Logistic Regression

Machine Learning – 10701/15781

Carlos Guestrin

Carnegie Mellon University

September 24th, 2007

Generative v. Discriminative classifiers – Intuition

generate spam: sample (or set) $P(Y = spam)$

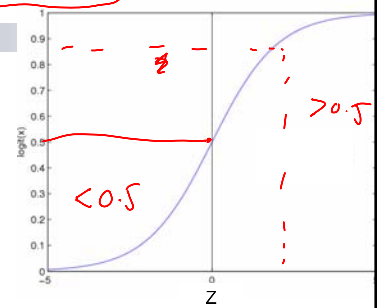
sample words: $P(X_i | Y = spam)$

- **Want to Learn**: h:**X** ↦ Y ∈ {1,2,3,…,k}
  - **X** – features
  - Y – target classes
- exactly
- **Bayes optimal classifier** – P(Y|**X**)
- **Generative classifier**, e.g., Naïve Bayes:
  - e.g., NB: $P(X|Y) = \prod_i P(X_i|Y)$ $P(Y)$
  - Assume some **functional form for P(X|Y), P(Y)**
  - learn $P(Y,X)$
  - Estimate parameters of P(~~X~~|Y), P(Y) directly from training data
  - Use Bayes rule to calculate P(Y|X= x) $= \frac{P(Y, X=x)}{P(X=x)}$
  - This is a **'*generative*' model**
    - **Indirect** computation of P(Y|X) through Bayes rule
    - But, **can 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 the **'*discriminative*' model**
    - Directly learn P(Y|X)
    - But **cannot obtain a sample of the data**, because P(X) is not available

at classification time: input x answer $P(Y|X=x)$

Logistic Regression

Logistic function (or Sigmoid): $\frac{1}{1+exp(-z)}$



- Learn P(Y|**X**) directly!
  - Assume a particular functional form
  - Sigmoid applied to a linear function of the data:

true

$$P(Y=1|X) = \frac{1}{1+\exp(w_0+\sum_{i=1}^{n} w_i X_i)}$$

Features can be discrete or continuous!

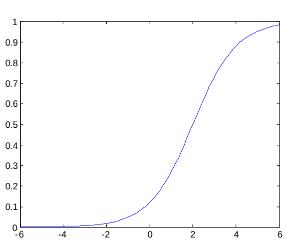
Understanding the sigmoid

$$g(w_0+\sum_i w_i x_i) = \frac{1}{1+e^{w_0+\sum_i w_i x_i}}$$

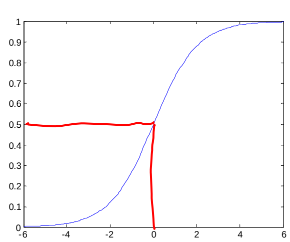
constant

$w_0 + w_1 x_1$

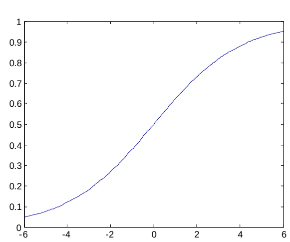
w_0=-2, w_1=-1



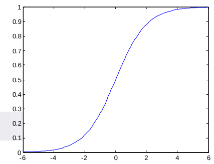
w_0=0, w_1=-1



w_0=0, w_1=-0.5



Logistic Regression – a Linear classifier



$$g(w_0+\sum_{i=1}^{n} w_i x_i) = \frac{1}{1+e^{w_0+\sum_i w_i x_i}}$$

$-\infty$

$w_0 + \sum_i w_i x_i < 0$

x: $g > 0.5$

$+$ true

$w_0 + \sum_i w_i x_i = 0$

$g = 0.5 = \frac{1}{1+1}$

$w_0 + \sum_i w_i x_i > 0$

x: $g < 0.5$

$-$ false

n-dimensional space

5

Very convenient!

$\ln 1 = 0$

$$P(\underline{Y=1}|X=<X_1,...X_n>, w) = \frac{1}{1+exp(w_0+\sum_i w_i X_i)}$$

implies

$= 1 - P(Y=1|X,w)$

$$P(Y=0|X=<X_1,...X_n>, w) = \frac{exp(w_0+\sum_i w_i X_i)}{1+exp(w_0+\sum_i w_i X_i)}$$

implies

output $Y=1$, if $1 >$

$$\frac{P(Y=0|X)}{P(Y=1|X)} = exp(w_0+\sum_i w_i X_i)$$

linear classification rule!

implies

$$\ln\frac{P(Y=0|X)}{P(Y=1|X)} = w_0+\sum_i w_i X_i < 0$$

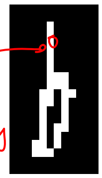
return $Y=1$

6

What if we have continuous X_i ?

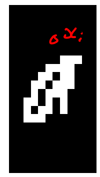
Eg., character recognition: X_i is i^{th} pixel

x_i pixel intensity



if class = b, x_i high

a, x_i low



x_i

Gaussian Naïve Bayes (GNB):

mean pixel i class k

$$P(X_i = x \mid \underbrace{Y = y_k}_{\text{label}}) = \frac{1}{\sigma_{ik}\sqrt{2\pi}} \; e^{\frac{-(x-\mu_{ik})^2}{2\sigma_{ik}^2}}$$

variance pixel i class k

Sometimes assume variance

- is independent of Y (i.e., σ_i),
- or independent of X_i (i.e., σ_k)
- or both (i.e., σ)

Example: GNB for classifying mental states

[Mitchell et al.]

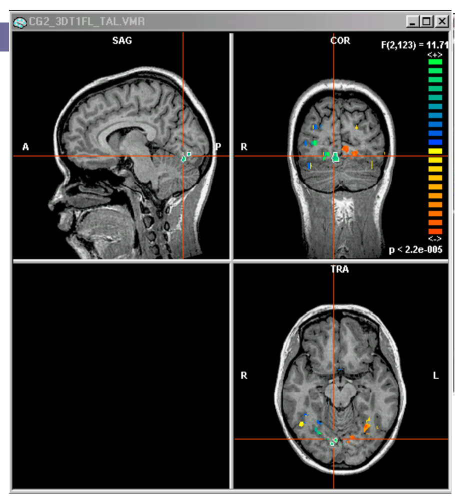
~1 mm resolution

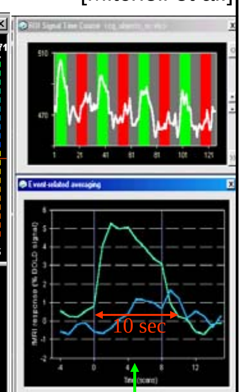
~2 images per sec.

15,000 voxels/image

non-invasive, safe

measures Blood Oxygen Level Dependent (BOLD) response





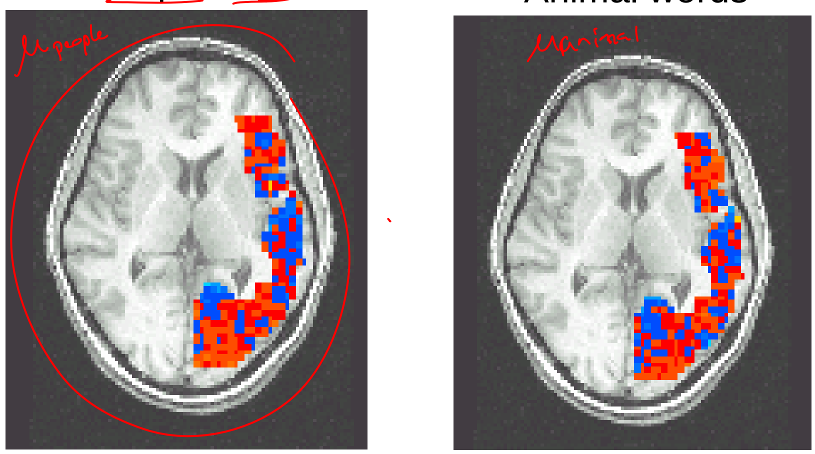
Typical impulse response

Learned Bayes Models – Means for P(BrainActivity | WordCategory)

[Mitchell et al.]

Pairwise classification accuracy: 85%

People words   –5 0 +5   Animal words



Logistic regression v. Naïve Bayes

- Consider learning f: X → Y, where
  - X is a vector of real-valued features, < X_1 … X_n >
  - Y is boolean
- Could use a Gaussian Naïve Bayes classifier
  - assume all X_i are conditionally independent given Y
  - model $P(X_i \mid Y = y_k)$ as Gaussian $N(\mu_{ik},\sigma_i)$
  - model P(Y) as Bernoulli(θ,1-θ)

variance only depends on x_i on pixel i, not on class

- What does that imply about the form of P(Y|X)?

$$P(Y = 1|X =< X_1, ...X_n >) = \frac{1}{1 + exp(w_0 + \sum_i w_i X_i)}$$

Cool!!!!

Derive form for P(Y|X) for continuous X_i

$e^{\ln x} = x$ $\qquad$ $P(y=1|X) = \frac{1}{1+e^{w_0+\sum_i w_i x_i}}$

Bayes rule

$$P(Y=1|X) = \frac{P(Y=1)P(X|Y=1)}{P(Y=1)P(X|Y=1)+P(Y=0)P(X|Y=0)}$$

$$= \frac{1}{1+\frac{P(Y=0)P(X|Y=0)}{P(Y=1)P(X|Y=1)}}$$

$$= \frac{1}{1+\exp(\ln \frac{P(Y=0)P(X|Y=0)}{P(Y=1)P(X|Y=1)})}$$

$$= \frac{1}{1+\exp(\underbrace{(\ln \frac{1-\theta}{\theta})}_{\text{"looks like } w_0\text{"}} + \boxed{\underbrace{\sum_i \ln \frac{P(X_i|Y=0)}{P(X_i|Y=1)}}_{\text{expand}}})}$$

"looks like w_0"

* independent of x_i

©Carlos Guestrin 2005-2007

11

Ratio of class-conditional probabilities

$\ln \frac{1}{e^{-x}} = x$ $\qquad$ $P(y=1|x) = \frac{1}{1+e^{w_0+\sum_i w_i x_i}}$

i indexes over features

$$\ln \frac{P(X_i|Y=0)}{P(X_i|Y=1)} = \qquad \boxed{P(X_i = x_i \mid Y = y_k) = \frac{1}{\sigma_i\sqrt{2\pi}} \; e^{\frac{-(x_i-\mu_{ik})^2}{2\sigma_i^2}}}$$

var doesn't depend on class k

$$\ln \frac{\frac{1}{\cancel{\sigma_i\sqrt{2\pi}}} e^{-\frac{(x_i-\mu_{i0})^2}{2\sigma_i^2}}}{\frac{1}{\cancel{\sigma_i\sqrt{2\pi}}} e^{-\frac{(x_i-\mu_{i1})^2}{2\sigma_i^2}}} =$$

$$-\frac{(x_i-\mu_{i0})^2}{2\sigma_i^2} + \frac{(x_i-\mu_{i1})^2}{2\sigma_i^2} = \overbrace{\frac{(\mu_{i0}-\mu_{i1})}{\sigma_i^2} x_i}^{w_i} + \overbrace{\frac{(\mu_{i1}^2-\mu_{i0}^2)}{2\sigma_i^2}}^{\text{also part of } w_0}$$

$$= \frac{-\cancel{x_i^2} + 2x_i\mu_{i0} - \mu_{i0}^2 + \cancel{x_i^2} - 2\mu_{i1}x_i + \mu_{i1}^2}{2\sigma_i^2}$$

©Carlos Guestrin 2005-2007

12

Derive form for P(Y|X) for continuous X_i

$$P(Y=1|X) = \frac{P(Y=1)P(X|Y=1)}{P(Y=1)P(X|Y=1)+P(Y=0)P(X|Y=0)}$$

$$= \frac{1}{1+\exp\left(\left(\ln\frac{1-\theta}{\theta}\right)+\boxed{\sum_i \ln\frac{P(X_i|Y=0)}{P(X_i|Y=1)}}\right)}$$

$$\boxed{\sum_i \left(\frac{\mu_{i0}-\mu_{i1}}{\sigma_i^2}X_i + \frac{\mu_{i1}^2-\mu_{i0}^2}{2\sigma_i^2}\right)}$$

$$P(Y=1|X) = \frac{1}{1+\exp(w_0+\sum_{i=1}^n w_i X_i)}$$

$$w_0 = \ln\frac{1-\theta}{\theta} + \sum_i \frac{\mu_{i1}^2-\mu_{i0}^2}{2\sigma_i^2}$$

Gaussian Naïve Bayes v. Logistic Regression

transform into parameterization of LR

Set of Gaussian Naïve Bayes parameters (feature variance independent of class label)

Set of Logistic Regression parameters

∃ some w's to NB, but not all w's

transform

- Representation equivalence
  - **But only in a special case!!!** (GNB with class-independent variances)
- But what's the difference???
- **LR makes no assumptions about** P(**X**|Y) **in learning**!!!
- **Loss function!!!**
  - Optimize different functions → Obtain different solutions

does not assume independence

assume form for P(Y|X)

Logistic regression for more than 2 classes

- Logistic regression in more general case, where $Y \in \{Y_1 \ldots Y_R\}$: learn *R-1* sets of weights

4 class: 3 sets of params

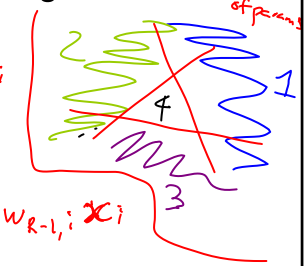
$$P(Y_1=1|x,w_1) \propto e^{w_{10}+\sum_i w_{1i}x_i}$$

$$P(Y=2|x,w_2) \propto e^{w_{20}+\sum_i w_{2i}x_i}$$

$$\vdots$$

$$P(Y=R-1|x,w_{R-1}) \propto e^{w_{R-1,0}+\sum_i w_{R-1,i}x_i}$$

$$P(Y=R|x) = 1-\sum_{j=1}^{R-1} P(Y=j|x) \propto 1-\sum_{j=1}^{R-1} e^{w_{j0}+\sum_i w_{ji}x_i}$$



Logistic regression more generally

- Logistic regression in more general case, where $Y \in \{Y_1 \ldots Y_R\}$: learn *R-1* sets of weights

for *k<R*

$$P(Y=y_k|X) = \frac{\exp(w_{k0}+\sum_{i=1}^{n} w_{ki}X_i)}{1+\sum_{j=1}^{R-1}\exp(w_{j0}+\sum_{i=1}^{n} w_{ji}X_i)}$$

for *k=R* (normalization, so no weights for this class)

$$P(Y=y_R|X) = \frac{1}{1+\sum_{j=1}^{R-1}\exp(w_{j0}+\sum_{i=1}^{n} w_{ji}X_i)}$$

Features can be discrete or continuous!

x_i = letter grade in 10701

$x_i = \begin{cases} A=1 \\ B=2 \\ C=3 \end{cases}$

Loss functions: Likelihood v. Conditional Likelihood

$P(x,y|w) = P(y|x,w) \cdot P(x|w)$

$D = \langle x^j, y^j \rangle_{j:1 \ldots N}$

- Generative (Naïve Bayes) Loss function:
  Data likelihood

$$\ln P(\mathcal{D} \mid \mathbf{w}) = \sum_{j=1}^{N} \ln P(\mathbf{x}^j, y^j \mid \mathbf{w})$$
$$= \sum_{j=1}^{N} \underbrace{\ln P(y^j \mid \mathbf{x}^j, \mathbf{w})}_{\text{classification}} + \sum_{j=1}^{N} \underbrace{\ln P(\mathbf{x}^j \mid \mathbf{w})}_{\text{for generating data, not importants for classification}}$$

- Discriminative models cannot compute $P(\mathbf{x}^j|\mathbf{w})$!
- But, discriminative (logistic regression) loss function:
  Conditional Data Likelihood

$$\ln P(\mathcal{D}_Y \mid \mathcal{D}_{\mathbf{X}}, \mathbf{w}) = \overbrace{\sum_{j=1}^{N} \ln P(y^j \mid \mathbf{x}^j, \mathbf{w})}^{\text{discriminative likelihood}}$$

j = training example
$y^j = 1$ if spam
$= 0$ if ¬spam
x^j = list of words

- Doesn't "waste effort" learning P(X) – focuses on P(Y|**X**) all that matters for classification

Expressing Conditional Log Likelihood

$$\max_w \; l(\mathbf{w}) \equiv \sum_j \ln P(y^j|\mathbf{x}^j, \mathbf{w})$$

j-th component: $P(Y = \text{spam} \mid x^j, w)$ if j-th was spam
$P(Y = \text{not spam} \mid x^j, w)$ if j was not spam

$$P(Y=0|\mathbf{X},\mathbf{w}) = \frac{1}{1+exp(w_0+\sum_i w_i X_i)}$$

$$P(Y=1|\mathbf{X},\mathbf{w}) = \frac{exp(w_0+\sum_i w_i X_i)}{1+exp(w_0+\sum_i w_i X_i)}$$

$$l(\mathbf{w}) = \sum_j \left[y^j \ln P(y=1|\mathbf{x}^j,\mathbf{w}) + (1-y^j) \ln P(y=0|\mathbf{x}^j,\mathbf{w}) \right]$$

if $y^j = 1$: $\ln P(y=1|x^j, w)$ + 0

if $y^j = 0$: 0 + $\ln P(y=0|x^j, w)$

$$l(w) = \sum_j \left[y_j \left[w_0 + \sum_i w_i x_i - \ln\left(1 + e^{w_0 + \sum_i w_i x_i}\right) \right] + (1-y_j)\left[-\ln\left(1+e^{w_0+\sum_i w_i x_i}\right) \right] \right]$$

Maximizing Conditional Log Likelihood

$$P(Y=0|X,W) = \frac{1}{1+exp(w_0+\sum_i w_i X_i)}$$
$$P(Y=1|X,W) = \frac{exp(w_0+\sum_i w_i X_i)}{1+exp(w_0+\sum_i w_i X_i)}$$

$$\max_{w}\quad l(\mathbf{w}) \equiv \ln \prod_j P(y^j|\mathbf{x}^j,\mathbf{w})$$

linear part

$$= \sum_j \left[y^j (w_0 + \sum_i^n w_i x_i^j) - \ln(1+exp(w_0+\sum_i^n w_i x_i^j)) \right]$$

Good news: $l(\mathbf{w})$ is concave function of **w** → no locally optimal solutions

w

Bad news: no closed-form solution to maximize $l(\mathbf{w})$

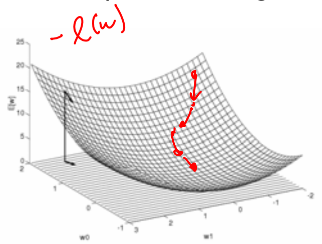
Good news: concave functions easy to optimize

Optimizing concave function – Gradient ascent

(Conjugate G.D.) better.

- Conditional likelihood for Logistic Regression is concave → Find optimum with gradient ascent

- l(w)



Gradient: $\nabla_{\mathbf{w}} l(\mathbf{w}) = [\frac{\partial l(\mathbf{w})}{\partial w_0}, \ldots, \frac{\partial l(\mathbf{w})}{\partial w_n}]'$

step size

Learning rate, η>0

0.01

Update rule: $\Delta \mathbf{w} = \eta \nabla_{\mathbf{w}} l(\mathbf{w})$

$$w_i^{(t+1)} \leftarrow w_i^{(t)} + \eta \frac{\partial l(\mathbf{w})}{\partial w_i}$$

- Gradient ascent is simplest of optimization approaches
  - e.g., Conjugate gradient ascent much better (see reading)

Maximize Conditional Log Likelihood: Gradient ascent

$\frac{\partial \ln f(w)}{\partial w} = \frac{\frac{\partial f(w)}{\partial w}}{f(w)}$

$$P(Y=0|X,W) = \frac{1}{1+exp(w_0+\sum_i w_i X_i)}$$

$$P(Y=1|X,W) = \frac{exp(w_0+\sum_i w_i X_i)}{1+exp(w_0+\sum_i w_i X_i)}$$

$\frac{\partial}{\partial w_1} = x_1^j$ $\frac{\partial e^{f(w)}}{\partial w} = \frac{\partial f(w)}{\partial w} e^{f(w)}$

$$l(\mathbf{w}) = \sum_j y^j (w_0 + \sum_i^n w_i x_i^j) - \ln(1 + exp(w_0 + \sum_i^n w_i x_i^j))$$

$$\frac{\partial l(w)}{\partial w_1} = \sum_j \left[y^j x_1^j - \frac{\partial}{\partial w_1} \ln(1 + e^{w_0 + \sum w_i x_i^j}) \right]$$

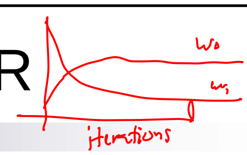
$$= \sum_j \left[y^j x_1^j - \underbrace{\frac{x_1^j e^{w_0 + \sum w_i x_i^j}}{1 + e^{w_0 + \sum w_i x_i^j}}}_{P(Y=1|X,W)} \right]$$

$$= \sum_j x_1^j [y^j - P(Y=1|X,W)]$$

if jth example is positive; if x_1^j is positive want to make w_1 large

if jth " is negative: if x_1^j is positive want to make w_1 small

Gradient Descent for LR



Gradient ascent algorithm: iterate until change < ε

$w(t)$: w at t'th iteration

$$\underbrace{w_0^{(t+1)} \leftarrow w_0^{(t)} + \eta \sum_j [y^j - \hat{P}(Y^j = 1 \mid \mathbf{x}^j, \mathbf{w}^{(t)})]}_{\text{no } x_0^j,\ x_0^j = 1}$$

For $i = 1 \ldots n$,

$$w_i^{(t+1)} \leftarrow w_i^{(t)} + \eta \sum_j x_i^j [y^j - \hat{P}(Y^j = 1 \mid \mathbf{x}^j, \mathbf{w}^{(t)})]$$

$$\frac{e^{w_0 + \sum_i w_i x_i^j}}{1 + e^{w_0 + \sum_i w_i x_i^j}}$$

repeat