Universidad Técnica Federico Santa María
Departamento de Informática

Control 3 - Máquinas de Aprendizaje (INF-393)

Semestre II 2017 - Viernes 22.12.17

PAUTA

1. (10 puntos) Explique cómo difiere la estrategia implementada por una red neuronal para abordar problemas linealmente inseparables de aquella implementada por un árbol de clasificación.

   Respuesta: La estrategia implementada por un árbol de decisión consiste en dividir el espacio recursivamente en regiones hiper-rectangulares de manera tal que el patrón remanente en cada una de ellas sea suficientemente simple de modelar. La estrategia implementada por una red neuronal consiste en cambio aprender una nueva representación (en general) sobre la cual se pueda implementar un modelo lineal. A esta última estrategia corresponde, en general, una partición no rectangular del espacio original.

2. (10 puntos) Considere el algoritmo Adaboost. ¿Es cierto que el algoritmo convergerá a error de entrenamiento cero a medida que aumentamos el número de clasificadores utilizados? ¿Se requiere que los éstos satisfagan alguna propiedad?

   Respuesta: No necesariamente. Por ejemplo si combinamos clasificadores lineales la frontera de decisión será lineal, por lo que no podremos obtener error de entrenamiento cero si los datos no son linealmente separables. Se requiere que cada predictor exhiba un error de generalización menor a 0.5.

3. (20 puntos) Derive las ecuaciones del backward pass correspondientes a una red neuronal feed-forward de tres capas, entrenada con la función de pérdida cross-entropy y con función de activación softmax en la capa de salida (asuma que se minimiza el error de entrenamiento).

   Respuesta: Denotemos la salida de la red neuronal como $f(\mathbf{x})$. Si la red tiene tres capas, la secuencia de transformaciones ejecutadas por el modelo (forward pass) se resumen en las siguienes ecuaciones:

   $$\begin{aligned} f(\mathbf{x}^{(\ell)}) &= g\left(\boldsymbol{W}\mathbf{z}^{(\ell)} + \boldsymbol{b}\right) \\ \mathbf{z}^{(\ell)} &= h\left(\boldsymbol{V}\mathbf{x}^{(\ell)} + \boldsymbol{c}\right), \end{aligned} \tag{1}$$

   donde h es la función de activación de la capa oculta y g denota la transformación softmax

   $$g_k(\boldsymbol{\xi}) = \frac{\exp(\xi_k)}{\sum_{k'} \exp(\xi_{k'})}. \tag{2}$$

   Notemos que

   $$g_k'(\boldsymbol{\xi}) = g_k(\boldsymbol{\xi})(1 - g_k(\boldsymbol{\xi})). \tag{3}$$

   Ahora, como la red se entrena para minimizar

   $$J(\Theta) = -\sum_{\ell}\sum_{k} y_k^{(\ell)} \log f_k(\mathbf{x}^{(\ell)}), \tag{4}$$

las ecuaciones correspondientes al backward pass quedan como sigue. Para la capa de salida,

$$\frac{\partial J(\Theta)}{\partial W_{kj}} = \sum_{\ell} \frac{\partial J(\Theta)}{\partial f_k(\mathbf{x}^{(\ell)})} \frac{\partial f_k(\mathbf{x}^{(\ell)})}{\partial W_{kj}}\,, \tag{5}$$

con

$$\frac{\partial J(\Theta)}{\partial f_k(\mathbf{x}^{(\ell)})} = -\frac{y_k^{(\ell)}}{f_k(\mathbf{x}^{(\ell)})}, \quad \frac{\partial f_k(\mathbf{x}^{(\ell)})}{\partial W_{kj}} = \mathbf{z}_j^{(\ell)} g_k'(\cdot)\,, \tag{6}$$

Para la capa oculta, tenemos

$$\frac{\partial J(\Theta)}{\partial V_{ji}} = \sum_{\ell} \frac{\partial J(\Theta)}{\partial \mathbf{z}_j^{(\ell)}} \frac{\partial \mathbf{z}_j^{(\ell)}}{\partial V_{ji}}\,, \tag{7}$$

El primer término del lado derecho se obtiene usando la regla de la cadena del cálculo multivariado,

$$\frac{\partial J(\Theta)}{\partial \mathbf{z}_j^{(\ell)}} = \sum_{k} \left(\frac{\partial J(\Theta)}{\partial f_k(\mathbf{x}^{(\ell)})} \frac{\partial f_k(\mathbf{x}^{(\ell)})}{\partial \mathbf{z}_j^{(\ell)}} \right)\,, \tag{8}$$

El término $\partial J(\Theta)/\partial f_k$ se encuentra ya disponible de los cálculos correspondientes a la capa sucesiva. Los demás términos se obtienen fácilmente de las ecuaciones del forward pass,

$$\frac{\partial f_k(\mathbf{x}^{(\ell)})}{\partial \mathbf{z}_j^{(\ell)}} = W_{kj} g_k'(\cdot), \quad \frac{\partial \mathbf{z}_j^{(\ell)}}{\partial V_{ji}} = \mathbf{x}_i^{(\ell)} h_j'(\cdot)\,. \tag{9}$$