CHAPTER 10:

Logistic Regression

Logistic Regression - Motivation

- Lets now focus on the binary classification problem in which
  - y can take on only two values, 0 and 1.
  - x is a vector of real-valued features, $< x_1 \ldots x_n >$

- We could approach the classification problem ignoring the fact that y is discrete-valued, and use our old linear regression algorithm to try to predict y given x.
  - However, it doesn't make sense for f(x) to possibly take values larger than 1 or smaller than 0 when we know that $y \in \{0, 1\}$.

- Since the output must be 0 or 1, we cannot directly use a linear model to estimate f(x).

- Furthermore, we would like to f(x) to represent the probability $P(C_1|x)$. Lets call it p.

- We will model the log of the odds of the probability p as a linear function of the input x.

$$odds = \frac{p}{1-p}$$

ln (odds of p) = ln (p/(1-p)) = **w**.x

If there is a 75% chance that it will rain tomorrow, then the odds of it raining tomorrow are 3 to 1. $(\frac{3}{4})/\frac{1}{4}=3/1$.

- This is the logit function. I.e. logit(p) = ln (p/(1-p))

We want: $f(x) = P(C_1 \mid \mathbf{x}) = p$
We will model as: $\ln (p/(1-p)) = \mathbf{w.x}$

- By applying the *inverse of the logit function, that is* the logistic function, on both sides, we get:

$$\text{logit}^{-1} (\ln (p/(1-p)))= \text{sigmoid} (\ln (p/(1-p))) = p$$

- Applying it on the RHS as well, we get

$$p = \text{logit}^{-1} (\mathbf{w.x}) = 1 / (1 + e^{-\mathbf{w.x}})$$

- Thus: $f(x) = 1 / (1 + e^{-w.x})$ and we will interpret it as $p = P(C_1 \mid \mathbf{x})$
- $= P (y=1 \mid \mathbf{x})$

Odds & Odds Ratios

The odds has a range of 0 to ∞ with values :

- greater than 1 associated with an event being more likely to occur than not to occur and
- values less than 1 associated with an event that is less likely to occur than not occur.

$$\ln(odds) = \ln\left(\frac{p}{1-p}\right) = \ln(p) - \ln(1-p)$$

- The **logit** is defined as the log of the odds (-∞ to +∞)

As **β.x** gets really big, p approaches 1

As **β.x** gets really small, p approaches 0

The Logistic Regression Model

$$\ln[p/(1-p)] = \beta_0 + \beta_1 X$$

- p is the probability that the event Y occurs, p(Y=1)
  - [range=0 to 1]
- p/(1-p) is the "odds ratio"
  - [range=0 to ∞]
- ln[p/(1-p)]: log odds ratio, or "logit“
  - [range=-∞ to +∞]

- We have:

  $f(\mathbf{x}) = 1 / (1 + e^{-\mathbf{w}.\mathbf{x}})$ and we will interpret it as $f(\mathbf{x}) = P(y=1 \mid \mathbf{x})$

  (in short p)

  Thus we have:

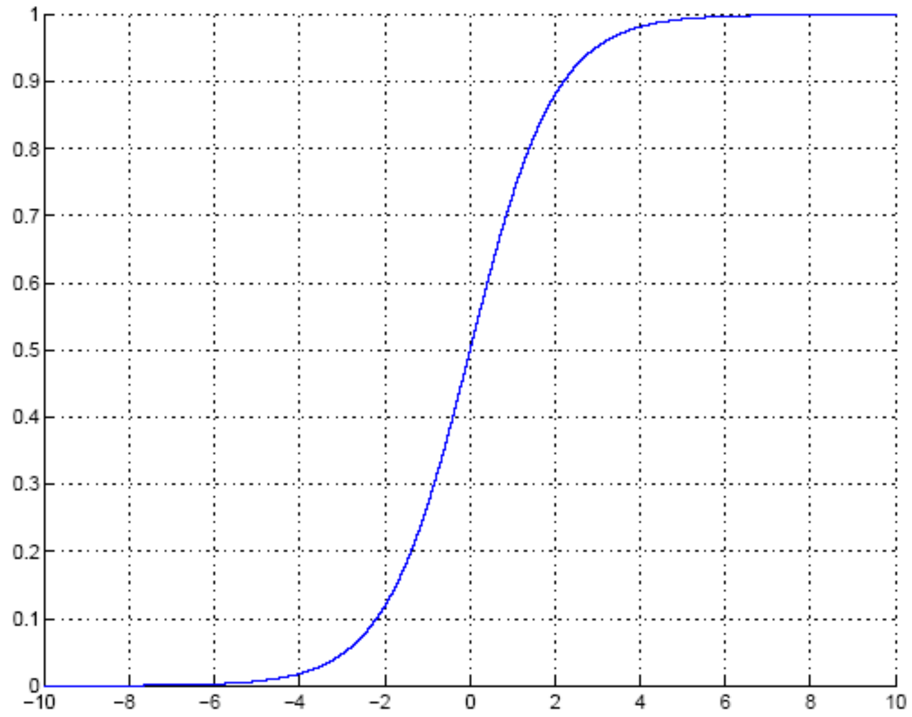
  $P(y=1 \mid \mathbf{x}) = f(\mathbf{x})$

  $P(y=0 \mid \mathbf{x}) = 1 - f(\mathbf{x})$

  - Which can be written more compactly by unifying the two rules :

$$P(y \mid \mathbf{x}) = (f(\mathbf{x}))^{y} (1 - f(\mathbf{x}))^{1-y} \quad \text{where } y \in \{0, 1\}$$

Logistic Regression Decision



1. Calculate $\mathbf{w}^T\mathbf{x}$ and choose C_1 if $\mathbf{w}^T\mathbf{x} > 0$, or

2. Calculate $\mathrm{f(x)=sigmoid}\left(\mathbf{w}^T\mathbf{x}\right)$ and choose C_1 if $f(\mathbf{x}) > 0.5$

Logistic Regression Decision

- Properties
  - **Linear Decision boundary**

  - **Need for scaling input features:**
    - Strictly speaking not needed, but useful in regularized version where we add the weight vector norm (which in turn depends on the scale of the input dimensions) as penalty.

- $P(y \mid \mathbf{x}; \mathbf{w}) = (f(\mathbf{x}))^y (1 - f(\mathbf{x}))^{1-y}$

- Find **w** that maximizes the log likelihood of data

  Equivalently, minimizes the negative log likelihood of data

$$\mathcal{X} = \{\mathbf{x}^t, y^t\}_t \quad y^t | \mathbf{x}^t \sim \text{Bernoulli}(p)$$

$$f(\mathbf{x}) = P(y = 1|\mathbf{x}) = \frac{1}{1 + \exp\left[-\left(\mathbf{w}^T\mathbf{x} + w_0\right)\right]}$$

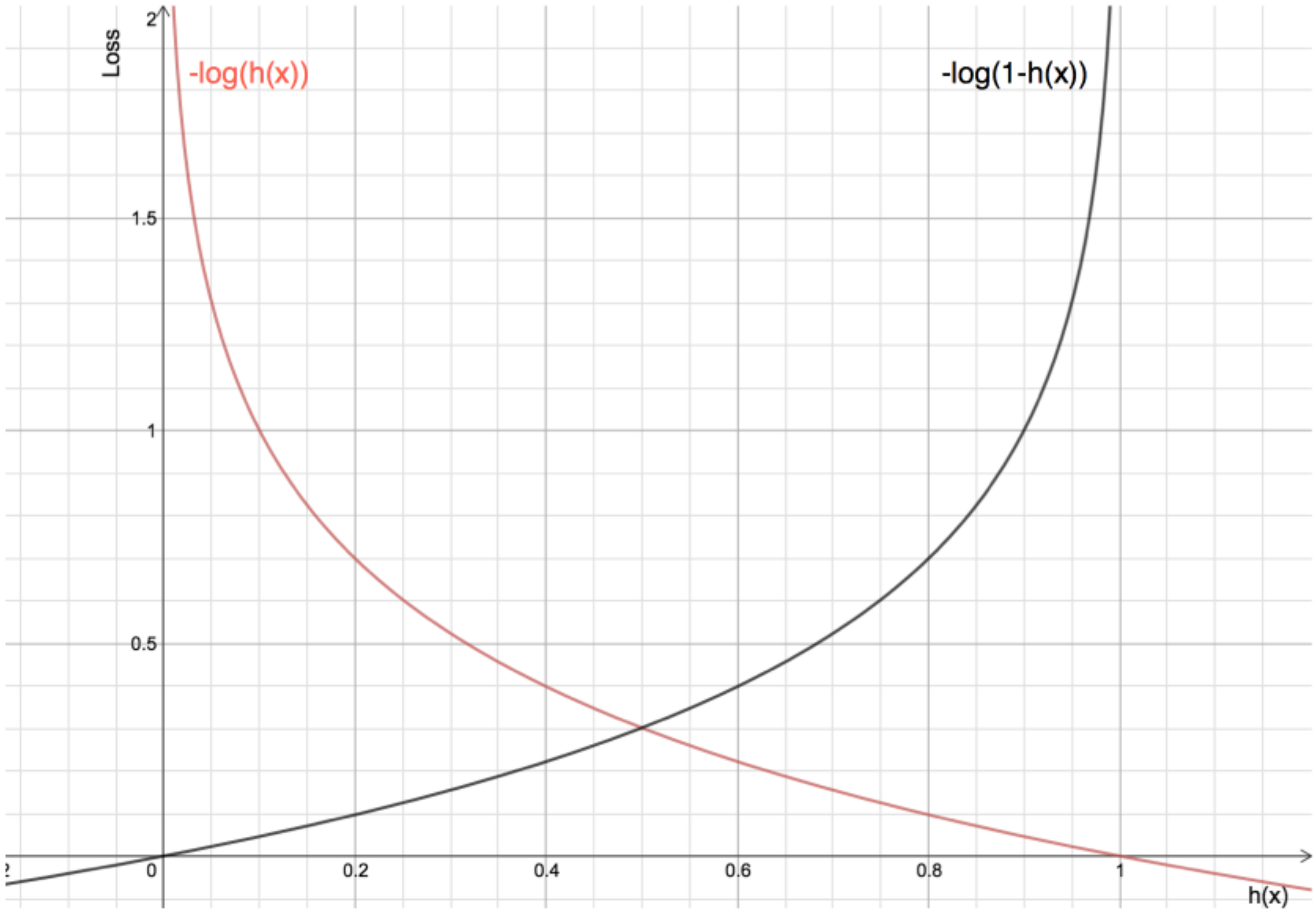
$$l(\mathbf{w}, w_0|\mathcal{X}) = \prod_t \left(f(x^t)\right)^{\left(y^t\right)} \left(1 - f(x^t)\right)^{\left(1-y^t\right)}$$

$$E = -\log l$$

$$E(\mathbf{w}, w_0|\mathcal{X}) = -\sum_t y^t \log f(x^t) + \left(1 - y^t\right) \log \left(1 - f(x^t)\right)$$

cross-entropy loss

Cross-entropy loss



Softmax Regression

Multinomial Logistic Regression

MaxEnt Classifier

Softmax Regression

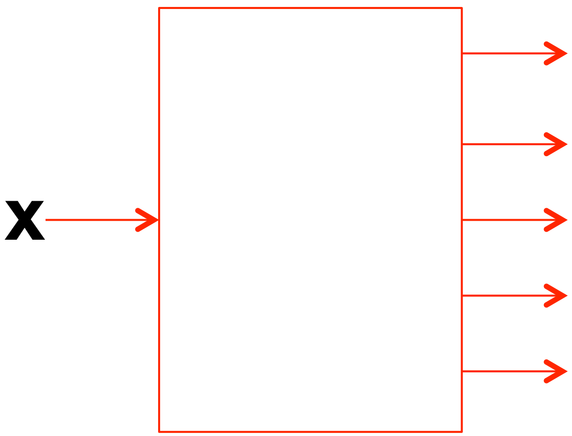
- Softmax regression model generalizes logistic regression to classification problems where **the class label *y* can take on more than two possible values.**

  - The response variable y can take on any one of k values, so $y \in \{1, 2, . . . , k\}$.

Softmax Regression

- Softmax regression model generalizes logistic regression to classification problems where **the class label *y* can take on more than two possible values.**

  - The response variable y can take on any one of k values, so $y \in \{1, 2, . . . , K\}$.



$$f(\mathbf{x}) = \begin{bmatrix} P(y=1|\mathbf{x}) \\ P(y=2|\mathbf{x}) \\ \cdots \\ P(y=K|\mathbf{x}) \end{bmatrix} = \frac{1}{\sum_{j=1}^{K} \exp\left[\mathbf{w}_j^T \mathbf{x}\right]} \begin{bmatrix} \exp\left[\mathbf{w}_1^T \mathbf{x}\right] \\ \exp\left[\mathbf{w}_2^T \mathbf{x}\right] \\ .. \\ \exp\left[\mathbf{w}_k^T \mathbf{x}\right] \end{bmatrix}$$

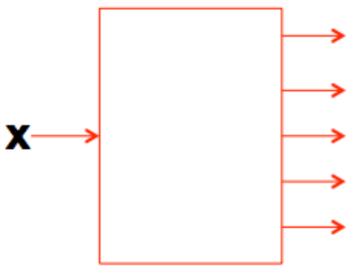
$$\mathcal{X} = \{\mathbf{x}^t, y^t\}_t \qquad y^t | \mathbf{x}^t \sim \text{Multinomial}(\ldots)$$

$$o_k = \hat{P}(y = k | \mathbf{x}) = \frac{\exp\left[\mathbf{w}_k^T \mathbf{x}\right]}{\sum_{j=1}^{K} \exp\left[\mathbf{w}_j^T \mathbf{x}\right]}, k = 1, \ldots, K$$

Maximizing the likelihood is equivalent to minimizing the negative log likelihood (cross-entropy error)

$$l(\{\mathbf{w}_k\} | \mathcal{X}) = \prod_t \prod_k \left(o_k^t\right)^{\left(y_k^t\right)}$$

$$E(\{\mathbf{w}_k\} | \mathcal{X}) = -\sum_{t=1} \sum_{k=1}^{K} 1\{y^t = k\} \log o_k^t = -\sum_{t=1} \sum_{k=1}^{K} 1\{y^t = k\} \log \frac{\exp\left[\mathbf{w}_k^T \mathbf{x}^t\right]}{\sum_{j=1}^{K} \exp\left[\mathbf{w}_j^T \mathbf{x}^t\right]}$$



O_k where k is the correct class