

Algoritmia/Algoritmos probabilísticos

Los **algoritmos probabilísticos** o probabilistas son aquellos que basan el resultado devuelto en decisiones aleatorias, de tal forma que, en promedio se obtienen una buena solución al problema planteado, dada una distribución de datos de entrada. Un problema típico para hacer ver el funcionamiento de este tipo de algoritmos es el siguiente:

Se conocen dos determinados emplazamientos lo suficientemente alejados el uno del otro, al menos igual a la distancia entre cada emplazamiento y el lugar de partida. Se sabe también que en uno de los dos lugares existe un importante botín. Sin embargo, no es posible explorar un sitio primero y otro después, pues cada día que pasa, el botín se reduce en una cantidad fija. Si se hace uso de la inteligencia, podría calcularse con exactitud el lugar del botín, pero el tiempo empleado en el cálculo haría perder parte de las ganancias. Supóngase ahora que alguien ofreciera la solución a cambio de parte de las ganancias, algo inferior al tiempo de cálculo. La duda planteada sería la siguiente: ¿Cuál es la mejor solución: calcular la ruta de forma independiente o aceptar el trato ofrecido? La solución es ninguna de las dos, pues hay una solución mejor, elegir aleatoriamente uno de los lugares.

Concretando el ejemplo, supongamos que cada localización está separada por cinco días de viaje, el cálculo de la ruta adecuada cuesta cuatro días y el trato ofrecido es dar una ganancia equivalente a tres días de pérdida. Supóngase x como el valor del botín e y como la cantidad diaria que se disminuye. Así, en el primero de los casos, se obtiene una ganancia de $x-9y$, mientras que si se acepta el trato, se obtiene una ganancia de $x-8y$. El segundo trato es claramente mejor, pero podría mejorarse. Si se escoge al azar un camino a seguir, podría acertarse o fallarse en la elección. Si se acierta, se obtiene un botín equivalente a $x-5y$, pero si se falla, se obtendría $x-10y$. Sin embargo, al haber sólo dos opciones, el caso promedio nos dice que se obtiene una ganancia de $x-7,5y$, mejorando los dos casos deterministas.

Otra ventaja de los algoritmos probabilistas sobre los deterministas consiste en que, si existen varias soluciones a un mismo problema, pueden devolver diferentes soluciones en diferentes ejecuciones sobre el mismo conjunto de datos, mientras que uno determinista ofrecerá siempre la misma solución. Así pues, tanto el tiempo de ejecución como el resultado pueden variar de una ejecución a otra.

Clasificación de los algoritmos probabilísticos

Se pueden distinguir fundamentalmente cuatro grandes categorías:

- **Algoritmos numéricos**, que devuelvan una aproximación al resultado, frecuentemente en forma de intervalo. Son útiles cuando la solución exacta es demasiado costosa (o directamente imposible de calcular, como por ejemplo, para números irracionales) y una aproximación es lo suficientemente buena. Considérese que se desea calcular el resultado de una complicada integral de varias dimensiones. Tal vez sólo se necesite una precisión de cuatro decimales, aunque la solución exacta conste de varias decenas de los mismos. Este tipo de algoritmos suelen ofrecer resultados más precisos cuanto mayor tiempo se dedica a su cálculo.
- **Algoritmos de Monte Carlo**, que siempre devuelven una solución aunque esta a veces no sea correcta. Son útiles cuando una aproximación no es suficiente (por ejemplo, en un problema de decisión). Cuentan con el inconveniente de no saber con exactitud si la respuesta es acertada, pues sólo existe una cierta probabilidad de éxito. Cuantas más veces se ejecute, más seguro se estará de la corrección de la solución.
- **Algoritmos de Las Vegas**, similares a los de Monte Carlo pero que nunca devuelven una solución errónea, con el inconveniente de que pueden no terminar o devolver solución. Esto garantiza que la respuesta sea la buena, pero no garantiza que el algoritmo funcione. Como en los casos anteriores, cuanto mayor es el tiempo dedicado al cálculo, más fiable es la solución. No debe confundirse este tipo de algoritmos con aquellos deterministas, como el simplex en programación lineal, que son muy eficientes para la mayoría de los posibles datos de entrada pero desastrosos para unos pocos. Nótese que dichos algoritmos siempre devuelven una solución correcta.

- **Algoritmos de Sherwood**, los cuales devuelven siempre una respuesta, la cual es forzosamente exacta. Aparecen cuando un algoritmo determinista conocido es más rápido en el caso medio que en el peor. El uso del azar permite reducir, e incluso eliminar, la diferencia entre buenos y malos ejemplares.

Problemas resueltos

Problema de la primalidad de números enteros

Se sugiere ir al libro de Brassard & Bratley

Fuentes y contribuyentes del artículo

Algoritmia/Algoritmos probabilísticos *Fuente:* <http://es.wikibooks.org/w/index.php?oldid=151976> *Contribuyentes:* Gothmog, Luckas Blade, Morza, 5 ediciones anónimas

Licencia

Creative Commons Attribution-Share Alike 3.0 Unported
[//creativecommons.org/licenses/by-sa/3.0/](https://creativecommons.org/licenses/by-sa/3.0/)
