Logistic Regression

Sargur N. Srihari
University at Buffalo, State University of New York
USA

Topics in Linear Classification using Probabilistic Discriminative Models

- Generative vs Discriminative

1. Fixed basis functions
2. Logistic Regression (two-class)
3. Iterative Reweighted Least Squares (IRLS)
4. Multiclass Logistic Regression
5. Probit Regression
6. Canonical Link Functions

Topics in Logistic Regression

- Logistic Sigmoid and Logit Functions
- Parameters in discriminative approach
- Determining logistic regression parameters
  - Error function
  - Gradient of error function
  - Simple sequential algorithm
  - An example
- Generative vs Discriminative Training
  - Naiive Bayes vs Logistic Regression

Logistic Sigmoid and Logit Functions

- In two-class case, *posterior* of class C_1 can be written as as a logistic sigmoid of feature vector $\boldsymbol{\phi}=[\phi_1,..\phi_M]^{\mathrm{T}}$

  $$p(C_1|\boldsymbol{\phi}) = y(\boldsymbol{\phi}) = \sigma\,(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\phi})$$

  with $p(C_2|\boldsymbol{\phi}) = 1 - p(C_1|\boldsymbol{\phi})$

  Here $\sigma\,(.)$ is the logistic sigmoid function

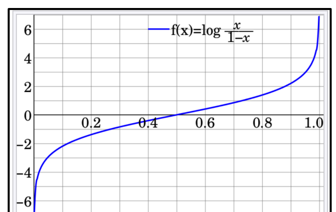
  – Known as logistic regression in statistics

  - Although a model for classification rather than for regression

- Logit function:

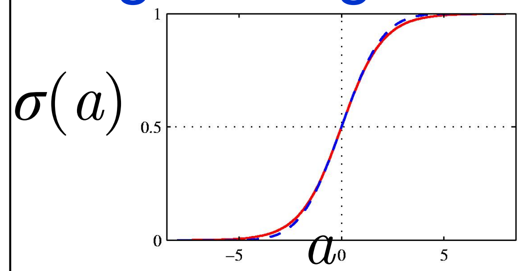
  – It is the log of the odds ratio

  - It links the probability to the predictor variables



Plot of logit(p) in the domain of 0 to 1, where the base of logarithm is e

Logistic Sigmoid



Properties:

<u>A. Symmetry</u>

$$\sigma\,(-a) = 1 - \sigma\,(a)$$

<u>B. Inverse</u>

$$a = \ln(\sigma/1-\sigma)$$

known as *logit.* Also known as *log odds* since it is the ratio

$$\ln[p(C_1|\phi)/p(C_2|\phi)]$$

<u>C. Derivative</u>

$$d\sigma/da = \sigma\,(1-\sigma)$$

Fewer Parameters in Linear Discriminative Model

- Discriminative approach (Logistic Regression)
  - For M -dim feature space ϕ:
  - M adjustable parameters
- Generative based on Gaussians (Bayes/NB)
  - $2M$ parameters for mean
  - $M(M+1)/2$ parameters for shared covariance matrix
  - Two class priors
  - Total of $M(M+5)/2 + 1$ parameters
    - Grows quadratically with M
  - If features assumed independent (naïve Bayes) still needs $M+3$ parameters

Determining Logistic Regression parameters

- Maximum Likelihood Approach for Two classes
- For a data set $(\boldsymbol{\phi}_n, t_n)$ where $t_n \varepsilon \{0,1\}$ and $\boldsymbol{\phi}_n = \boldsymbol{\phi}(\boldsymbol{x}_n)$, $n = 1,..,N$

- Likelihood function can be written as

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

where $\mathrm{t} = (t_1,..,t_N)^{\mathrm{T}}$ and $y_n = p(C_1 | \boldsymbol{\phi}_n)$

y_n is the probability that $t_n = 1$

Error Fn for Logistic Regression

- Likelihood function is

$$\boxed{p(\mathrm{t} \mid \boldsymbol{w}) = \prod_{n=1}^{N} y_n^{t_n} \left\{1 - y_n\right\}^{1-t_n}}$$

- By taking negative logarithm we get the *Cross-entropy Error Function*

$$\boxed{E(\boldsymbol{w}) = -\ln p(t \mid \boldsymbol{w}) = -\sum_{n=1}^{N} \left\{ t_n \ln y_n + (1 - t_n) \ln(1 - y_n) \right\}}$$

where $y_n = \sigma(a_n)$ and $a_n = \boldsymbol{w}^T \boldsymbol{\phi}_n$

- We need to minimize $E(\boldsymbol{w})$

At its minimum, derivative of $E(\boldsymbol{w})$ is zero

So we need to solve for w in the equation

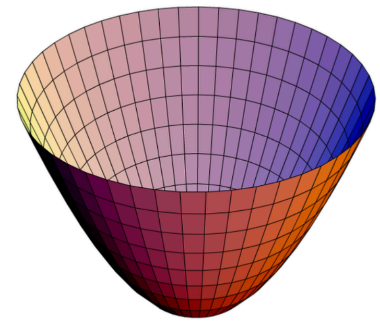
$$\boxed{\nabla E(\boldsymbol{w}) = 0}$$

Gradient of Error Function

Error function

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

where $y_n = \sigma(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\phi}_n)$



Using Derivative of logistic sigmoid $d\sigma/da = \sigma(1-\sigma)$

Gradient of the error function

$$\nabla E(\boldsymbol{w}) = \sum_{n=1}^{N} \underbrace{(y_n - t_n)\phi_n}_{\text{Error x Feature Vector}}$$

Error x Feature Vector

Contribution to gradient by data point n is error between target t_n and prediction $y_n = \sigma(\mathrm{w}^{\mathrm{T}}\phi_n)$ times basis ϕ_n

Proof of gradient expression

$Let \ z = z_1 + z_2$

where $z_1 = t \ln \sigma(w\phi)$ and $z_2 = (1-t)\ln[1-\sigma(w\phi)]$

$$\frac{dz_1}{dw} = \frac{t\sigma(w\phi)[1-\sigma(w\phi)]\phi}{\sigma(w\phi)}$$

$$\frac{d\sigma}{da} = \sigma(1-\sigma)$$

and

Using $\frac{d}{dx}(\ln ax) = \frac{a}{x}$

$$\frac{dz2}{dw} = \frac{(1-t)\sigma(w\phi)[1-\sigma(w\phi)](-\phi)}{[1-\sigma(w\phi)]}$$

Therefore $\frac{dz}{dw} = (\sigma(w\phi) - t)\phi$

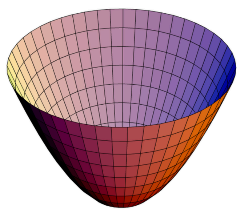
Simple Sequential Algorithm

- Given Gradient of error function

$$\nabla E(\boldsymbol{w}) = \sum_{n=1}^{N} \left(y_n - t_n\right)\phi_n$$

where $y_n = \sigma(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\phi}_n)$

- Solve using an iterative approach



$$\boldsymbol{w}^{\tau+1} = \boldsymbol{w}^{\tau} - \eta \nabla E_n$$

- where

$$\nabla E_n = \underbrace{(y_n - t_n)\phi_n}_{\text{Error x Feature Vector}}$$

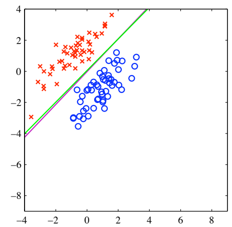
Takes precisely same form as Gradient of Sum-of-squares error for linear regression

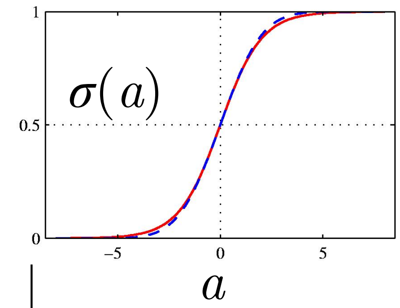
Samples are presented one at a time in which each each of the weight vectors is updated

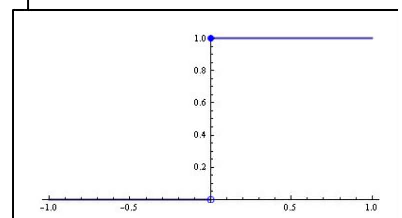
ML solution can over-fit

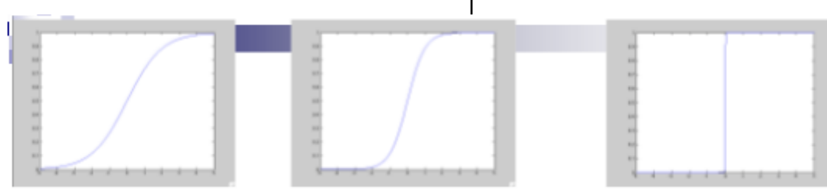
- Severe over-fitting for linearly separable data
  - Because ML solution occurs at $\sigma = 0.5$
    - With $\sigma > 0.5$ and $\sigma < 0.5$ for the two classes
    - Solution equivalent to $a = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{\phi} = 0$
  - Logistic sigmoid becomes infinitely steep
    - A Heavyside step function

      $||\boldsymbol{w}||$ goes to infinity
    - Penalizing wts can avoid this









$$\frac{1}{1+e^{-x}} \qquad \frac{1}{1+e^{-2x}} \qquad \frac{1}{1+e^{-100x}}$$

An Example of 2-class Logistic Regression

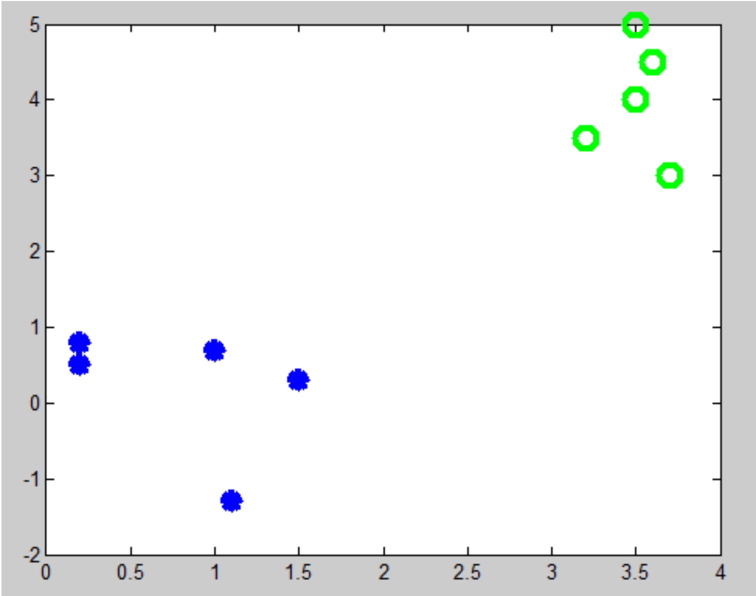
- Input Data

C1 =

3.7000	3.0000
3.2000	3.5000
3.5000	5.0000
3.6000	4.5000
3.5000	4.0000

C2 =

1.1000	-1.3000
0.2000	0.5000
1.5000	0.3000
0.2000	0.8000
1.0000	0.7000



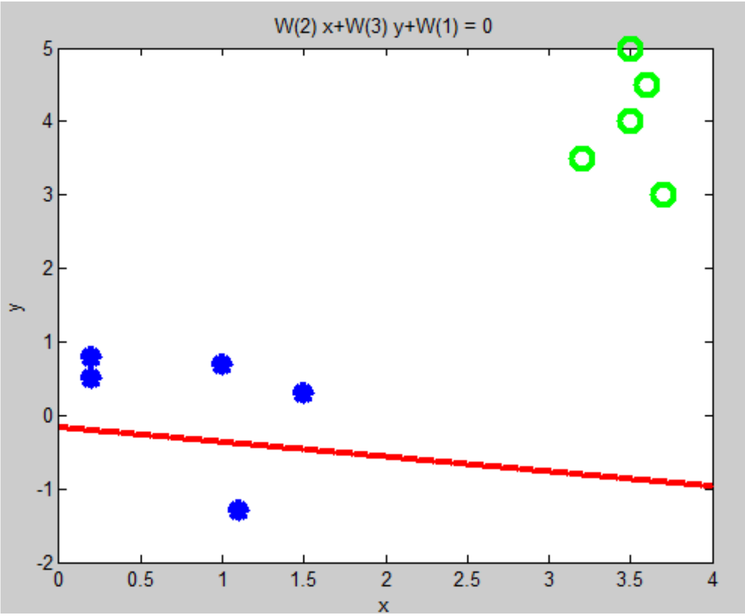
$\phi_0(\boldsymbol{x})=1$, dummy feature

Initial Weight Vector, Gradient and Hessian (2-class)

- Weight vector

```
W =

    0.1117
    0.1363
    0.6787
```



- Gradient

```
Delta_E =

         0
    6.7500
    9.5000
```

- Hessian

```
H =

    3.5000    5.3750    5.2500
    5.3750   17.4825   17.4950
    5.2500   17.4950   22.4150
```

Final Weight Vector, Gradient and Hessian (2-class)

- Weight Vector

```
W =

  704.5915
  -20.9086
 -337.6170
```

- Gradient

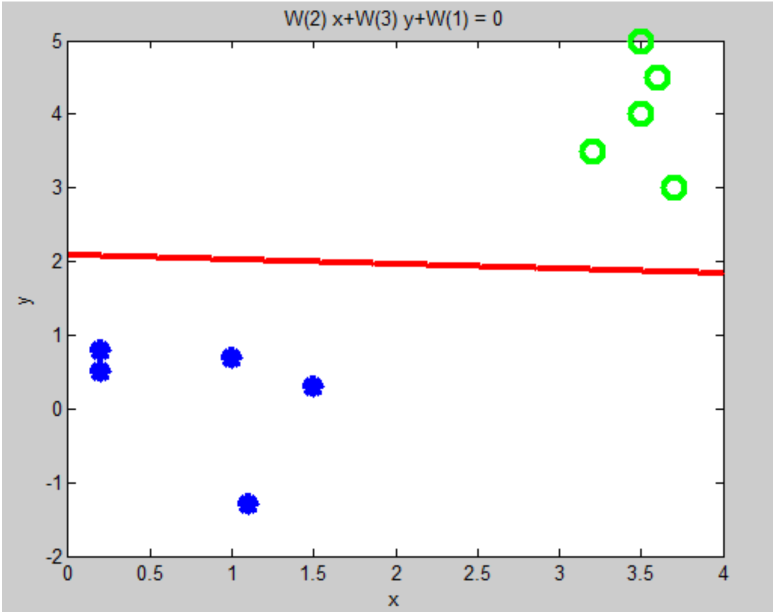
```
Delta_E =

  -12.3917
   -1.6321
    4.9025
```

- Hessian

```
H =

    1.0000    0.0000    0.0000
    0.0000    1.0000    0.0000
    0.0000    0.0000    1.0000
```



Number of iterations : 10

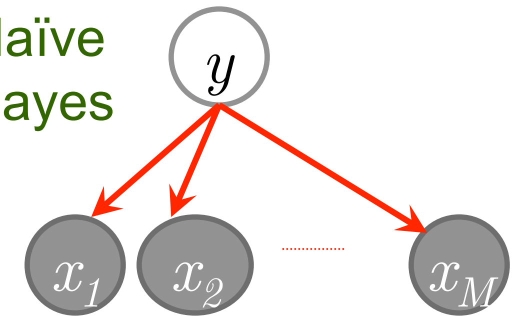
Error (Initial and Final): 15.0642, 1.0000e-009

Generative vs Discriminative Training

Variables $\boldsymbol{x} = \{x_1, \ldots x_M\}$ and classifier target y

1. Generative: estimate parameters of variables independently

Naïve Bayes



For classification:
Determine joint:

$$p(y,\boldsymbol{x}) = p(y)\prod_{i=1}^{M} p(x_i \mid y)$$

From joint get required conditional $p(y|\boldsymbol{x})$

Simple estimation

independently estimate M sets of parameters

But independence is usually false

We can estimate $M(M+1)/2$ covariance matrix

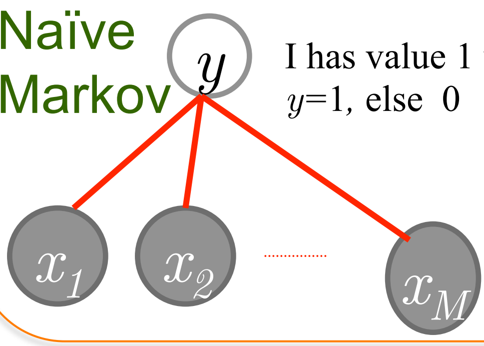
2. Discriminative: estimate joint parameters w_i

Potential Functions (log-linear)

$\phi_i(x_i, y) = \exp\{w_i x_i \, \mathrm{I}\{y=1\}\}$,

$\phi_0(y) = \exp\{w_0 \, \mathrm{I}\{y=1\}\}$

Naïve Markov

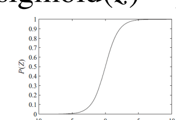


I has value 1 when $y=1$, else 0

For classification:

Unnormalized $\quad \tilde{P}(y=1 \mid \boldsymbol{x}) = \exp\left\{w_0 + \sum_{i=1}^{M} w_i x_i\right\} \qquad \tilde{P}(y=0 \mid \boldsymbol{x}) = \exp\{0\} = 1$

Normalized $\quad P(y=1 \mid \boldsymbol{x}) = \mathrm{sigmoid}\left\{w_0 + \sum_{i=1}^{M} w_i x_i\right\}$ where $\mathrm{sigmoid}(z) = \dfrac{e^z}{1+e^z}$



Logistic Regression

<u>Jointly</u> optimize M parameters

More complex estimation but correlations accounted for

Can use much richer features:

Edges, image patches sharing same pixels

multiclass

$$p(y_i \mid \phi) = y_i(\phi) = \frac{\exp(a_i)}{\sum_j \exp(a_j)}$$

where $a_j = \boldsymbol{w}_j^{\mathrm{T}}\phi$

Logistic Regression is a special architecture of a neural network