Outline

Naive Bayes
- Laplace Smoothing
- Event Models

Kernel Methods

Recap:

$$X = \begin{array}{l} \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 1 \\ \\ \end{bmatrix} \end{array} \begin{array}{l} \text{a} \\ \text{aardvark} \\ \vdots \\ \text{buy} \\ \\ \end{array} \quad \updownarrow d \qquad n \text{ examples}$$

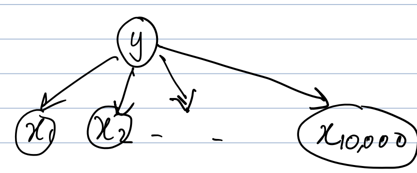
$$x_j = \mathbb{1}\{\text{word } j \text{ appears in email}\}$$

Generative Model

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

$$y = \begin{cases} 0 & \text{not spam} \\ 1 & \text{spam} \end{cases}$$

$$p(x|y) = \prod_{j=1}^{d} p(x_j|y)$$



Parameters

$$\phi_{j|y=1} = P(x_j = 1 \mid y = 1)$$

$$\phi_{j|y=0} = P(x_j = 1 \mid y = 0)$$

$$\phi_y = P(y = 1)$$

Joint Likelihood

$$\mathcal{L}(\phi_y, \phi_{j|y}) = \prod_{i=1}^{n} P(x^{(i)}, y^{(i)}; \phi_y, \phi_{j|y})$$

MLE

$$\phi_y = \frac{\sum_{i=1}^{n} \mathbb{1}\{y^{(i)} = 1\}}{n}$$

$$\phi_{j|y=1} = \frac{\sum_{i=1}^{n} \mathbb{1}\{x_j^{(i)} = 1, y^{(i)} = 1\}}{\sum_{i=1}^{n} \mathbb{1}\{y^{(i)} = 1\}}$$

Prediction :

$$P(y=1|x) = \frac{\overbrace{P(x|y=1)}^{\phi_{j|y=1}} \cdot P(y=1)}{P(x|y=1)P(y=1) + \underbrace{P(x|y=0)}_{\phi_{j|y=0}} \cdot P(y=0)}$$

COVID $\quad j = 1273$

$$P(x_{1273}=1|y=1) = \frac{0}{\#\{y=1\}} = \phi_{1273|y=1}$$

$$P(x_{1273}=1|y=0) = \frac{0}{\#\{y=0\}} = \phi_{1273|y=0}$$

$$P(x|y=1) = \prod_{j=1}^{10,000} P(x_j|y=1)$$

$$P(y=1|x) = \frac{\overbrace{P(x|y=1)}^{=0} \cdot P(y=1)}{\underbrace{P(x|y=1)}_{=0} \cdot P(y=1) + \underbrace{P(x|y=0)}_{=0} \cdot P(y=0)}$$

$\phi_{1273|y=1}$

$\phi_{1273|y=0}$

	Won?
Wake forest	0
OSU	0
Arizona	0
Caltech	0
Oklahoma	??

$$P(x=1) = \frac{\#\text{"1"s}}{\#\text{"1"s} + \#\text{"0"s}} \quad \frac{+1}{+2}$$

$$= \frac{0}{0+4} \quad \frac{+1}{+2} = \frac{1}{6}$$

$$= 0$$

Laplace Smoothing

$$\frac{\#\text{"1"s} + 1}{\#\text{"0"s} + 1}$$

$x_i \in \{1 \ldots |V|\}$

size	< 400 feet^2	400–800	800–1200	>1200
x	1	2	3	4

$$P(x \mid y) = \prod_{i=1}^{d} \underbrace{P(x_j \mid y)}_{}$$

multinomial (vs. bernoulli)

$$X = \begin{array}{c} \\ \\ \begin{bmatrix} \\ 1 \\ 1 \\ 1 \end{bmatrix} \end{array} \quad \begin{array}{l} \text{a} \\ \text{aardvark} \\ \text{account} \\ \vdots \\ \text{bank} \\ \text{beneficiary} \end{array} \quad \begin{array}{c} 1 \\ 2 \\ 800 \\ \\ 1600 \\ \vdots \end{array}$$

$x_i \in \{0, 1\}$

" bank -- account -- bank "

$$X \in \begin{bmatrix} 1600 \\ \vdots \\ 800 \\ \vdots \\ 1600 \\ 6200 \end{bmatrix} \in \mathbb{R}^{d_i}$$

$x_j \in \{1 \ldots |V|\}$ $\qquad |V| = 10{,}000$

d_i : length of email i

Multivariate Bernoulli event model

Multinomial event model

$$P(x, y) = P(x \mid y) \cdot P(y)$$

assume : $P(x \mid y) = \prod_{j=1}^{d} P(x_j \mid y)$

$x_j \in \{1, \ldots, |V|\}$

Parameters

$\phi_y = P(y = 1)$

$\phi_{k \mid y=0} = P(x_j = k \mid y = 0)$

Chance that word j is k^{th} word in dictionary if $y = 0$

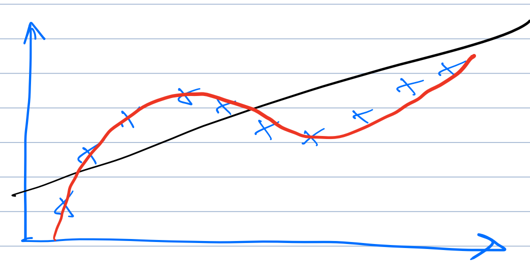
MLE $\phi_{k|y=0} = \dfrac{\sum_{i=1}^{n} \left(\mathbb{1}\{y^{(i)}=0\} \sum_{j=1}^{d_i} \mathbb{1}\{x_j^{(i)}=k\} \right)}{\sum_{i=1}^{n} \mathbb{1}\{y^{(i)}=0\} \cdot d_i}$

Laplace Smoothing: $+1$ to numerator
$+|V|$ to denominator
10,000

Map rare words to "UNK"

- mortgage
  mϕrtg/\ge
  UNK
- spoofed headers
- fetching URL

Kernel Methods



$x \in \mathbb{R}$

Linear models : $\theta^T x$

$h_\theta(x) = \theta_3 x^3 + \theta_2 x^2 + \theta_1 x + \theta_0$

$\phi(x) = \begin{bmatrix} 1 \\ x \\ x^2 \\ x^3 \end{bmatrix}$ $\phi : \mathbb{R} \to \mathbb{R}^4$

$h_\theta(x) = [\theta_0, \theta_1, \theta_2, \theta_3] \begin{bmatrix} 1 \\ x \\ x^2 \\ x^3 \end{bmatrix} = \theta^T \phi(x)$

$$\theta = \begin{bmatrix} \theta_0 \\ \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix}$$

$h_\theta(x)$ = linear in θ, $\phi(x)$

$$\{(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}) \ldots (x^{(n)}, y^{(n)})\}$$

$$\Downarrow$$

$$\{(\phi(x^{(1)}), y^{(1)}), (\phi(x^{(2)}), y^{(2)}) \ldots (\phi(x^{(n)}), y^{(n)})\}$$

cubic polynomial for old dataset

$\iff$ linear on new dataset

LMS on new dataset

$$\min_\theta \frac{1}{2} \sum_{i=1}^{n} (y^{(i)} - \theta^T \phi(x^{(i)}))^2$$

Gradient Descent :

Loop $\{$ $\theta := \theta + \alpha \sum_{i=1}^{n} (y^{(i)} - \theta^T \phi(x^{(i)})) \phi(x^{(i)})$

$\}$

$\in \mathbb{R}^p$ $\quad$ $\in \mathbb{R}^p$ $\quad$ $O(np)$

Terminology

$\phi : \mathbb{R}^d \longrightarrow \mathbb{R}^p$ feature map

attributes $\quad$ features

x : attributes

$\phi(x)$: "features"

What to do if p is very large?

$d > 1$ fit cubic polynomial

$$\phi(x) = \begin{bmatrix} 1 \\ x_1 \\ \vdots \\ x_d \\ x_1^2 \\ \vdots \\ x_i x_j \\ \vdots \\ x_d^2 \\ x_1^3 \\ \vdots \\ x_i x_j x_k \\ \vdots \\ x_d^3 \end{bmatrix} \begin{matrix} \left.\vphantom{\begin{matrix} x_1 \\ \vdots \\ x_d \end{matrix}}\right\} d \\ \left.\vphantom{\begin{matrix} x_1^2 \\ \vdots \\ x_i x_j \\ \vdots \\ x_d^2 \end{matrix}}\right\} d^2 \\ \left.\vphantom{\begin{matrix} x_1^3 \\ \vdots \\ x_i x_j x_k \\ \vdots \\ x_d^3 \end{matrix}}\right\} d^3 \end{matrix}$$

$$\theta^T \phi(x)$$

$$= \_ \cdot 1 + \_ x_1 + \_ x_2$$

$$+ \_ x_i x_j$$

$$+ \_ x_i x_j x_k$$

Problem: $\phi(x)$ is high dimensional!

$$p = 1 + d + d^2 + d^3 \qquad O(d^3)$$

$$d = 10^3 \qquad p \sim 10^9$$

$$\theta := \theta + \alpha \sum_{i=1}^{n} \left(y^{(i)} - \theta^T \phi(x^{(i)})\right) \phi(x^{(i)})$$

Runtime for 1 iteration of GD is $O(np)$

Key observation

If θ initialized at 0,

then at any time, θ can be written as

$$\underset{\in \mathbb{R}^p}{\theta} = \sum_{i=1}^{n} \beta_i \, \phi(x^{(i)}) \quad \text{for some } \underset{\in \mathbb{R}^n}{\beta_1 \ldots \beta_n \in \mathbb{R}}$$

Proof of observation:

By induction on # iterations

Base case: iteration 0

$$\theta = 0 = \sum_{i=1}^{n} \underbrace{0}_{\beta_i} \cdot \phi(x^{(i)})$$

Assume at iteration t, $\theta = \sum_{i=1}^{n} \beta_i \phi(x^{(i)})$

Next iteration:

$$\theta := \theta + \alpha \sum_{i=1}^{n} (y^{(i)} - \theta^T \phi(x^{(i)})) \phi(x^{(i)})$$

$$= \sum_{i=1}^{n} \underbrace{\left(\beta_i + \alpha (\underbrace{y^{(i)}}_{\text{scalar}} - \underbrace{\theta^T \phi(x^{(i)})}_{\text{scalar}}) \right)}_{\text{new } \beta_i} \phi(x^{(i)})$$

New Algo: represent $\theta \in \mathbb{R}^p$ by $\beta \in \mathbb{R}^n$

p param $\longrightarrow$ n param

$$\beta_i := \beta_i + \alpha (y^{(i)} - \theta^T \phi(x^{(i)}))$$

$$= \beta_i + \alpha \left(y^{(i)} - \left(\sum_{j=1}^{n} \beta_j \phi(x^{(j)}) \right)^T \phi(x^{(i)}) \right)$$

$$= \beta_i + \alpha \left(y^{(i)} - \underbrace{\sum_{j=1}^{n}}_{n} \beta_j \underbrace{\langle \phi(x^{(j)}), \phi(x^{(i)}) \rangle}_{p} \right)$$

① $\langle \phi(x^{(i)}), \phi(x^{(j)}) \rangle$ can be precomputed

② $\langle \phi(x^{(i)}), \phi(x^{(j)}) \rangle$ can often be computed much faster without explicitly computing $\phi(\,)$