

# Simulación de Sistemas

## Práctica 3

Teoría de colas  
1455175: Angel Moreno

28 de agosto de 2018

### Resumen

## 1. Introducción

La *teoría de colas* es un área de las matemáticas que estudia el comportamiento de líneas de espera. Las líneas de espera es nuestra medida de interés. Esta practica nos ayuda a estudiar el efecto del orden de ejecución de trabajos. Se utiliza la examinación de un número si es ó no un número primo como nuestro proceso y los trabajos que asignaremos serán los dicho números.

El siguiente código es el se usa para nuestro proceso a realizar con el trabajo asignado (número entero), el cual es un proceso simple de comparaciones pero eficiente para ilustrar el objetivo de esta práctica.

```
primo <- function(n) {  
  if (n == 1 || n == 2) {  
    return(TRUE)  
  }  
  if (n %% 2 == 0) {  
    return(FALSE)  
  }  
  for (i in seq(3, max(3, ceiling(sqrt(n))), 2)) {  
    if ((n %% i) == 0) {  
      return(FALSE)  
    }  
  }  
  return(TRUE)  
}
```

## 2. Tarea

Se examina cómo las diferencias en los tiempos de ejecución de los diferentes ordenamientos cambian cuando se varía el número de núcleos asignados al cluster y se investiga también el efecto de la proporción de primos y no primos en el vector igual como la magnitud de los números incluidos en el vector. El primer reto es argumentar, apoyándose con visualizaciones, las posibles causas a las diferencias en los tiempos de ejecuciones observadas en la tarea base y razonar cómo y por qué detrás de las diferencias observadas.[1]

## 3. Simulación

Se tomaron los siguientes parámetros para la simulación:

- Cantidad de 2, 3 y 4 cluster.
- Magnitud de los números de 4, 5 y 6 dígitos.
- Cantidad de números primos y no primos (proporción).

- Ordenamientos de los trabajos (números primos).

Los números primos se tomaron de una pagina de internet[2] y se generaron los números no primos de una secuencia de números pares, múltiplos de 3 y múltiplos de 5. El vector que se utilizo se genero apartir de estos números primos y no primos.

Una vez programada la simulación con los parámetros dados obtuvimos los siguientes resultados.

## 4. Resultados

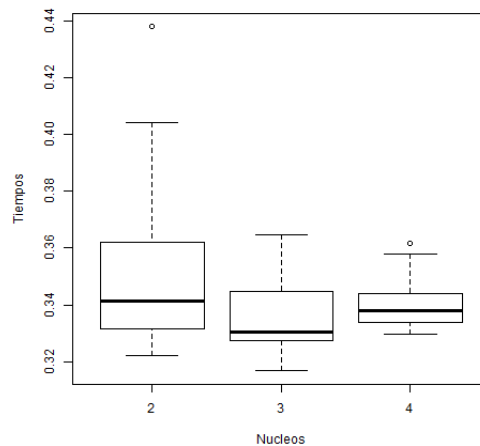


Figura 1: Caja bigotes de núcleos contra tiempos.

Se puede observar en la figura 1 como al momento de variar los núcleos de la computadora uno esperaría cambios radicales en los tiempos de espera pero notamos que no varia mucho y al contrario el aumento de núcleos no garantiza disminuir nuestros tiempos de espera.

Al momento de también variar la magnitud de los dígitos obtenemos la figura 2 donde se observa que efectivamente si se cambia la magnitud de los dígitos a explorar , el tiempo de ejecución aumenta por la magnitud del dígito.

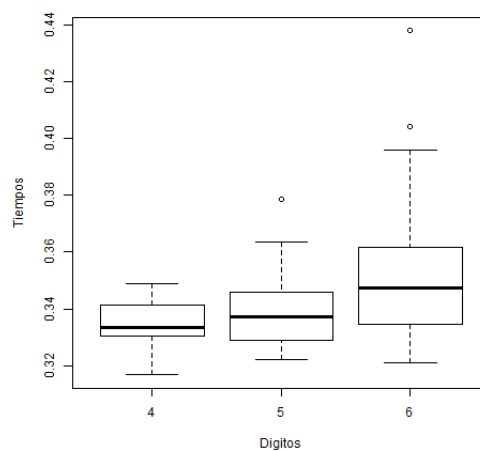


Figura 2: Caja bigotes de dígitos contra tiempos.

Y por último la figura 3 y la figura 4 muestran las gráficas de proporciones, orden contra tiempos respectivamente. Observe que las orden de que se entrega el trabajo (numero entero) no afecta al momento de la ejecución de la simulación mientras tanto las proporciones de, por así decirlo, "trabajos mas difíciles" "trabajos fáciles" si afectan nuestra simulación.

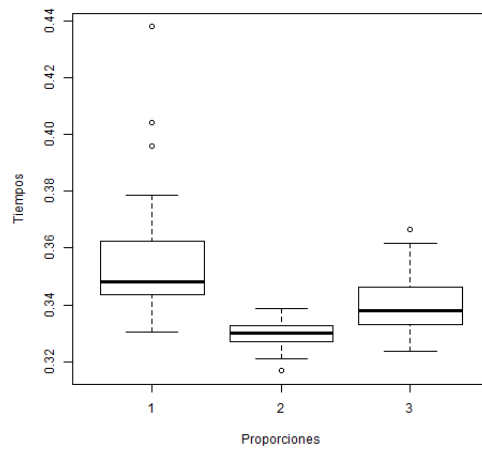


Figura 3: Caja bigotes de proporciones contra tiempos.

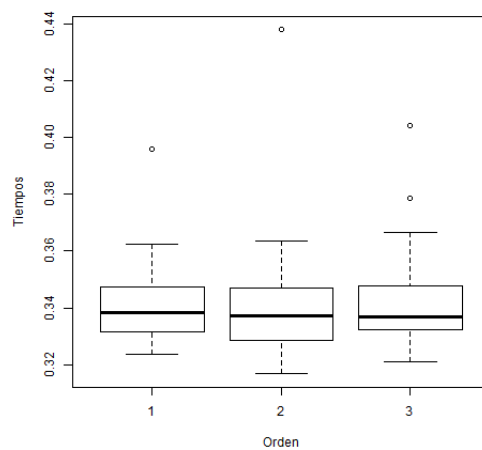


Figura 4: Caja bigotes de orden contra tiempos.

## Referencias

- [1] SCHAEFFER E., *R paralelo: simulación and análisis de datos*, 2018., <https://elisa.dyndns-web.com/teaching/comp/par/>
- [2] SCHAEFFER E., *Probabilidad y estadística*, 2018., <https://elisa.dyndns-web.com/teaching/prob/prob.html>
- [3] ANÓNIMO., *The first fifty million primes*, 2017., <https://primes.utm.edu/lists/small/millions/>