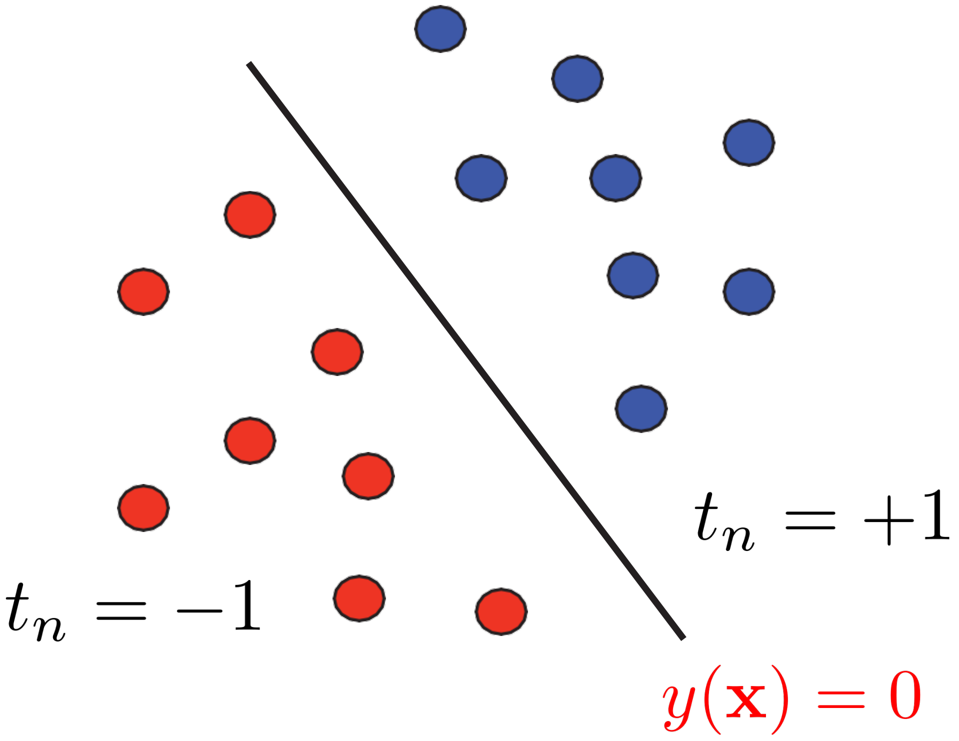
Classification: Three Different Methods

- **Discriminant models**
  - Given training data, assign each data x to one class C_k via a discriminant function
  - *Do not consider distribution* of the training data
- **Probabilistic discriminant models**
  - Given training data, model the posterior class distribution $p(C_k|x)$
  - Use the distribution $p(C_k|x)$ to perform classification for testing data
- **Probabilistic generative models**
  - Given training data, model the joint (data, class) distribution $p(x,C_k)$
  - Find class-conditional distribution $p(x| C_k)$ and class prior distribution $p(C_k)$
  - Then use Bayes rule to compute $p(C_k|x) \sim p(x| C_k)\ p(C_k)$

Probabilistic Discriminant Models: Logistic Regression & Generalized Linear Models

Recap: Discriminative Models

$$y(\mathbf{x}) = \mathbf{w}^T\mathbf{x} \quad t = \{+1, -1\}$$



Fisher LDA $\mathbf{w}^T\mathbf{x}_n \geq y_0$

Perceptron $\mathbf{w}^T\mathbf{x}_n \cdot t_n > 0$

SVM $\mathbf{w}^T\mathbf{x}_n \cdot t_n \geq 1$

How confident a data point is classified as a label +1/-1?

$$p(t = +1|\mathbf{x}; \mathbf{w}) = ?$$

Posterior distribution of t given x

Binary-Class Logistic Regression

Binary-Class Logistic Regression

$$y(\mathbf{x}) = \mathbf{w}^T\mathbf{x}$$

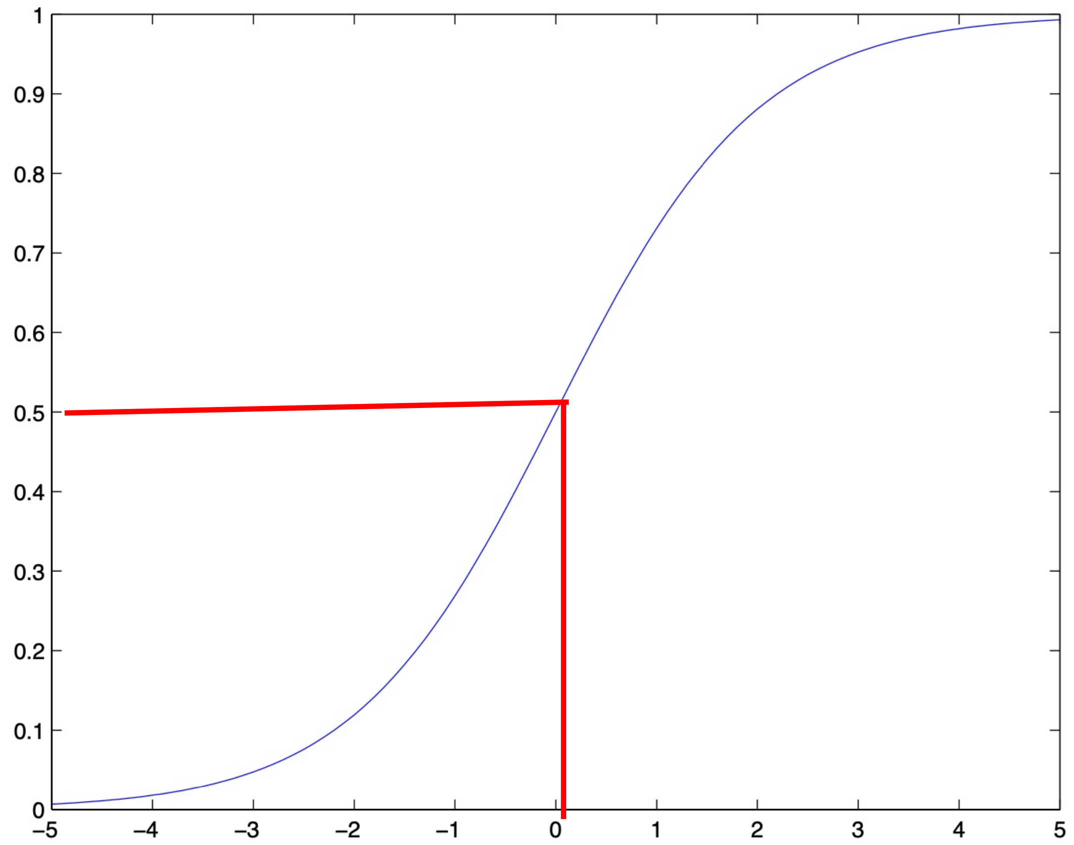
$$p(t=+1|\mathbf{x};\mathbf{w}) \; = \sigma(y(\mathbf{x}))$$

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

Logistic/Sigmoid/S-Shaped function

$$\sigma(y(\mathbf{x})) \; = \frac{1}{1+\exp(-\mathbf{w}^T\mathbf{x})} = h_{\mathbf{w}}(\mathbf{x})$$

$$p(t=0|\mathbf{x};\mathbf{w}) \; = 1 - h_{\mathbf{w}}(\mathbf{x})$$



Properties of Logistic/Sigmoid Function

Symmetric property $\sigma(-a) = 1 - \sigma(a)$

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

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

Logit function $a = \ln\left(\frac{\sigma}{1-\sigma}\right)$

$$\frac{1}{1+\exp(-a)} \Big/ \frac{\exp(-a)}{1+\exp(-a)} = \exp(a)$$

Derivative $\frac{\partial \sigma(a)}{\partial a} = \sigma(a)(1-\sigma(a))$

$$\begin{aligned}
\frac{\partial \sigma(a)}{\partial a} &= \frac{\partial}{\partial a}\frac{1}{1+\exp(-a)} \\
&= \frac{\exp(-a)}{(1+\exp(-a))^2} \\
&= \frac{1}{1+\exp(-a)} \cdot \frac{\exp(-a)}{1+\exp(-a)} \\
&= \sigma(a)(1-\sigma(a))
\end{aligned}$$

Logistic Regression via (Log-)Likelihood

$$p(t=+1|\mathbf{x};\mathbf{w}) = h_{\mathbf{w}}(\mathbf{x}) \qquad p(t=0|\mathbf{x};\mathbf{w}) = 1 - h_{\mathbf{w}}(\mathbf{x})$$

$$\Downarrow$$

$$p(t|\mathbf{x};\mathbf{w}) = h_{\mathbf{w}}(\mathbf{x})^t \cdot (1 - h_{\mathbf{w}}(\mathbf{x}))^{1-t}$$

Assume N training data points independently sampled

$$L(\mathbf{w}) \ = \prod_{n=1}^{N} p(t_n|\mathbf{x}_n;\mathbf{w}) \ = \prod_{n=1}^{N} h_{\mathbf{w}}(\mathbf{x}_n)^{t_n} \cdot (1 - h_{\mathbf{w}}(\mathbf{x}_n))^{1-t_n}$$

$$\ell(\mathbf{w}) = \ln L(\mathbf{w}) \ = \sum_{n=1}^{N} t_n \ln h_{\mathbf{w}}(\mathbf{x}_n) + (1 - t_n) \ln(1 - h_{\mathbf{w}}(\mathbf{x}_n))$$

Chain Rule of Derivative

Let $\boldsymbol{z}$ be a function of $\boldsymbol{y}$ and $\boldsymbol{y}$ be a function of $\boldsymbol{x}$

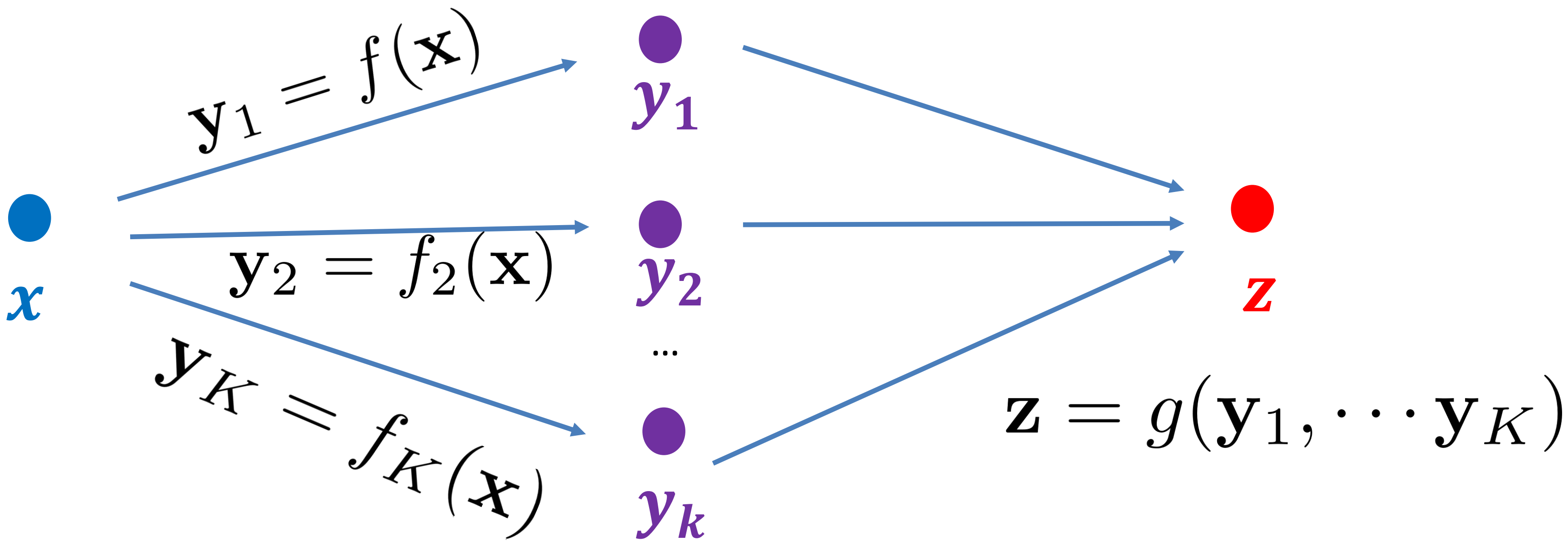
$$\boldsymbol{x} \xrightarrow[\mathbf{y} = f(\mathbf{x})]{} \boldsymbol{y} \xrightarrow[\mathbf{z} = g(\mathbf{y})]{} \boldsymbol{z}$$

$$\frac{\partial \mathbf{z}}{\partial \mathbf{x}} = \frac{\partial \mathbf{z}}{\partial \mathbf{y}} \cdot \frac{\partial \mathbf{y}}{\partial \mathbf{x}}$$

$$\nabla_{\mathbf{x}} \mathbf{z} = \nabla_{\mathbf{y}} \mathbf{z} \cdot \nabla_{\mathbf{x}} \mathbf{y}$$

Chain Rule of Derivative

Let z be a function of y_1 … y_K and each y_k be a function of x



$$\frac{\partial \mathbf{z}}{\partial \mathbf{x}} = \sum_j \frac{\partial \mathbf{z}}{\partial \mathbf{y}_j} \cdot \frac{\partial \mathbf{y}_j}{\partial \mathbf{x}}$$

Maximum Likelihood Estimation

$$\max_{\mathbf{w}} \ell(\mathbf{w}) = \sum_{n=1}^{N} t_n \ln h_{\mathbf{w}}(\mathbf{x}_n) + (1 - t_n) \ln(1 - h_{\mathbf{w}}(\mathbf{x}_n))$$

$$\Longrightarrow \nabla_{\mathbf{w}} \ell(\mathbf{w}) = \sum_{n=1}^{N} \boxed{(t_n - h_{\mathbf{w}}(\mathbf{x}_n)) \cdot \mathbf{x}_n}$$

$$h_{\mathbf{w}}(\mathbf{x}_n) = \frac{1}{1+\exp(-\mathbf{w}^T \mathbf{x})}$$

Error multiplies data features

Identical form as solving least square loss in linear regression

Linear Regression

$$\min_{\mathbf{w}} L(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^{N} \left(t_n - y(\mathbf{x}_n)\right)^2 \quad \nabla_{\mathbf{w}} L(\mathbf{w}) = -\sum_{n=1}^{N} \boxed{(t_n - y(\mathbf{x}_n))\mathbf{x}_n} \quad y(\mathbf{x}_n) = \mathbf{w}^T \mathbf{x}_n$$

Difference: **non-linear function** of w in LogReg, while linear of w in LinReg

Deeper reason: Generalized Linear Models

Iteratively Reweighted Least Square (IRLS)

$$\nabla_{\mathbf{w}}\ell(\mathbf{w}) = \sum_{n=1}^{N}(t_n - h_{\mathbf{w}}(\mathbf{x}_n)) \cdot \mathbf{x}_n = 0 \qquad h_{\mathbf{w}}(\mathbf{x}_n) = \frac{1}{1+\exp(-\mathbf{w}^T\mathbf{x})}$$

No close-form solution due to nonlinearity of the sigmoid function!

Newton's method (Assignment 1: P3)

$$\mathbf{w}^{(new)} = \mathbf{w}^{(old)} - \mathbf{H}(\mathbf{w})^{-1}\nabla_{\mathbf{w}}E(\mathbf{w}) \qquad E(\mathbf{w}) = -\ell(\mathbf{w})$$

Hessian matrix: $\mathbf{H}(\mathbf{w}) = \nabla^2_{\mathbf{w}}E(\mathbf{w})$

$$\nabla_{\mathbf{w}}E(\mathbf{w}) = \textstyle\sum_{n=1}^{N}(h_{\mathbf{w}}(\mathbf{x}_n) - t_n)\mathbf{x}_n = \mathbf{X}(\mathbf{h}_{\mathbf{w}} - \mathbf{t}) \quad \mathbf{h}_{\mathbf{w}} = [h_{\mathbf{w}}(\mathbf{x}_1); \cdots; h_{\mathbf{w}}(\mathbf{x}_N)]$$

$$\mathbf{H}(\mathbf{w}) = \textstyle\sum_{n=1}^{N} h_{\mathbf{w}}(\mathbf{x}_n)(1 - h_{\mathbf{w}}(\mathbf{x}_n)\mathbf{x}_n\mathbf{x}_n^T = \mathbf{XRX}^T$$

$$\mathbf{R} = \mathrm{diag}(h_{\mathbf{w}}(\mathbf{x}_1)(1 - h_{\mathbf{w}}(\mathbf{x}_1), \cdots, h_{\mathbf{w}}(\mathbf{x}_N)(1 - h_{\mathbf{w}}(\mathbf{x}_N))$$

Iteratively Reweighted Least Square (IRLS)

Iteratively Reweighted Least Square

$$\mathbf{w}^{(new)} = \mathbf{w}^{(old)} - \mathbf{H}(\mathbf{w})^{-1}\nabla_{\mathbf{w}}E(\mathbf{w})$$

$$= \mathbf{w}^{(old)} - \left(\mathbf{X}\mathbf{R}\mathbf{X}^T\right)^{-1} \cdot \mathbf{X}(\mathbf{h}_{\mathbf{w}} - \mathbf{t})$$

$$= \left(\mathbf{X}\mathbf{R}\mathbf{X}^T\right)^{-1}\left(\mathbf{X}\mathbf{R}\mathbf{X}^T\mathbf{w}^{(old)} - \mathbf{X}(\mathbf{h}_{\mathbf{w}} - \mathbf{t})\right)$$

Normal equation

$$= \left(\mathbf{X}\mathbf{R}\mathbf{X}^T\right)^{-1}\mathbf{X}\mathbf{R}\mathbf{Z} \qquad \mathbf{Z} = \mathbf{X}^T\mathbf{w}^{(old)} - \mathbf{R}^{-1}(\mathbf{h}_{\mathbf{w}} - \mathbf{t}))$$

$$\mathbf{X}\mathbf{X}^T\mathbf{w} = \mathbf{X}\mathbf{y}$$

$$= \left(\tilde{\mathbf{X}}\tilde{\mathbf{X}}^T\right)^{-1}\tilde{\mathbf{X}}\mathbf{R}^{1/2}\mathbf{Z} \qquad \tilde{\mathbf{X}} = \mathbf{X}\mathbf{R}^{1/2}$$

$$\mathbf{w}^\star = \left(\mathbf{X}\mathbf{X}^T\right)^{-1}\mathbf{X}\mathbf{y}$$

$$= \left(\tilde{\mathbf{X}}\tilde{\mathbf{X}}^T\right)^{-1}\tilde{\mathbf{X}}\tilde{\mathbf{Y}} \qquad \tilde{\mathbf{Y}} = \mathbf{R}^{1/2}\mathbf{Z}$$

Normal equations for a **weighted** least square, with weights defined in **R**

R is not constant and depends on **w,** and is iteratively updated

$$\mathbf{R} = \mathrm{diag}(h_{\mathbf{w}}(\mathbf{x}_1)(1 - h_{\mathbf{w}}(\mathbf{x}_1), \cdots, h_{\mathbf{w}}(\mathbf{x}_N)(1 - h_{\mathbf{w}}(\mathbf{x}_N))$$

Logistic Regression: Classification

$\mathbf{w}^*$ $\mathbf{x}$

$$h_{\mathbf{w}^*}(\mathbf{x}) = p(t = +1|\mathbf{x}; \mathbf{w}^*)$$

≥ 0.5	+1
< 0.5	0

$\mathbf{x}_A, \mathbf{x}_B$

$$h_{\mathbf{w}^*}(\mathbf{x}_A) > h_{\mathbf{w}^*}(\mathbf{x}_B)$$

A is more confident than B
to be classified as +1

Multi-Class Logistic (Softmax) Regression

Multi-Class Classification

Face recognition

#classes = #people

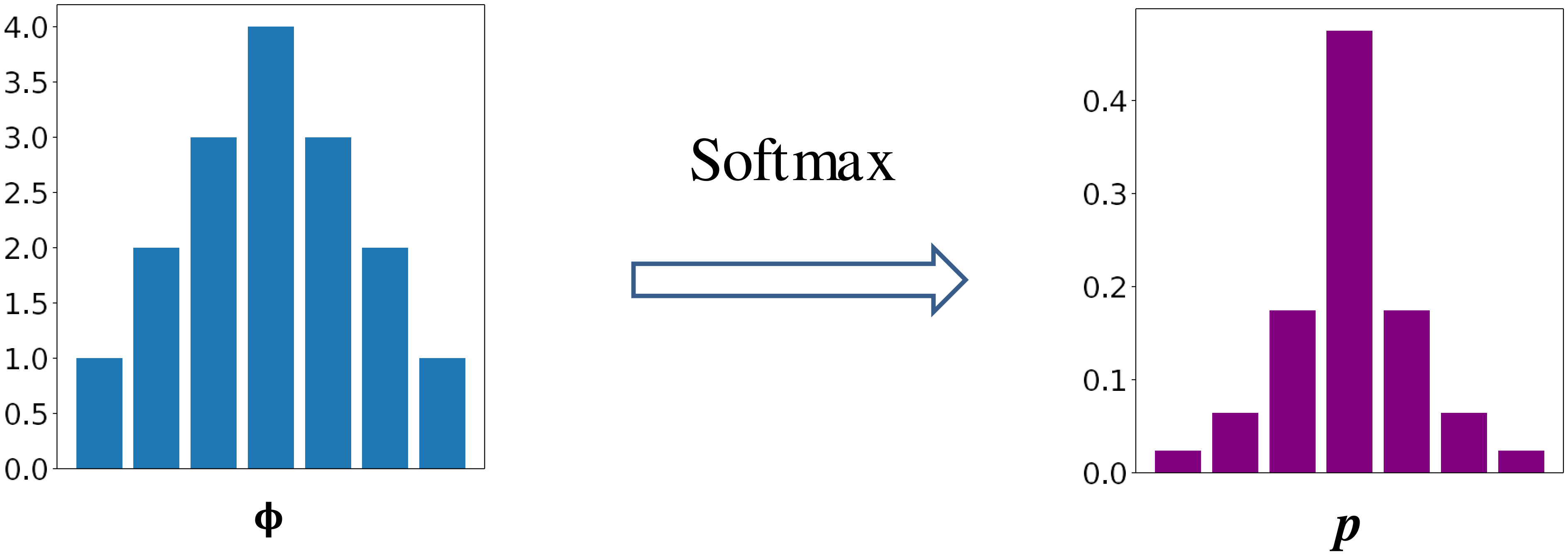
Hand-written digit recognition

#classes = 10

Softmax Function

$$\boldsymbol{\phi} \in \mathbb{R}^K; \; \mathbf{p} = \mathrm{Softmax}(\boldsymbol{\phi}) \in \mathbb{R}^K$$

$$p_k = \frac{\exp(\phi_k)}{\sum_{j=1}^{K} \exp(\phi_j)} \qquad \sum_{k=1}^{K} p_k = 1$$



One-Hot Encoding

- Binary classification
  - t = +1 or -1 (0)

- Multiclass classification:
  - Label t: One-hot encoding (1-of-K binary coding)
  - #classes=10 (e.g., in digit recognition)
    - t_n = 3 is $\mathbf{t_n}$ = [0,0,0,1,0,0,0,0,0,0] $\in \{0,1\}^{10}$

Cross-Entropy

- The vectors $\mathbf{y}$ and $\mathbf{p}$ are K-dim vectors with nonnegative entries

$$y_1 + \cdots + y_K = 1 \quad \text{and} \quad p_1 + \cdots + p_K = 1.$$

- Cross-entropy between $\mathbf{y}$ and $\mathbf{p}$

$$H(\mathbf{y}, \mathbf{p}) = -\sum_{k=1}^{K} y_k \log p_k \geq 0$$

- Cross-entropy measures the ***dissimilarity*** between $\mathbf{y}$ and $\mathbf{p}$
  - $\mathbf{y}$=[1;0] and $\mathbf{p}$=[0.99; 0.01] => $H(\mathbf{y}, \mathbf{p}) \approx 0$
  - $\mathbf{y}$=[1;0] and $\mathbf{p}$=[0.5; 0.5] => $H(\mathbf{y}, \mathbf{p}) = \log 2$

From Logistic Function to Softmax Function

For binary classification, logistic function to define the posterior probability

$$p(t=+1|\mathbf{x};\mathbf{w})=\sigma(y(\mathbf{x}))$$

$$p(t=0|\mathbf{x};\mathbf{w})=1-\sigma(y(\mathbf{x}))$$

$$y(\mathbf{x})=\mathbf{w}^T\mathbf{x}$$

For K-class classification, we expect to have a function such that

$$\sum_{k=1}^{K}p(t=k|\mathbf{x};\mathbf{W})=1$$

$$\mathbf{W}=[\mathbf{w}_1,\mathbf{w}_2,\cdots,\mathbf{w}_K]$$

$$\mathbf{y}(\mathbf{x})=\mathbf{W}^T\mathbf{x}$$

Softmax function

$$p(t=k|\mathbf{x};\mathbf{W})=\frac{\exp(y_k(\mathbf{x}))}{\sum_j\exp(y_j(\mathbf{x}))}$$

Likelihood of Multi-Class Logistic Regression

$$p(t = k|\mathbf{x};\mathbf{W}) = \frac{\exp(y_k(\mathbf{x}))}{\sum_j \exp(y_j(\mathbf{x}))} = p_k(\mathbf{x}) \qquad \mathbf{y}(\mathbf{x}) = \mathbf{W}^T\mathbf{x}$$

Assume N training data points independently sampled

$$L(\mathbf{W}) = \prod_{n=1}^{N} p(\mathbf{t}_n|\mathbf{x}_n;\mathbf{W}) = \prod_{n=1}^{N}\prod_{k=1}^{K} p(t = k|\mathbf{x}_n;\mathbf{W})^{t_{nk}}$$

$$\ell(\mathbf{W}) = \ln L(\mathbf{W}) = \sum_{n=1}^{N}\sum_{k=1}^{K} t_{nk} \ln p_k(\mathbf{x}_n)$$

Negative cross entropy

Maximizing (log-)likelihood equals to minimizing cross entropy loss

Maximum Likelihood Estimation

$$\max_{\mathbf{W}} \ell(\mathbf{W}) = \sum_{n=1}^{N} \sum_{k=1}^{K} t_{nk} \ln p_k(\mathbf{x}_n)$$

$$p_k(\mathbf{x}) = \frac{\exp(y_k(\mathbf{x}))}{\sum_j \exp(y_j(\mathbf{x}))} \qquad \mathbf{y}(\mathbf{x}) = \mathbf{W}^T \mathbf{x}$$

$$\nabla_{\mathbf{w}_i} \ell(\mathbf{W}) = \sum_{n=1}^{N} \sum_{k=1}^{K} t_{nk} \frac{\nabla_{\mathbf{w}_i} p_k(\mathbf{x}_n)}{p_k(\mathbf{x}_n)}$$

$$= \sum_{n=1}^{N} \boxed{(t_{ni} - y_{ni})\mathbf{x}_n}$$

Still error multiplies features

$$\nabla_{\mathbf{w}_i} p_k(\mathbf{x}_n) = \sum_{j=1}^{K} \nabla_{y_j(\mathbf{x}_n)} p_k \cdot \nabla_{\mathbf{w}_i} y_j(\mathbf{x}_n)$$

$$\nabla_{y_j(\mathbf{x}_n)} p_k = \begin{cases} p_k(1 - p_k), & \text{if } j = k \\ -p_j p_k, & \text{if } j \neq k \end{cases}$$

$$\nabla_{\mathbf{w}_i} y_j(\mathbf{x}_n) = \begin{cases} \mathbf{x}_n, & \text{if } i = j, \\ \mathbf{0}, & \text{if } i \neq j \end{cases}$$

Multi-Class Logistic Regression: Classification

$$\mathbf{W}^* \quad \mathbf{x} \qquad \mathbf{h}_{\mathbf{W}^*}(\mathbf{x}) = \begin{bmatrix} h_{\mathbf{w_1^*}}(\mathbf{x}) \\ h_{\mathbf{w_2^*}}(\mathbf{x}) \\ \vdots \\ h_{\mathbf{w_k^*}}(\mathbf{x}) \end{bmatrix} = \begin{bmatrix} p(t=1|\mathbf{x};\mathbf{W}^*) \\ p(t=2|\mathbf{x};\mathbf{W}^*) \\ \vdots \\ p(t=K|\mathbf{x};\mathbf{W}^*) \end{bmatrix}$$

$$\tilde{t} = \arg \max_{k\in\{1,2,\cdots,K\}} \mathbf{h}_{\mathbf{w}_k^*}(\mathbf{x})$$

$$\mathbf{x}_A,\ \mathbf{x}_B \qquad h_{\mathbf{w}_{\tilde{t}}^*}(\mathbf{x}_A) > h_{\mathbf{w}_{\tilde{t}}^*}(\mathbf{x}_B)$$

A is more confident than B
to be classified as $\tilde{t}$

Generalized Linear Models (GLMs)

The Exponential Family

Exponential family distribution

$$p(y;\eta) = b(y)\exp\left(\eta^T T(y) - a(\eta)\right)$$

y: random variable of the distribution

η: natural parameter of the distribution

T(y): sufficient statistic for the distribution, often T(y) = y

a(η): log partition function (exp(- *a*(η) makes sure the distribution p(y; η) rational)

Fixed *T*, *a* and *b* defines a family of distributions that is parameterized by η

Example: Bernoulli Distribution

$$p(y;\eta) = b(y)\exp\left(\eta^T T(y) - a(\eta)\right)$$

$$\text{Bernoulli dist.} Ber(y;\phi) : p(y=1;\phi) = \phi;\ p(y=0;\phi) = 1-\phi$$

$$p(y;\phi) = \phi^y (1-\phi)^{1-y}$$

$$= \exp\left(y \log\phi + (1-y)\log(1-\phi)\right)$$

$$= \exp\left(\log\left(\tfrac{\phi}{1-\phi}\right)y + \log(1-\phi)\right)$$

$$b(y) = 1$$

$$T(y) = y$$

$$\eta = \log\left(\tfrac{\phi}{1-\phi}\right)$$

$$a(\eta) = -\log(1-\phi)$$
$$= \log(1+\exp(\eta))$$

Example: Gaussian Distribution

$$p(y;\eta) = b(y)\exp\big(\eta^T T(y) - a(\eta)\big)$$

Gaussian dist $\mathcal{N}(y;\mu,1) : p(y;\mu) = \frac{1}{\sqrt{2\pi}}\exp\big(-\frac{(y-\mu)^2}{2}\big)$

$$p(y;\mu) = \frac{1}{\sqrt{2\pi}}\exp\big(-\tfrac{1}{2}y^2\big)\exp(\mu y - \tfrac{1}{2}\mu^2)$$

$$b(y) = \frac{1}{\sqrt{2\pi}}\exp(-y^2/2)$$

$$T(y) = y$$

$$\eta = \mu$$

$$a(\eta) = -\mu^2/2 = \eta^2/2$$

Constructing GLMs

Consider a classification/regression problem where we predict the value of some random variable *t* as a function of *x*.

To derive a GLM for this problem, we make three assumptions about the conditional posterior distribution of t given x and about model:

- 1. t | x; **w** ~ ExpFamily(η). I.e., given x and ***w***, the distribution of t follows some exponential family distribution, with parameter η

- 2. Given x, our goal is to predict the expected value of t. This means we would like the prediction y(x) outputted by the learned model satisfies y(x) = E[t|x; **w**].

- 3. The natural parameter η and the input x are related linearly: $\eta = w^T x$ (or if η is vector-valued, then $\eta = W^T x$)

GLM: Linear Regression

Assumption#1: t | x; **w** ~ ExpFamily(η), Gaussian dist. $N(\mu, \sigma^2)$ and *η=μ*

$$p(t|\mathbf{x};\mathbf{w}) \sim \mathcal{N}(\mu, \sigma^2)$$

Assumption#2: y(x) = E[t|x; **w**]

$$y_{\mathbf{w}}(\mathbf{x}) = \mathbb{E}[t|\mathbf{x};\mathbf{w}] = \mu = \eta$$

Assumption#3: natural parameter η and the inputs x are related linearly

$$\eta = \mathbf{w}^T\mathbf{x}$$

$$\Longrightarrow \quad y_{\mathbf{w}}(\mathbf{x}) = \mathbf{w}^T\mathbf{x}$$

GLM: Logistic Regression

Assumption#1: t | x; **w** ~ ExpFamily(η), Bernoulli dist. $Ber(\phi)$ with $\phi = 1/(1+\exp(-\eta))$

$$p(t|\mathbf{x};\mathbf{w}) \sim Ber(\phi)$$

Assumption#2: y(x) = E[t|x; **w**]

$$h_{\mathbf{w}}(\mathbf{x}) = \mathbb{E}[t|\mathbf{x};\mathbf{w}] = \phi = 1/(1+\exp(-\eta))$$

Assumption#3: natural parameter η and the inputs x are related linearly

$$\eta = \mathbf{w}^T\mathbf{x}$$

$$\Longrightarrow \quad h_{\mathbf{w}}(\mathbf{x}) = \frac{1}{1+\exp(-\mathbf{w}^T\mathbf{x})}$$

Summary

- Logistic regression & Multiclass logistic (softmax) regression
  - Probabilistic discriminant models
  - Model label's posterior probability p(t | x; w)
  - Logistic/Sigmoid function & softmax function

- (Multiclass) logistic regression vs. linear regression
  - Identical parameter update (error * data features)
  - No closed-form solution due to its nonlinearity
  - Solution: Iteratively Reweighted Least Square (IRLS)

- Generalized linear models (GLMs)
  - Exponential family distributions
    - Bernoulli distribution, Gaussian distribution, Poisson distribution, etc.
  - (Multiclass) logistic regression & linear regression are special cases