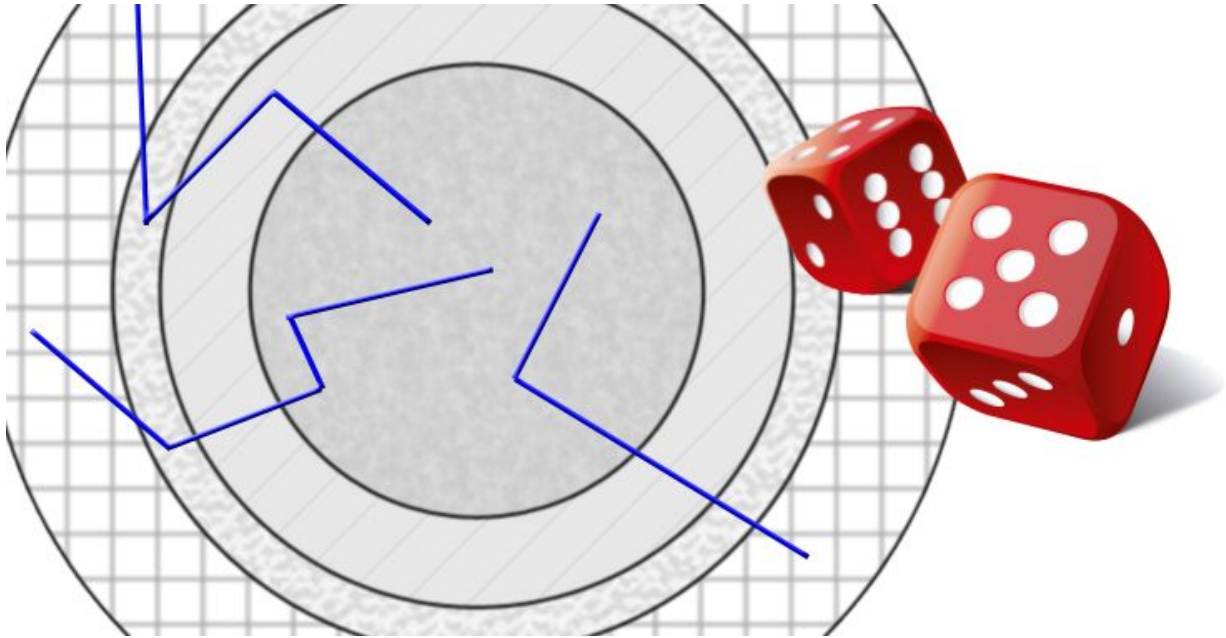


ALGORITMO DE MONTE CARLO

Tratamiento Inteligente de Datos.

Máster en Ingeniería Informática.



Lidia Sánchez Mérida.

Fernando Roldán Zafra.

ÍNDICE

RESUMEN	3
INTRODUCCIÓN	3
ORIGEN	4
APLICACIONES	5
FUNCIONAMIENTO	11
¿POR QUÉ SE USA?	13
VARIANTES	13
CONCLUSIONES	18
BIBLIOGRAFÍA	20

ÍNDICE DE FIGURAS

Figura 1. Ordenador FERMIAC.	4
Figura 2. Estimación del número π con el método <i>Monte Carlo</i> .	6
Figura 3. Estimación de π con 10 experimentos.	7
Figura 4. Estimación de π con 100 experimentos.	7
Figura 5. Estimación de π con 1.000 experimentos.	8
Figura 6. Estimación de π con 10.000 experimentos.	8
Figura 7. Resultado de la estimación del precio de un activo durante un mes.	9
Figura 8. Resultado del modelo <i>Nagel-Schreckenberg</i> .	10
Figura 9. Planificación de la duración de las tareas del proyecto.	12
Figura 10. Resultados obtenidos con el método <i>Monte Carlo</i> .	13
Figura 11. Cálculo de la distribución a posteriori en inferencia bayesiana.	15
Figura 12. Cadena de Markov.	15
Figura 13. Cálculo del punto que se añadirá a la cadena.	16
Figura 14. Estimación de la distribución a posteriori de por el método de MCMC.	17
Figura 15. Comparación entre la distribución estándar y la secuencia de Sobol.	18

RESUMEN

El algoritmo de *Monte Carlo* es un método estadístico que surgió para solventar los inconvenientes de aplicar métodos determinísticos a problemas caracterizados por la incertidumbre y por la influencia de variables aleatorias. Con el paso del tiempo, se ha continuado estudiando y ampliando este método hasta conseguir integrarlo en una gran variedad de aplicaciones reales, además de desarrollar múltiples variantes con el objetivo de extender más aún su uso y mejorar su precisión. Es, por todo ello, uno de los algoritmos más conocidos en el ámbito de la estadística y es comúnmente utilizado en el modelado de sistemas complejos, en los que la gestión de los riesgos y el cálculo de las probabilidades para cada posible resultado son requisitos indispensables para su correcta predicción.

INTRODUCCIÓN

Desde que comenzó la era digital los ordenadores han ido evolucionando considerablemente, mejorando sus prestaciones y convirtiéndose en herramientas de estudio con un gran potencial. Esto ha derivado en el uso de este tipo de dispositivos para, entre otros, resolver problemas matemáticos pertenecientes a diversas áreas, como por ejemplo la estadística. Su velocidad de cómputo y la optimización de la gran mayoría de algoritmos facilita el cálculo de un conjunto masivo de probabilidades con el objetivo de obtener modelos predictivos que sean capaces de proporcionar información útil para posteriormente tomar decisiones. Con ellos se pueden considerar todas las posibilidades existentes así como el grado asociado a su cumplimiento.

Si bien el tipo de variables que se encuentran en un problema dependen de su naturaleza particular en la gran mayoría de los casos se les puede asociar un valor estimado para comenzar a trabajar sobre el modelo predictivo. Sin embargo, existen una amplia variedad de aplicaciones en las que intervienen una serie de parámetros cuantitativos aleatorios de los que desconocemos su valor. Un ejemplo de ello puede ser el tráfico existente en una determinada carretera dada una hora. Esta variable puede tener unos valores muy diferentes dependiendo del contexto.

Este tipo de problemas en los que se encuentra muy presente la incertidumbre no son abordables mediante algoritmos tradicionales puesto que no han sido diseñados para crear modelos predictivos caracterizados por esta propiedad. Asimismo proporcionar un valor cualquiera a este tipo de variables cuantitativas no es la mejor solución para

comenzar a trabajar en la definición de un sistema de predicción. Es por ello por lo que se han diseñado algoritmos como *Monte Carlo*, que es capaz de calcular el posible rango de valores para este tipo de variables así como su probabilidad asociada. Este algoritmo ayuda a entender el impacto que tendrían los riesgos asociados a un determinado valor así como tratar la incertidumbre a la hora de realizar las predicciones [1][2][3]. En este trabajo se profundizará más acerca de su funcionamiento, sus variantes así como las diversas aplicaciones de la vida real en las que se utiliza este algoritmo en particular.

ORIGEN

Una primera variante del método de *Monte Carlo* fue diseñada a través del denominado experimento *La aguja de Buffon*, con el cual se pretendía estimar la constante π mediante el lanzamiento, de manera aleatoria, de agujas de longitud igual a la distancia entre estas las líneas paralelas que componían un plano. El objetivo era calcular la probabilidad de que cada aguja lanzada fuese capaz de intersectar una línea [4].

Si bien los fundamentos del algoritmo ya habían sido definidos, aún no se conocía como tal. Sin embargo, se siguió experimentando con él aplicándolo, en este caso, al ámbito de la física nuclear. Fue *Enrico Fermi*, uno de los colaboradores del proyecto *Manhattan* cuyo objetivo consistía en construir bombas atómicas [6], quien en los años treinta se dedicó a definir este método estadístico para estudiar el movimiento de los neutrones. Para ello se fabricó un ordenador analógico capaz de repetir un mismo experimento consistente en generar y estudiar trayectorias de neutrones dirigidos hacia un material con el objetivo de traspasarlo.



Figura 1. Ordenador FERMIAC.

Pese a que existen registros de que fue *Fermi* quien determinó el método estadístico *Monte Carlo* como no realizó ningún tipo de publicación no quedó constancia de este invento hasta que el matemático *Stanislaw Ulam*, otro de los colaboradores del proyecto

Manhattan, se le ocurrió la siguiente idea. En ella se reflexionaba sobre la probabilidad de finalizar el juego de cartas *el solitario* de forma satisfactoria. Al principio intentó resolverlo de forma manual proponiendo como solución la repetición simultánea de este juego y el posterior conteo de las partidas que terminaban satisfactoriamente. Sin embargo, la era digital ya había comenzado y los ordenadores disponían de una mayor capacidad de cómputo, lo que los convertía en potentes herramientas para desarrollar modelos de predicción en base a un conjunto de probabilidades.

Fue entonces cuando esta nueva idea fue aplicada en el proyecto mencionado anteriormente para realizar un estudio de la trayectoria de los neutrones así como de la distancia a la que colisionaban explicando, de este modo, el proceso de fisión nuclear que podría originar una arma nuclear.

En 1946 con la colaboración de *John von Neumann*, fueron capaces de definir un modelo digital que explicaba la teoría de la difusión de neutrones en un material fisionable mediante la repetición de experimentos cuya base era el muestreo estadístico. Este modelo consistía en producir varias generaciones de neutrones con el objetivo de estudiar sus trayectorias dirigidas para atravesar un determinado material. Para cada generación se establecen las probabilidades asociadas a cada factor que influye en este experimento, como por ejemplo la velocidad del neutrón, y se realizaban tantas repeticiones como fuesen necesarias para obtener suficientes datos con los que ajustar todas las variables pertinentes.

En aquella época el desarrollo y aplicación de este método probabilístico de forma digital fue un gran avance en tanto en cuanto simplificaba de forma considerable realizar cálculos sumamente complejos como el consistente en estimar la velocidad de los neutrones, así como considerar los posibles efectos termodinámicos de un proceso físico tan enrevesado como es la fisión [5] [7].

Tras los buenos resultados proporcionados por este algoritmo, pronto se extendió su aplicación a otros ámbitos de muy diversa naturaleza, tales como finanzas, modelado del tráfico, entre otros, algunos de los cuales se detallarán en mayor profundidad en la siguiente sección. Sin embargo, el origen de su nombre proviene de la popularidad de los juegos de azar en la ciudad de Montecarlo, en los que era fundamental conocer las probabilidades asociadas a ganar o perder la apuesta con el objetivo de maximizar los beneficios [1].

APLICACIONES

Una de las aplicaciones más conocidas de este método consiste en **estimar el número π** especificando el número de decimales que deseamos encontrar. Para ello se utiliza la definición formal de esta constante, la cual indica que el número π representa la relación existente entre la longitud de la circunferencia y su propio diámetro. Asimismo, considerando que esta constante dispone de un número infinito de decimales, de los que no se ha podido encontrar ningún patrón para generarlos, el método de *Monte Carlo* es una buena herramienta para aplicar en este problema. El planteamiento de este método estadístico consiste en realizar varias partidas de dardos para, posteriormente, contar el número de dardos que se encuentran en la diana [2].

Matemáticamente, realmente se trata de un plano 2D de una unidad de longitud en el que se encuentra un círculo concéntrico cuyo radio dispone de la misma longitud que el plano. A continuación se generan una cantidad masiva de números aleatorios, los cuales pueden resultar dentro del círculo o fuera. Este planteamiento se ilustra con la siguiente figura.

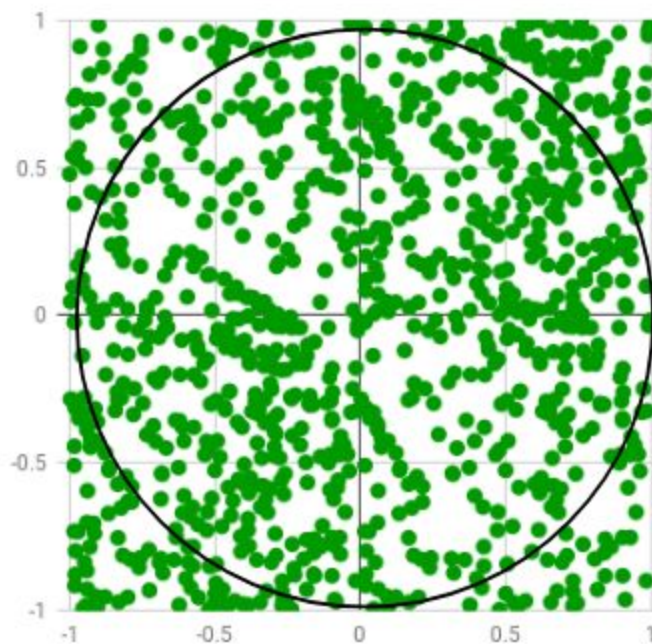


Figura 2. Estimación del número π con el método *Monte Carlo*.

A continuación se cuentan todos aquellos puntos que “han dado en la diana” y se realiza el siguiente razonamiento matemático en el que se parte de que el área del círculo que se

encuentra circunscrito dentro del plano es $\frac{\pi}{4}$ y por lo tanto

$$\frac{\text{Área del círculo}}{\text{Área del plano}} = \frac{N^\circ \text{ de puntos dentro del círculo}}{N^\circ \text{ total de puntos generados}} \Rightarrow \pi = 4 \cdot \frac{N^\circ \text{ de puntos dentro del círculo}}{N^\circ \text{ de puntos dentro del plano}}$$

Para conocer cuántos puntos se sitúan dentro del círculo basta con comprobar si sus coordenadas cumplen la siguiente condición $x^2 + y^2 \leq 1$. Como el método de *Monte Carlo* es probabilístico, cuantas más partidas de dardos se jueguen, más datos se obtienen para aumentar la precisión con la que calcula los decimales de esta constante [8]. Este hecho se puede ver reflejado en la siguiente captura en la que se grafican los resultados obtenidos tras 10, 100, 1.000 y 10.000 repeticiones [2].

Estimate of pi: 3.116

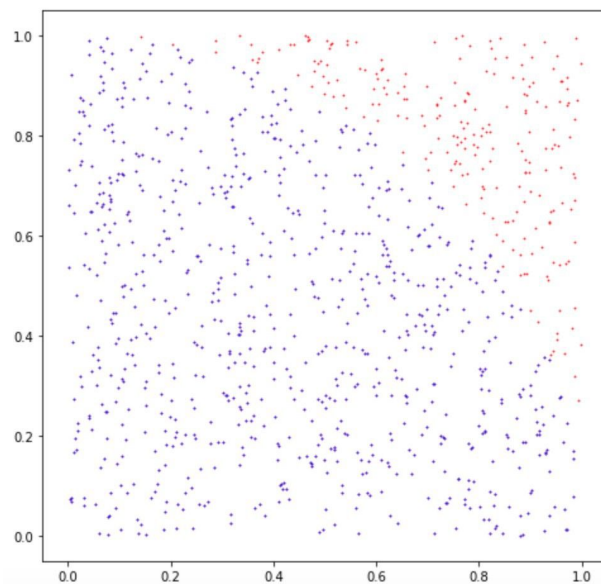


Figura 3. Estimación de π con 10 experimentos.

Estimate of pi: 3.142

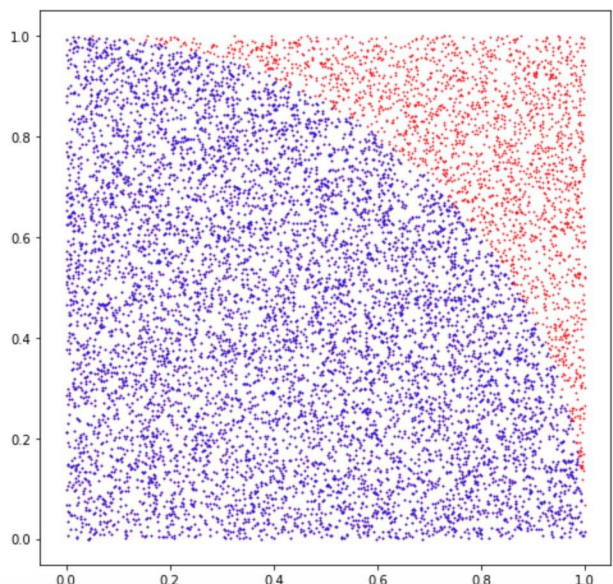


Figura 4. Estimación de π con 100 experimentos.

Estimate of pi: 3.13872

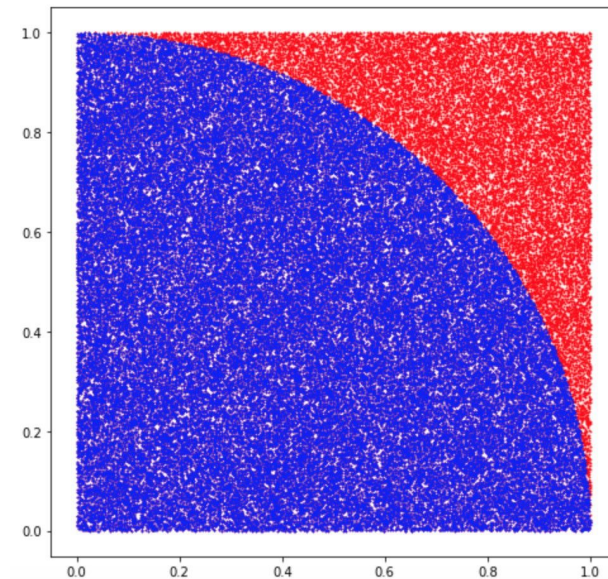


Figura 5. Estimación de π con 1.000 experimentos.

Estimate of pi: 3.141932

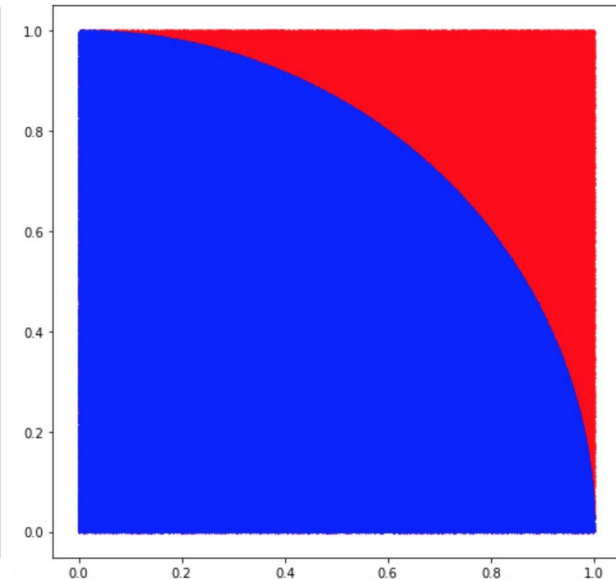


Figura 6. Estimación de π con 10.000 experimentos.

Otro de los ámbitos en los que se aplica este método estadístico es en el mundo **financiero**, en particular se utiliza para realizar predicciones en el mercado económico. Los buenos resultados proporcionados por este algoritmo en problemas cuya principal característica es la incertidumbre, lo ha convertido en una potente herramienta capaz de gestionar los riesgos ocultos tras las transacciones monetarias con el fin de conocer si estas son beneficiosas o no para un determinado cliente. El siguiente ejemplo relacionado con esta temática consiste en **estimar la trayectoria del precio de un producto** utilizando herramientas tan sencillas como *Excel* y considerando las variables que principalmente influyen en esta operación, como la volatilidad del mercado, la cual, además, es aleatoria.

A continuación, utilizando el histórico de precios del producto y los valores de las variables establecidos calculamos una serie de datos estadísticos, tales como la media, la desviación estándar, entre otros. Con toda información se podrá estimar el precio del activo para un día. Por lo tanto, basta con repetir este mismo proceso tantas veces como precios por día queramos predecir. Un caso práctico de este tipo de experimento consiste en calcular el valor de un activo para todo el mes [1].

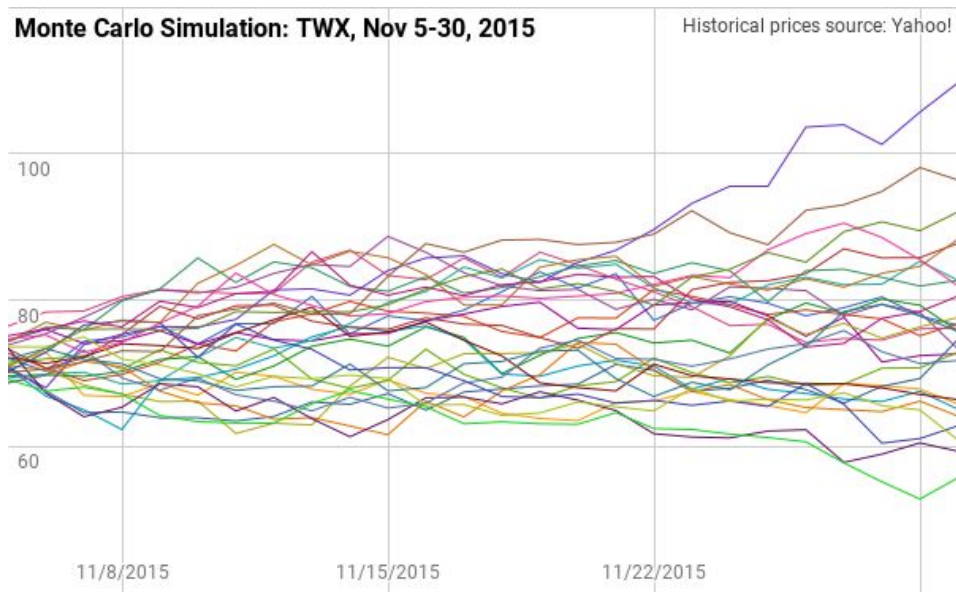


Figura 7. Resultado de la estimación del precio de un activo durante un mes.

Las probabilidades calculadas para cada uno de los experimentos conforman una distribución normal en forma de campana, por lo que los valores más probables se concentran en la mitad de esta. No obstante, hay que tener en cuenta que todos estos valores son estimados y que por tanto no existen garantías de que la trayectoria más probable sea la que finalmente ocurra [1].

Como tercera aplicación de este método se encuentra el **estudio del tráfico de vehículos**, en concreto, los atascos. Sus causas pueden ser de muy diversa naturaleza y los factores que influyen en este tipo de sucesos son completamente aleatorios y dependientes de las condiciones tanto humanas como materiales, como las características de los vehículos o el estado de las carreteras. Por ello, con el fin de comprender los desencadenantes que originan un atasco se utiliza la técnica de *Monte Carlo*, en particular se ha desarrollado el modelo *Nagel-Schreckenberg*.

Para comprender los fundamentos de este modelo, procedemos a explicar su versión más sencilla. En ella se establecen M zonas distintas en las que se divide una carretera circular y un vehículo solo podrá circular por una en concreto. Teniendo en cuenta que hay N vehículos y que cada uno lleva una velocidad V se asume que en el siguiente experimento un vehículo se ha desplazado por V zonas de la carretera.

Para cada experimento se realizarán cuatro pasos. El primero de ellos es incrementar la velocidad en una unidad hasta alcanzar la velocidad máxima, que todos los vehículos

respetan. Como segundo paso se deberá comprobar la distancia existente entre un vehículo y el que se encuentra delante. Si supera un determinado umbral, entonces reducirá la velocidad para no colisionar con el vehículo que le precede. En la tercera fase se proporciona cierta aleatoriedad disminuyendo la velocidad actual según una probabilidad P . Por último, el vehículo se desplazará V unidades para completar el experimento actual. En resumen, este modelo trata de estimar la velocidad adecuada, teniendo en cuenta los factores comentados anteriormente, y a partir de ella efectuará su desplazamiento para repetir de nuevo el mismo experimento partiendo de la posición calculada. A continuación se muestra el resultado de ejecutar 100 veces el experimento detallado anteriormente con el modelo *Nagel-Schreckenberg*.

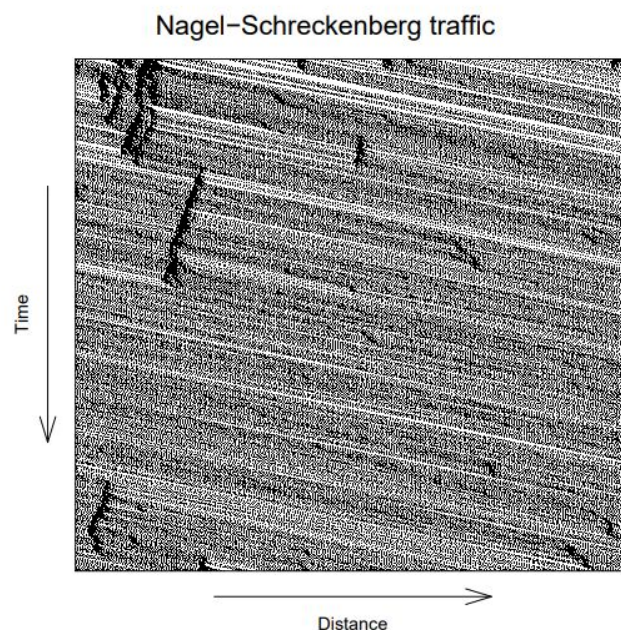


Figura 8. Resultado del modelo *Nagel-Schreckenberg*.

Tal y como se puede apreciar, las zonas resaltadas en negro representan los atascos que se producen al principio, cuando aún los parámetros influyentes no disponen de los valores correctos para predecir este tipo de situaciones y actuar para que no ocurran. Sin embargo, podemos comprobar que conforme los vehículos recorren más distancia estos sucesos remiten originando un tráfico más fluido puesto que el algoritmo es capaz de predecir el recorrido apropiado para cada vehículo dependiendo de sus condiciones y de las del tráfico que le rodea.

Tanto este modelo como otros han sido desarrollados para adaptarse a la complejidad que entraña el tráfico en la vida real con el objetivo de estimar el impacto que tendrían

medidas tales como añadir un carril más a una carretera, cortar una calle, entre otras. Asimismo, estos modelos más elaborados son capaces de realizar sus predicciones teniendo en cuenta una serie de cualidades específicas para cada vehículo, como el destino al que se dirige y desde donde inicia su camino [9].

FUNCIONAMIENTO

En esta sección se detalla el funcionamiento del método *Monte Carlo* en su versión más primitiva para entender los conceptos básicos con el objetivo de explicar, en el siguiente capítulo, algunas de las variantes más populares. Esta versión clásica se conoce como el **algoritmo de simulación de Monte Carlo Crudo o Puro**. En este primer enfoque se presume que las variables no están relacionadas, y por lo tanto, esta primera versión se aplica realizando los siguientes pasos:

1. En primer lugar deberemos determinar las variables aleatorias del modelo que queremos estudiar y sus distribuciones acumuladas a partir de la probabilidad de ocurrencia de cada una de las clases.
2. Como segundo paso inicializamos las variables aleatorias identificadas generando valores al azar, comúnmente situados en el rango entre 0 y 1.
3. Calculamos valores estadísticos como la media, varianza, error, entre otras, y dibujamos el histograma de las clases. Esto último nos permitirá deducir cuál es la distribución de probabilidad del modelo que estemos estudiando.
4. Por último se realizará un análisis de los resultados obtenidos para cada una de las muestras participantes.

Tanto el proceso número dos como el tres se deberán repetir el número de veces necesarias para obtener un conjunto de resultados suficientemente amplio como para definir un modelo predictivo caracterizado por una buena capacidad de predicción y que disponga de un alto grado de precisión.

No obstante, como todo algoritmo se deben tener en cuenta un conjunto de aspectos que influyen en la obtención de resultados. Los principales se detallan a continuación:

- El sistema que se intenta aprender debe ser estimado por más de una función de distribución probabilística, lo que se traduce a que se deben realizar varios experimentos para definir los valores que influyen en el sistema de manera precisa.
- La forma de generar los números aleatorios con los que al comienzo inicializamos las variables también influyen puesto que se pueden generar valores

correlacionados y con ello establecer relaciones entre variables que realmente no se encuentran asociadas.

- Del mismo modo se deben establecer un rango particular para cada una de las variables del sistema con el objetivo de conocer qué conjunto de valores puede tomar. Esta condición está directamente relacionada con la anterior puesto que el valor aleatorio generado para una variable en particular deberá de pertenecer al rango de esta.
- Paralelización de procesadores para realizar una simulación en la que intervienen cantidades masivas de variables con el objetivo de optimizar el tiempo computacional del algoritmo [10].

Un ejemplo de la necesidad de aplicar este algoritmo es el siguiente. Supongamos que estamos trabajando en la planificación de un proyecto informático y realizamos un desglose de las actividades que debemos llevar a cabo para obtener el producto final. Para conocer la duración de cada una de las tareas asociamos la duración particular para el mejor caso, el peor y el más probable. Los resultados obtenidos se pueden observar en la siguiente captura.

Activity	Optimistic	Most Likely	Pessimistic
A	8	10	12
B	10	12	14
C	12	14	16
Total	30	36	42

Figura 9. Planificación de la duración de las tareas del proyecto.

Tal y como se puede comprobar la estimación de la duración total del proyecto es de unos 30 días en el mejor de los casos, de 36 días en el más probable y de 42 días en la peor situación. Sin embargo, no conocemos la probabilidad de ocurrencia para cada una de las diferentes posibilidades. Para calcularlas podemos aplicar el algoritmo de *Monte Carlo* del siguiente modo. En primer lugar definimos el rango de duración para las tres tareas que debemos realizar en el proyecto. Para la tarea *A* se le asigna un rango entre 8 y 12 días, para la tarea *B* el rango será de entre 10 y 14 días, y para la última tarea se le asignará un rango de valores entre 12 y 16 días de duración.

Una vez hemos asignado el rango de probabilidades para cada una de las tareas del proyecto se realizan diversas simulaciones del algoritmo *Monte Carlo*, que en este ejemplo han sido 300, y se obtienen los siguientes resultados.

Duration (in days)	Likelihood of Completion
30 - 32	7%
33 - 35	36%
36 - 38	42%
39 - 42	15%

Figura 10. Resultados obtenidos con el método *Monte Carlo*.

Si analizamos los resultados obtenidos podemos comprobar cómo el método ha sido capaz de proporcionar la probabilidad asociada a cada una de las posibles duraciones para el proyecto, siendo la más probable la asociada a terminarlo entre 36 y 38 días. De este modo, como gestor del proyecto, te permite conocer cuál es el período más probable sobre el que se terminará el proyecto con el objetivo de tomar mejores decisiones para ajustar una planificación realista [11].

¿POR QUÉ SE USA?

- Este algoritmo se caracteriza por simplificar el modelado de sistemas sumamente complejos a un proceso de iteración consistente en realizar varios experimentos con el objetivo de asociar una probabilidad a cada una de las posibles salidas existentes considerando cada parámetro influyente.
- Es fácilmente paralelizable si existe la posibilidad de ejecutar, por separado, cada uno de los procesos que influyen en un experimento. De este modo se reduce considerablemente el tiempo de computación invertido.
- Es un método capaz de predecir el comportamiento de variables aleatorias que influyen en el funcionamiento de un sistema incorporando una probabilidad para cada una de las posibles salidas del sistema además de considerar los posibles valores que pueden llegar a alcanzar estos parámetros.
- Al ser un método principalmente estadístico, dispone de un conjunto de teorías matemáticas y estadísticas que respaldan su funcionamiento, variantes y resultados [3].

VARIANTES

Dentro del método de Monte Carlo podemos encontrar diversas variantes, todas ellas

basadas en el algoritmo básico descrito anteriormente. Dos de ellas son la del método de Monte Carlo por cadenas de Markov y el Quasi Monte Carlo.

Monte Carlo por cadenas de Markov

Una de las variantes más relevantes del método de Monte Carlo viene dado en el uso de las cadenas de Markov. Pero antes de esto, para explicar cuál es el verdadero funcionamiento y el porqué del uso de las cadenas de Markov deberemos introducir brevemente el concepto de **estadística bayesiana**.

La estadística bayesiana es un tipo de inferencia estadística la cual se basa en que, a partir de una suposición previa que llamaremos suposición a *priori* y una predicción de lo que es más probable que ocurra, estima qué es lo que probablemente ocurrirá a continuación, a lo que llamaremos suposición a *posteriori*. En caso de la estimación a *priori*, se trata de una estimación que realizaremos sin tener en cuenta ningún tipo de dato observado. Por otro lado la estimación de lo más probable vendrá dada por los datos estadísticos que recojamos. Y por último, la estimación a *posteriori* de lo que es más probable que ocurra en el futuro proviene de la conjunción de ambas probabilidades. Otra característica de este tipo de inferencias estadísticas consiste en que la probabilidad de que ocurra algo no se encuentra descrita por una probabilidad en concreto, sino que se trata de una distribución de probabilidades.

Como podemos imaginar el método de Monte Carlo descrito anteriormente no puede ser usado para este tipo de sistemas y es que en el caso anterior tratábamos sistemas en los cuales las variables aleatorias no estaban correlacionadas entre sí.

Este modelo, es fácilmente resoluble si tanto la distribución a *priori* como la distribución más probable tienen ambas una distribución normal, en tal caso, la distribución a *posteriori* vendrá dada como una conjunción de las dos distribuciones, tal y como se puede ver en la imagen siguiente:

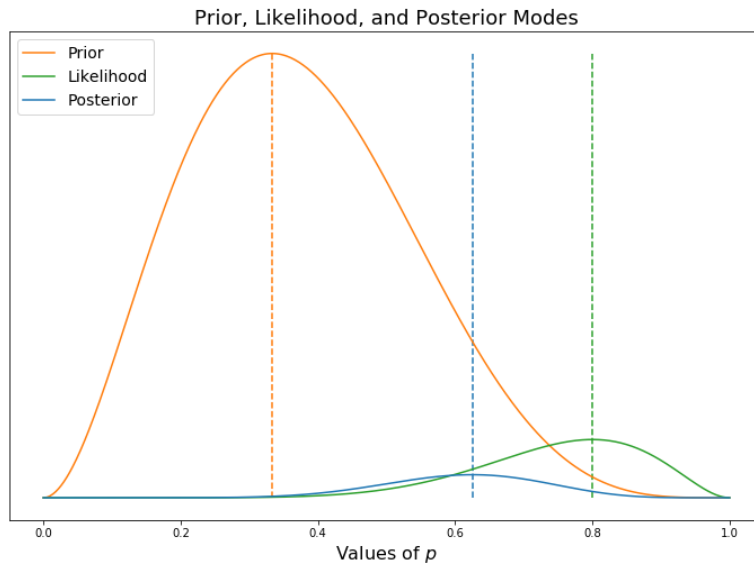


Figura 11. Cálculo de la distribución a posteriori en inferencia bayesiana.

Sin embargo, para distribuciones distintas a la normal, el cálculo de la situación a posteriori no será tan inmediata y será mucho más compleja de resolver. Es en este punto donde entran tanto las cadenas de Markov [12] como el método de Monte Carlo.

Dado que el método de Monte Carlo fue descrito anteriormente, empezaremos hablando de las **cadenas de Markov**. Estas cadenas, simplificandolas enormemente, son un mecanismo de estudio probabilístico en el que la posibilidad de que ocurra un evento depende del conjunto de estados de los eventos anteriores. De igual forma, la probabilidad de que se presente un estado siguiente vendrá dado por los estados actuales del modelo que se esté estudiando. Todas estos estados, vendrán dados además de por los estados anteriores, por una serie de probabilidades asociadas a cada uno. Una representación de las cadenas de Markov se puede ver a continuación:

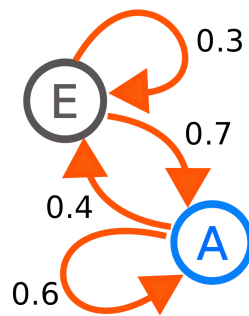


Figura 12. Cadena de Markov.

Un ejemplo de las cadenas de Markov y también el más conocido, lo propuso el propio Markov mientras intentaba estudiar las probabilidades de ocurrencia de sucesos no aleatorios. Esto lo hizo mediante el estudio de miles de pares de letras de un libro de poesía ruso, en el que fue capaz de definir un modelo para calcular la probabilidad de ocurrencia de una letra en función de la letra anterior. De esta forma fue capaz de crear largas cadenas de caracteres, las cuales seguían una distribución acorde a la media demostrando que a largo plazo estas cadenas seguían un patrón.

Este tipo de cadenas puede parecer que no tienen mucha utilidad pero tras varias iteraciones (separándolos de los valores iniciales, los cuales no son útiles por ser especificados de forma aleatoria) seremos capaces de determinar cuál es el estado más probable de nuestro modelo, aportando ahora sí, una información muy valiosa.

Una vez explicado esto, podemos intuir la relación entre la inferencia bayesiana y las cadenas de Markov. Llegados a este punto podemos especificar cuál es la metodología usada por esta técnica. Primero, está la parte del algoritmo de Monte Carlo, la cual generará números aleatorios sujetos a una determinada regla, la cual nos permitirá definir qué números de un par de valores generados aleatoriamente explican mejor nuestro modelo *a priori*.

De esta forma si obtenemos un valor que se ajusta mejor a nuestro modelo, este será añadido a una cadena de parámetros junto con una probabilidad, que representará el nivel de adaptación de este valor a nuestro modelo con respecto al anterior. Una vez se han computado suficientes muestras aleatorias, se realiza un histograma sobre las concentraciones de los puntos más frecuentes y de esta forma podremos obtener la distribución *a posteriori* de nuestro modelo. Un ejemplo de esto se puede ver en las siguientes imágenes:

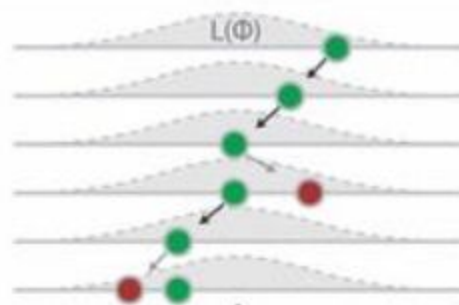


Figura 13. Cálculo del punto que se añadirá a la cadena.

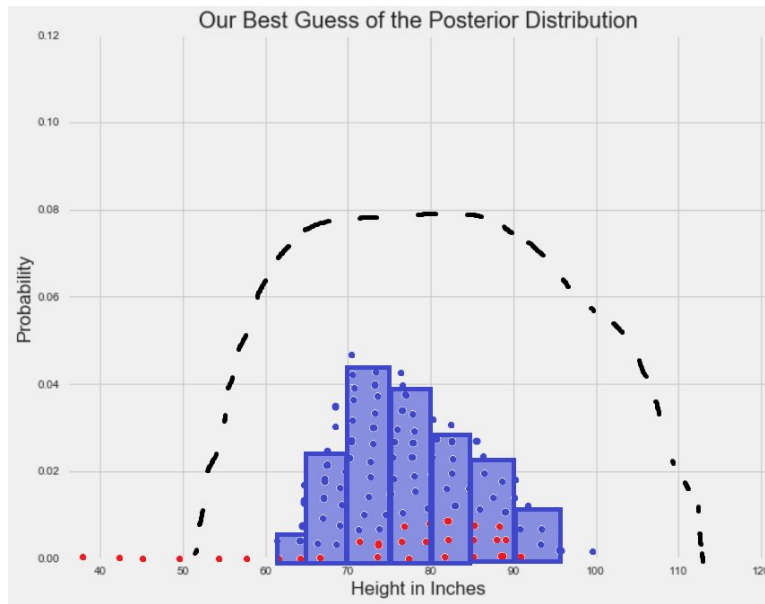


Figura 14. Estimación de la distribución a posteriori de por el método de MCMC.

En las dos imágenes anteriores podemos ver ejemplificado el proceso. En la primera de ellas podemos ver cómo se computan dos puntos aleatorios. Uno de ellos, será el que se añadirá a la cadena de dígitos, que será el que más se ajuste a la distribución a *priori*, tal y como se explicó anteriormente. Por otro lado, en la segunda imagen tenemos que una vez se han computado varios de estos puntos se realiza un histograma de la distribución de estos, obteniendo a partir de ahí el mejor modelo posible a *posteriori*.

Quasi Monte Carlo

Es otra de las variantes de este método estadístico, la cual se encuentra principalmente enfocada a la integración numérica, es decir, uno de los problemas sobre el que se puede aplicar esta variante consiste en aproximar la integral de una función a través de la media de los valores resultantes de evaluar cada una de las funciones que componen la integral en un conjunto de puntos. La principal diferencia entre el método estándar y esta variante reside en la generación de estos puntos, ya que *Quasi Monte Carlo* utiliza **secuencias de baja discrepancia** [15], en la cual los valores también se generan al azar pero estableciendo esta última condición.

Algunas de las técnicas que se pueden utilizar para aplicar esta metodología son *Halton sequence* o **Sobol sequence**. Este proceso en particular genera cada uno de los números

aleatorios teniendo en cuenta la posición de los valores anteriormente generados, con el objetivo de obtener una distribución más equilibrada que si los muestreamos aleatoriamente mediante el método tradicional [19]. Añadiendo esta nueva cualidad podemos aumentar la velocidad de convergencia del algoritmo [16][17]. Un ejemplo representativo de la principal diferencia comentada se muestra en la siguiente captura.

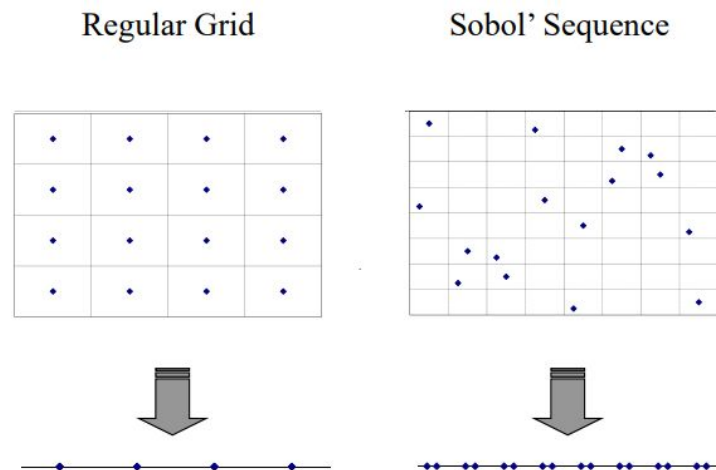


Figura 15. Comparación entre la distribución estándar y la secuencia de Sobol.

Una de las principales aplicaciones de esta variante consiste en modelar problemas financieros mediante integrales de gran dimensión. La principal razón de ello reside en que casi todos los planteamientos económicos se pueden representar mediante una función de pago que genera valores cuantitativos, como puede ser la definición de un modelo centrado en el precio de acciones.

La dimensión de la integral con la cual se realiza el planteamiento de un aspecto financiero concreto depende directamente del número de pasos en la discretización así como el número de movimientos *brownianos*. Este término es comúnmente utilizado tanto en el ámbito de la física como el que nos ocupa actualmente y representa el comportamiento de los parámetros aleatorios en el tiempo. En este caso concreto, podría representar, por ejemplo, la fluctuación del precio de un activo [18].

CONCLUSIONES

Como hemos visto en este trabajo, el método de *Monte Carlo* es un método estadístico que es capaz de aportar estimaciones sobre un sistema en concreto, aumentando su precisión mediante la repetición masiva de un experimento particular en los que influye un conjunto de parámetros aleatorios sobre los que estimar su valor con el objetivo de

definir el modelo que queremos calcular.

Esto tiene otro tipo de aplicaciones más complejas, aparte de las que hemos explicado anteriormente, pudiendo variar estas enormemente. Un ejemplo de ello se puede ver en el método del ***Quantum Monte Carlo***, el cual usando diferentes metodologías basadas en el algoritmo de *Monte Carlo* pretende resolver la ecuación de *Schrodinger* para el problema de los n-cuerpos [19]. Otro ejemplo podría ser el cálculo de integrales multidimensionales computacionalmente complejas [20] mediante este método, y es que aunque no permita un cálculo exacto, aporta una aproximación más que aceptable si aportamos los suficientes números aleatorios.

Asimismo este método también es utilizado en programas de modelado gráfico y animación por ordenador como *3dsMax*, con el objetivo de realizar diversas simulaciones utilizadas para diversos propósitos, como el estudio de células microscópicas [22].

Por lo tanto, a través de la realización de este trabajo hemos podido comprender los beneficios que tiene este modelo en diferentes ámbitos de la ciencia. Siendo como no, uno de estos ámbitos la informática y el tratamiento de la información. Y es que una de las funciones que tiene este método es la estimación del estado de un sistema en un momento futuro. Y si bien, antiguamente, se podía calcular para ciertos problemas, actualmente, considerando la potencia de cómputo de los ordenadores, y la complejidad que han adquirido los cálculos necesarios para realizar las estimaciones de las diversas variables influyentes en un problema, es necesario el uso de este tipo de técnicas de predicción, que ofrecen tanto unos buenos resultados como un rendimiento excelente en comparación con otras técnicas.

BIBLIOGRAFÍA

1. Will Kenton, Monte Carlo Simulation Definition, 2019,
<https://www.investopedia.com/terms/m/montecarlosimulation.asp>
2. Christopher Pease, An Overview of Monte Carlo Methods, 2018
<https://towardsdatascience.com/an-overview-of-monte-carlo-methods-675384eb1694>
3. Dirk P. Kroese, Tim Brereton, Thomas Taimre, Zdravko I. Botev, *Why the Monte Carlo method is so important today*,
https://people.smp.uq.edu.au/DirkKroese/ps/whyMCM_final.pdf
4. Maximilian J. Wang, Jin Wang, *Removing the inherent paradox of the Buffon's needle Monte Carlo simulation using fixed-point iteration method*,
<https://informs-sim.org/wsc14papers/includes/files/327.pdf>
5. N. Metropolis, *The beginning of the Monte Carlo method*,
<https://permalink.lanl.gov/object/tr?what=info:lanl-repo/lareport/LA-UR-88-9067>
6. Atomic Heritage Foundation, *The Manhattan Project*, 2017,
<https://www.atomicheritage.org/history/manhattan-project>
7. Roger Eckhardt, *Stan Ulam, John Von Neumann, and the Monte Carlo Method*,
http://www-star.st-and.ac.uk/~kw25/teaching/mcrt/MC_history_3.pdf
8. GeeksforGeeks, *Estimating the value of Pi using Monte Carlo*,
<https://www.geeksforgeeks.org/estimating-value-pi-using-monte-carlo/>
9. Art B. Owen, "Monte Carlo theory, methods and examples", 2013,
<https://statweb.stanford.edu/~owen/mc/>
10. Universidad nacional del centro de la provincia de buenos aires, "Simulación, Método Monte Carlo",
<https://jaimesotou.files.wordpress.com/2011/05/metodo-montecarlo-03.pdf>
11. Sonya Siderova, *Navem Quantifying the Uncertainty: Monte Carlo Simulation*, 2018
<https://getnave.com/blog/monte-carlo-simulation/>
12. Sánchez G. Javier, "Cadenas de Markov",
<https://economipedia.com/definiciones/cadena-de-markov.html>

13. Mauri Kathan, “Markov chain Monte Carlo”,
<https://towardsdatascience.com/markov-chain-monte-carlo-291d8a5975ae>
14. Ben Shaver, “A zero-math Introduction to markov chain Monte Carlo Methods”,
2016
<https://towardsdatascience.com/a-zero-math-introduction-to-markov-chain-monte-carlo-methods-dcba889e0c50>
15. Sergei Kucherenko Imperial College London UK Imperial College London,
Application of Quasi Monte Carlo methods in finance,
http://www.broda.co.uk/doc/Aplication_QMC_Finance.pdf
16. Frances Y. Kuo, Dirk Nuyens, *A practical guide to Quasi-Monte Carlo Methods*
<https://people.cs.kuleuven.be/~dirk.nuyens/taiwan/QMC-practical-guide-20161107-1up.pdf>
17. Jorgen Veisdal, Medium, *Brownian Motion in Financial Markets*, 2012
<https://medium.com/cantors-paradise/brownian-motion-in-financial-markets-ea5f02204b14>
18. Antoine Savine, Medium, *Sobol Sequence Explained by Antoine Savine*, 2018,
https://medium.com/@antoine_savine/sobol-sequence-explained-188f422b246b
19. Florian Hartig, *A simple Metropolis-Hastings MCMC in R*,
<https://theoreticalecology.wordpress.com/2010/09/17/metropolis-hastings-mcmc-in-r/>
20. Fabien Alet, Univ. Paul Savatier Toulouse, *A quick introduction to Quantum Monte Carlo methods*, <http://alps.comp-phys.org/mediawiki/images/d/d8/QMC.PSI.pdf>
21. Patricia Saavedra Barrera, Departamento de Matemáticas, Universidad Autónoma Metropolitana-Iztapalapa, *Integración numérica por Monte-Carlo*,
http://sgpwe.izt.uam.mx/pages/cbi/psb/anatum/material_adicional/mcarlo.pdf
22. National Center of Multiscale Modeling of Biological System, *Cell Modeling*,
<https://mmbios.pitt.edu/research/technology-research-and-development/cell-modeling>