

Tarea 12

Simulación de Sistemas

Beatriz Alejandra García Ramos

A 31 de Octubre de 2017

1. Red neuronal

Trabajando como se trabaja con redes neuronales, en esta práctica se tendrán cuatro perceptrones, los cuales serán los elementos básicos de la red neuronal utilizada para poder generar números de manera correcta a partir de una plantilla establecida en donde se le dan tres tonalidades distintas: negro, gris y blanco. Estas tonalidades son tomadas con una cierta probabilidad y dada ésta se indica en la salida qué pixel de los quince que utilizan los números en la plantilla serán marcados. Aquellos que son negros en la plantilla casi siempre se colocarán en el pixel, mientras que los grises y blancos se pondrán ocasionalmente y como un ruido aleatorio de baja frecuencia, respectivamente.

Los cuatro perceptrones utilizados son de acuerdo a los cuatro bits necesarios para la salida, estos reciben la misma entrada y al producir la salida lo hacen de manera independiente. Dado que las tonalidades no tienen un cien por ciento de probabilidad al crearse la salida es muy probable que al menos uno de los números haya sido confundido por otro y se marque como si no fuera el deseado.

2. Tarea base

Analizando el código y su proceso es posible darse cuenta que aquello que es favorable paralelizar es la etapa en la que se realiza la prueba donde se clasifican las coordenadas que se reciben de los verdaderos y falsos a partir de la etapa de entrenamiento.

Se utilizó la función `parSapply` para realizar la paralelización y lo que se seleccionó como la función `prueba` fue aquello que se utiliza para generar los contadores. Además se incrementó el número de iteraciones que se realizaban para ese proceso ya que al dejarlo con la cantidad de trescientas iteraciones los tiempos no reflejaban mejoras al momento de tener la paralelización, así que en la fase de prueba del secuencial y por consecuencia en la paralelización se realizan diez mil iteraciones para lograr ver el cambio verdadero entre lo secuencial y lo paralelo, ya que trescientos pasos es muy poco si se toma en cuenta que la paralelización realiza su proceso a la par.

Como se muestra en la figura 2 el tiempo de ejecución de aquello que se paralelizó tiene una disminución notable con respecto a lo secuencial. Se realizaron veinte repeticiones de cada uno de los procesos para tener una buena cantidad de muestras que permitieran por medio de la gráfica de caja-bigote tener una noción del comportamiento de los tiempos con respecto a su proceso considerando que podría haber ciertas diferencias entre una repetición y otra.

Aproximadamente, al ser paralelizado el proceso, disminuyó su tiempo de ejecución por cinco segundos, lo cual en programación es un buen ahorro de tiempo. Se puede asegurar que la paralelización fue buena al tener resultados como estos y la implementación y el proceso que se paralelizó fue el necesario para que se tuviera una mejora.

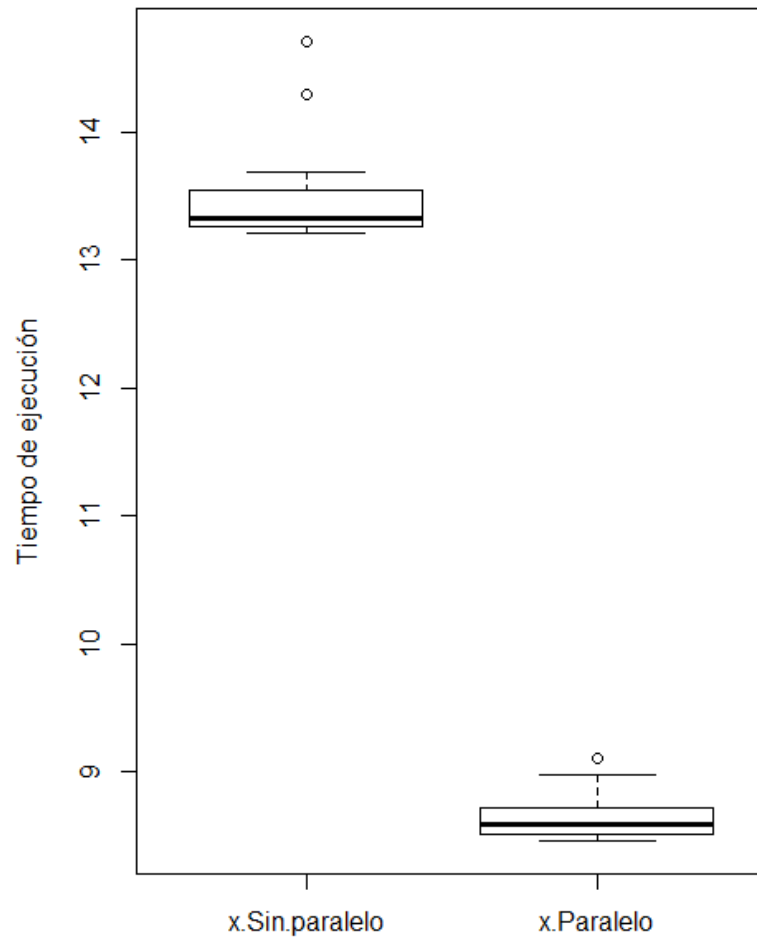


Figura 1: Tiempos de ejecución con veinte repeticiones