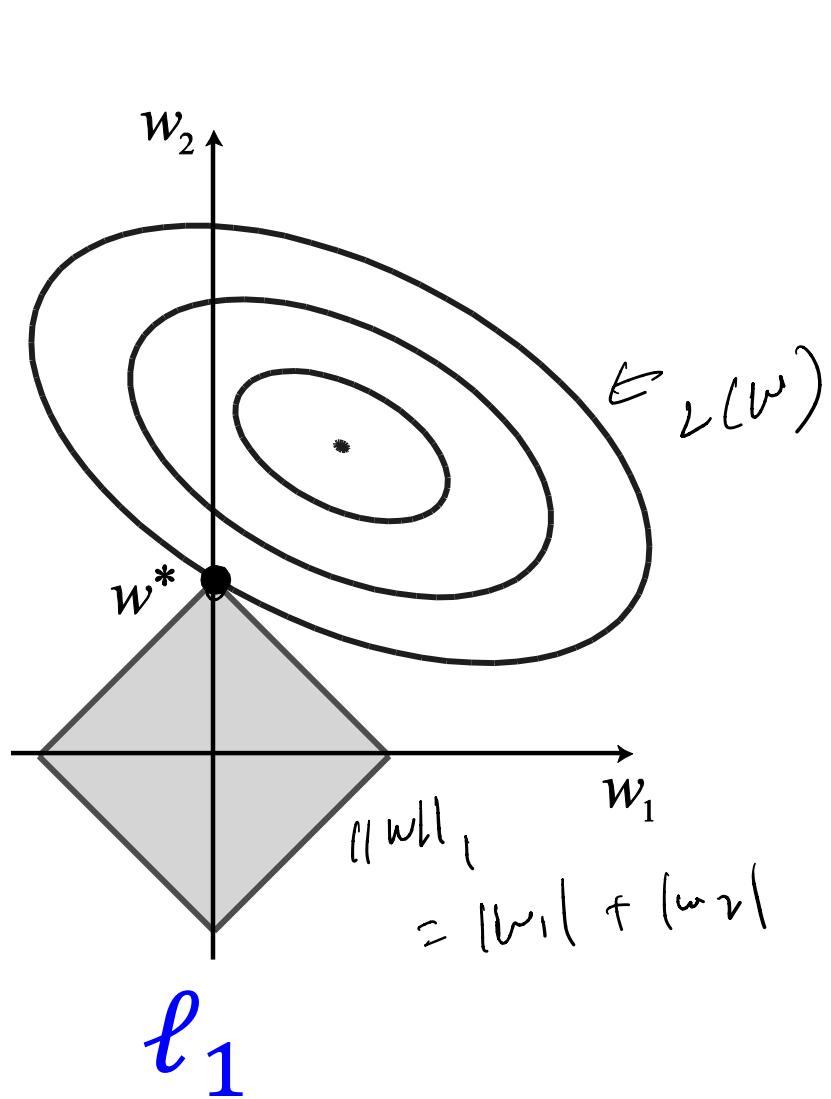
$$\min_{w, b} \sum_{i=1}^n L(x_i, y_i, w, b) + \lambda \|w\|_1 + \gamma |b|$$

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

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



Regularization

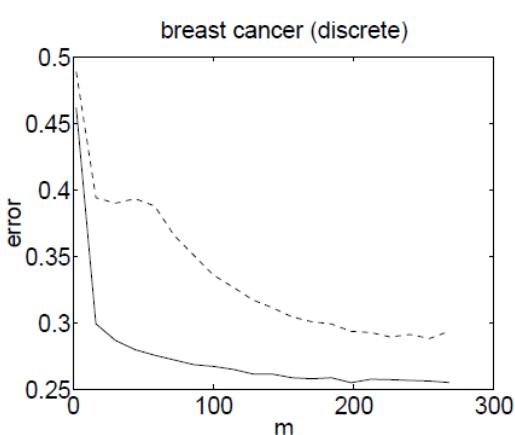
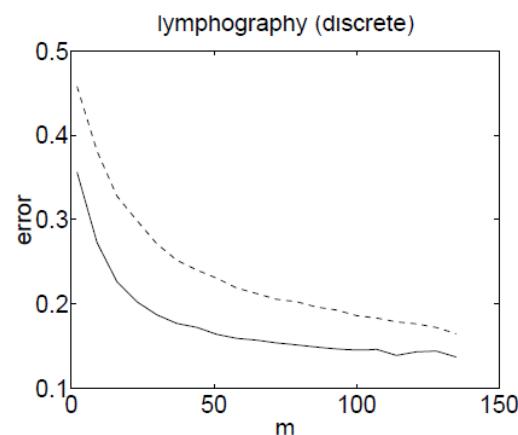
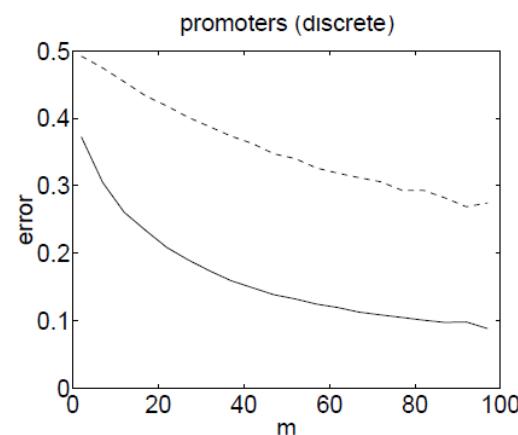
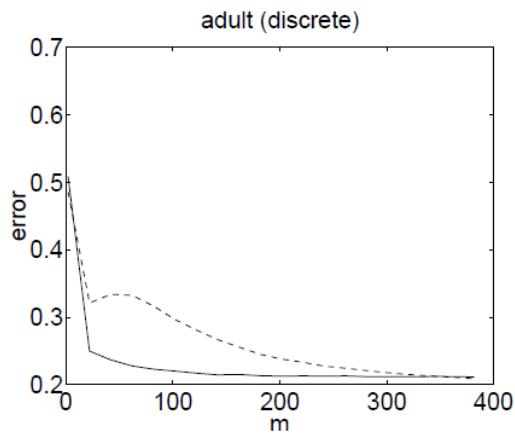
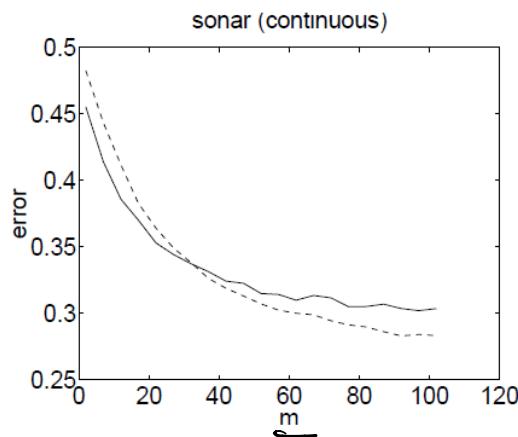
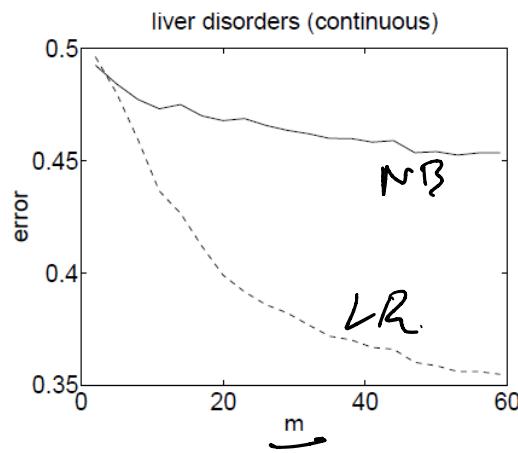


Naïve Bayes vs. Logistic Regression



- 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)

NB vs. LR (on UCI datasets)



— Naïve bayes
..... Logistic Regression

Sample size m

LR in General

- Suppose that $y \in \{1, \dots, R\}$, i.e., that there are R different class labels

$$x^{(i)} \in \mathbb{R}^d$$

- Can define a collection of weights and biases as follows

- Choose a vector of biases and a matrix of weights such that for $y \neq R$

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

$$w_k \in \mathbb{R}^d$$

$$b_k \in \mathbb{R}$$

and

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

$$\arg \max_y$$

Inference : $\arg \max_{y=1:R} p(y|x)$

$$P(Y=k|x) \propto \exp(b_k + w_k^T x)$$

$$P(Y=R|x) \propto \exp(b_R + w_R^T x)$$

$$P(Y=k|x) = \frac{\exp(b_k + w_k^T x)}{\sum_{j=1}^R \exp(b_j + w_j^T x)}$$

$$P'(Y=k|x) = \frac{\exp(b_k - b_R + (w_k - w_R)^T x)}{1 + \sum_{j < R} \exp(b_j - b_R + (w_j - w_R)^T x)}$$

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

$\underbrace{\quad}_{\text{obs prob}}$

webpage

features

webpage

feats

$\underline{w}^T \underline{x} + b$

click. } classifn
0/1 }

CTR. }
0.1 } regression

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