Ampliación de Investigación Operativa.
Ejercicio práctico a desarrollar en clase.

26 de marzo de 2007

1. Simulación

Ejercicio 1. Simular el número de lanzamientos necesarios a realizar de una moneda perfecta hasta conseguir tres caras.

- Diseñar el modelo probabilístico que sigue el sistema e implementarlo en R.
- Realizar un número grande de simulaciones (> 1000).
- Análizar los resultados obtenidos. En concreto:
  - Representar la función de distribución o de densidad obtenida para el número de lanzamientos.
  - Determinar el valor medio de lanzamientos.
  - Determinar el intervalo de confianza al nivel $0,95$.
  - Tratar de determinar si la función de densidad obtenida para el número de lanzamientos se ajusta a algún modelo conocido.
  - Analizar la sensibilidad del modelo cuando la probabilidad de conseguir cara en una moneda recorre $(0,1)$ (es decir, dividir el intervalo y dar los intervalos de confianza para cada división). Representar los intervalos de confianza para cada uno de los valores.

Para este primer ejercicio se recomienda seguir el guión del Tema 6 visto en clase.

Modificar el anterior ejercicio para simular el número de lanzamientos necesarios a realizar hasta obtener 3 caras consecutivas.

Ejercicio 2.

Elegir uno de los siguientes problemas para simular:

1. Se pretende implementar para una base de datos un algoritmo de ordenación. Como la base de datos se modifica diáriamente, pero se pretende que la ordenación se realice una vez a la semana.

   Supondremos que sólo hay 1000 datos en la base de datos y el campo que se pretende usar para ordenar contiene números entre 0 y 1000000, totalmente aleatorios. Cada semana se modifican $p = 100$ datos (determinados aleatoriamente de modo independiente, es decir, el mismo dato se puede modificar dos veces), asignando al campo para ordenar un número aleatorio entre 0 y 1000000.

   El método utilizado para ordenar será el de la burbuja.

   Se pretende estimar el tiempo (para simplificar, podemos suponer que el coste de intercambiar dos valores es 1 y todos los demás costes son 0).

2. Queremos comparar dos estrategias de mus. En concreto, quedarnos sólo con cerdos en el primer descarte, frente a quedarnos con cerdos y caballos. El número de cartas de la baraja es $p = 40$, con 8 cerdos y 4 caballos.

   Se pretende estimar las probabilidades de obtener media de cerdos o caballos y de obtener duples.

Para cada problema, seguir el guión del ejercicio anterior. La sensibilidad se determinará respecto al parámetro p. En el primer caso se considerará p entre 1 y 1000 y para el segundo, entre 30 y 52.

2. Programación Dinámica

Ejercicio 1. La cadena de supermercado Safeco, compra a una lechería local galones de leche a un precio de 1 euro/galón. Cada galón se vende en las tres tiendas de la cadena a 2 euro/galón. La lechería compra la leche sobrante a 0.5 euro/galón al término del día. Desafortunadamente para Safeco, es incierta la demanda en cada una de sus tres tiendas. Los datos acumulados indican que la demanda diaria en cada tienda es como la que se muestra en la siguiente tabla:

	Demanda diaria (galones)	Probabilidad
Tienda 1	1	0.6
	3	0.4
Tienda 2	1	0.5
	2	0.4
	3	0.1
Tienda 3	1	0.4
	2	0.3
	3	0.3

Además, debido a la confianza con el cliente, se estima que por cada galón de leche faltante en una tienda, se penaliza con 1 euro (costo por faltante). Safeco desea determinar el número de galones a comprar, así como el reparto entre las tres tiendas con el fin de maximizar la ganancia neta diaria (ingresos menos costos) esperada que da la leche.

Describir el comportamiento del sistema anteriormente planteado y determinar cuales son las soluciones factibles del mismo.

¿Qué propiedad verifica la solución o soluciones óptimas del problema anterior?

Implementar el algoritmo de Programación Dinámica para este sistema. Como datos del problema se considerarán el número de tiendas, el vector de posibles ventas y la matriz de probabilidades de cada venta para cada tienda. Todos los demás datos intermedios deberán calcularse mediante funciones.

Comprobad cómo varía la solución óptima del sistema si agregamos una cuarta tienda que vende entre 1 y 4 galones con probabilidades 0,3, 0,3, 0,3 y 0,1 y disponemos de 10 galones para asignar entre las tiendas.

Ejercicio 2. Imagine que tiene 5000 euros para invertir y que tendrá la oportunidad de hacerlo en cualesquiera de dos inversiones (A o B) al principio de cada uno de los próximos tres años. Existe incertidumbre respecto al rendimiento de ambas inversiones. Si se invierte en A, se puede perder todo el dinero o (con probabilidad más alta) obtener 10000 euros (una ganancia de 5000) al final del año. Si se invierte en B, se pueden obtener los mismos 5000 o (con probabilidad más baja) 10000 al terminar el año. Las probabilidades para estos eventos son las siguientes:

Inversión	Cantidad obtenida	Probabilidad
A	0	0.3
	10000	0.7
B	5000	0.9
	1000	0.1

Se le permite hacer (a lo sumo) una inversión al año y sólo puede invertir 5000 cada vez.

1. Utilice programación dinámica para encontrar la política de inversión que maximice la cantidad de dinero esperada que tendrá después de los tres años.

2. Utilice programación dinámica para encontrar la política de inversión que maximice la probabilidad de tener por lo menos 10000 después de los 3 años.

3. Simulación de Programación Dinámica

Ejercicio 1. Se pretende obtener la solución óptima del problema del ejercicio 1 mediante un proceso de simulación del sistema. Para ello, sigue los siguientes pasos:

1. Implementar una función para generar distribuciones de probabilidades discretas con espacio de estados finitos.

   Argumentos: número de datos a simular (n), vector de estados (est), vector de probabilidades (prob)

   Salida: un vector de datos de longitud n

2. Implementar una función para obtener el beneficio neto diario, conocida la demanda en las distintas tiendas, así como la asignación diaria.

   Argumentos: vector de cantidades demandadas en las tiendas (demandas), vector de cantidades asignadas en las tiendas (asignadas)

   Salida: beneficio neto diario cuando se produce dicha demanda y asignación

3. Implementar una función que simule el comportamiento del sistema a partir del conocimiento del vector de cantidades asignadas y de las funciones desarrolladas en los apartados 1 y 2.

   Argumentos: número de réplicas del sistema (n), vector de cantidades asignadas en las tiendas (asignadas), lista de los vectores de estados de las tiendas (estados), lista de los vectores de probabilidades de las tiendas (probabilidades)

   Salida: vector de beneficios netos

4. Implementar una función que obtenga las posibles soluciones factibles del problema, cuando se conocen las posibles demandas de cada tienda. (Use la filosofía de la programación dinámica, de salida a entrada).

   Argumentos: lista de los vectores de estados de las tiendas (estados)

   Salida: Matriz con tantas columnas como tiendas y donde las filas representan una solución factible.

5. Implementar una función que simule el comportamiento del sistema para todas las soluciones factibles del mismo, a partir de las funciones desarrolladas en los apartados 3 y 4.

   Argumentos: número de réplicas del sistema (n), lista de los vectores de estados de las tiendas (estados), lista de los vectores de probabilidades de las tiendas (probabilidades)

Salida: matriz con n columnas y tantas filas como soluciones factibles, siendo cada fila el vector de beneficios netos para su solución factible de cantidades a asignar correspondiente

6. A partir de la simulación realizada en el apartado anterior, implementar una función que determine la solución óptima al problema. Comparar los resultados con la solución teórica obtenida. Comparar el tiempo necesario para obtener una y otra solución.

   Argumentos: salida de la función del apartado 5, lista de los vectores de estados de las tiendas (estados)

   Salida: Número de galones a comprar, así como el vector de asignación óptimo y beneficio neto estimado.

4. Documentación a entregar

Para cada ejercicio, se entregará un fichero ".r" con las funciones implementadas en R y debidamente comentadas y un fichero ".txt" (o ".rft", ".doc", etc) con los datos de salida y los comentarios necesarios. En caso de elegir un formato que lo permita, se pueden incorporar las gráficas en lugar de los datos cuando sea pertinente.

Fecha de entrega: 18 de Abril.