# Lecture 9: 2/15/17

## Logistic Regression

**Model:**

$$p(y=1 \mid \vec{x}, \vec{\theta}) \triangleq \text{sigmoid}(\theta^T x)$$
$$= \text{logistic}(\theta^T x)$$
$$= \sigma(\theta^T x) \qquad \text{where } \sigma(u) = \frac{1}{1+\exp(-u)}$$
$$= \frac{1}{1+\exp(-\vec{\theta}^T \vec{x})}$$

$$p(y=0 \mid \vec{x}, \vec{\theta}) = 1 - p(y=1 \mid \vec{x}, \vec{\theta})$$
$$= \frac{\exp(-\theta^T x)}{1+\exp(-\theta^T x)}$$

**Decision Boundary:**

The set of points $\vec{x}$ where $p(y=1 \mid \vec{x}, \vec{\theta}) = 0.5$

$$\hat{y} = h(\vec{x}) = \underset{y \in \{0,1\}}{\text{argmax}}\; p(y \mid x, \theta)$$

**Solve:** $\frac{1}{1+\exp(\theta^T x)} = 0.5$

$$\Rightarrow 1 = 0.5(1+\exp(-\theta^T x))$$
$$\Rightarrow 2 = 1+\exp(-\theta^T x)$$
$$\Rightarrow 1 = \exp(-\theta^T x)$$
$$\Rightarrow 0 = -\theta^T x$$
$$\Rightarrow 0 = \vec{\theta}^T \vec{x}$$

with $x_0 = 1$, this is the defn of a hyperplane

☆ The decision boundary is linear.

**Bernoulli Interp.**

**Story:** (generate $y$)

$\phi = ?$

$y \sim \text{Bernoulli}(\phi)$

$$\Rightarrow P(Y=y) = \begin{cases} \phi & \text{if } y=1 \\ 1-\phi & \text{if } y=0 \end{cases}$$

**Story:** (generate $y$ given $\vec{x}$)

$\phi = \sigma(\vec{\theta}^T \vec{x})$

$y \sim \text{Bernoulli}(\phi)$

$$P(Y=y \mid x, \theta) = \begin{cases} \sigma(\theta^T x) & \text{if } y=1 \\ 1-\sigma(\theta^T x) & \text{if } y=0 \end{cases}$$

# Gradient for Logistic Regression

## Conditional Log Likelihood

Recall: If generative:

$$\ell(\theta) \triangleq \log p(\mathcal{D} \mid \vec{\theta})$$
$$= \log p(x^{(1)}, \dots, x^{(N)}, y^{(1)}, \dots, y^{(N)} \mid \vec{\theta})$$
$$= \sum_{i=1}^{N} \log p(x^{(i)}, y^{(i)} \mid \vec{\theta})$$

If discriminative:

$$\ell(\theta) \triangleq \log p(y^{(1)}, \dots, y^{(N)} \mid \vec{x}^{(1)}, \dots, \vec{x}^{(N)}, \vec{\theta})$$
$$= \sum_{i=1}^{N} \log p(y^{(i)} \mid x^{(i)}, \vec{\theta})$$ b/c iid

$$\Rightarrow J(\vec{\theta}) \triangleq -\frac{1}{N}\ell(\vec{\theta}) = \frac{1}{N}\sum_{i=1}^{N} J_i(\vec{\theta})$$
$$\Rightarrow J_i(\vec{\theta}) \triangleq -\log p(y^{(i)} \mid \vec{x}^{(i)}, \vec{\theta})$$
$$= \begin{cases} -\log \mu_i(\theta) & \text{if } y^{(i)} = 1 \\ -\log(1-\mu_i(\theta)) & \text{if } y^{(i)} = 0 \end{cases}$$
where $\mu_i(\theta) = \frac{1}{1+\exp(-\vec{\theta}^T \vec{x}^{(i)})}$

Caution:
$$\hat{\theta} = \operatorname{argmin}_{\theta} J(\vec{\theta})$$
$$= \operatorname{argmax}_{\theta} -J(\vec{\theta})$$
$$= \operatorname{argmax}_{\theta} \ell(\theta)$$

Partials:

$$\frac{dJ_i(\vec{\theta})}{d\theta_j} = \begin{cases} \frac{d}{d\theta_j}\left[-\log \mu_i(\theta)\right] & \text{if } y^{(i)} = 1 \\ \frac{d}{d\theta_j}\left[-\log(1-\mu_i(\theta))\right] & \text{if } y^{(i)} = 0 \end{cases}$$
$$= \dots$$
$$= \dots$$
$$= \dots$$
Homework
$$= -\left(y^{(i)} - \mu_i(\vec{\theta})\right) x_j^{(i)}$$

Recall: for Linear Reg.
$$\frac{dJ_i(\vec{\theta})}{d\theta_j} = -\left(y^{(i)} - \vec{\theta}^T \vec{x}\right) x_j^{(i)}$$

$$\Rightarrow \nabla J_i(\theta) = -\left(y^{(i)} - \mu_i(\vec{\theta})\right) \vec{x}^{(i)}$$
$$\Rightarrow \nabla J(\theta) = \frac{1}{N}\sum_{i=1}^{N} \nabla J_i(\theta)$$

★ We can follow these gradients using GD or SGD

# Nonlinear Features

Also "nonlinear basis functions"

So far, input was always $\vec{x} = [x_1, \ldots, x_M]^T$

**Key idea:** Do some preprocessing of $\vec{x}$.

Let input be some function, $b$, of $\vec{x}$.

original input: $\vec{x} \in \mathbb{R}^M$ where $M' > M$ usually

new input: $\vec{x}' \in \mathbb{R}^{M'}$

$$\vec{x}' \triangleq b(\vec{x})$$

**Ex#1:** Polynomial basis functions.

$M = 1$ $\quad x \in \mathbb{R}$ $\quad \vec{x}' = b(\vec{x}) \triangleq \begin{bmatrix} 1 \\ x \\ x^2 \\ x^3 \\ x^4 \\ x^5 \\ x^6 \end{bmatrix}$ $\quad x_0'$

$M' = 7$

**Ex#2:**

$M = 2$ $\quad \vec{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$ $\quad \vec{x}' = b(\vec{x}) \triangleq \begin{bmatrix} 1 \\ x_1 \\ x_2 \\ x_1 \cdot x_2 \\ \log(x_1) \\ \log(x_2) \\ \exp(x_1) \\ \exp(x_2) \end{bmatrix}$

$M' = 8$

**For Linear Regression:**

- Nothing changes, just treat $\vec{x}'$ as the input
- $y$ still a linear function of $\vec{x}'$, even though a nonlinear function of $\vec{x}$

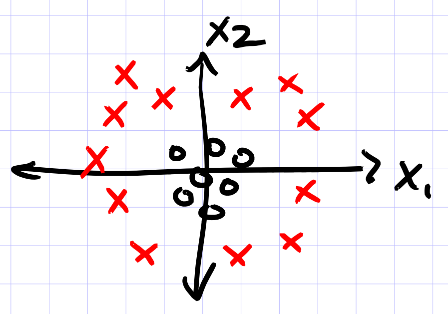
$$\hat{y} = h(b(\vec{x})) = \vec{\theta}^T b(\vec{x}) = \theta^T \vec{x}' \quad \Rightarrow \theta \in \mathbb{R}^{M'}$$

- Computing $b(\vec{x})$ is just preprocessing.

**For Logistic Regression:**

$$p(y=1 \mid b(\vec{x}), \theta) = \frac{1}{1+\exp(-\theta^T b(x))} = \frac{1}{1+\exp(-\theta^T \vec{x}')}$$

- Decision Boundary is linear in $\vec{x}' = b(\vec{x})$, nonlinear in $\vec{x}$




$$\vec{x}' = b(\vec{x}) = \begin{bmatrix} 1 \\ \sqrt{x_1^2 + x_2^2} \\ x_1 \\ x_2 \end{bmatrix}$$

**For KNN:**

Same...

**For NB:**

Same...

**Ex: HW2**

$\vec{x}$ was HTML of articles

$\vec{x}' = b(\vec{x})$ was word indicator vectors.