Regressão Logística

Prof. Danilo Silva

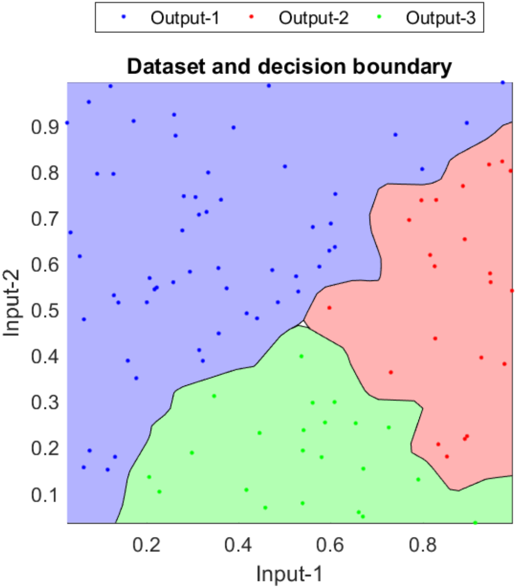
EEL7514/EEL7513 - Tópico Avançado em Processamento de Sinais:
Introdução ao Aprendizado de Máquina

EEL / CTC / UFSC

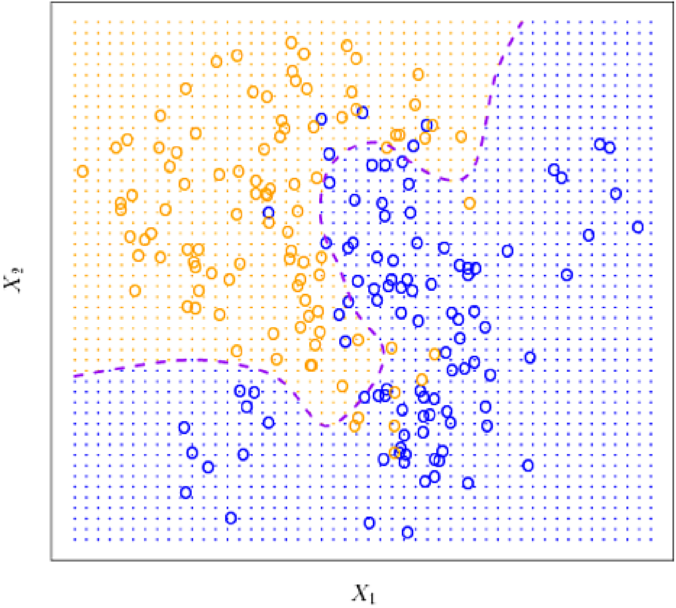
Classificação

- Problema de classificação com K classes:
  - $\mathbf{x} = (x_1, \ldots, x_n)^T \in \mathbb{R}^n$ é o vetor de atributos
  - $y \in \mathcal{Y} = \{1, 2, \ldots, K\}$ é o rótulo que indica a classe a qual $\mathbf{x}$ pertence
  - Um classificador é uma função $g : \mathbb{R}^n \to \mathcal{Y} = \{1, 2, \ldots, K\}$
  - O desempenho de um classificador depende do custo $L(k, y)$ de se classificar $g(\mathbf{x}) = k$ quando a classe correta é y
- Dado um conjunto de treinamento, desejamos encontrar um classificador que obtenha bom desempenho em novos dados

Exemplo


Output-1
Output-2
Output-3
Dataset and decision boundary
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
Input-2
0.2
0.4
0.6
0.8
Input-1

Exemplo



Codificação de Rótulos

- De maneira geral, definimos eventos correspondentes às K classes:

$$\mathcal{C}_k = \{\mathbf{x} \text{ pertence à classe } k\}, \quad k = 1, \dots, K$$

  - No entanto, o mapeamento específico em uma v.a. y é arbitrário

- Classificação binária ($K = 2$):
  - $\mathcal{C}_1 = \{y = 1\}, \mathcal{C}_2 = \{y = 2\}$
  - $\mathcal{C}_0 = \{y = 0\}$ (classe negativa), $\mathcal{C}_1 = \{y = 1\}$ (classe positiva)
  - $\mathcal{C}_0 = \{y = -1\}$ (classe negativa), $\mathcal{C}_1 = \{y = +1\}$ (classe positiva)

- Classificação multi-classe ($K > 2$):
  - $\mathcal{C}_k = \{y = k\}, k = 1, \dots, K$
  - Codificação 1-de-K / função indicadora / "*One Hot Encoder*":

$$\begin{aligned}
\mathcal{C}_1 &= \{y = (1, 0, 0, 0, \dots, 0)\} \\
\mathcal{C}_2 &= \{y = (0, 1, 0, 0, \dots, 0)\} \\
\mathcal{C}_3 &= \{y = (0, 0, 1, 0, \dots, 0)\} \\
&\vdots
\end{aligned}$$

Funções Discriminantes

- Muitos métodos de classificação (exceto classificadores hierárquicos) são baseados em funções discriminantes (também chamados de preditores ou *scores* de confiança):

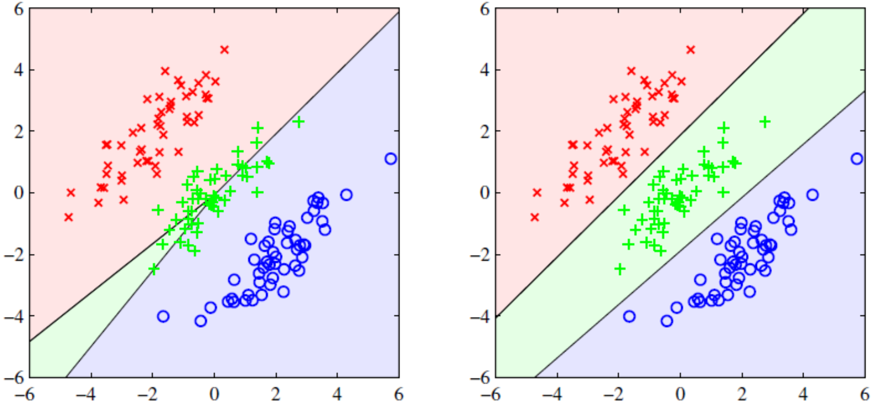
$$f_k(\mathbf{x}) \in \mathbb{R}$$

- Decide-se pela classe $\mathcal{C}_k$ que maximiza o discriminante:

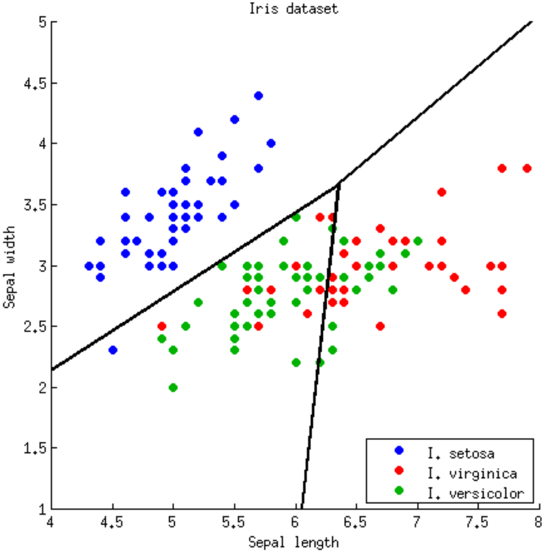
$$g(\mathbf{x}) = k \iff f_k(\mathbf{x}) = \max_{k' \in \{1,\ldots,K\}} f_{k'}(\mathbf{x})$$

- Assim, o problema de classificação é transformado em K problemas de regressão
- Discriminante linear: $f_k(\mathbf{x}) = \mathbf{w}_k^T \mathbf{x}$
  - Nesse caso, o classificador é dito ser um classificador linear

Exemplo



Exemplo



Iris dataset
Sepal width
Sepal length
5
4.5
4
3.5
3
2.5
2
1.5
1
4
4.5
5
5.5
6
6.5
7
7.5
8
I. setosa
I. virginica
I. versicolor

Classificação Binária

- Se $K = 2$ (classes $\mathcal{C}_0$ e $\mathcal{C}_1$), é suficiente usar um único discriminante:

$$g(\mathbf{x}) = 1 \text{ (decide-se por } \mathcal{C}_1) \iff f(\mathbf{x}) = f_1(\mathbf{x}) - f_0(\mathbf{x}) \geq 0$$

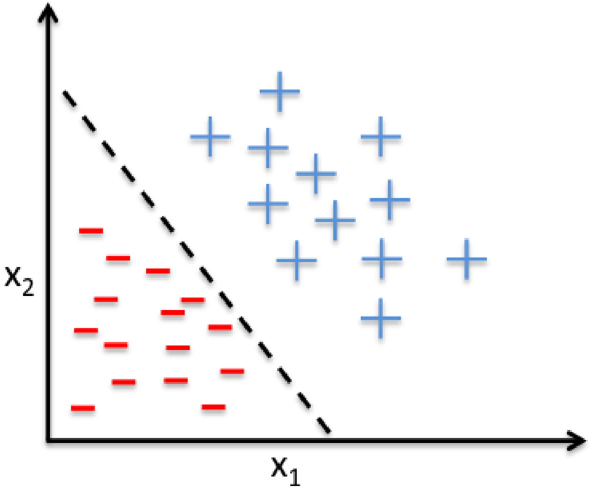
  (Consequentemente decide-se por $\mathcal{C}_0$ se $f(\mathbf{x}) < 0$)

- No caso de um classificador linear, temos

$$g(\mathbf{x}) = 1 \iff \mathbf{w}^T\mathbf{x} \geq 0$$

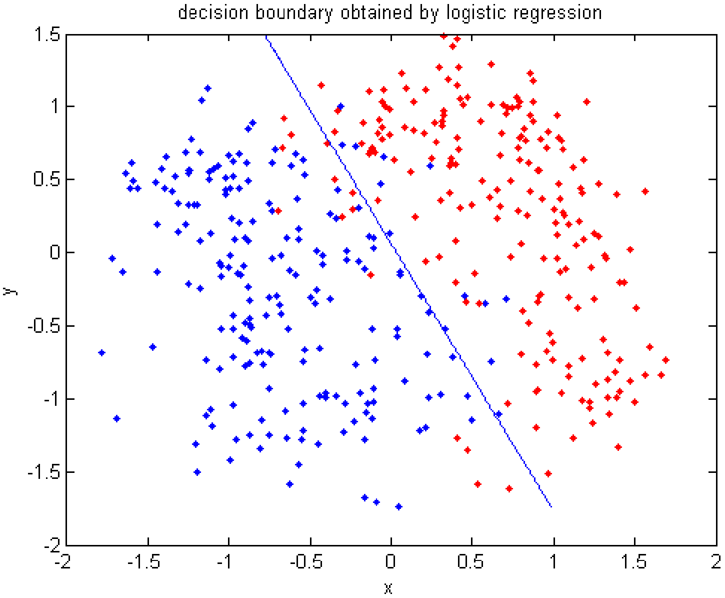
- Geometricamente, o vetor $\mathbf{w}$ define a direção de um hiperplano (perpendicular a $\mathbf{w}$) que passa pela origem
- Uma amostra $\mathbf{x}$ é classificada como positiva (classe $\mathcal{C}_1$) se estiver no semi-espaço do lado positivo do hiperplano (no sentido da projeção em $\mathbf{w}$)

Exemplo

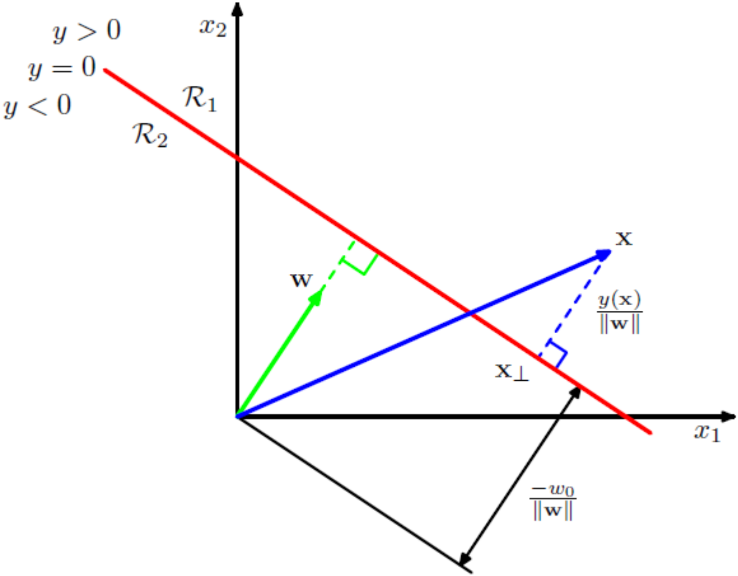


Example of a linear decision boundary for binary classification.

Exemplo


decision boundary obtained by logistic regression
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-2
-1.5
-1
-0.5
0
0.5
1
1.5
2
x
y

Exemplo



$y > 0$
$y = 0$
$y < 0$
x_2
$\mathcal{R}_1$
$\mathcal{R}_2$
$\mathbf{x}$
$\mathbf{w}$
$\frac{y(\mathbf{x})}{\|\mathbf{w}\|}$
$\mathbf{x}_\perp$
x_1
$\frac{-w_0}{\|\mathbf{w}\|}$

Classificação Binária via Regressão Linear

- Uma forma simples de determinar $\mathbf{w}$ é usando regressão linear. Desejamos ajustar um modelo

$$\hat{y} = \mathbf{w}^T \mathbf{x}$$

a partir de exemplos de treinamento rotulados como $y \in \{-1, +1\}$

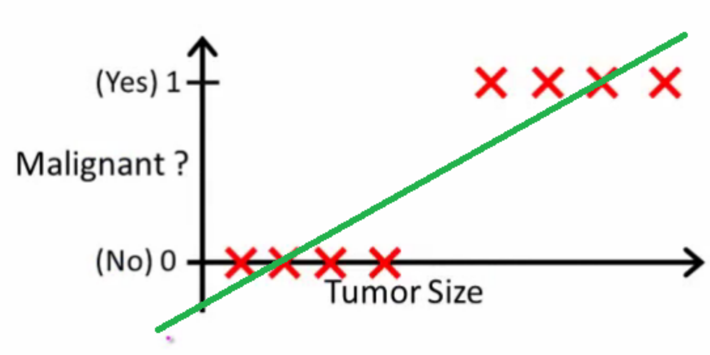
- Solução via mínimos quadrados:

$$J(\mathbf{w}) = \frac{1}{2m} \sum_{i=1}^{m} (\mathbf{w}^T \mathbf{x}^{(i)} - y^{(i)})^2 = \frac{1}{2m} \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2$$

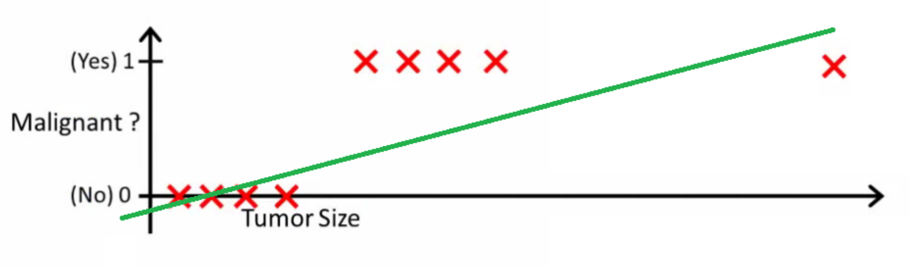
$$\mathbf{w} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}$$

- A classificação em si é dada por $g(\mathbf{x}) = \operatorname{sgn}(\hat{y}) = \operatorname{sgn}(\mathbf{w}^T \mathbf{x})$

Exemplo



Exemplo



Problema: Sensibilidade a Outliers

- Um problema desta solução é que o uso do erro quadrático

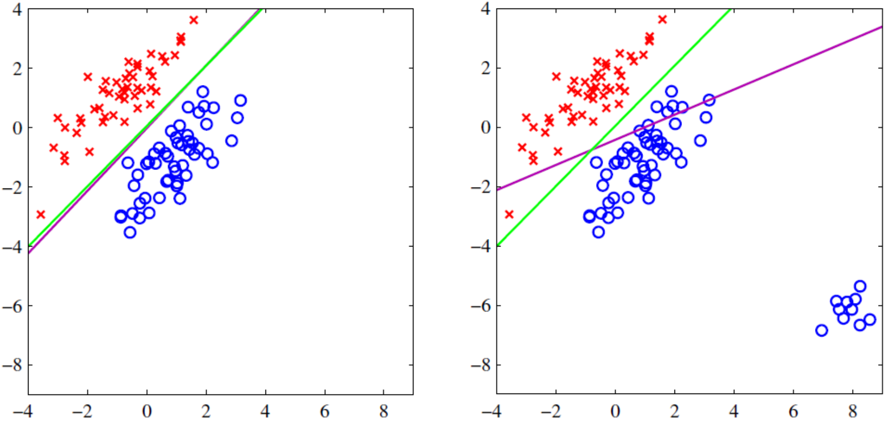
$$L(\hat{y}, y) = \frac{1}{2}(\hat{y} - y)^2$$

  como função perda penaliza predições que estão “certas demais”

- Por exemplo, assumindo a classe correta $y = 1$, um score de $\hat{y} = 100$ (alta confiança) tem um custo $L(\hat{y}, y) = 4900.05$ muito mais elevado que $\hat{y} = 0$ (baixa confiança), cujo custo é $L(\hat{y}, y) = 0.5$

- Consequentemente, valores altos de $\hat{y} = \mathbf{w}^T\mathbf{x}$ influenciam excessivamente o modelo

Exemplo



Regressão Logística

- Uma solução para esse problema é o modelo de regressão logística

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

com rótulos codificados como $y \in \{0, 1\}$, onde

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

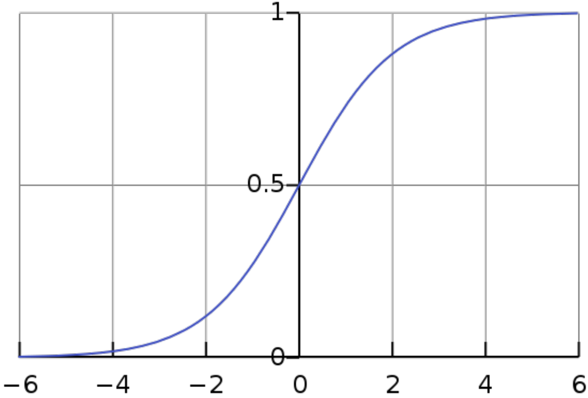
é a função sigmóide logística padrão

- Note que

$$\lim_{z \to -\infty} \sigma(z) = 0, \qquad \sigma(0) = \frac{1}{2}, \qquad \lim_{z \to \infty} \sigma(z) = 1$$

- Decide-se por $\mathcal{C}_1 \iff \hat{y} = \sigma(\mathbf{w}^T \mathbf{x}) > 1/2 \iff \mathbf{w}^T \mathbf{x} > 0$

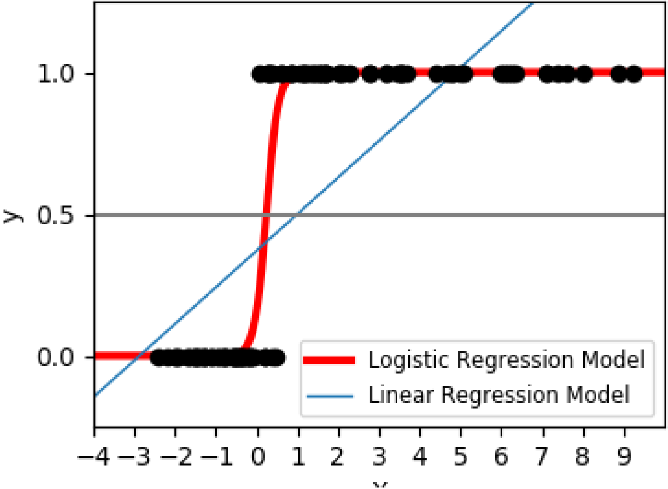
Função Logística



Propriedades:

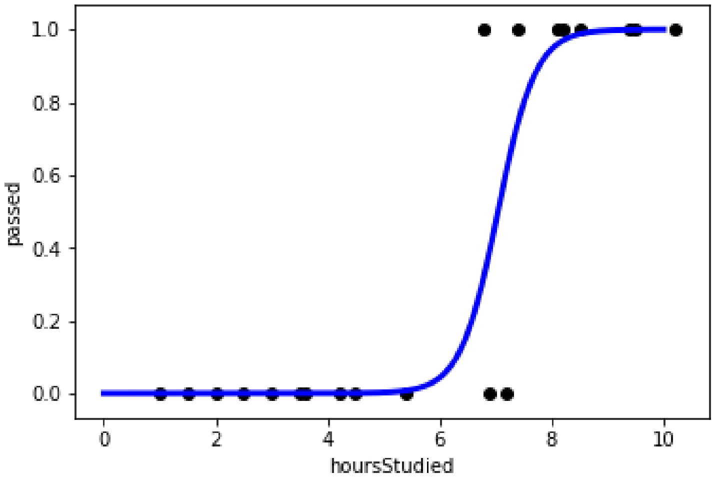
$$\sigma(-x) = 1 - \sigma(x)$$
$$\sigma'(x) = \sigma(x)(1 - \sigma(x))$$

Exemplo



1.0
0.5
0.0
y
Logistic Regression Model
Linear Regression Model
−4 −3 −2 −1 0 1 2 3 4 5 6 7 8 9

Exemplo



Função Custo

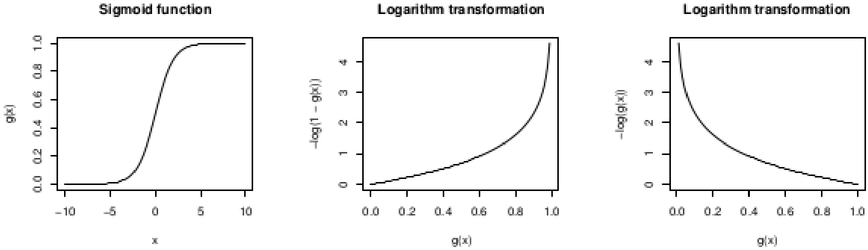
- Mesmo com o modelo de regressão logística, o uso do erro quadrático ainda é problemático:
  - Penaliza pouco um score de confiança $z = \mathbf{w}^T\mathbf{x}$ muito errado ($L(\hat{y}, y) < 1/2$)
  - Resulta em uma função custo $J(\mathbf{w})$ não-convexa
- É usual adotar como função perda a entropia cruzada:

$$L(\hat{y}, y) = -y \log \hat{y} - (1 - y) \log(1 - \hat{y})$$

  - Note que $L(0, 1) = L(1, 0) = \infty$, enquanto $L(0, 0) = L(1, 1) = 0$
  - Resulta em uma função custo $J(\mathbf{w})$ convexa

$$J(\mathbf{w}) = -\frac{1}{m} \sum_{i=1}^{m} y^{(i)} \log \hat{y}^{(i)} + (1 - y^{(i)}) \log(1 - \hat{y}^{(i)})$$

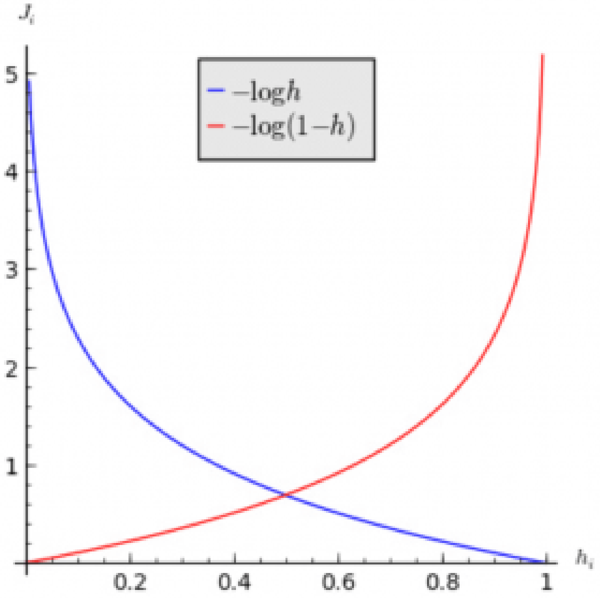
Função Custo: Exemplo



(a) *Sigmoid function.* **(b)** *Cost for* $y = 0$. **(c)** *Cost for* $y = 1$.

Figure B.1: *Logarithmic transformation of the sigmoid function.*

Função Custo: Exemplo



Otimização

- Função custo:

$$J(\mathbf{w}) = -\frac{1}{m}\sum_{i=1}^{m} y^{(i)} \log \hat{y}^{(i)} + (1 - y^{(i)}) \log(1 - \hat{y}^{(i)})$$
$$= -\frac{1}{m}\left(\mathbf{y}^T \log \hat{\mathbf{y}} + (1 - \mathbf{y})^T \log(1 - \hat{\mathbf{y}})\right)$$

onde $\hat{y}^{(i)} = \sigma(\mathbf{w}^T \mathbf{x}^{(i)})$ e $\hat{\mathbf{y}} = \sigma(\mathbf{X}\mathbf{w})$

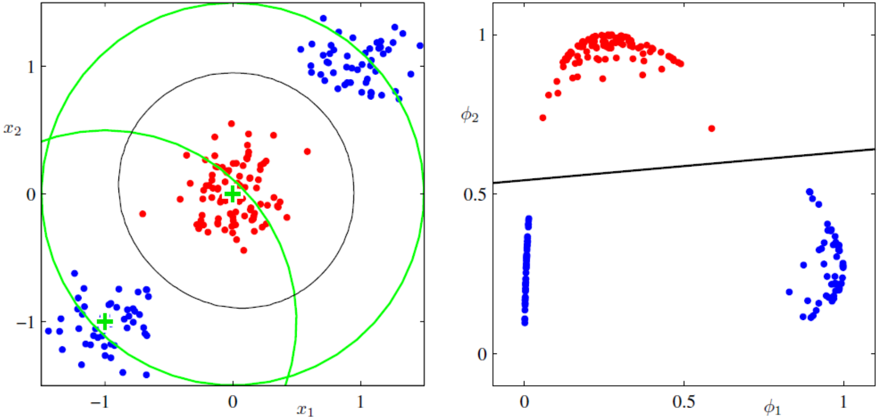
- Gradiente:

$$\nabla J(\mathbf{w}) = \frac{1}{m}\mathbf{X}^T(\hat{\mathbf{y}} - \mathbf{y}) = \frac{1}{m}\mathbf{X}^T(\sigma(\mathbf{X}\mathbf{w}) - \mathbf{y})$$

Extensão com Funções de Base

- Assim como no caso de regressão linear, o modelo básico de regressão logística pode ser estendido com funções de base, isto é, utilizando como atributos $x_j = \varphi_j(\mathbf{u})$, $j = 1, \ldots, n$, funções não-lineares dos atributos originais $\mathbf{u} = (u_1, \ldots, u_N)^T$
- O treinamento é idêntico a partir da matriz de projeto $\mathbf{X}$, entretanto a visualização a partir dos atributos originais ($\mathbf{u}$) será diferente
  - Em particular, permite uma separação não-linear entre as classes

Exemplo



Notação (Bishop): Atributos originais: x_1, x_2; Atributos transformados: ϕ_1, ϕ_2

Regularização

- Com o aumento no número de atributos, aumenta também a tendência a overfitting no conjunto de treinamento, tornando-se importante usar regularização para garantir uma boa generalização
- Regularização ℓ_2: $\Omega(\mathbf{w}) = \frac{1}{2m}\sum_{j=1}^{n} w_j^2 = \frac{1}{2m}\mathbf{w}^T\mathbf{L}\mathbf{w}$
- Função custo:

$$\begin{aligned} J(\mathbf{w}) &= J_{\text{train}}(\mathbf{w}) + \lambda\frac{1}{2m}\mathbf{w}^T\mathbf{L}\mathbf{w} \\ &= -\frac{1}{m}\mathbf{y}^T\log\hat{\mathbf{y}} + (1-\mathbf{y})^T\log(1-\hat{\mathbf{y}}) + \lambda\frac{1}{2m}\mathbf{w}^T\mathbf{L}\mathbf{w} \end{aligned}$$

- Gradiente:

$$\nabla J(\mathbf{w}) = \frac{1}{m}\mathbf{X}^T(\sigma(\mathbf{X}\mathbf{w}) - \mathbf{y}) + \lambda\frac{1}{m}\mathbf{L}\mathbf{w}$$

- λ é um hiperparâmetro a ser determinado na etapa de validação

Classificação Multi-Classe

- A regressão logística é, na verdade, um método de encontrar uma função discriminante $f(\mathbf{x}) = \mathbf{w}^T\mathbf{x}$ (classificador linear)
- A extensão para um problema multi-classe pode ser feita treinando-se, para cada classe k, um classificador "um contra todos" (*one-vs-rest*):

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

  onde o rótulo $y_k \in \{0, 1\}$ indica se $\mathbf{x}$ pertence à classe $\mathcal{C}_k$
  - Codificação 1-de-K (*One Hot Encoding*): $y = (y_1, \ldots, y_K)$
- Decide-se por $\mathcal{C}_k \iff f_k(\mathbf{x}) = \max_{k'} f_{k'}(\mathbf{x})$
  - Equivalentemente, decide-se por $\mathcal{C}_k \iff \hat{y}_k = \max_{k'} \hat{y}_{k'}$, onde $\hat{y}_k = \sigma(\mathbf{w}_k^T\mathbf{x})$

Avaliação do modelo

- A função custo usada no treinamento (mesmo sem regularização) não necessariamente é representativa do verdadeiro custo do modelo em uma aplicação real
  - Ex: acurácia = $1 -$ taxa de erro
- De maneira geral, um classificador é avaliado em termos de sua matriz de confusão

$$p(\hat{k}, k) = P[g(\mathbf{x}) = \hat{k}, y = k]$$

- Para um classificador binário:

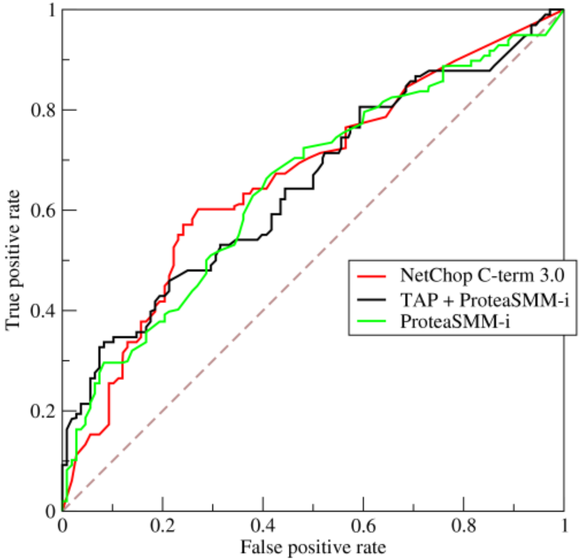
$$TPR = \frac{TP}{TP + FN} \quad \text{true positive rate}$$

$$FPR = \frac{FP}{FP + TN} \quad \text{false positive rate}$$

Matriz de confusão

		Predicted class	
		P	*N*
Actual Class	*P*	True Positives (TP)	False Negatives (FN)
	N	False Positives (FP)	True Negatives (TN)

Curva ROC (Receiver Operating Characteristic)



Métrica de Avaliação

- Na teoria da decisão são estipulados custos $L(\hat{k}, k)$ para cada entrada da matriz de confusão, e a decisão ótima minimiza o custo médio
- Na prática é difícil estipular ou estimar $L(\hat{k}, k)$, mas para facilitar a comparação entre diferentes modelos, é altamente recomendável definir uma métrica única de avaliação (*single-real-number metric*)
- Exemplos:
  - Acurácia
  - AUC (*Area Under [ROC] Curve*)
  - F1 score
  - Matthews Correlation Coefficient