|  |  |
| --- | --- |
|  | **Pontificia Universidad CatÓlica de Chile**  **Escuela de IngenierÍa**  **DEPARTAMENTO DE INGENIERÍA INDUSTRIAL Y DE SISTEMAS**  **PROFESOR: PEDRO GAZMURI S.**  **ICS 3723 – SIMULACIÓN**  **1/2017** |

# **Tarea N°4**

|  |  |
| --- | --- |
| **Alumnos**:  Ignacio Acevedo |  |
| Ignacio Barría |  |
| Daniel Carrasco |  |
| Kevin Johnson |  |
|  |  |

**Fecha Entrega**: 12 de junio del 2017

Tabla de Contenidos

**Índice 0**

**Pregunta 1 0**

Generación del input 0

Marco Teórico0

Procedimiento0

Resultados0

Validación 0

**Pregunta 2 0**

Generación del input 0

Procedimiento0

Resultados0

Validación 0

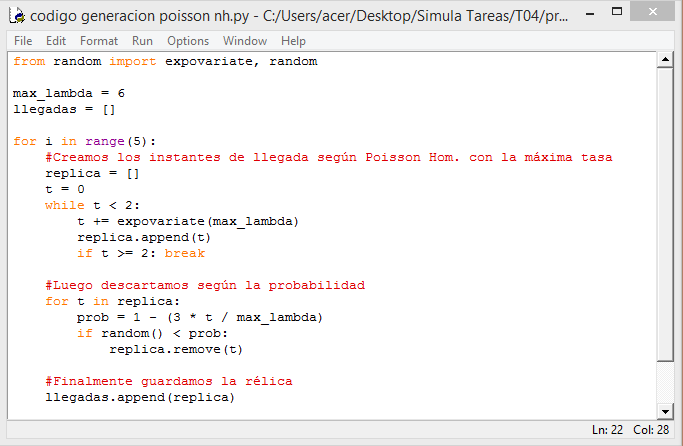
**Pregunta 3 0**

lalala 0

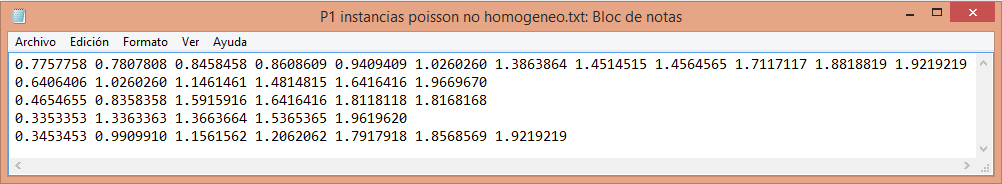
**Pregunta 1**

**Generación del input**

Para generar instantes de llegada según un proceso de Poisson No Homogéneo se utilizó el método visto en clases, específicamente en la última parte de la presentación *Clase N°6.pptx.* Se programó el siguiente código en Python:

  
Imagen 1: Código en Python para generar 5 réplicas de instantes de llegada según Poisson NH.

Se obtuvieron las siguientes 5 réplicas para los instantes de llegada de los pasajeros:

  
Imagen 2: instantes de llegada obtenidos con el código anterior.

**Marco Teórico**

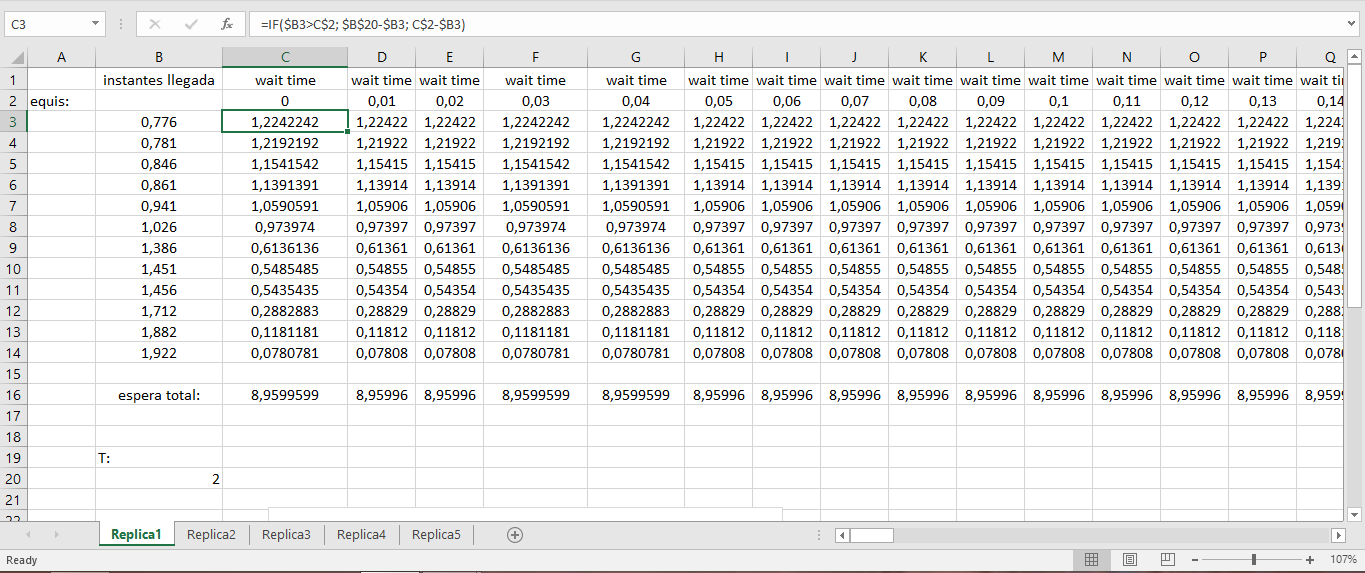
Sea **x** el parámetro de diseño del modelo e **Y** las variables aleatorias de entrada. Se definen las siguientes variables:

Dado que **x** es el instante en que pasará el segundo bus al paradero,

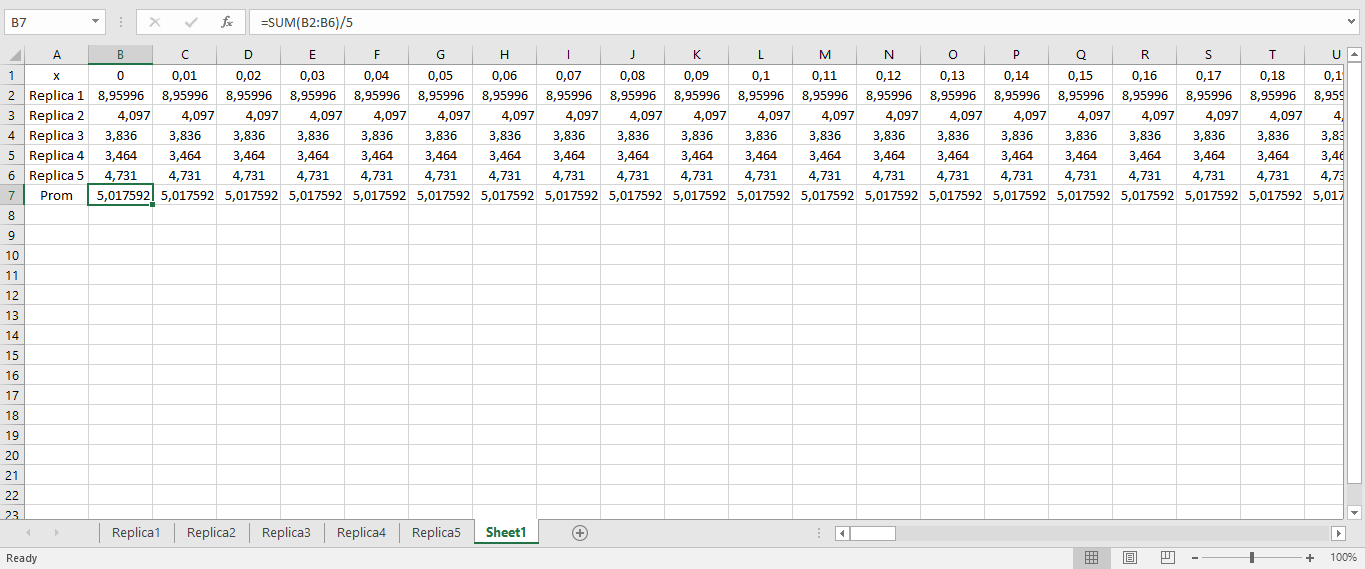
Luego, la función a optimizar es:

**Procedimiento**

Se trabajó en un Excel donde cada hoja corresponde a una réplica. La segunda columna corresponde a los instantes de llegada mientras que la segunda fila corresponde a valores de x desde 0 a 2 con un paso de 0,01. La última fila corresponde a la espera total de los pasajeros, para cada valor de x.

  
Imagen 3: Excel de trabajo para una réplica.

Luego se construyó una hoja para obtener el promedio de las 5 réplicas:

  
Imagen 4: Excel de trabajo, hoja de los promedios.

**Resultados**

El tiempo de espera total en función de **x** se graficó para cada réplica:

Gráfico 1: tiempo de espera total para cada valor de x, réplica 1.

Gráfico 2: tiempo de espera total para cada valor de x, réplica 2.

Gráfico 3: tiempo de espera total para cada valor de x, réplica 3.

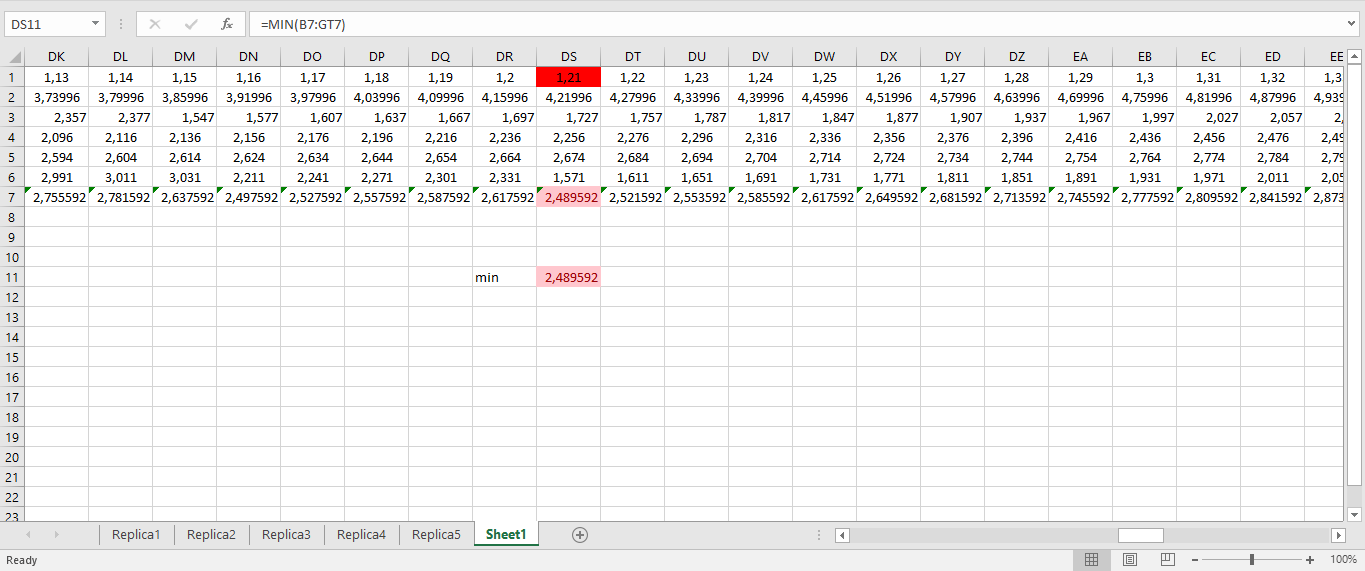
Gráfico 4: tiempo de espera total para cada valor de x, réplica 4.

Gráfico 5: tiempo de espera total para cada valor de x, réplica 5.

Luego se procedió a graficar el promedio de los tiempos de espera total sobre las 5 réplicas:

Gráfico 6: tiempo de espera total para cada valor de x, promedio sobre las 5 réplicas.

Finalmente se encontró el instante x que minimiza el tiempo de espera total promedio sobre las 5 réplicas, que resultó ser

  
Imagen 5: Excel de trabajo, hoja de los promedios, mínimo tiempo de espera.

**Validación**

Se puede observar en cada gráfico que la curva de espera total termina a la misma altura a la que empieza. Esto es lógico ya que los pasajeros esperarán lo mismo si el segundo bus pasa en el instante 0 (sin pasajeros) que en el instante T (junto con el primer bus).

**Pregunta 2**

**Generación del input**

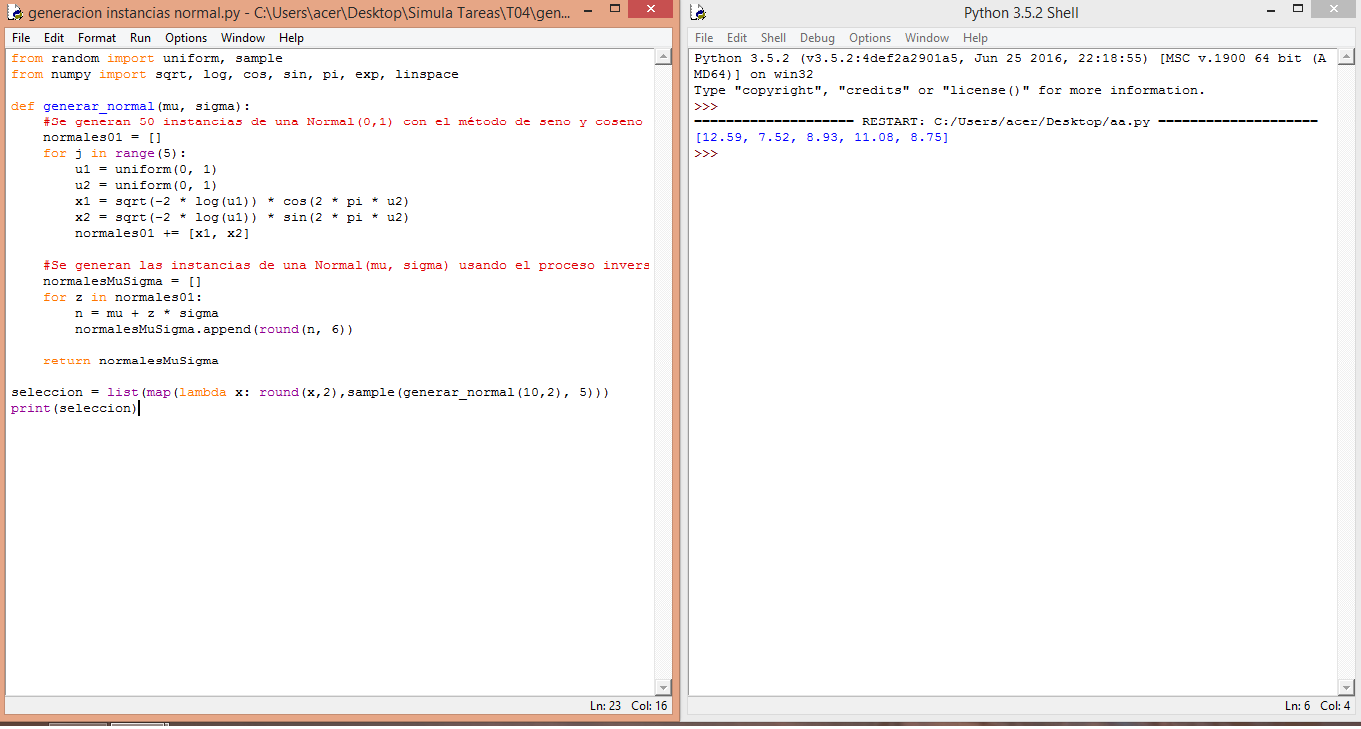
Para generar instancias de una distribución Normal(µ, σ) se utilizó el método de Box-Muller, visto en clases y que se explica a continuación:

Sean instancias de una distribución Uniforme(0, 1), entonces

son instancias de una distribución Normal(0, 1).

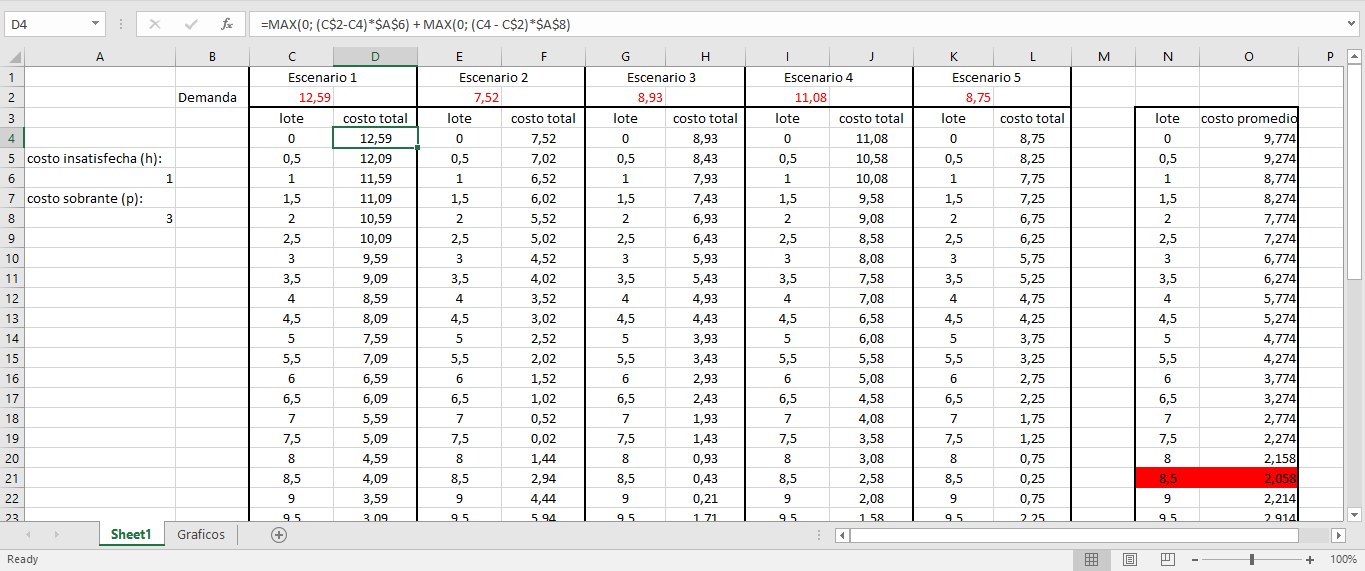
Además, sea entonces .

El código utilizado para este método se muestra a continuación:

****Imagen 6: código en Python para generar 5 instancias de una Normal(10, 2).

**Procedimiento**

Se trabajó en un Excel cada uno de los escenarios, donde la demanda se muestra en rojo en la segunda fila. La primera columna del escenario corresponde al lote inicial Q, desde 0 hasta 20 en intervalos de 0,5. La segunda columna corresponde al costo total, que considera el costo de insatisfacción de la demanda y el costo de las unidades que sobran al final del periodo.

****Imagen 7: Excel de trabajo para la pregunta 2.

**Resultados**

Se graficó el costo total para cada una de las 5 réplicas.

Gráfico 7: costo total para distintos lotes iniciales, réplica 1.

Gráfico 8: costo total para distintos lotes iniciales, réplica 2.

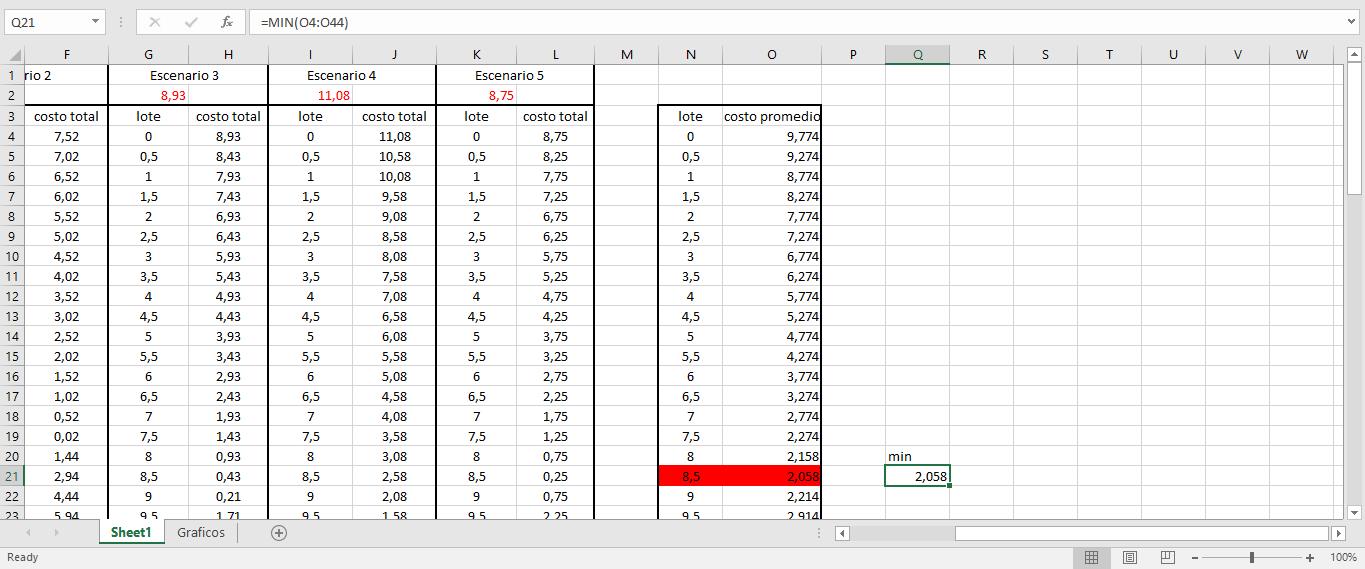
Gráfico 9: costo total para distintos lotes iniciales, réplica 3.

Gráfico 10: costo total para distintos lotes iniciales, réplica 4.

Gráfico 11: costo total para distintos lotes iniciales, réplica 5.

Finalmente se graficó el costo total promedio sobre las 5 réplicas para luego encontrar el mínimo utilizando la función MIN en el último recuadro del Excel de trabajo.

Gráfico 12: costo total para distintos lotes iniciales, promedio sobre las réplicas.

  
Imagen 8: Excel de trabajo, costo promedio mínimo.

Por lo que el lote inicial que minimiza el costo de insatisfacción de demanda y el costo de sobrestock es:

**Validación**

En cada gráfico, la curva de costo total comienza a la misma altura que la demanda. Esto es completamente lógico, ya que, si el lote inicial es cero, el costo total será la propia demanda multiplicada por el costo de insatisfacción de la demanda, que para este caso es 1.

Además, el costo total es cero cuando el lote inicial es exactamente igual a la demanda. Esto es lógico, ya que en ese caso no habría demanda insatisfecha ni tampoco unidades sobrantes al final del periodo.

**Demanda Exponencial(1/10)**

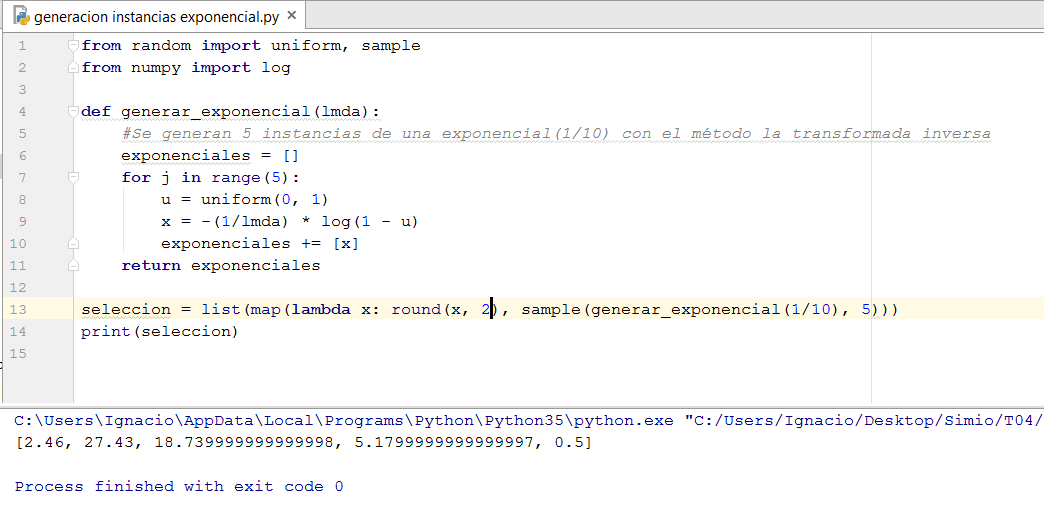
Al igual que en caso anterior no se puede resolver de forma analítica, ya que la demanda es estocástica y no se tiene certeza absoluta de cuál será su valor, de esta manera se procedió a resolver utilizando los mismos pasos que con la demanda como distribución normal.

**Generación del input**

Para generar instancias de una distribución Exponencial() se utilizó el método de la transformada inversa, visto en clases y que se explica a continuación:

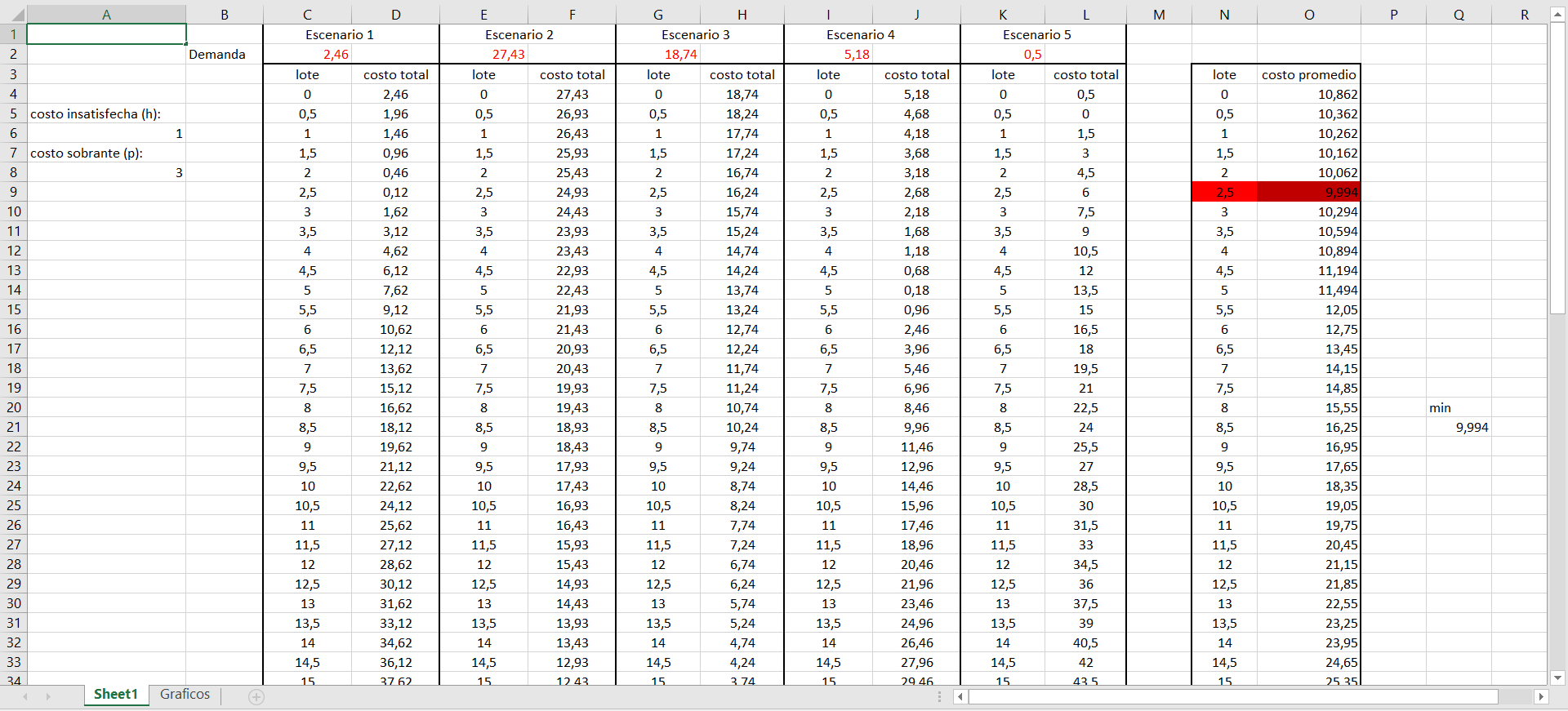
Sea instancia de una distribución Uniforme(0, 1), entonces es una instancia de una distribución Exponencial() con:

El código utilizado para este método se muestra a continuación:

Imagen 9: código en Python para generar 5 instancias de una Exponencial (1/10).

**Procedimiento**

Al igual que en el caso anterior se trabajó en un Excel cada uno de los escenarios, donde la demanda se muestra en rojo en la segunda fila. La primera columna del escenario corresponde al lote inicial Q, desde 0 hasta 20 en intervalos de 0,5. La segunda columna corresponde al costo total, que considera el costo de insatisfacción de la demanda y el costo de las unidades que sobran al final del periodo.

****Imagen 10: Excel de trabajo para la pregunta 2 (demanda exponencial).

**Resultados**

Se graficó el costo total para cada una de las 5 réplicas.

Gráfico 13: costo total para distintos lotes iniciales, réplica 1.

Gráfico 14: costo total para distintos lotes iniciales, réplica 2.

Gráfico 15: costo total para distintos lotes iniciales, réplica 3.

Gráfico 16: costo total para distintos lotes iniciales, réplica 4.

Gráfico 17: costo total para distintos lotes iniciales, réplica 5.

Finalmente se graficó el costo total promedio sobre las 5 réplicas para luego encontrar el mínimo utilizando la función MIN en el último recuadro del Excel de trabajo.

Gráfico 18: costo total para distintos lotes iniciales, promedio sobre las réplicas.

Por lo que el lote inicial que minimiza el costo de insatisfacción de demanda y el costo de sobrestock es:

El gran problema que se presenta respecto a este Stock es que la distribución exponencial varia demasiado, entonces una muestra con 5 escenarios es poco representativa. Si notamos bien el promedio de la distribución es 10, sin embargo, estamos muy alejados de ese promedio debido a la gran variabilidad de esta distribución.

**Validación**

De la misma forma que en el primer caso, la validación es igual.

En cada gráfico, la curva de costo total comienza a la misma altura que la demanda. Esto es completamente lógico, ya que, si el lote inicial es cero, el costo total será la propia demanda multiplicada por el costo de insatisfacción de la demanda, que para este caso es 1.

Además, el costo total es cero cuando el lote inicial es exactamente igual a la demanda. Esto es lógico, ya que en ese caso no habría demanda insatisfecha ni tampoco unidades sobrantes al final del periodo.

**Pregunta** **3**

Sea A={1..M}, con M lo suficientemente grande como para satisfacer la demanda en un caso extremo.

Para cada x en A queremos encontrar , que es el valor de que para x=S-s minimiza los costos E(), para lo que hacemos *sample path optimization*. Con el fin de realizar esto hacemos una cantidad K de réplicas (dependiente de nuestra capacidad computacional), con lo que tenemos las funciones , donde representa las variables aleatorias de input instanciadas en la réplica i. Con esto obtenemos .

Del enunciado sabemos que la función es lineal convexa, por lo que podemos graficarla y obtener .

Finalmente, sea y sean y los valores donde dicho óptimo se alcanza. De aquí entonces que la política óptima es:

Para nuestra estrategia necesitamos definir la cantidad x:

Utilizando dicha cantidad, el costo para cada periodo es el siguiente:

Donde es el inventario del periodo y es la demanda del periodo . Asumiendo un inventario inicial , que cumple que .