Sentiment analysis

Logistic Regression (LR)

Generative vs. Discriminative

NB is a Generative Model!

$$P(H|E) = \frac{P(E|H)P(H)}{P(E)} = \frac{P(E|H)P(H)}{\sum_{H \in \mathcal{H}} P(E|H)\,P(H)}$$

If we choose H, $P(H)$, how can we generate instances of event/evidences $P(E|H)$
If we are in H4: Winter and in Canada, generate a day → it would be mostly no sun!

LR is Discriminative!

Logistic Regression

A step forward to Naïve Bayes!

there may be some correlation in the input features!

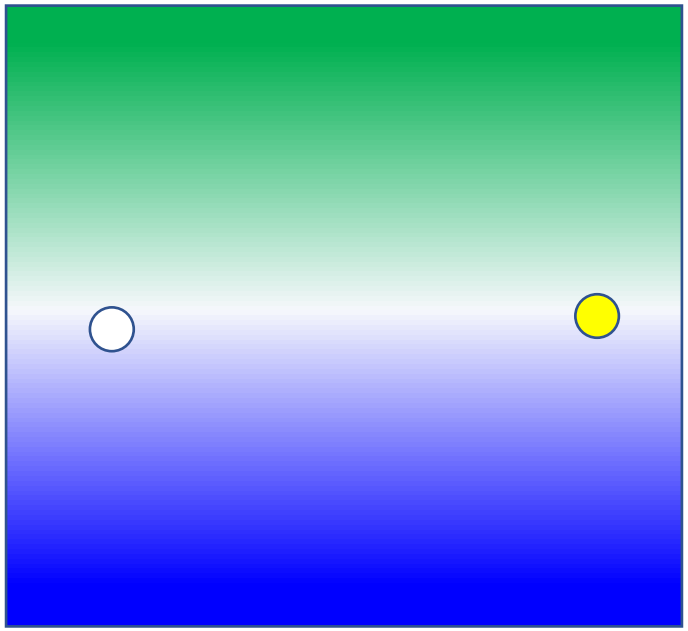
Logistic Regression

a neural network can be viewed as a series of logistic regression classifiers stacked on top of each other.

Logistic Regression

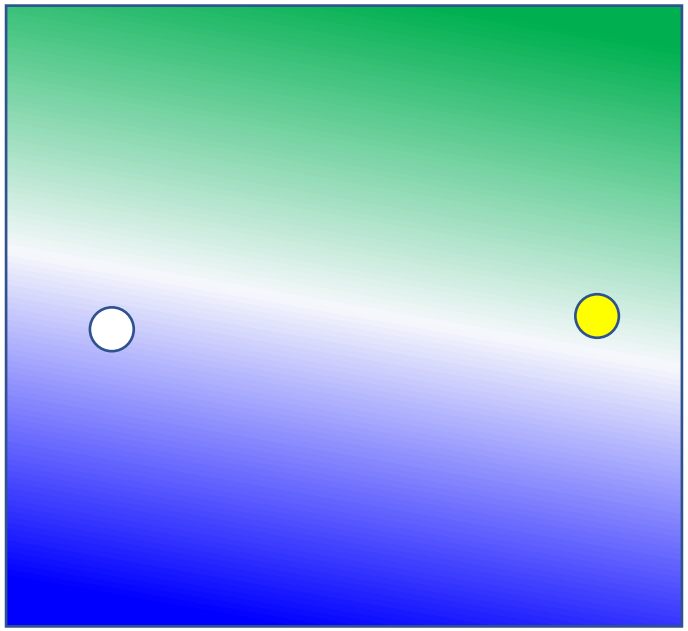
Don’t care about hypothesis and their probabilities.
Try to maximize the distance of data from different classes.
Try to discriminate the most!

Logistic Regression



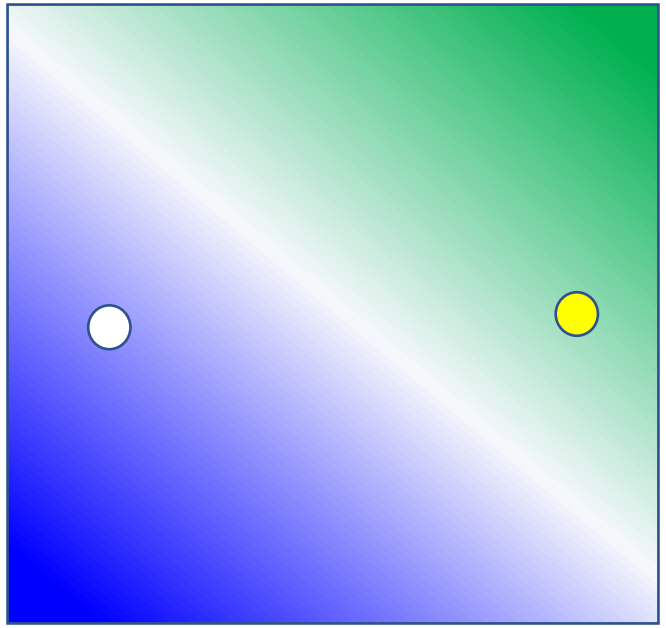
$P(+|x_+)=0.50$ $P(-|x_+)=0.50$
$P(-|x_-)=0.50$ $P(+|x_-)=0.50$

Logistic Regression



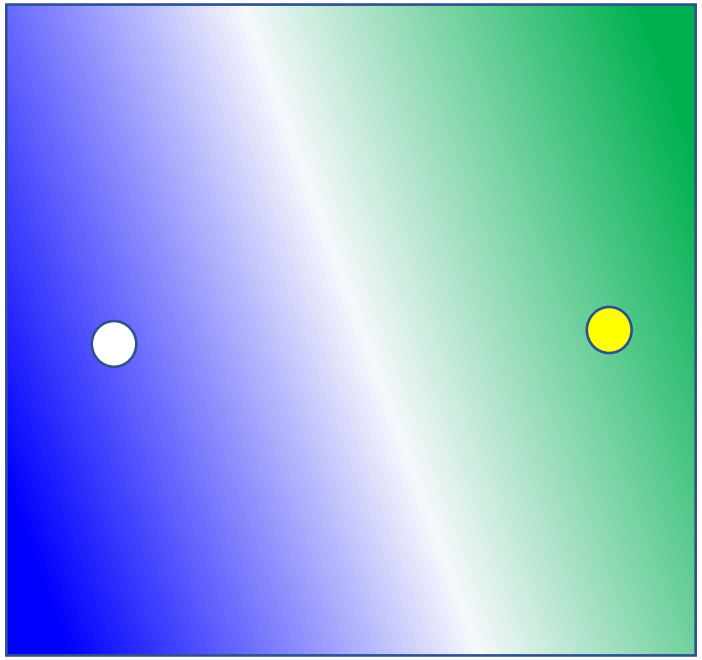
$P(+|x_+)=0.55$ $P(-|x_+)=0.45$
$P(-|x_-)=0.55$ $P(+|x_-)=0.45$

Logistic Regression



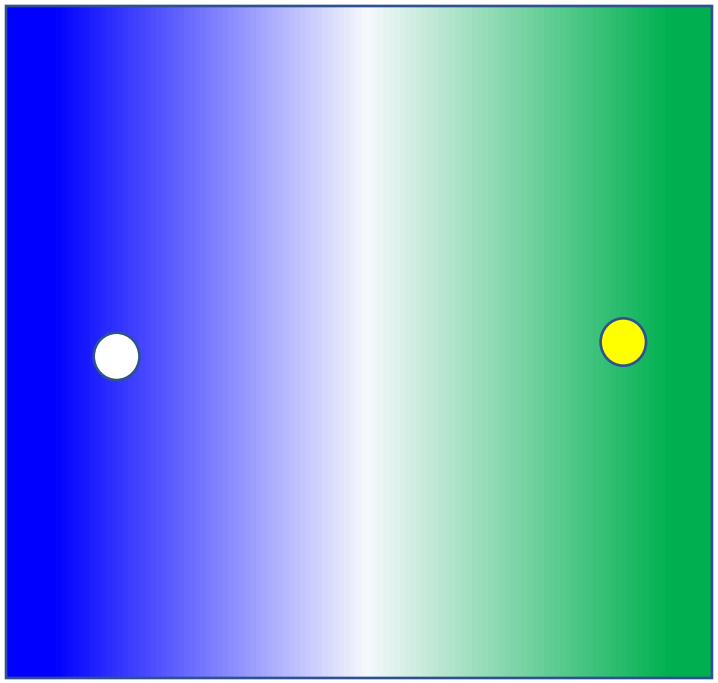
$P(+|x_+)=0.65$ $P(-|x_+)=0.35$
$P(-|x_-)=0.65$ $P(+|x_-)=0.35$

Logistic Regression



$P(+|x_+)=0.75$ $P(-|x_+)=0.25$
$P(-|x_-)=0.75$ $P(+|x_-)=0.25$

Logistic Regression



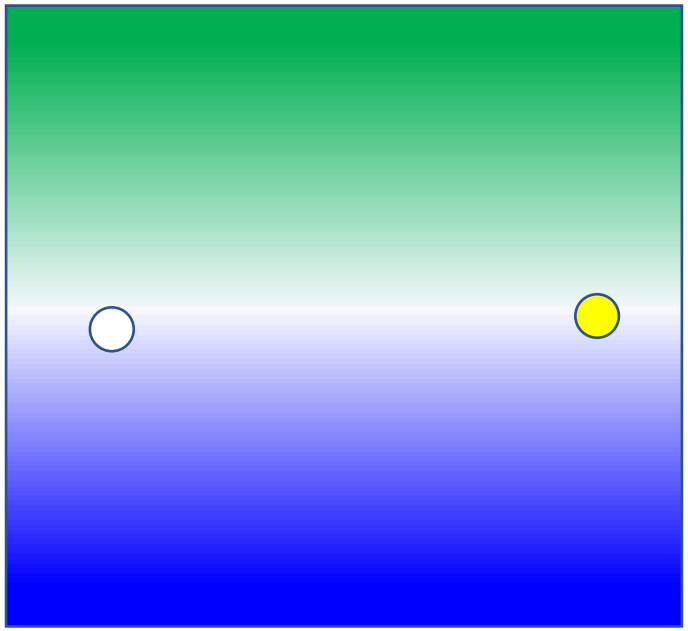
$P(+|x_+)=0.85$ $P(-|x_+)=0.15$
$P(-|x_-)=0.85$ $P(+|x_-)=0.15$

Logistic Regression

(I) Iterative Process
(II) Optimization = Discriminate classes the most

Logistic Regression

More for $P(-|x_-) \uparrow$

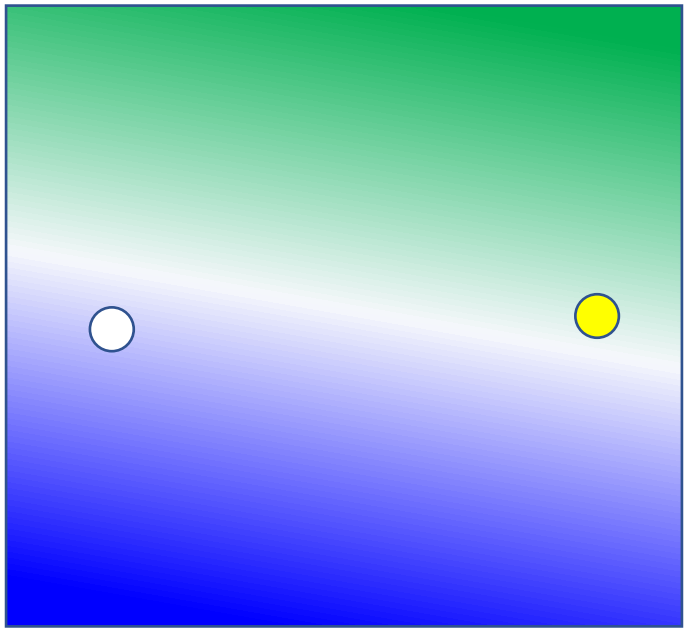


More for $P(+|x_+) \uparrow$

$P(+|x_+)=0.50$ $P(-|x_+)=0.50$
$P(-|x_-)=0.50$ $P(+|x_-)=0.50$

Logistic Regression

Even more for $P(-|x_-) \uparrow$

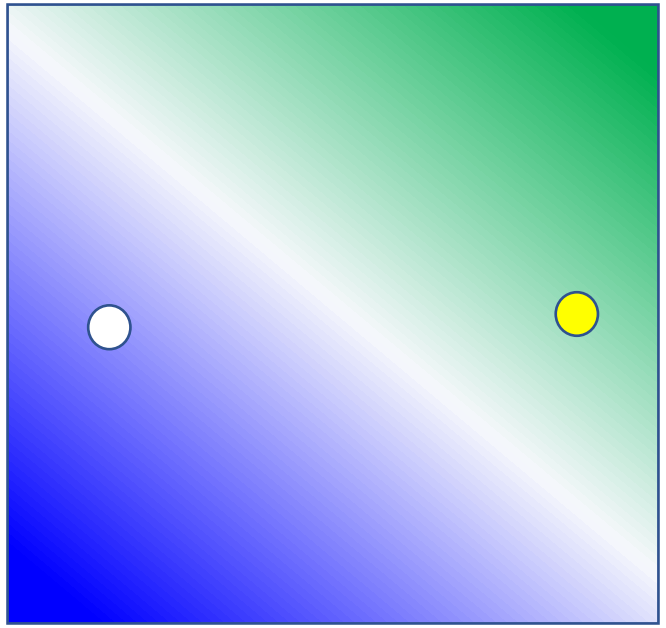


Even more for $P(+|x_+) \uparrow$

$P(+|x_+)=0.55$ $P(-|x_+)=0.45$
$P(-|x_-)=0.55$ $P(+|x_-)=0.45$

Logistic Regression

Even more for $P(-|x_-) \uparrow$

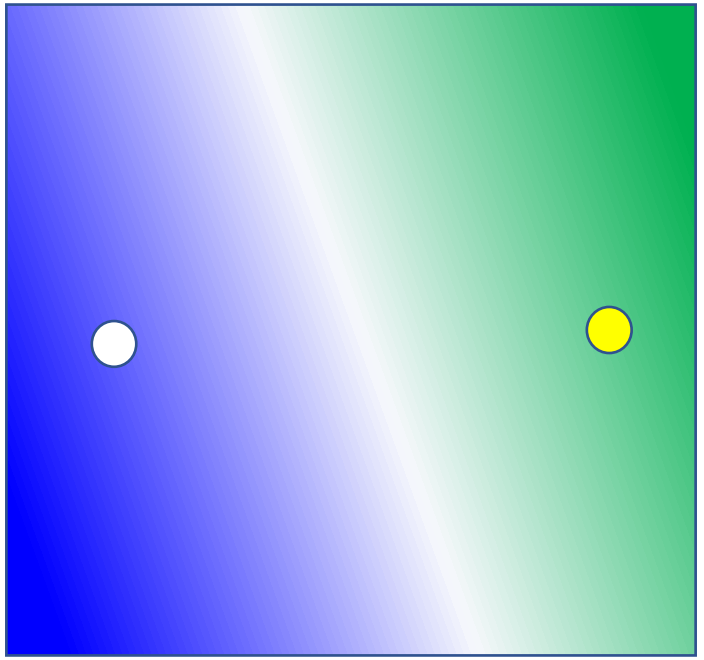


Even more for $P(+|x_+) \uparrow$

$P(+|x_+)=0.65$ $P(-|x_+)=0.35$
$P(-|x_-)=0.65$ $P(+|x_-)=0.35$

Logistic Regression

Even more for $P(-|x_-) \uparrow$

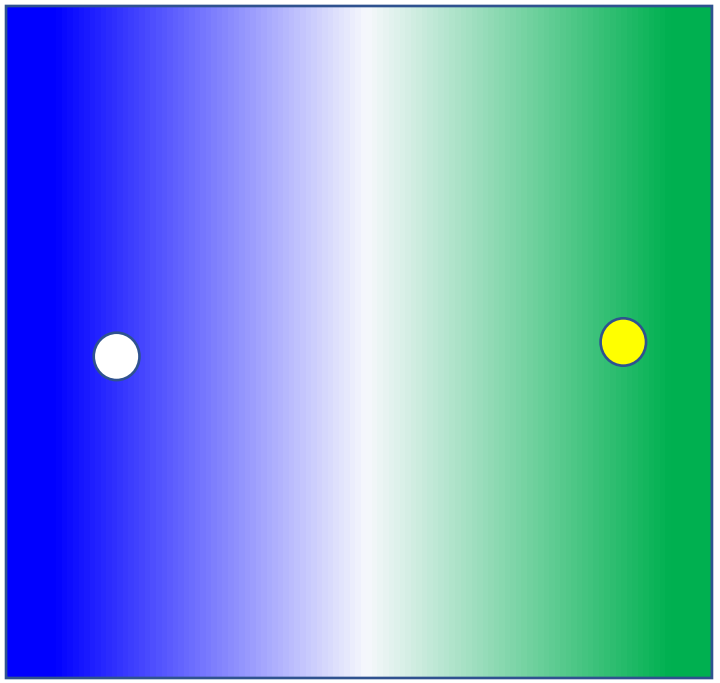


Even more for $P(+|x_+) \uparrow$

$P(+|x_+)=0.75$ $P(-|x_+)=0.25$
$P(-|x_-)=0.75$ $P(+|x_-)=0.25$

Logistic Regression

Max for $P(-|x_-) \uparrow$



Max for $P(+|x_+) \uparrow$

$P(+|x_+)=0.85$ $P(-|x_+)=0.15$

$P(-|x_-)=0.85$ $P(+|x_-)=0.15$

Logistic Regression

(I) Iterative Process: We can change the line that discriminate

(II) Optimization = Discriminate classes the most

  (I) Maximizing the $P(+|x_+) + P(-|x_-)$

Logistic Regression

(I) What is that line?
(II) What if we have more than two inputs?

Logistic Regression

(I) Iterative Process: We can change the line that discriminate

(II) Optimization

(I) Max ($\prod_{x \in \{+\}} P(+|x_+) + \prod_{x \in \{-\}} P(-|x_-)$)

Logistic Regression

(I) Iterative Process: We can change the line that discriminate

(II) Optimization

(I) Max $(\sum_{x \in \{+\}} \text{Log } P(+|x_+) + \sum_{x \in \{-\}} \text{Log } P(-|x_-))$

Logistic Regression

(I) Iterative Process: We can change the line that discriminate

(II) Optimization

(I) Min $-(\sum_{x\in\{+\}} \text{Log } P(+|x_+) + \sum_{x\in\{-\}} \text{Log } P(-|x_-))$

Logistic Regression
c= {-, +} → y={0, 1}

(I) Iterative Process: We can change the line that discriminate

(II) Optimization

(I) Min $-\sum_{(x,y)\in \mathbf{D}} [(y)\text{Log } P(y|x_y) + (1-y) \text{ Log } P(y|x_y)]$

Logistic Regression

c= {-, +} → y={0, 1}

(I) Iterative Process: We can change the line that discriminate

(II) Optimization

(I) Min $-\sum_{(x,y)\in \mathbf{D}} \text{Log } P(y|x_y)]$

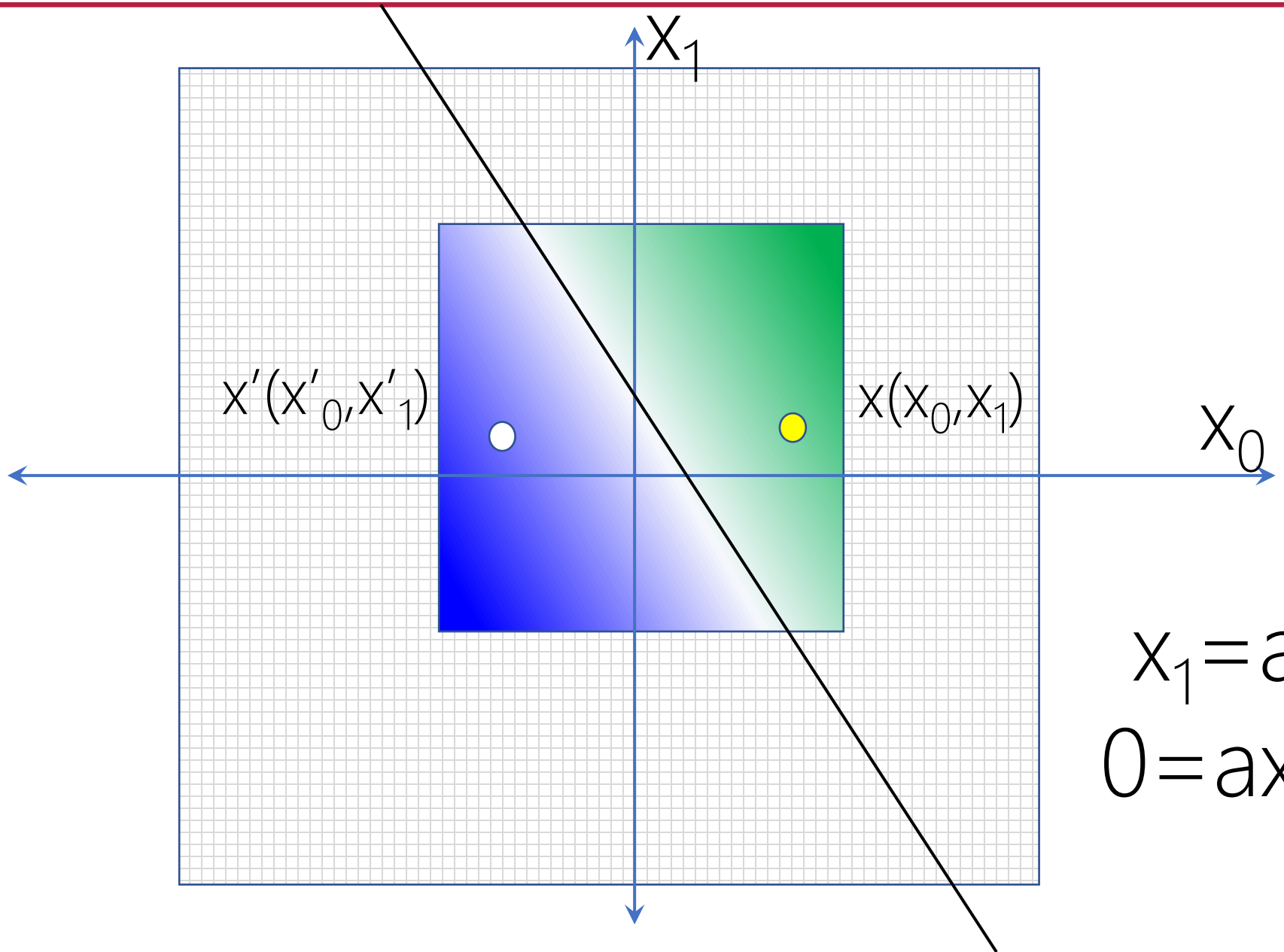
Logistic Regression
$c = \{-, +\} \rightarrow y = \{0, 1\}$

(I) Iterative Process: We can change the line that discriminate

(II) Optimization

(I) Min $-\sum_{(x,y)\in D} [(y) \text{Log } P(+|x_+) + (1-y) \text{ Log } P(-|x_-)]$

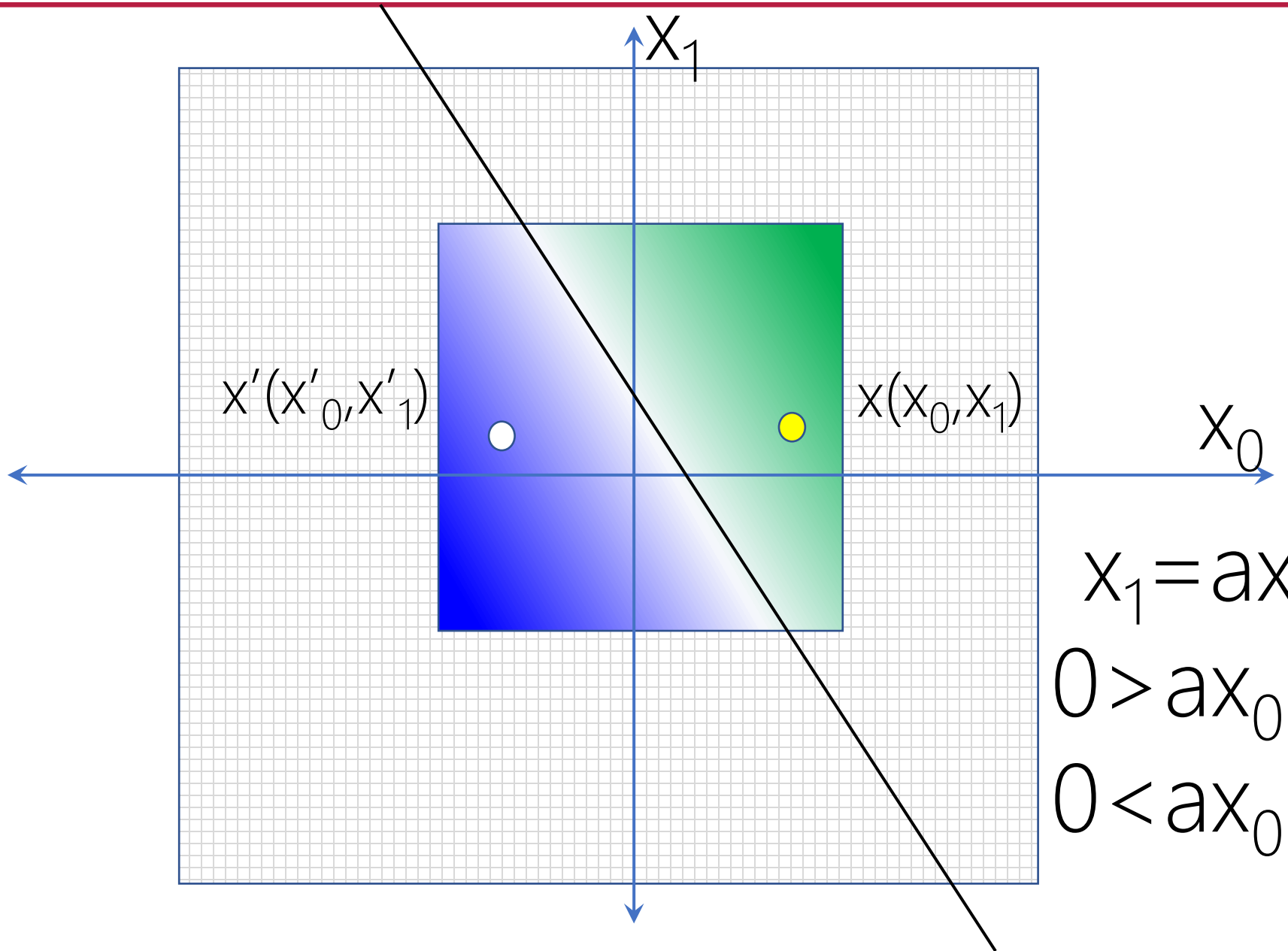
Logistic Regression



$$x_1 = ax_0 + b$$

$$0 = ax_0 - x_1 - b$$

Logistic Regression

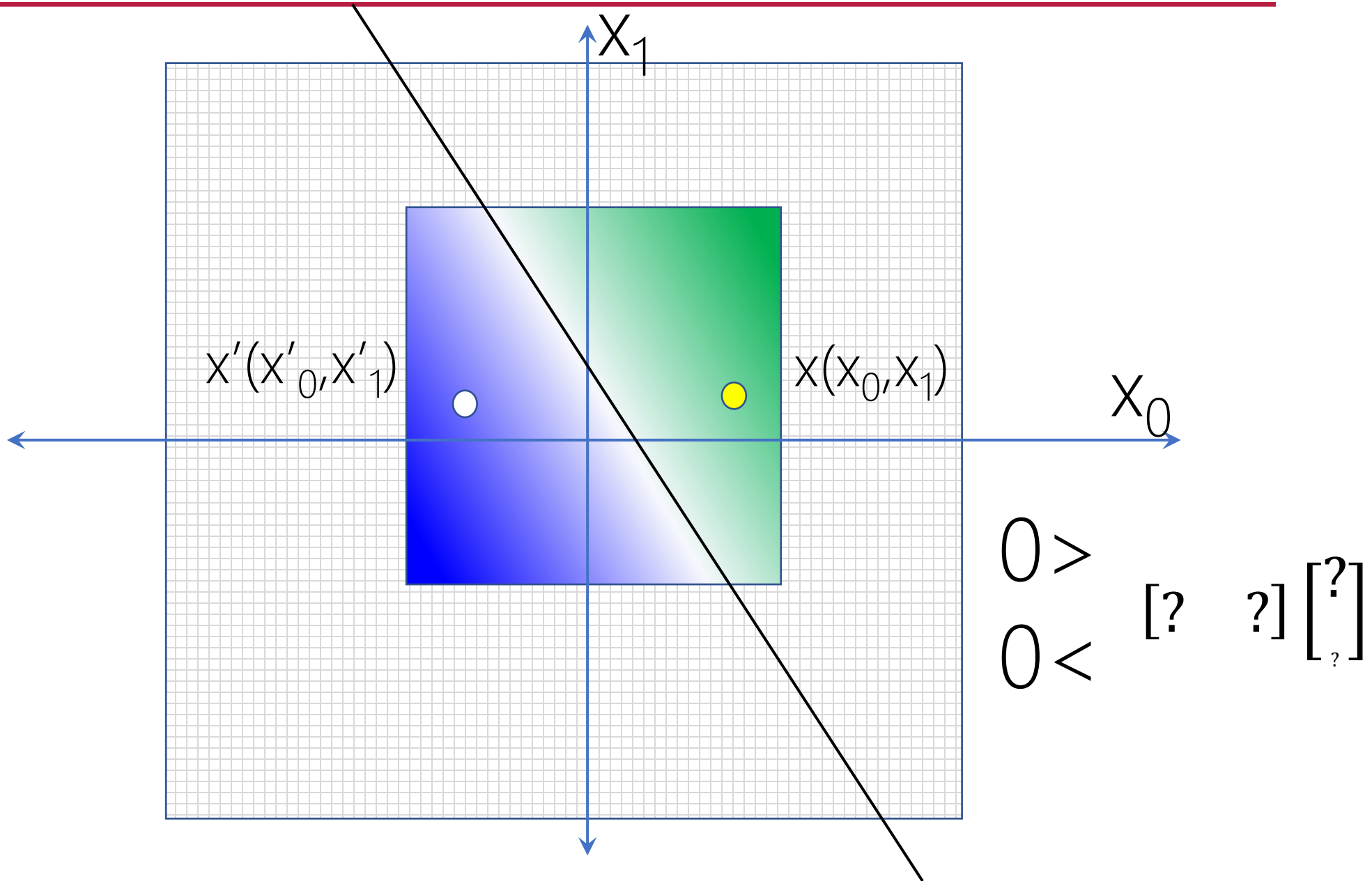


$$x_1 = ax_0 + b$$

$$0 > ax_0 - x_1 - b$$

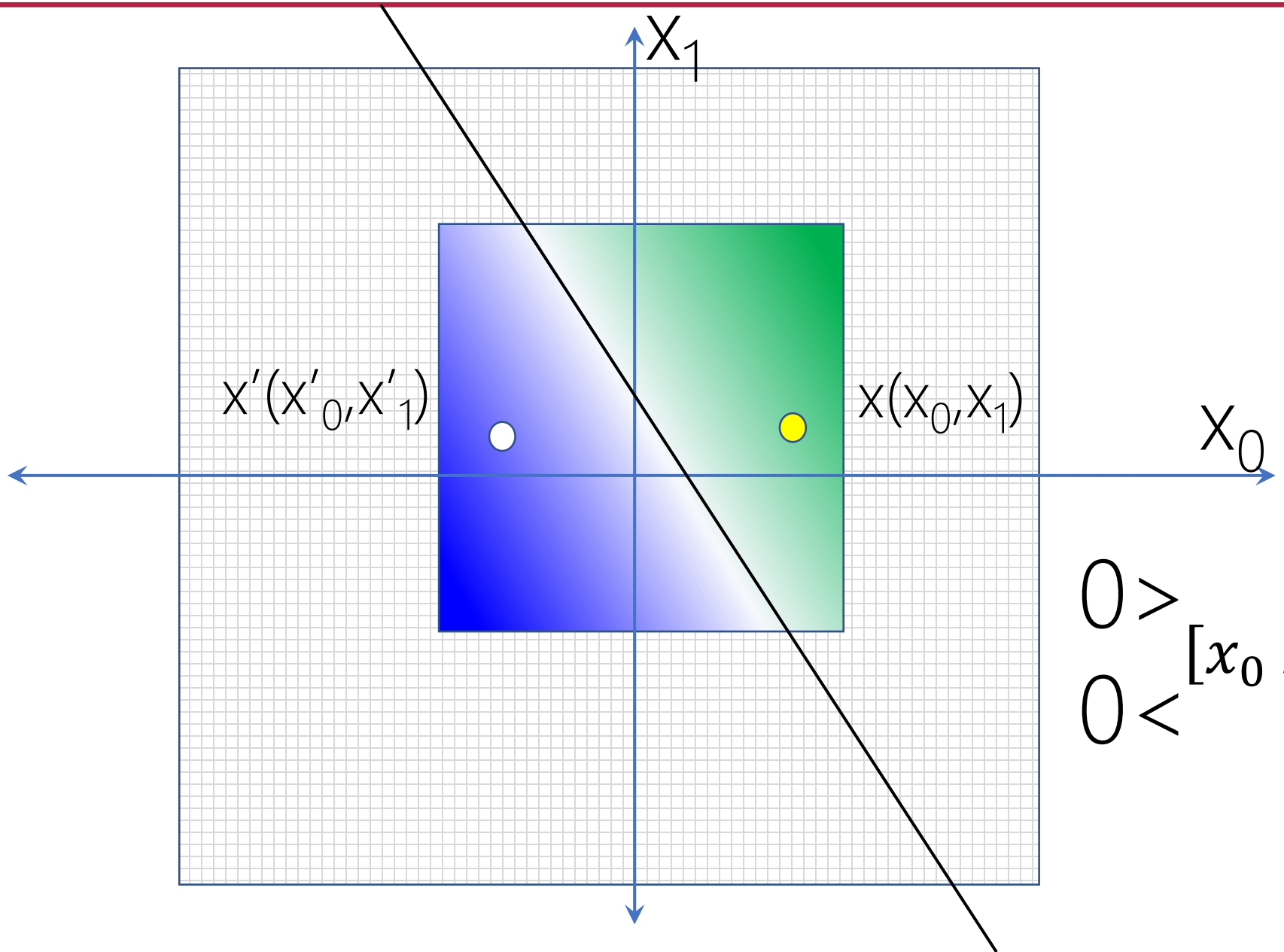
$$0 < ax_0 - x_1 - b$$

Logistic Regression



$$0 > \quad [? \quad ?]\begin{bmatrix}?\\?\end{bmatrix}$$
$$0 <$$

Logistic Regression



$$0 > [x_0 \; x_1 \; 1] \begin{bmatrix} a \\ -1 \\ b \end{bmatrix}$$

$$0 <$$

Logistic Regression

$$0 > \quad [x_0\ x_1\ \ldots x_d 1] \begin{bmatrix} w_1 \\ w_2 \\ \ldots \\ w_{d+1} \end{bmatrix} = [?]$$

$$0 <$$

d = 1 → Line in 2-dimension

d = 2 → Plane in 3 dimension

d = n → Hyperplane in (n+1) dimension

Logistic Regression

$$0 > f(X) \longrightarrow -\infty$$

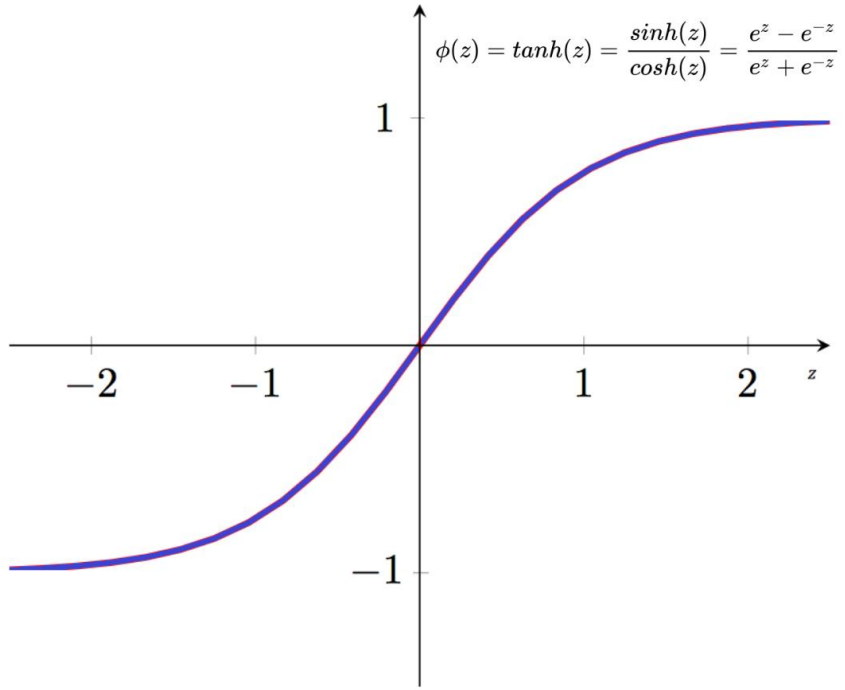
$$0 < f(X) \longrightarrow +\infty$$

d = 1 → Line in 2-dimension
d = 2 → Plane in 3 dimension
d = n → Hyperplane in (n+1) dimension

Logistic Regression: Squish by Tanh

$$0 > \quad f(X) \longrightarrow -1$$

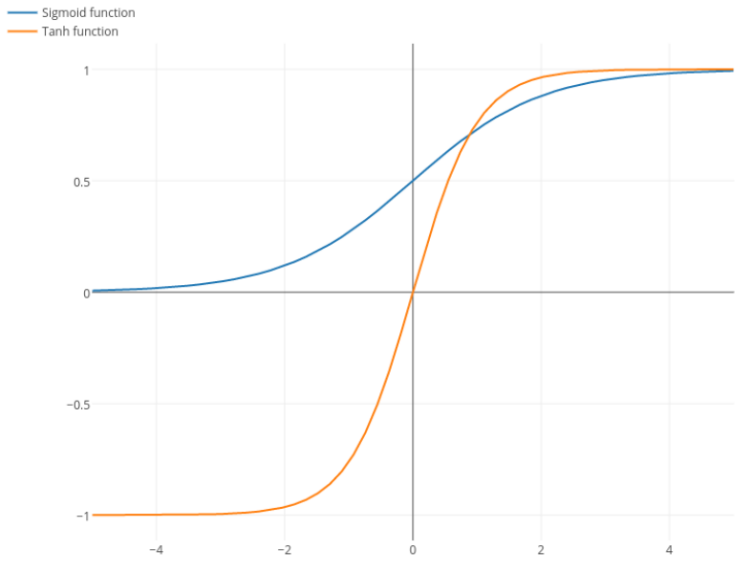
$$0 < \quad f(X) \longrightarrow +1$$



Logistic Regression: Squish by Sigmoid

$$0 > \quad f(X) \longrightarrow 0$$

$$0 < \quad f(X) \longrightarrow +1$$



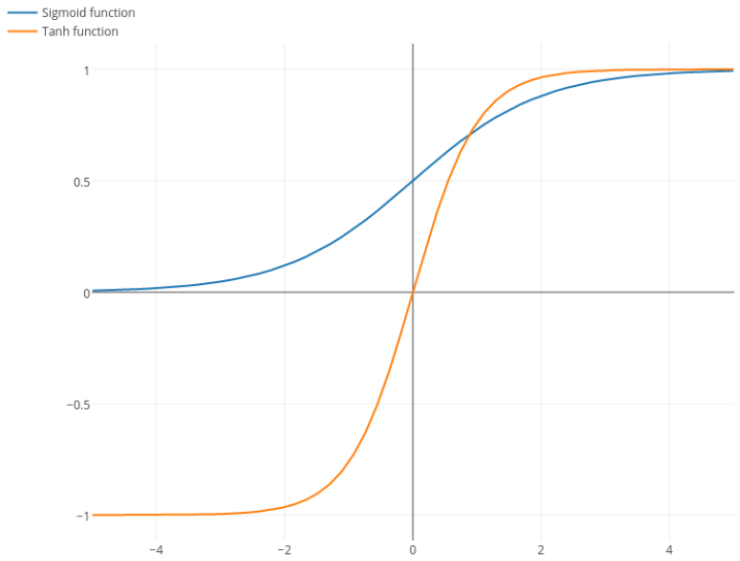
$$S(f(X)) = \frac{1}{1+e^{-f(X)}}$$

Becomes very similar to probability values*!*

Logistic Regression: Squish

$$0 > \quad f(X) \longrightarrow 0$$

$$0 < \quad f(X) \longrightarrow +1$$



$$S(f(X)) = \frac{1}{1+e^{-f(X)}}$$

Becomes very similar to probability values!

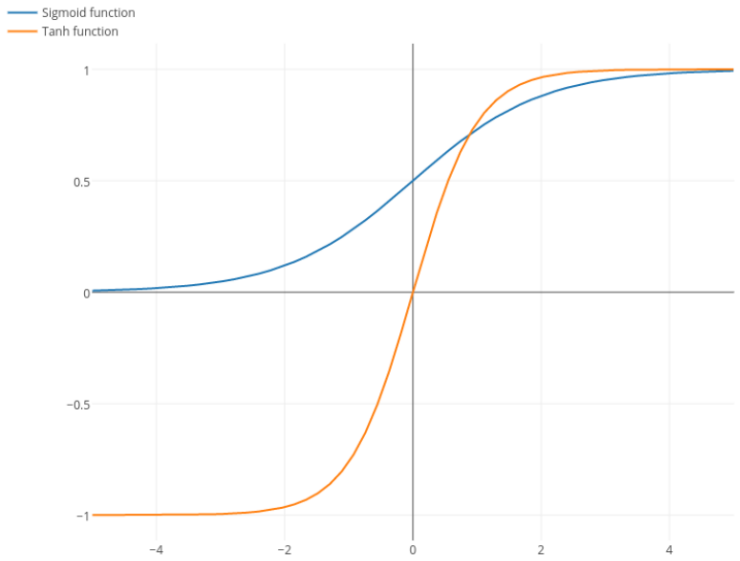
But only for positive class (+)

y=1 → P(y|x) = P(+|x) = Sigmoid(x)

Logistic Regression: Squish

$0 > \quad f(X) \longrightarrow 0$

$0 < \quad f(X) \longrightarrow +1$



$$S(f(X)) = \frac{1}{1+e^{-f(X)}}$$

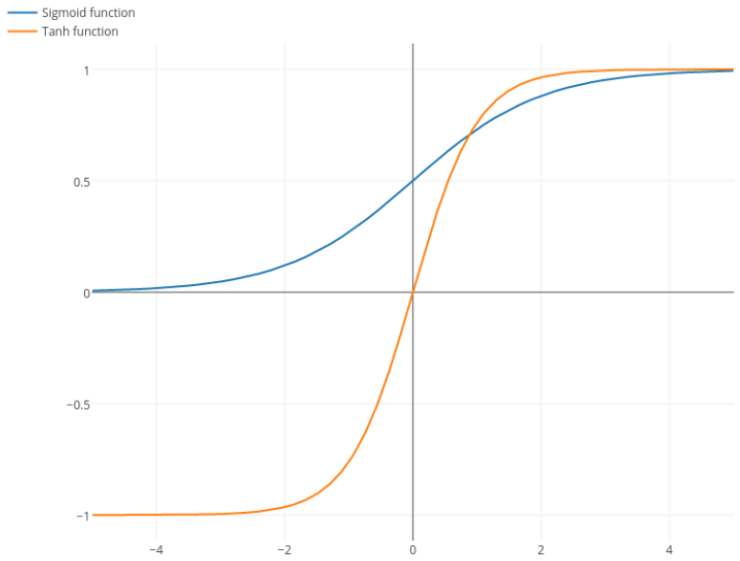
Becomes very similar to probability values!

For negative class (-)?

Logistic Regression: Squish

$0 > \quad f(X) \longrightarrow 0$

$0 < \quad f(X) \longrightarrow +1$



$$S(f(X)) = \frac{1}{1+e^{-f(X)}}$$

Becomes very similar to probability values!

For negative class (-)

P(+|x) + P(-|x) = 1 → P(-|x) = 1 - P(+|x)

Logistic Regression
c= {-, +} → y={0, 1}

(I) Iterative Process: We can change the *f* that discriminate

(II) Optimization

(I) Min $-\sum_{(x,y)\in \mathbf{D}} \text{Log } P(y|x_y)] \rightarrow \begin{cases} y = 1: P(+|x_+) = Sigmoid(f(x_+)) \\ y = 0: P(-|x_-) = 1 - Sigmoid(f(x_+)) \end{cases}$

Logistic Regression

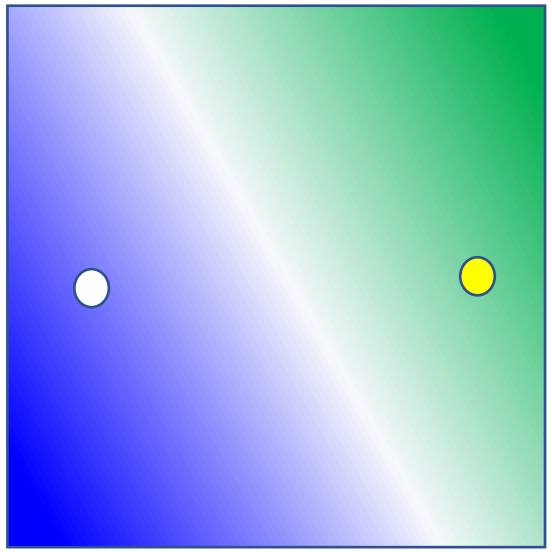
Optimization: Min $-\sum_{(x,y)\in D}$ Log P(y|x_y)] → $\begin{cases} y=1:\ P(+|x_+) = Sigmoid(f(x_+)) \\ y=0: P(-|x_-) = 1 - Sigmoid(f(x_+)) \end{cases}$

- Derivatives per function weights (Gradients)
  - Update on each input data
  - Update on batches of input data (Stochastic Gradient Descend)
  - Update on multiple rounds (epoch) of ALL data

Logistic Regression

Optimization: Min $-\sum_{(x,y)\in D}$ Log P(y|x_y)] → $\begin{cases} y=1:\ P(+|x_+) = Sigmoid(f(x_+)) \\ y=0: P(-|x_-) = 1 - Sigmoid(f(x_+)) \end{cases}$

- Function *f* is linear function of weights

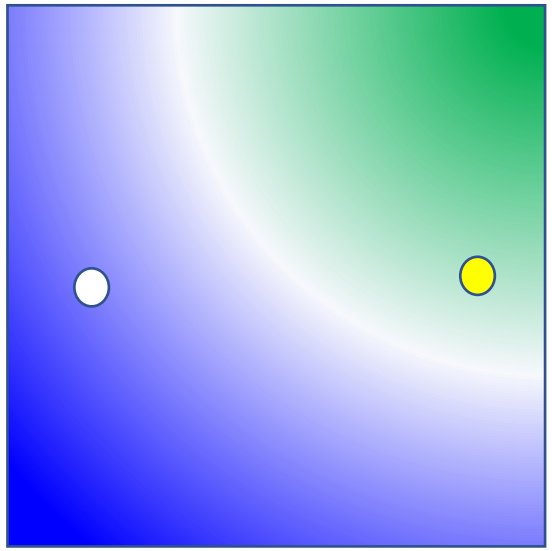


$$[x_0\ x_1\ \ldots x_d 1]\begin{bmatrix} w_1 \\ w_2 \\ \ldots \\ w_{d+1} \end{bmatrix} = [?]$$

Logistic Regression

Optimization: Min $-\sum_{(x,y)\in \mathbf{D}} \text{Log } P(y|x_y)] \rightarrow \begin{cases} y = 1: P(+|x_+) = Sigmoid(f(x_+)) \\ y = 0: P(-|x_-) = 1 - Sigmoid(f(x_+)) \end{cases}$

- Can we have *f* as a non-linear function of weights?

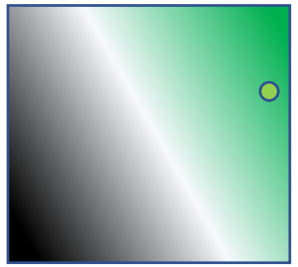


Multiclass with Logistic Regression

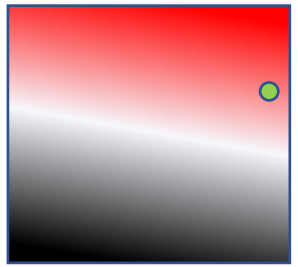
Multi-target (Multi-label)

Optimization: Min $-\sum_{(x,y)\in \mathbf{D}} \text{Log } P(y|x_y)]$

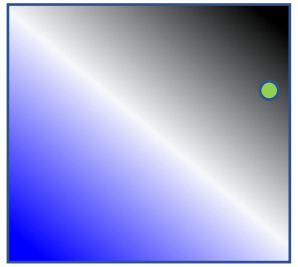
- Data point can be member of more than one class



green class vs. else: $[x_0\ x_1\ \ldots x_d 1]\begin{bmatrix} w_1 \\ w_2 \\ \ldots \\ w_{d+1} \end{bmatrix} = [?]$



red class vs. else: $[x_0\ x_1\ \ldots x_d 1]\begin{bmatrix} w'_1 \\ w'_2 \\ \ldots \\ w'_{d+1} \end{bmatrix} = [?]$



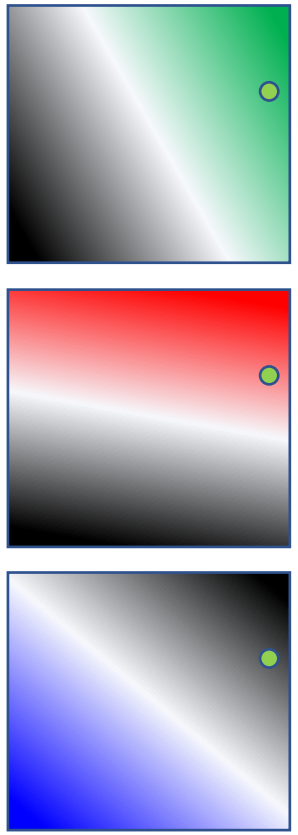
blue class vs. else: $[x_0\ x_1\ \ldots x_d 1]\begin{bmatrix} w''_1 \\ w''_2 \\ \ldots \\ w''_{d+1} \end{bmatrix} = [?]$

Multiple Binary Classification

Multi-target (Multi-label)

Optimization: Min $-\sum_{(x,y)\in \mathbf{D}} \text{Log } P(y|x_y)]$

- Data point can be member of more than one class



$$[x_0\ x_1\ \ldots x_d 1]\begin{bmatrix} w_1 & w'_1 & w''_1 \\ w_2 & w'_2 & w''_2 \\ \ldots & \ldots & \ldots \\ w_{d+1} & w'_{d+1} & w''_{d+1} \end{bmatrix} = [?, ?, ?] = [1,0,1]$$

Multinomial Logistic Regression

Logistic Regression → Softmax Regression

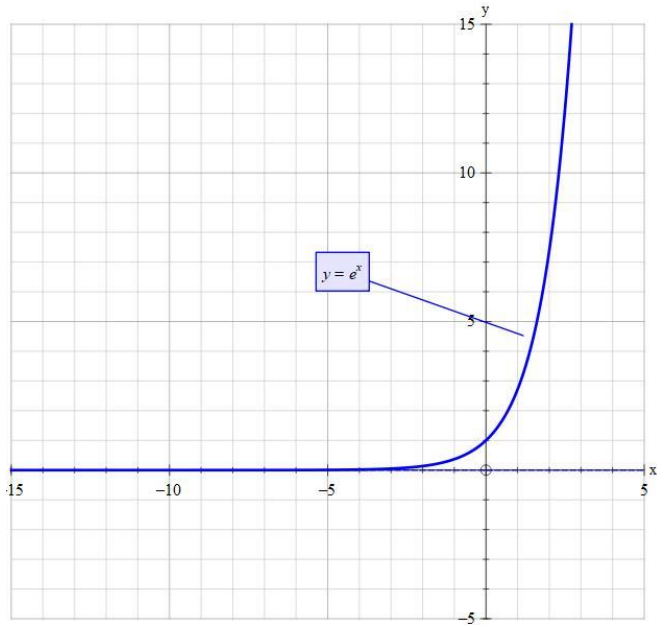
Softmax Regression

$$softmax(\boldsymbol{z} = [z_1, z_2, \dots, z_d]) = \frac{1}{\sum_{i=1}^{d} e^{z_i}} [e^{z_1}, e^{z_2}, \dots, e^{z_d}]$$

$$0 \leq \frac{e^{z_j}}{\sum_{i=1}^{d} e^{z_i}} \leq 1$$

$$\sum_{j=1}^{d} \frac{e^{z_j}}{\sum_{i=1}^{d} e^{z_i}} = 1$$

```
In[2]: import torch
In[3]: sample_vector = torch.rand(*(1,10))
In[4]: sample_vector
Out[4]:
tensor([[0.6084, 0.8744, 0.8193, 0.8784, 0.2906, 0.7169, 0.0126, 0.8783, 0.3822,
         0.3883]])
In[5]: torch.softmax(sample_vector, dim=1)
Out[5]:
tensor([[0.0985, 0.1285, 0.1216, 0.1290, 0.0717, 0.1098, 0.0543, 0.1290, 0.0786,
         0.0790]])
In[6]: torch.sum(torch.softmax(sample_vector, dim=1))
Out[6]: tensor(1.0000)
```



Multi-target (Multi-label)

Optimization: Min $-\sum_{(x,y)\in D} \text{Log } P(y|x_y)]$

$$[x_0\ x_1\ \ldots x_d 1]\begin{bmatrix} w_1 & w'_1 & w''_1 \\ w_2 & w'_2 & w''_2 \\ \ldots & \ldots & \ldots \\ w_{d+1} & w'_{d+1} & w''_{d+1} \end{bmatrix} = [1.5, -1.4, 10] \Rightarrow Softmax$$

$$\Rightarrow [2.0342e-04, 1.1193e-05, 9.9979e-01]$$

```
In[12]: torch.softmax(torch.as_tensor([1.5, -1.4, 10.0]).view(-1), dim=0)
Out[12]: tensor([2.0342e-04, 1.1193e-05, 9.9979e-01])
```

$$[x_0\ x_1\ \ldots x_d 1]\begin{bmatrix} w_1 & w'_1 & w''_1 \\ w_2 & w'_2 & w''_2 \\ \ldots & \ldots & \ldots \\ w_{d+1} & w'_{d+1} & w''_{d+1} \end{bmatrix} = [1.5, -1.4, 10]$$

$$\Rightarrow sigmoid = [0.8176, 0.1978, 1.0000]$$
$$\Rightarrow softmax \Rightarrow [0.3652, 0.1965, 0.4383]$$

```
In[14]: torch.sigmoid(torch.as_tensor([1.5, -1.4, 10.0]).view(-1))
Out[14]: tensor([0.8176, 0.1978, 1.0000])
In[15]: torch.softmax(torch.sigmoid(torch.as_tensor([1.5, -1.4, 10.0]).view(-1)), dim=0)
Out[15]: tensor([0.3652, 0.1965, 0.4383])
```

Multi-target (Multi-label)

Optimization: Min $-\sum_{(x,y)\in \mathbf{D}} \text{Log } P(y|x_y)]$

Multitarget: $(x, [1, 0, 1]) \overset{?}{\leftrightarrow} (x, [0.8176, 0.1978, 1.0])$

$(x, \{0\}) \overset{?}{\leftrightarrow} (x, [0.8176, \blacksquare, \blacksquare])$

$(x, \{2\}) \overset{?}{\leftrightarrow} (x, [\blacksquare, \blacksquare, 1.0])$

$-$ (Log $P(c=0|x_{c=0})]$ + Log $P(c=2|x_{c=2})]$)

$-$ (Log 0.8176 + Log 1.0)

Multi-target (Multi-label)

Optimization: Min $-\sum_{(x,y)\in D}$ Log P(y|x_y)]

Multitarget: $(x, [1, 0, 1]) \overset{?}{\leftrightarrow} (x, [0.8176, 0.1978, 1.0])$

$$-\left([1, 0, 1] \log \begin{bmatrix} 0.8176 \\ 0.1978 \\ 1.0 \end{bmatrix}\right) = ?$$

Multi-target (Multi-label)

Optimization: Min $-\sum_{(x,y)\in D} \text{Log } P(y|x_y)]$

Multitarget: $(x, [1, 0, 1]) \overset{?}{\leftrightarrow} (x, [0.8176, 0.1978, 1.0])$

$$-\left([1,0,1]\log\begin{bmatrix}0\\1\\0\end{bmatrix}\right) = +\infty$$

$$-\left([1,0,1]\log\begin{bmatrix}0.8176\\0.1978\\1.0\end{bmatrix}\right) = ?$$

$$-\left([1,0,1]\log\begin{bmatrix}1\\0\\1\end{bmatrix}\right) = 0$$

Multiclass

Optimization: Min $-\sum_{(x,y)\in \mathbf{D}}$ Log P(y|x_y)]

Multiclass:$(x, [0, 1, 0]) \overset{?}{\leftrightarrow} (x, [0.3652, 0.1965, 0.4383])$

$(x, \{1\}) \overset{?}{\leftrightarrow} (x, [\blacksquare, 0.1965, \blacksquare])$

– Log P(c=1|$x_{c=1}$)
– (Log 0.1965)

Multiclass

Optimization: Min $-\sum_{(x,y)\in D}$ Log P($y|x_y$)]

Multiclass: $(x, [0,1,0]) \overset{?}{\leftrightarrow} (x, [0.3652, 0.1965, 0.4383])$

$$-\left([0,1,0]\log\begin{bmatrix}1\\0\\0\end{bmatrix}\right) = +\infty$$

$$-\left([0,1,0]\log\begin{bmatrix}0.3652\\0.1965\\0.4383\end{bmatrix}\right) = ?$$

$$-\left([0,1,0]\log\begin{bmatrix}0\\1\\0\end{bmatrix}\right) = 0$$

Evaluation

Transparent (Interpretable) Model