ACTIVIDAD 2 CONFIABILIDAD DE REDES

Ramirez, Fernando R-3888/1

Introducción

El trabajo pretende demostrar el uso del algoritmo de Montecarlo para generar un estimador confiable de una variable aleatoria con una distribución determinada.

Para esto, se hace uso de los archivos provistos por la catedra: mc_std.c y mt.c, siendo el primero el código de Montecarlo y el segundo un generador de números aleatorios utilizando el método de Mersenne-Twister.

Desarrollo

Primero se trabaja con una variable aleatoria X de distribución exponencial con parámetro λ=4.

Revisando el código de mc_std.c se notan como partes principales, el bloque for que hace las iteraciones de la variable aleatoria (X), donde además se generan dos variables más: "S" que se utilizara posteriormente en el calculo de la media y que es meramente la suma de los valores de la variable aleatoria X, y "V" que se utilizará para el calculo de la varianza y que es la suma de los sucesivos X al cuadrado. Luego del for se procede a calcular el estimador crudo y la varianza utilizando S, V y las fórmulas correspondientes de cálculo.

Antes del For principal tenemos una sentencia que realiza el registro de clock() que en la sentencia que esta después, del cálculo de la media y la varianza, se utiliza para calcular el tiempo de ejecución del algoritmo.

Y por ultimo el display de los valores parámetro del Montecarlo (repeticiones) y de los que arrojó, como ser, Tiempo de ejecución, Media, Varianza, Desviación estándar y error relativo.

Realizando un primer RUN al algoritmo se obtienen los siguientes resultados:

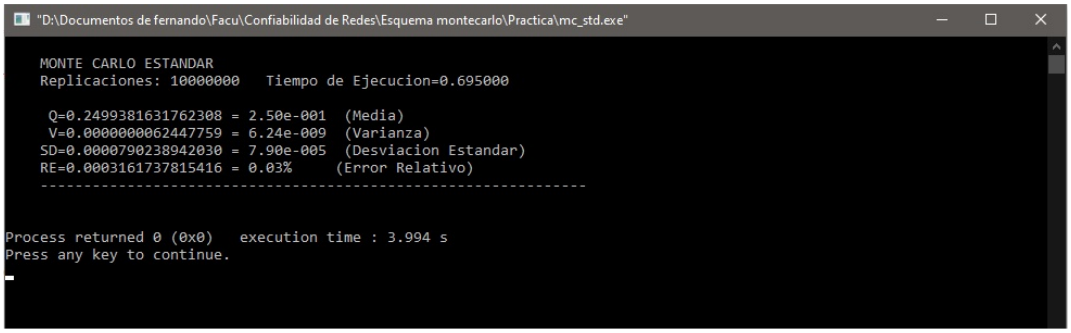

Fig 1. Resultado Montecarlo Estándar para una Var. Alea. con dist. Expo y λ=4

Toca ahora, reemplazar la variable X de distribución Exponencial por la brindada en la Actividad 1.

Para ello se modifica el código de la siguiente manera:

```
48
49    for(k=0;k<size;k++){
50
51      C=U;
52      if (C<=0.1){X=50;}
53      if (C>0.1 && C<=0.4){X=80;}
54      if (C>0.4 && C<=1){X=10;}
55      //Expo(4.0);          // elijo lambda=4
56      S+=X;
57      V+=X*X;
58    }
59
60    Q=S/size;
61    V=(V/size-Q*Q)/(size-1);
62
63    //--------------------------------------
64
```

Fig. 2 - Modificación software Montecarlo Crudo

Obviamente se declaro la Variable C del mismo tipo que X pero por razones obvias no se muestra dicha declaratoria.

Se compila y ejecuta el algoritmo y se obtienen los siguientes resultados:

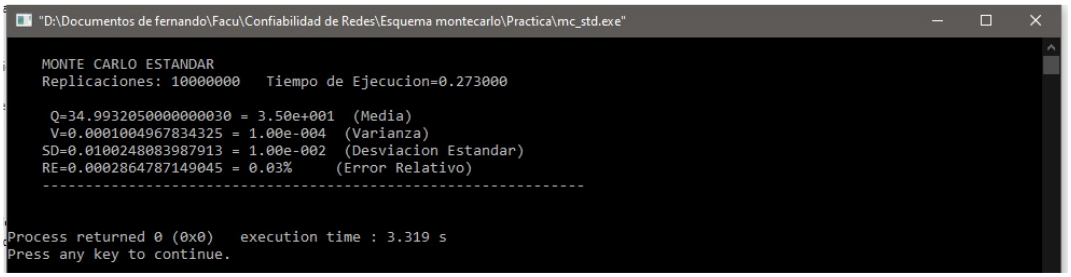

Fig. 3 - Resultado de aplicar Montecarlo a la Var. Alea. de la Actividad 1

Si nos remitimos a los resultados obtenidos en la actividad 1 los cuales fueron:

$$E(x) = 50 * 0.1 + 80 * 0.3 + 10 * 0.6 = 35$$

Se puede observar que el método de Montecarlo generó un estimador que solo difiere en 0.01 del resultado que debería de tener. Sin embargo, se nota qué, para el mismo nivel de iteraciones, el método por frecuencias relativas y Montecarlo arrojan los mismos errores frente a la esperanza.

Nada se puede decir respecto de los otros indicadores, dado que no fueron calculados en la actividad 1.

No obstante, si se puede destacar algo muy importante. El método de Montecarlo pretende generar un estimador de una variable aleatoria de distribución desconocida y se puede ver con claridad que es lo logra muy bien. Es claro que en este ejercicio la distribución de probabilidades de la variable aleatoria se conocía, pero es exactamente

esto lo que nos permite asegurar que el programa funciono correcto, al menos para una distribución discreta.