Probabilistic classification

rather than a hard rule, e.g. sign$(f(x,w))$
what if we model $p(y_n = C_k \mid x_n)$

<u>Approach 1</u>: <u>Discriminative</u> modeling. We need to choose some <u>model</u> for $p(y_n = C_k \mid x_n)$

Binary case: sigmoid commonly used.

$$\sigma(z) = \frac{1}{1+\exp\{-z\}} =$$

(logit function)

put in a linear function $z = w^T x + w_0$
logistic regression (model for <u>continuous</u> probs)

turn the crank:

$$\hat{p}(y=1 \mid x) = \frac{1}{1+\exp\{-w^T x - w_0\}}$$

$$\hat{p}(y=0 \mid x) = 1 - \hat{p}(y=1 \mid x) = \frac{\cancel{1+}\exp\{-w^T x - w_0\} \cancel{-1}}{1+\exp\{-w^T x - w_0\}}$$

$$= \frac{1}{1+\exp\{w^T x + w_0\}}$$

$\hat{p}(1)^{y_n} \hat{p}(0)^{1-y_n}$

$$\text{llik}\ (w) = \sum_{n=1}^{N} \underbrace{y_n \log \hat{p}(y=1 \mid x_n)} + \underbrace{(1-y_n) \log \hat{p}(y=0 \mid x_n)}$$ goal: maximize

$$\nabla ll(w) = \sum_{n=1}^{N} -y_n x_n \hat{p}(y_n = 0 \mid x_n) + (1-y_n) x_n \hat{p}(y_n = 1 \mid x_n)$$

apply gradient descent/ascent

Note: very similar to the perceptron alg:

$$\nabla_p = \begin{cases} y_n = 1 & x_n \\ y_n = 0 & -x_n \end{cases} \quad \text{vs.} \quad \underbrace{\begin{matrix} x_n \hat{p}(y_n=0) \\ -x_n \hat{p}(y_n=1) \end{matrix}}_{\text{recipe for SGD w/ logit loss}}$$

if wrong

recipe for SGD w/ hinge loss

LR: $x \to y$

$w^T x + w_0$ coming into our sigmoid

defined prob. of y.

⇒ if I wanted to create a toy data set to <u>test</u> logistic regression.

⇒ create a bunch of x's
⇒ choose w, w_0
⇒ compute $\hat{p}(y \mid x)$
⇒ sample $y \sim \hat{p}(y \mid x)$

Approach 2: Generative Model (model joint $p(x,y)$ rather than)

Approach 2: generati... (... just $p(y|x)$ ...)

prior on y $\rightarrow$ (y) $\rightarrow$ (x)

$\Rightarrow$ $y \sim p(y)$
$\Rightarrow$ choose some parameters for $p(x|y)$
$\Rightarrow$ sample $x \sim p(x|y)$

if y is binary $p(y)$ can be a Bernoulli (easy)

if we want to infer $y|x$, we need Bayes rule:

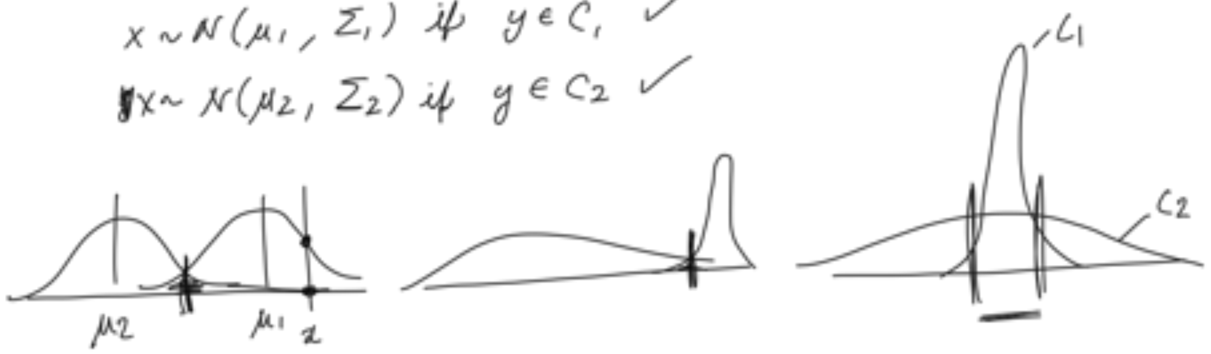
$$p(y=c_k|x) \propto \underbrace{\boxed{p(x|y=c_k)}}_{\text{form will depend on type of var. } x} \underbrace{\boxed{p(y=c_k)}}_{\text{prior (e.g. Bernoulli if binary)}} = p(x,y)$$

before: regressed on this

Suppose that x is continuous: can assume

$x \sim \mathcal{N}(\mu_1, \Sigma_1)$ if $y \in C_1$ ✓

$x \sim \mathcal{N}(\mu_2, \Sigma_2)$ if $y \in C_2$ ✓



(use $\log \frac{p(c_1|x)}{p(c_2|x)}$ as the boundary

$$\frac{1}{\sqrt{2\pi}|\Sigma|^{\frac{1}{2}}} \exp\left\{-\frac{1}{2}(x-\mu)\Sigma^{-1}(x-\mu)^T\right\}$$

$$\log\left(\frac{p(c_1)(2\pi)^{D/2}|\Sigma_2|^{1/2}}{p(c_2)(2\pi)^{D/2}|\Sigma_1|^{1/2}} \exp\left\{-\frac{1}{2}\left[(x-\mu_1)\Sigma_1^{-1}(x-\mu_1)^T - (x-\mu_2)\Sigma_2^{-1}(x-\mu_2)\right]\right\}\right)$$

$$= \text{a lot of terms that don't depend on } x - \frac{1}{2}\left[(x)(\Sigma_1^{-1} - \Sigma_2^{-1})(x)^T - 2x(\Sigma_1^{-1}\mu_1^T + \Sigma_2^{-1}\mu_2^T) + \text{more terms without } x\right.$$

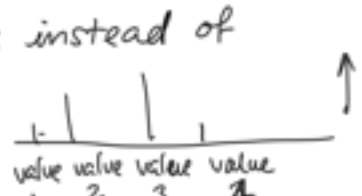
$$= \text{a lot of terms w/o } x + \underbrace{x(\Sigma_1^{-1} - \Sigma_2^{-1})x^T}_{} + x(\text{something})$$

if $\Sigma_1 = \Sigma_2$
then: we have
a linear boundary!

That was the x is continuous case...

$p(y \in C_1|x)$ $\quad$ $p(y \in C_2|x)$

discrete case: now we can multinomials instead of Gaussians to define $\underset{\text{class}}{p(x|y)}$. Multinomial:



class label

↳ explode w/ number of dims (R^D)

make an assumption! $p(x|c) = \prod_{d=1}^{D} p(x_d|c)$

Naive Bayes assumption

We can write down likelihood: $\pi_{dj}^{k} = Pr(x_d \text{ takes value } j \text{ in } y = C_k)$

$p(x|\{\pi\},y)$: $\prod_{n=1}^{N} \prod_{d=1}^{D} \pi_{dj}^{k\,(y_n=k)}$, $x_{nd}=j$

$$\log p(x|\{\pi\},y) = \sum_n \sum_d \log \prod_k \pi_{dj}^{k\,\boxed{\mathbb{I}(y_n=k)}\,\mathbb{I}(x_{nd}=j)} = \sum_n \sum_d \sum_k \mathbb{I}(y_n=k)\,\mathbb{I}(x_{nd}=j) \log \pi_{dj}^{k}$$

tip: group by class.

ASIDE: / NOTES:

1) we can do full Bayes (e.g. priors over params) if we wish
2) we can do multi-class easily for generative case
   (a little harder in the discrim case: separate w_k for each class...)

3) generative model: easy to deal w/ missing data

$p(x|y)\ p(y)$

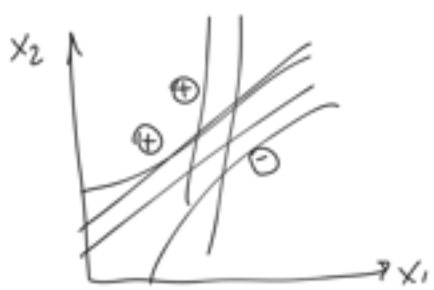
$$\underbrace{\left[\prod_{d=\text{we observe}} p(x_d|y)\right] \cdot p(y)}_{\text{Bayes Rule}}$$

$\sigma(w^T x + w_0)$

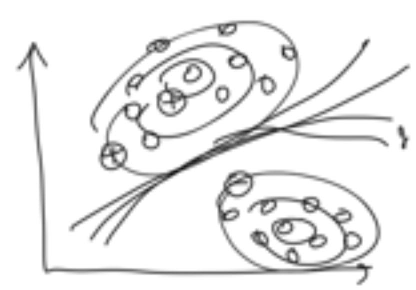
$p(y|x)$

Concept Check : Semi-Supervised Learning

How can unlabeled data help?



what boundaries might you fit?



does unlabeled help?

are they both as easy to use

generative vs. discriminative models: are they [illegible]
in this semi-supervised setting? why or why not?

assume: density of $\underline{\underline{x}}$ reflects label $\underline{\underline{y}}$

$$\begin{array}{c} y \\ \downarrow \\ x \end{array}$$

$\underline{p(y \mid x)}$