EL7006 Redes Neuronales y Teoría de Información para el Aprendizaje
CONTROL 1

Profesor: Pablo Estévez — Semestre Primavera 2015
Prof. Auxiliar: Pablo Huijse
Ayudante: Pablo Huentelemu — Tiempo: 1.5 horas

Atención: Utilice los archivos ipynb adjuntos al control, los cuales tienen en cada uno de sus bloques los contenidos del control previamente programados. Al final de este documento encontrará instrucciones generales para usar *ipython notebook*.

Problema 1: Regresor MCC

Considere un modelo general

$$Z = f(X) + \nu,$$

donde $f(\cdot)$ es una función desconocida, X es un vector de entrada, ν es la componente asociada al ruido y Z son las observaciones. Usted tiene a su disposición los pares entrada-salida del modelo $\{(x_i, z_i)\}_{i=1,\ldots,N}$. El objetivo es entrenar un aproximador paramétrico $g(x; w)$ para recuperar la función desconocida $f(\cdot)$. En este caso la función de densidad de probabilidad (fdp) del ruido está dada por $p_\nu(\nu) = 0.8\,\mathcal{N}(0, \varepsilon) + 0.2\,\mathcal{N}(10.0 \cdot \varepsilon, \varepsilon)$, que corresponde a una mezcla impulsiva de Gaussianas (modelo Middleton), donde $\varepsilon = 0.25 \cdot \mathrm{std}(f(X))$. Entrene regresores usando como funciones de costo el MSE y el "Maximum Correntropy Criterion" (MCC). La función de costo del MCC está dada por:

$$\max_w J(w) = \frac{1}{N} \sum_{i=1}^{N} G_\sigma\left(e_i\right),$$

donde $G_\sigma(\cdot)$ es un kernel Gaussiano con ancho de banda σ.
A continuación se detalla paso a paso lo que usted debe realizar.

1. Genere las observaciones contaminadas Z, considerando $f(X) = a + bx + cx^2$ con parámetros $a = 10$, $b = -1.0 \cdot \mathrm{RUT}_1$ y $c = \mathrm{RUT}_2$, donde RUT_1 y RUT_2 son el primer y segundo dígito de su RUT, respectivamente (si alguno de estos dígitos es cero, reemplácelo por 10).

2. Entrene un aproximador paramétrico usando gradiente descendente y el MSE como función de costo. Considere un aproximador con tres pesos definido como $g(x; w) = w_1 + w_2 x + w_3 x^2$. Utilice 1000 épocas y una tasa de aprendizaje $\mu = 0.01$. Estudie la evolución del error intrínseco $\mathbb{E}\left[(g(x; w) - f(x))^2\right]$ y visualice los resultados del regresor en el espacio de entrada. Considere estos resultados como referencia para comparar con el regresor MCC.

3. Entrene un aproximador paramétrico usando gradiente descendente y el MCC como función de costo. Considere un aproximador equivalente al usado en el punto anterior. Entrene el aproximador usando MSE durante 200 épocas. Luego entrene usando MCC por 800 épocas a partir de los resultados del anterior. Utilice una tasa de aprendizaje $\mu = 1.0$. Pruebe los siguientes valores para el ancho de banda del kernel Gaussiano: $\sigma = 0.2,\ 2.0,\ y\ 20.0$. Describa lo que observa justificando apropiadamente. Compare los pesos obtenidos por los regresores MCC con los parámetros intrínsecos del sistema. ¿Con cuál valor de σ se obtiene un mejor desempeño (mínimo error intrínseco al final del entrenamiento)? ¿Por qué? ¿Cuál es la función del ancho de banda? ¿Qué criterios existen para seleccionar este parámetro? Justifique sus respuestas basándose en la teoría.

4. Compare el mejor regresor MCC con el regresor MSE del punto (2) en términos de la evolución del error intrínseco, parámetros obtenidos y calidad del regresor. ¿Cuál regresor es más robusto al ruido impulsivo? ¿Por qué?

5. Estudie las gráficas de la función de densidad de probabilidad (FDP) del error para el MSE y MCC (mejor caso). Comente sobre las diferencias observadas justificando según la teoría. ¿Cuál es la relación entre el criterio MCC y la esperanza de la FDP del error?

6. Compare el criterio MCC con el criterio de mínima entropía del error (MEE)

$$\min_{w} J(w) = -\log\left(\frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N} G_\sigma\left(e_i - e_j\right)\right),$$

¿Por qué no es necesario compensar el término constante (sesgo) cuando se usa MCC?

Problema 2: Basis pursuit

Considere la función twinsine, más un componente adicional, definida como

$$x_t = \sin(2\pi 1.0t) + \sin(2\pi(1.0 + df)t) + 0.5\cos(2\pi 0.3t), \tag{1}$$

muestreada en una grilla regular de 100 puntos entre 0 y 10 segundos (Frecuencia de muestreo 10 [Hz]).

1. Describa el método de basis pursuit e indique un procedimiento paso a paso que utilice basis pursuit para obtener un espectro de frecuencias de la señal twinsine. Reformule el problema de basis pursuit tal que se pueda resolver con un algoritmo de programación lineal (simplex).

2. Genere la señal de la Eq. (1) considerando $df = 0.1$ Hz. Obtenga y compare los espectros de Fourier y Basis pursuit. Aumente progresivamente el parámetro de sobre-completitud (L) del diccionario de frecuencias. Considere al menos $L = 1, 2, 5, 10, 20$. ¿Pueden diferenciarse los componentes frecuenciales individuales usando el espectro de Fourier o Basis pursuit? Compare los espectros y comente. ¿Qué se espera considerando el espaciado de frecuencias y el límite de Rayleigh ?

3. Genere una nueva señal considerando $df = 0.05$ Hz. Nuevamente, obtenga y compare los espectros de Fourier y Basis pursuit de la señal, utilizando distintos valores de L. Considere al menos $L = 1, 2, 5, 10, 20$ ¿Pueden diferenciarse los componentes frecuenciales individuales usando el espectro de Fourier o Basis pursuit? Compare los espectros y comente. ¿Qué se espera considerando el espaciado de frecuencias y el límite de Rayleigh ?

4. Responda, justificando adecuadamente. ¿Qué ventajas tiene basis pursuit frente a los métodos clásicos de Fourier en términos de diccionarios, resolución y fuga espectral? ¿Qué es basis pursuit denoising, en que se diferencia con basis pursuit y que aplicaciones tiene?

IPYTHON NOTEBOOK: Para iniciar un servidor de *ipython notebook*, abra una consola de anaconda (Menú de inicio, anaconda, open anaconda shell) y escriba *ipython notebook*. Se debería abrir un browser en la dirección http://127.0.0.1:8888/. Una vez en la interfaz del *ipython notebook* utilice el explorador de carpetas y navegue hasta encontrar los archivos ipynb adjuntos. Se recomienda que guarde los archivos en C:\\Anaconda\Examples. Una vez seleccionado el notebook se desplegará en el browser. Para ejecutar cada bloque de código presione SHIFT+ENTER, el notebook producirá los resultados y avanzará al siguiente bloque automáticamente. Puede seleccionar manualmente un bloque y volver a correrlo (SHIFT+ENTER) si lo desea.