

Random machines

How algorithms emulate randomness

Jean-Marc Vincent¹

¹Laboratoire LIG
Jean-Marc.Vincent@imag.fr

Flip a coin with a computer

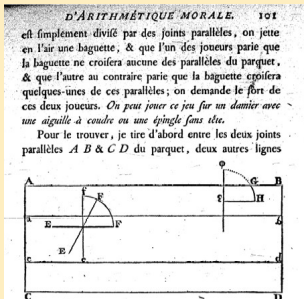
Outline of the lecture

- 1 Random machines**
 - Why generate random numbers ?
 - Random machines
 - Pseudo-random generators

- 2 and Human mind**
 - Randomness detection
 - Generate randomness

Randomized numerical algorithms

Buffon's needle



parallèles *ab* & *cd*, éloignées des premières de moitié de la longueur de la baguette *EF*, & je vois évidemment que tant que le milieu de la baguette sera entre ces deux secondes parallèles, jamais elle ne pourra croîer les premières dans quelque situation *EF*, *ef*, qu'elle puisse se trouver; & comme tout ce qui peut arriver au-dessus de *ab* arrive de même au-dessous de *cd*, il ne s'agit que de déterminer l'un ou l'autre; pour cela je remarque que toutes les situations de la baguette peuvent être

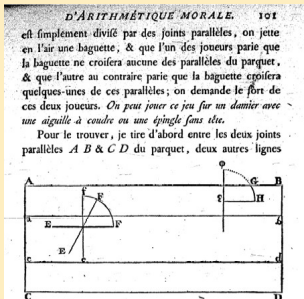
Model

$$\mathbb{P}(X + l \sin \alpha \leq a) = \frac{\int_0^\pi (a - l \sin \alpha) d\alpha}{\pi a} = \frac{2l}{\pi a}$$

Computation of π by repeating experiments
 → Monte-Carlo methods
 Computation of integrals (high dimension)

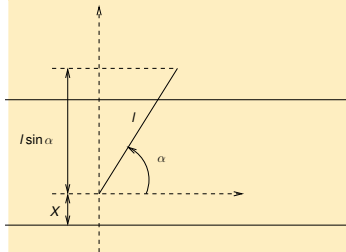
Randomized numerical algorithms

Buffon's needle



parallèles *ab* & *cd*, éloignées des premières de la moitié de la longueur de la baguette *EF*, & je vois évidemment que tant que le milieu de la baguette fera entre ces deux secondes parallèles, jamais elle ne pourra croîer les premières dans quelque situation *EF*, *ef*, qu'elle puisse se trouver; & comme tout ce qui peut arriver au-dessus de *ab* arrive de même au-dessous de *cd*, il ne s'agit que de déterminer l'un ou l'autre; pour cela je remarque que toutes les situations de la baguette peuvent être

Model



X de loi uniforme sur $[0 : l]$

α de loi uniforme sur $[0, \pi]$

a

$\mathbb{P}(X + l \sin \alpha \geq a)$

$$\mathbb{P}(X + l \sin \alpha \leq a) = \frac{\int_0^\pi (a - l \sin \alpha) d\alpha}{\pi a} = \frac{2l}{\pi a}$$

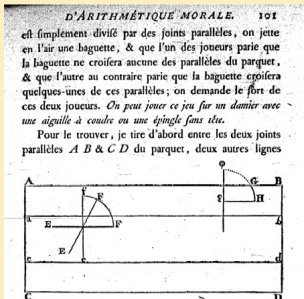
Computation of π by repeating experiments

→ Monte-Carlo methods

Computation of integrals (high dimension)

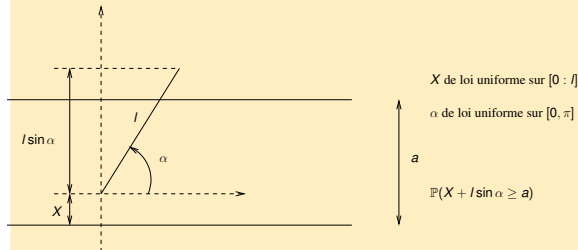
Randomized numerical algorithms

Buffon's needle



parallèles ab & cd , éloignées des premières de la moitié de la longueur de la baguette EF , & je vois évidemment que tant que le milieu de la baguette fera entre ces deux secondes parallèles, jamais elle ne pourra croîer les premières dans quelque situation EF , ef , qu'elle puisse se trouver; & comme tout ce qui peut arriver au-dessus de ab arrive de même au-dessous de cd , il ne s'agit que de déterminer l'un ou l'autre; pour cela je remarque que toutes les situations de la baguette peuvent être

Model



$$\mathbb{P}(X + l \sin \alpha \leq a) = \frac{\int_0^\pi (a - l \sin \alpha) d\alpha}{\pi a} = \frac{2l}{\pi a}$$

Computation of π by repeating experiments

→ Monte-Carlo methods

Computation of integrals (high dimension)

Georges Louis Leclerc Comte de Buffon (1707-1788)



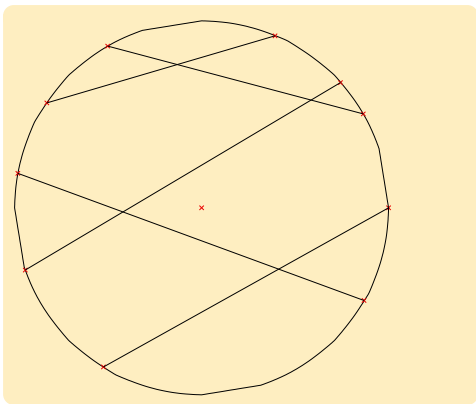
Georges-Louis Leclerc, comte de Buffon (7 septembre 1707 à Montbard - 16 avril 1788 à Paris), est un naturaliste, mathématicien, biologiste, cosmologiste et écrivain français. Ses théories ont influencé deux générations de naturalistes, parmi lesquels notamment Jean-Baptiste de Lamarck et Charles Darwin. La localité éponyme Buffon, dans la Côte-d'Or, fut la seigneurie de la famille Leclerc.

Les premiers travaux de Buffon ont été consacrés aux mathématiques. Il faut surtout signaler le Mémoire sur le jeu de franc carreau, qui présente l'originalité de faire intervenir le calcul infinitésimal dans le calcul des probabilités. Par la suite, Buffon utilisera les mathématiques dans ses recherches sur la résistance du bois et sur le refroidissement des planètes, ainsi que dans son Essai d'arithmétique morale (Supplément, t. IV, 1777), mais ces travaux montrent que, pour lui, les mathématiques ne sont qu'un moyen de préciser l'idée qu'il peut avoir des choses, et non une discipline autonome. Il est ingénieur plus que mathématicien.

Par contre, il est philosophe de tempérament. Le tome I de l'Histoire naturelle (1749) s'ouvre par un discours De la manière d'étudier et de traiter l'histoire naturelle, qui est une réflexion sur la valeur de la connaissance humaine. Rompant à la fois avec l'idéalisme rationaliste et l'empirisme sceptique, Buffon affirme la validité d'une science fondée sur les faits, mais sachant en dégager les lois, débarrassée de toute téléologie, d'une science qui sans doute ne vaut que pour l'homme, mais qui est la seule que l'homme puisse atteindre. Par la suite, Buffon admettra que l'homme peut découvrir les vraies lois de la nature (De la nature, 1re et 2e vues, Histoire naturelle, t. XII et XIII, 1764-1765). Son tempérament rationaliste l'emporte alors sur sa formation philosophique, d'inspiration sceptique.

Generation of geometrical objects

Joseph Bertrand : generate a random chord



Compute the probability that the length of the chord is greater than the length of the side of an equilateral triangle inscribed in the circle.

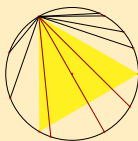
Alternatives

$$p = \frac{1}{2} \quad p = \frac{1}{3} \quad p = \frac{1}{4}$$

Generation of geometrical objects

Joseph Bertrand : generate a random chord

Circle



$$p = \frac{1}{3}.$$

Rays

$$p = \frac{1}{2}.$$

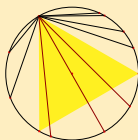
Disc

$$p = \frac{1}{4}.$$

Generation of geometrical objects

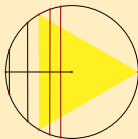
Joseph Bertrand : generate a random chord

Circle



$$p = \frac{1}{3}.$$

Rays



$$p = \frac{1}{2}.$$

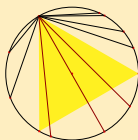
Disc

$$p = \frac{1}{4}.$$

Generation of geometrical objects

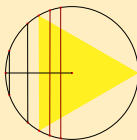
Joseph Bertrand : generate a random chord

Circle



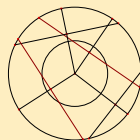
$$p = \frac{1}{3}.$$

Rays



$$p = \frac{1}{2}.$$

Disc



$$p = \frac{1}{4}.$$

Joseph Bertrand (1822-1900)



Joseph Louis François Bertrand, habituellement appelé Joseph Bertrand, né le 11 mars 1822 à Paris, mort le 3 avril 1900 à Paris, était un mathématicien, historien des sciences et académicien français.

Enfant prodige, à onze ans il suit les cours de l'École Polytechnique en auditeur libre. Entre onze et dix-sept ans il obtient deux baccalauréats, une licence et le doctorat ès sciences avec une thèse sur la théorie mathématique de l'électricité, puis est admis premier au concours d'entrée 1839 de l'École Polytechnique. Il est ensuite reçu au concours de l'agrégation de mathématiques des facultés et premier au premier concours d'agrégation de mathématiques des lycées avec Charles Briot, ainsi qu'à l'École des mines. Il fut professeur de mathématiques au lycée Saint-Louis, répétiteur, examinateur puis professeur d'analyse en 1852 à l'École polytechnique et titulaire de la chaire de physique et mathématiques au Collège de France en 1862 en remplacement de Jean-Baptiste Biot.

En 1845, en analysant une table de nombres premiers jusqu'à 6 000 000, il fait la conjecture qu'il y a toujours au moins un nombre premier entre n et $2n-2$ pour tout n plus grand que 3. Tchebychev a démontré cette conjecture, le postulat de Bertrand, en 1850.

Pour l'étude de la convergence des séries numériques, il mit au point un critère de comparaison plus fin que le critère de Riemann.

$$\sum \frac{1}{n^\alpha \log n^\beta} \text{ converge ssi } (\alpha, \beta) \geq (1, 1).$$

Outline of the lecture

- 1 Random machines**
 - Why generate random numbers ?
 - Random machines
 - Pseudo-random generators

- 2 and Human mind**
 - Randomness detection
 - Generate randomness

Physical instruments

Lottery

- Wheel, sectors
Dynamical amortized chaotic system
- Coins, dices
Chaotic system
- Bingo, roulette
- Card shuffling
repeated perturbed permutations

seed : human action

Machines

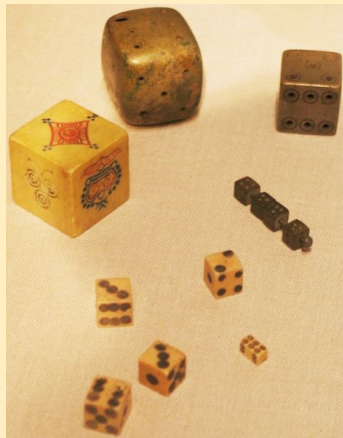
Physical instruments

Lottery

- Wheel, sectors
Dynamical amortized chaotic system
- Coins, dices
Chaotic system
- Bingo, roulette
- Card shuffling
repeated perturbed permutations

seed : human action

Machines



Pre-computed randomness

Rand corporation

Tables with random figures

TABLE OF RANDOM DIGITS 281

```

13000  31720 01273  20818 23248  66903 55238  72887 45155  43214 11849
13001  58971 74215  38773 26132  71727 21941  35303 70144  96569 88064
13002  82231 26139  04946 24667  78238 72766  41805 62206  57512 95123
13003  77346 73403  17182 45771  88843 81230  77118 22870  36126 75891
13004  78873 54500  82482 43730  34186 82778  12390 17114  72569 33784

13005  54233 80475  14686 71100  54283 41826  39952 10178  24692 78547
13006  87210 45684  55000 22878  82869 03192  96866 79801  84050 88751
13007  35977 82057  28434 83996  32874 27732  21866 28149  79258 08336
13008  01798 05027  16803 83295  24464 15327  34792 45274  26964 41283
13009  28687 32108  90542 03720  28884 65518  91983 16737  73430 00848

13010  94804 02297  22897 87687  68013 06657  01227 32046  80343 71107
13011  88487 22522  44274 70790  92271 14420  18770 36266  07202 18866
13012  44432 89730  24809 68228  88220 04781  33731 74739  72897 93390
13013  53126 00861  71284 46282  87522 98832  27128 38801  32863 89824
13014  30421 83366  20868 18839  11538 08733  50983 38275  89448 66605

13015  80669 17972  93222 42848  45247 44771  04393 05480  27082 05286
13016  74977 46460  10430 98540  22064 22468  82005 71738  10212 05245
13017  80133 11188  26128 77467  10073 45780  20987 96880  47497 28645
13018  73227 74518  22980 20160  92817 00211  42382 28120  74188 82650
13019  08833 55503  24680 77715  71866 11231  18399 43878  82068 70390

13020  10858 14687  48980 86271  66747 79393  00990 11562  20229 06821
13021  61208 44219  20763 02262  30728 84949  85428 29827  71869 37380
13022  08293 28609  01555 88948  77854 87817  82723 85737  54842 28987
13023  20233 25887  46117 77452  34887 81806  28825 10202  12861
13024  49751 04824  50224 90597  46374 18474  75780 12341  67264 82896

13025  14221 07948  62399 58232  22943 40205  40532 07550  26514 07805
13026  82181 90648  13932 07923  10622 44462  45481 74891  48398 28486
13027  55226 73208  17424 21926  51825 01428  13768 82177  78183 13488
13028  52120 24200  44858 49359  13224 8174  43489 84239  12299 81389
13029  22068 87173  21822 24229  88828 26404  13604 68394  60564 89827

13030  81463 87224  33416 48804  29256 21588  20259 79995  81974 53180
13031  07987 88348  87702 79829  09002 24181  80214 80144  72718 56286
13032  94622 82486  27027 88861  14114 20470  28768 42862  82002 17870
13033  43209 22040  89880 41708  22299 48913  02879 78328  91223 18807
13034  81724 77977  30411 83784  79874 81823  81246 08218  17238 87202

13035  21079 38714  11486 82300  88004 39467  85906 02664  01269 54713
13036  21282 98312  68234 24724  00572 87245  88920 09476  97899 88928
13037  84697 01189  08029 18019  23502 77165  07999 65278  80808 92523
13038  68041 23472  05870 22624  21281 20182  71805 87993  06472 78236
13039  72853 11689  78063 90264  42723 62670  28248 84828  27420 51240

13040  26682 72429  29736 24231  28109 45102  17893 49260  45204 27668
13041  71084 87884  70605 99049  42316 08041  26815 72842  41800 24487
13042  94987 48850  05402 29048  70885 46913  12760 90909  88240 09827
13043  13209 88602  04876 68283  20117 80277  47812 87109  07403 13562
13044  20740 20903  17977 02314  19017 81818  52720 00799  71490 71460

13045  64347 87866  87587 87503  05180 82208  01114 25888  24610 87780
13046  88289 31270  11217 41880  55920 27279  28915 88156  28033 01902
13047  16413 94371  11427 38423  20526 14061  88895 09249  48023 13489
13048  12406 73889  21809 82654  27259 90138  81207 16884  20683 08460
13049  26123 81333  90450 84161  00991 00861  22883 96828  64492 28914

```

Random bits

- RandomNumber.org
- Hotbits
<http://www.fourmilab.ch/hotbits/>
- Générateur de Marsaglia
<http://www.stat.fsu.edu/pub/diehard/>
- etc.

Pre-computed randomness

Rand computer

Tables with random figures

TABLE OF RANDOM DIGITS 281

```

13000  31720 01273 20018 23248 46903 50238 72887 45155 43214 11849
13001  59971 74215 38773 26132 71727 21941 35033 70144 96589 88064
13002  82231 26139 04946 24667 78238 72766 41805 62206 57512 90123
13003  77246 73403 17182 43771 88643 81230 77118 22870 36124 75891
13004  78572 54500 82482 43730 34186 82778 12390 17114 72589 33784

13005  54233 80475 16486 71100 54283 41828 39952 10178 24692 76547
13006  87210 45484 55000 22878 82848 03192 96866 79801 84050 88751
13007  35977 82057 28434 83896 32874 27732 21866 28149 79258 08336
13008  01798 05027 16803 83255 24464 15327 34792 45274 26964 41283
13009  28687 32108 90542 03720 28884 65518 91983 16737 73420 00848

13010  94804 02297 22897 87687 68013 06657 01227 30646 80343 71107
13011  88847 23522 44274 70730 92271 14420 18770 86266 07023 58866
13012  44432 89730 94809 68228 88220 04781 33731 74739 73897 93930
13013  53126 00861 71284 46282 87522 98832 27128 38801 32863 89824
13014  30421 83346 20868 18839 11538 08733 50983 38275 89448 66605

13015  80569 17972 93222 42848 42547 44771 04289 05480 37082 02288
13016  74977 46460 10420 88840 22064 22468 82005 71738 10212 05245
13017  80133 11188 26128 77467 10073 45780 20987 96880 47497 28645
13018  73227 74518 22980 20160 92817 00211 46282 28120 74188 82626
13019  08833 55503 24680 77715 71866 11231 18399 43878 82068 70390

13020  10858 14667 48980 86271 66747 79393 00990 11562 20229 06821
13021  61208 44219 20763 02262 20728 84549 85428 29827 71869 37380
13022  08293 28609 01555 88948 77854 87517 82723 85737 54842 28987
13023  20252 35838 49317 77482 34887 81884 78864 28525 82240 12821
13024  49751 04824 58224 90597 46374 18474 75790 12341 67264 82896

13025  14221 07948 62399 58232 25943 45205 45032 07850 26514 07805
13026  82181 90648 18932 07923 10622 44462 45481 74891 48388 28486
13027  55226 73208 17424 21926 51825 01428 13768 82177 78183 13488
13028  52120 24200 44858 49339 11224 8174 43489 84239 12299 21819
13029  22058 87173 21822 24229 80828 26404 13804 68394 66564 89227

13030  81463 87224 32416 48804 29256 21588 20259 79995 81974 53180
13031  07987 88348 87702 79829 09002 24181 80214 80144 72718 56288
13032  94622 82486 27027 88861 14114 20470 28768 42882 82002 17870
13033  43209 22040 89880 41708 22299 48913 02879 78328 91223 18807
13034  81724 77977 30411 83786 78874 81823 81246 08218 27238 87202

13035  21079 38714 11486 82300 88004 39467 85906 02664 01269 54713
13036  21282 98312 68234 24724 05872 87243 88920 09476 97899 88928
13037  84697 10189 08029 18019 23502 77165 07959 82178 88088 92522
13038  68041 23472 05870 22024 21281 20182 71805 87893 06472 76826
13039  72853 11689 78063 92064 42723 62670 28248 84828 27420 51240

13040  76682 72429 29756 24231 28109 45102 17892 49280 45204 27668
13041  71084 87884 70605 99049 42316 08041 28415 72842 41800 24887
13042  94987 48850 20402 29848 70885 46915 12700 76209 88240 09827
13043  13209 88602 04876 68283 20117 80277 47812 87109 07403 13562
13044  20740 20903 17977 02214 19027 81818 22720 00729 71490 71460

13045  61347 87866 87587 87503 05180 62206 01114 25888 24610 87780
13046  88289 32750 11217 41880 55920 27273 28915 88156 28033 01902
13047  16413 94371 11427 38423 20526 14061 88495 09549 28023 13489
13048  12406 73889 21809 82854 27759 90128 81207 14884 20883 08480
13049  26123 81333 92450 84161 00991 00861 22883 96828 64492 28914

```

Random bits

- RandomNumber.org
- Hotbits
<http://www.fourmilab.ch/hotbits/>
- Générateur de Marsaglia
<http://www.stat.fsu.edu/pub/diehard/>
- etc.

Outline of the lecture

- 1 Random machines**
 - Why generate random numbers ?
 - Random machines
 - Pseudo-random generators

- 2 and Human mind**
 - Randomness detection
 - Generate randomness

Monte-Carlo method

Numeric integration

Manhattan Project Simulation of nuclear chain reaction => solve partial differential equations

$$I = \int_a^b f(x) dx \simeq \frac{1}{N} \sum_{i=1}^N f(U_i).$$

Stanislaw Marcin Ulam (math)

Enrico Fermi (physic)

John von Neumann (app math)

Nicholas Metropolis (physic)

Computer : Eniac 1943

Electronic Numerical
Integrator And Computer
John Mauchly et J. Presper
Eckert

University of Pennsylvania.

Monte-Carlo method

Numeric integration

Manhattan Project Simulation of nuclear chain reaction => solve partial differential equations

$$I = \int_a^b f(x) dx \simeq \frac{1}{N} \sum_{i=1}^N f(U_i).$$

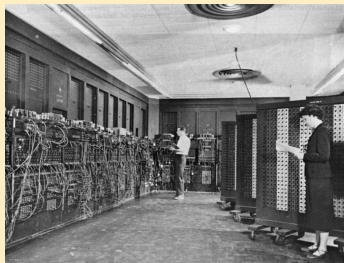
Stanislaw Marcin Ulam (math)

Enrico Fermi (physic)

John von Neumann (app math)

Nicholas Metropolis (physic)

Computer : Eniac 1943



Electronic Numerical Integrator And Computer
John Mauchly et J. Presper Eckert

University of Pennsylvania.

John von Neuman (1903-1957)



Mathématicien américain d'origine hongroise. Il a apporté d'importantes contributions tant en mécanique quantique, qu'en analyse fonctionnelle, en théorie des ensembles, en informatique, en sciences économiques ainsi que dans beaucoup d'autres domaines des mathématiques et de la physique. Il a de plus participé aux programmes militaires américains.

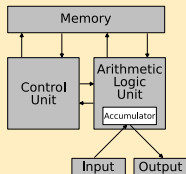
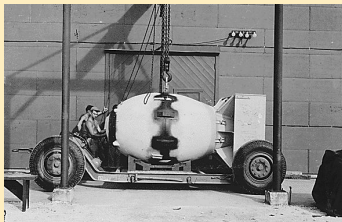
Computer Architecture

John von Neuman (1903-1957)

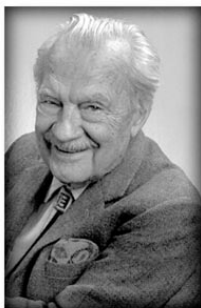


Mathématicien américain d'origine hongroise. Il a apporté d'importantes contributions tant en mécanique quantique, qu'en analyse fonctionnelle, en théorie des ensembles, en informatique, en sciences économiques ainsi que dans beaucoup d'autres domaines des mathématiques et de la physique. Il a de plus participé aux programmes militaires américains.

Computer Architecture



Nicholas Metropolis (1915-1999)



Nick Metropolis

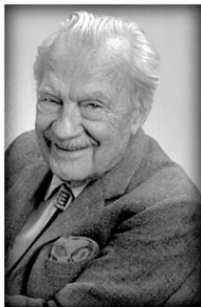
Metropolis contributed several original ideas to mathematics and physics. Perhaps the most widely known is the Monte Carlo method. Also, in 1953 Metropolis co-authored the first paper on a technique that was central to the method known now as simulated annealing. He also developed an algorithm (the Metropolis algorithm or Metropolis-Hastings algorithm) for generating samples from the Boltzmann distribution, later generalized by W.K. Hastings.

Simulated annealing

Convergence to a global minimum by a control stochastic gradient algorithm

$$X_{n+1} = X_n - \vec{\text{grad}}\Phi(X_n)\Delta(\text{Random}).$$

Nicholas Metropolis (1915-1999)



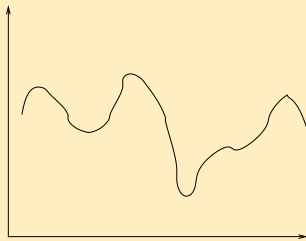
Nick Metropolis

Metropolis contributed several original ideas to mathematics and physics. Perhaps the most widely known is the Monte Carlo method. Also, in 1953 Metropolis co-authored the first paper on a technique that was central to the method known now as simulated annealing. He also developed an algorithm (the Metropolis algorithm or Metropolis-Hastings algorithm) for generating samples from the Boltzmann distribution, later generalized by W.K. Hastings.

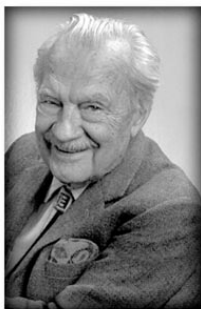
Simulated annealing

Convergence to a global minimum by a control stochastic gradient algorithm

$$X_{n+1} = X_n - \vec{\text{grad}}\Phi(X_n)\Delta(\text{Random}).$$



Nicholas Metropolis (1915-1999)



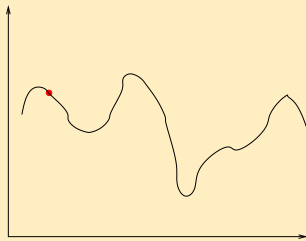
Nick Metropolis

Metropolis contributed several original ideas to mathematics and physics. Perhaps the most widely known is the Monte Carlo method. Also, in 1953 Metropolis co-authored the first paper on a technique that was central to the method known now as simulated annealing. He also developed an algorithm (the Metropolis algorithm or Metropolis-Hastings algorithm) for generating samples from the Boltzmann distribution, later generalized by W.K. Hastings.

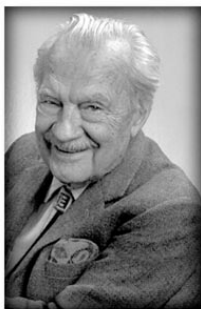
Simulated annealing

Convergence to a global minimum by a control stochastic gradient algorithm

$$X_{n+1} = X_n - \vec{\text{grad}}\Phi(X_n)\Delta(\text{Random}).$$



Nicholas Metropolis (1915-1999)



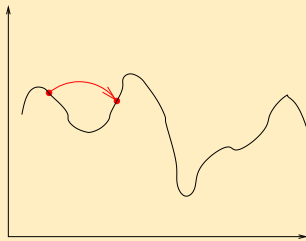
Nick Metropolis

Metropolis contributed several original ideas to mathematics and physics. Perhaps the most widely known is the Monte Carlo method. Also, in 1953 Metropolis co-authored the first paper on a technique that was central to the method known now as simulated annealing. He also developed an algorithm (the Metropolis algorithm or Metropolis-Hastings algorithm) for generating samples from the Boltzmann distribution, later generalized by W.K. Hastings.

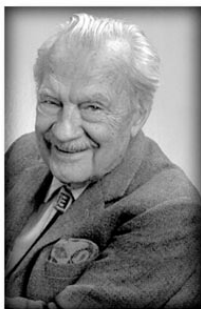
Simulated annealing

Convergence to a global minimum by a control stochastic gradient algorithm

$$X_{n+1} = X_n - \vec{\text{grad}}\Phi(X_n)\Delta(\text{Random}).$$



Nicholas Metropolis (1915-1999)



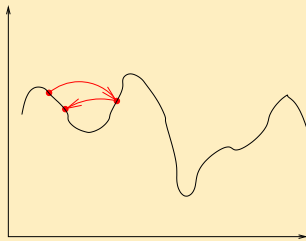
Nick Metropolis

Metropolis contributed several original ideas to mathematics and physics. Perhaps the most widely known is the Monte Carlo method. Also, in 1953 Metropolis co-authored the first paper on a technique that was central to the method known now as simulated annealing. He also developed an algorithm (the Metropolis algorithm or Metropolis-Hastings algorithm) for generating samples from the Boltzmann distribution, later generalized by W.K. Hastings.

Simulated annealing

Convergence to a global minimum by a control stochastic gradient algorithm

$$X_{n+1} = X_n - \vec{\text{grad}}\Phi(X_n)\Delta(\text{Random}).$$



Nicholas Metropolis (1915-1999)



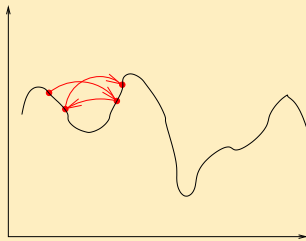
Nick Metropolis

Metropolis contributed several original ideas to mathematics and physics. Perhaps the most widely known is the Monte Carlo method. Also, in 1953 Metropolis co-authored the first paper on a technique that was central to the method known now as simulated annealing. He also developed an algorithm (the Metropolis algorithm or Metropolis-Hastings algorithm) for generating samples from the Boltzmann distribution, later generalized by W.K. Hastings.

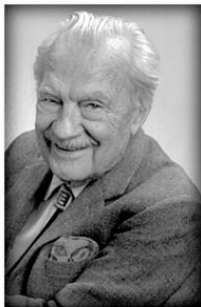
Simulated annealing

Convergence to a global minimum by a control stochastic gradient algorithm

$$X_{n+1} = X_n - \vec{\text{grad}}\Phi(X_n)\Delta(\text{Random}).$$



Nicholas Metropolis (1915-1999)



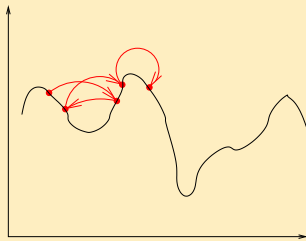
Nick Metropolis

Metropolis contributed several original ideas to mathematics and physics. Perhaps the most widely known is the Monte Carlo method. Also, in 1953 Metropolis co-authored the first paper on a technique that was central to the method known now as simulated annealing. He also developed an algorithm (the Metropolis algorithm or Metropolis-Hastings algorithm) for generating samples from the Boltzmann distribution, later generalized by W.K. Hastings.

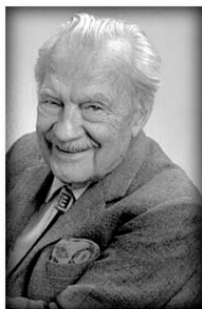
Simulated annealing

Convergence to a global minimum by a control stochastic gradient algorithm

$$X_{n+1} = X_n - \vec{\text{grad}}\Phi(X_n)\Delta(\text{Random}).$$



Nicholas Metropolis (1915-1999)



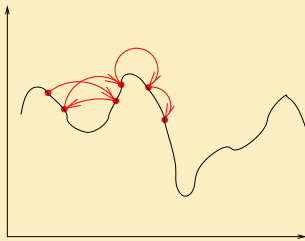
Nick Metropolis

Metropolis contributed several original ideas to mathematics and physics. Perhaps the most widely known is the Monte Carlo method. Also, in 1953 Metropolis co-authored the first paper on a technique that was central to the method known now as simulated annealing. He also developed an algorithm (the Metropolis algorithm or Metropolis-Hastings algorithm) for generating samples from the Boltzmann distribution, later generalized by W.K. Hastings.

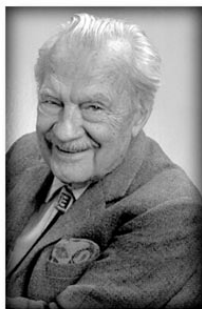
Simulated annealing

Convergence to a global minimum by a control stochastic gradient algorithm

$$X_{n+1} = X_n - \vec{\text{grad}}\Phi(X_n)\Delta(\text{Random}).$$



Nicholas Metropolis (1915-1999)



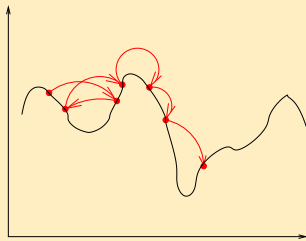
Nick Metropolis

Metropolis contributed several original ideas to mathematics and physics. Perhaps the most widely known is the Monte Carlo method. Also, in 1953 Metropolis co-authored the first paper on a technique that was central to the method known now as simulated annealing. He also developed an algorithm (the Metropolis algorithm or Metropolis-Hastings algorithm) for generating samples from the Boltzmann distribution, later generalized by W.K. Hastings.

Simulated annealing

Convergence to a global minimum by a control stochastic gradient algorithm

$$X_{n+1} = X_n - \vec{\text{grad}}\Phi(X_n)\Delta(\text{Random}).$$



Nicholas Metropolis (1915-1999)



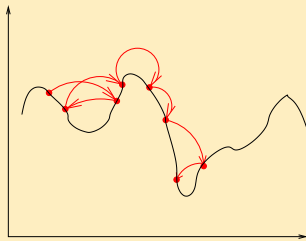
Nick Metropolis

Metropolis contributed several original ideas to mathematics and physics. Perhaps the most widely known is the Monte Carlo method. Also, in 1953 Metropolis co-authored the first paper on a technique that was central to the method known now as simulated annealing. He also developed an algorithm (the Metropolis algorithm or Metropolis-Hastings algorithm) for generating samples from the Boltzmann distribution, later generalized by W.K. Hastings.

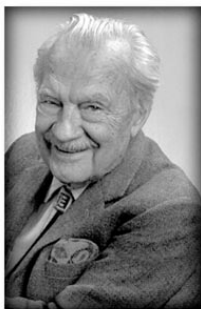
Simulated annealing

Convergence to a global minimum by a control stochastic gradient algorithm

$$X_{n+1} = X_n - \vec{\text{grad}}\Phi(X_n)\Delta(\text{Random}).$$



Nicholas Metropolis (1915-1999)



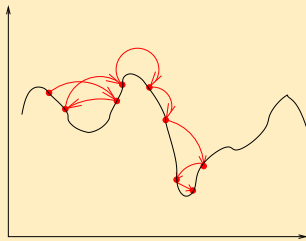
Nick Metropolis

Metropolis contributed several original ideas to mathematics and physics. Perhaps the most widely known is the Monte Carlo method. Also, in 1953 Metropolis co-authored the first paper on a technique that was central to the method known now as simulated annealing. He also developed an algorithm (the Metropolis algorithm or Metropolis-Hastings algorithm) for generating samples from the Boltzmann distribution, later generalized by W.K. Hastings.

Simulated annealing

Convergence to a global minimum by a control stochastic gradient algorithm

$$X_{n+1} = X_n - \vec{\text{grad}}\Phi(X_n)\Delta(\text{Random}).$$



Pseudo-random generator (1)

Middle of squares

Objects : integers

Idea : shuffle procedure

Generation algorithm

$x \leftarrow \textit{seed}$

repeat

$y \leftarrow x^2$

$x \leftarrow \textit{middle}(y)$

$\textit{write}(x)$

until End of simulation

Example

$$x_0 = 5869 \rightarrow x_0^2 = 34|4451|61$$

$$x_1 = 4451 \rightarrow x_1^2 = 19|8114|01$$

$$x_2 = 8114 \rightarrow x_2^2 = 65|8369|96$$

$$x_3 = 8369 \rightarrow x_3^2 = 70|0401|61$$

$$x_4 = 0401 \rightarrow x_4^2 = 00|1608|01$$

$$x_5 = 1608 \rightarrow x_5^2 = 02|5856|64$$

$$x_6 = 5856 \rightarrow x_6^2 = 34|2927|36$$

$$x_7 = 5856 \rightarrow \dots$$

Pseudo-random generator (1)

Middle of squares

Objects : integers

Idea : shuffle procedure

Generation algorithm

$x \leftarrow \textit{seed}$

repeat

$y \leftarrow x^2$

$x \leftarrow \textit{middle}(y)$

$\textit{write}(x)$

until End of simulation

Example

$$x_0 = 5869 \rightarrow x_0^2 = 34|4451|61$$

$$x_1 = 4451 \rightarrow x_1^2 = 19|8114|01$$

$$x_2 = 8114 \rightarrow x_2^2 = 65|8369|96$$

$$x_3 = 8369 \rightarrow x_3^2 = 70|0401|61$$

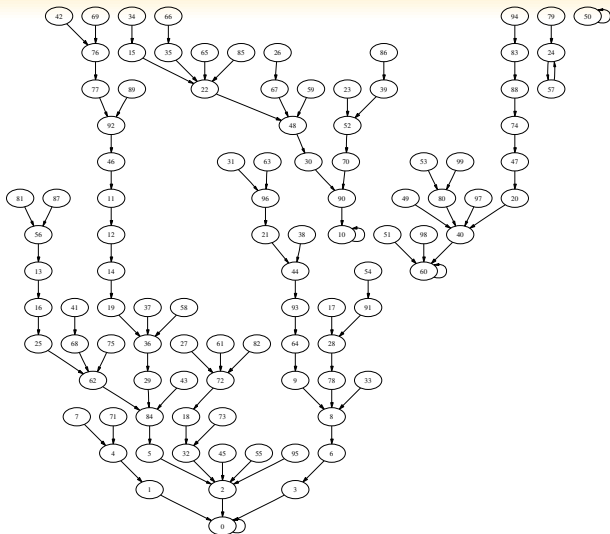
$$x_4 = 0401 \rightarrow x_4^2 = 00|1608|01$$

$$x_5 = 1608 \rightarrow x_5^2 = 02|5856|64$$

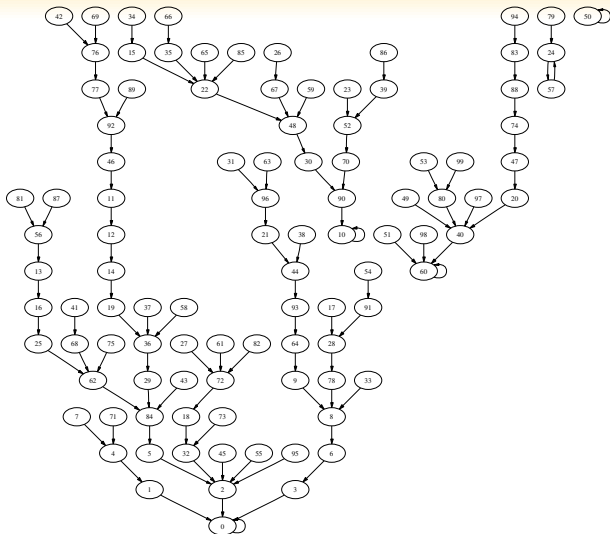
$$x_6 = 5856 \rightarrow x_6^2 = 34|2927|36$$

$$x_7 = 5856 \rightarrow \dots$$

Middle of squares



Middle of squares



Pseudo-random generator (2)

Modulo transform

Linear transform

Objects : integers

$\{0, \dots, m-1\}$

Parameters :

$a, b \in \{0, \dots, m-1\}$

$x_{n+1} = a * x_n + b \text{ mod } m$

$x \leftarrow \text{seed}$

repeat

$x \leftarrow a * x + b \text{ mod } m$

write(x)

until End of simulation

Example

$a = 11, b = 1, m = 71$

$17 \rightarrow 46 \rightarrow 10 \rightarrow 40 \rightarrow 15 \rightarrow 24 \rightarrow \dots$

Diagram $x_{n+1} = 3 * x_n + 4 \text{ mod } 11$

Pseudo-random generator (2)

Modulo transform

Linear transform

Objets : integers

$\{0, \dots, m-1\}$

Parameters :

$a, b \in \{0, \dots, m-1\}$

$x_{n+1} = a * x_n + b \text{ mod } m$

$x \leftarrow \text{seed}$

repeat

$x \leftarrow a * x + b \text{ mod } n$

$\text{write}(x)$

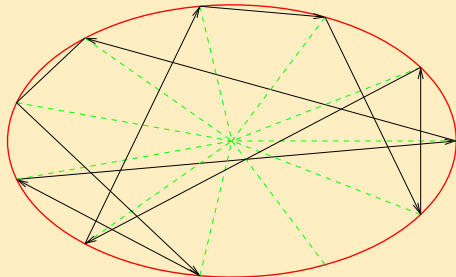
until End of simulation

Example

$a = 11, b = 1, m = 71$

$17 \rightarrow 46 \rightarrow 10 \rightarrow 40 \rightarrow 15 \rightarrow 24 \rightarrow \dots$

Diagram $x_{n+1} = 3 * x_n + 4 \text{ mod } 11$



Congruent generators

Find a maximum cycle length

Théorème

Hull-Dobell, (1962) Let $\{x_n\}$ the sequence defined by $x_{n+1} = ax_n + b \pmod m$. Then the maximal cycle has length m if and only if the 3 conditions are verified :

- 1 $GCD(a, m) = 1, GCD(b, m) = 1 ;$
- 2 *if a prime number p is a divisor of m , then p divide $a - 1 ;$*
- 3 *if 4 divide m , then 4 divide $a - 1$.*

Gives uniformity, but not the mixing property

Examples of congruent generators

$$x_{n+1} = 7^5 x_n \pmod{2^{31} - 1}, \text{ (IBM's generator)}$$

$$x_{n+1} = 427419669081 x_n \pmod{999999999989},$$

(Maple's generator 999999999989 is prime)

$$x_{n+1} = 3^{15} x_n \pmod{2^{32}},$$

$$x_{n+1} = 3 + 2^{16} x_n \pmod{2^{31}},$$

$$x_{n+1} = 13^{13} x_n \pmod{2^{59}},$$

$$x_{n+1} = 24298 x_n + 99991 \pmod{199017},$$

cycle length :

$$2^{30} = 1\,073\,741\,824,$$

$$2^{29} = 536\,870\,912,$$

$$2^{57} = 144\,115\,188\,075\,855\,872,$$

$$199\,017$$

Random bits

$x = \{x_1, x_2, \dots, x_n, \dots\}$ and $y = \{y_1, y_2, \dots, y_n, \dots\}$
sequences of random bits

Hourra for the *XOR*

The sequence $z = \{z_1, z_2, \dots, z_n, \dots\}$ with $z_i = x_i \text{ XOR } y_i$ is better than x and y .

Hypothesis x_i and y_i are independent.

Proof :

Random bits

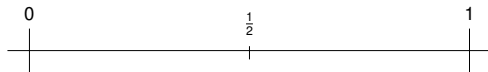
$x = \{x_1, x_2, \dots, x_n, \dots\}$ and $y = \{y_1, y_2, \dots, y_n, \dots\}$
sequences of random bits

Hourra for the *XOR*

The sequence $z = \{z_1, z_2, \dots, z_n, \dots\}$ with $z_i = x_i \text{ XOR } y_i$ is better than x and y .

Hypothesis x_i and y_i are independent.

Proof :



Random bits

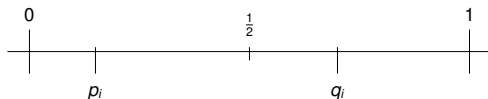
$x = \{x_1, x_2, \dots, x_n, \dots\}$ and $y = \{y_1, y_2, \dots, y_n, \dots\}$
sequences of random bits

Hourra for the *XOR*

The sequence $z = \{z_1, z_2, \dots, z_n, \dots\}$ with $z_i = x_i \text{ XOR } y_i$ is better than x and y .

Hypothesis x_i and y_i are independent.

Proof :



$$\mathbb{P}(X_i \text{ XOR } Y_i = 1) = p_i(1 - q_i) + (1 - p_i)q_i$$

Random bits

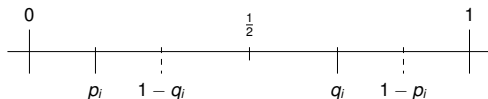
$x = \{x_1, x_2, \dots, x_n, \dots\}$ and $y = \{y_1, y_2, \dots, y_n, \dots\}$
sequences of random bits

Hourra for the *XOR*

The sequence $z = \{z_1, z_2, \dots, z_n, \dots\}$ with $z_i = x_i \text{ XOR } y_i$ is better than x and y .

Hypothesis x_i and y_i are independent.

Proof :



$$\mathbb{P}(X_i \text{ XOR } Y_i = 1) = p_i(1 - q_i) + (1 - p_i)q_i$$

Random bits

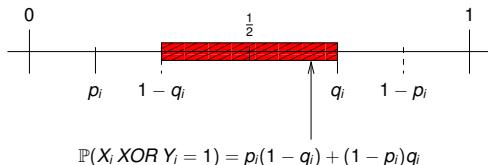
$x = \{x_1, x_2, \dots, x_n, \dots\}$ and $y = \{y_1, y_2, \dots, y_n, \dots\}$
sequences of random bits

Hourra for the *XOR*

The sequence $z = \{z_1, z_2, \dots, z_n, \dots\}$ with $z_i = x_i \text{ XOR } y_i$ is better than x and y .

Hypothesis x_i and y_i are independent.

Proof :



Random bits (2)

Biased generator of independent bits : $p = \mathbb{P}(x_i = 1)$

$$x = \{x_1, x_2, \dots, x_n, \dots\}$$

$$y_n = x_{nk+1} \text{ XOR } x_{nk+2} \text{ XOR } \dots \text{ XOR } x_{n(k+1)}$$

$$\mathbb{P}(y_n = 1) = \frac{1}{2} \left(1 - (1 - 2p)^k \right) \xrightarrow{\text{exponentially}} \frac{1}{2}.$$

Approximation of an unbiased coin (error control)

Ex : $p = \frac{1}{3}, k = 10$

$$\mathbb{P}(y_n = 1) \simeq \frac{1}{2} \pm 10^{-5}.$$

Random bits (the end)

Biased generator of independent bits : $p = \mathbb{P}(x_i = 1)$

$x = \{x_1, x_2, \dots, x_n, \dots\}$

Generate an unbiased coin :

repeat

$X = \text{coin}()$;

$Y = \text{coin}()$;

until ($X \neq Y$)

return X ;

Rejection base algorithm : $\mathbb{P}(\text{accept}) = 2p(1 - p)$

Mean number of iterations : $\bar{N} = \frac{1}{2p(1-p)}$

Ex : $p = \frac{1}{3}$,

$$\bar{N} = \frac{9}{4} = 2,25.$$

Random bits (the end)

Biased generator of independent bits : $p = \mathbb{P}(x_i = 1)$

$X = \{X_1, X_2, \dots, X_n, \dots\}$

Generate an unbiased coin :

repeat

$X = \text{coin}()$;

$Y = \text{coin}()$;

until ($X \neq Y$)

return X ;

Rejection base algorithm : $\mathbb{P}(\text{accept}) = 2p(1 - p)$

Mean number of iterations : $\bar{N} = \frac{1}{2p(1-p)}$

Ex : $p = \frac{1}{3}$,

$$\bar{N} = \frac{9}{4} = 2,25.$$

Pseudo-random generator

Use the *XOR* inside the algorithm

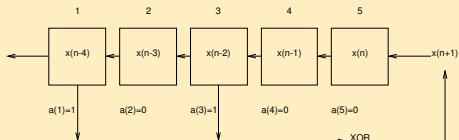
Tausworthe (1965)

initial binary vector (seed) $x^0 = (x_{-m+1}, \dots, x_{-1}, x_0)$,
multi-linear recurrence :

$$x_{n+1} = a_1 x_{n-m+1} + a_2 x_{n-m+1} + \dots + a_m x_n \pmod{2}$$

a_1, a_2, \dots, a_m fixed

Shifted loop register



Pseudo-random generator

Use *XOR* and modulo

Mersenne Twister (1998)

$$x_n = x_{n-(N-M)} \oplus (x_{n-N}^U | x_{n-N+1}^L) A$$

with a good parameter set :
cycle length = $2^{19937} - 1$

Blum Blum Shub Generator (1986)

$$x_{n+1} = x_n^2 \bmod M$$

Statistically bad
Excellent for cryptography

Outline of the lecture

1 Random machines

- Why generate random numbers ?
- Random machines
- Pseudo-random generators

2 and Human mind

- Randomness detection
- Generate randomness

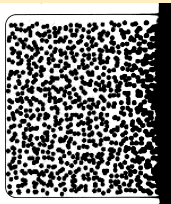
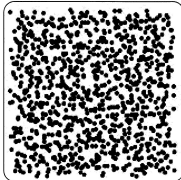
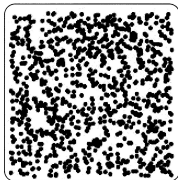
Randomness detection

René Magritte, Golconda. 1953.



Randomness detection (2)

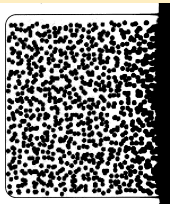
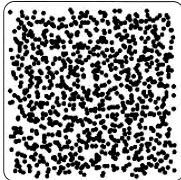
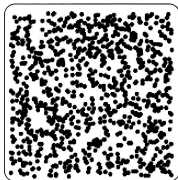
Which one is random ?



Rake effect : randomness = uniformity = equality among places

Randomness detection (2)

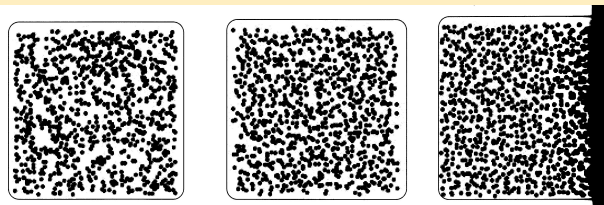
Which one is random ?



Rake effect : randomness = uniformity = equality among places

Randomness detection (2)

Which one is random ?



Rake effect : randomness = uniformity = equality among places

Randomness detection (3)

Pierre Bruegel : Children games



crowd : random (not under control)

Randomness detection (3)

Pierre Bruegel : Children games



crowd : random (not under control)

The Law of the Series

Das Gesetz der Serie (Paul Kammerer)

- 1 what happened is going to happen again rapidly
- 2 bad events are grouped
- 3 events from a same category happen together

causal explanation

rake effect (ex anniversary paradox)

focus of observation

true relation

The Law of the Series

Das Gesetz der Serie (Paul Kammerer)

- 1 what happened is going to happen again rapidly
- 2 bad events are grouped
- 3 events from a same category happen together

causal explanation

rake effect (ex anniversary paradox)

focus of observation

true relation

Outline of the lecture

1 Random machines

- Why generate random numbers ?
- Random machines
- Pseudo-random generators

2 and Human mind

- Randomness detection
- Generate randomness

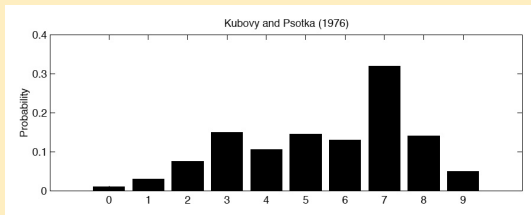
Generate randomness

Give a random figure between 0 and 9

We look for a number and considering the arithmetic properties of this number we choose the one with the less properties.

Generate randomness

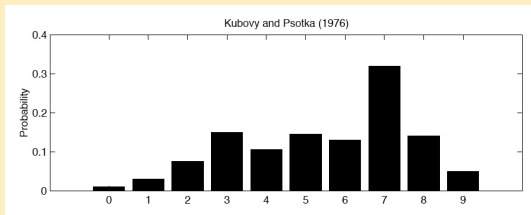
Give a random figure between 0 and 9



We look for a number and considering the arithmetic properties of this number we choose the one with the less properties.

Generate randomness

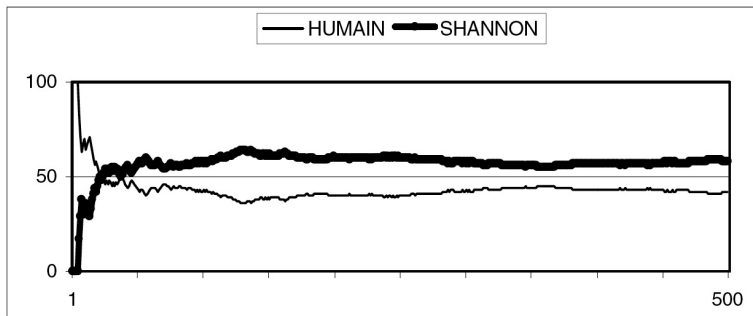
Give a random figure between 0 and 9



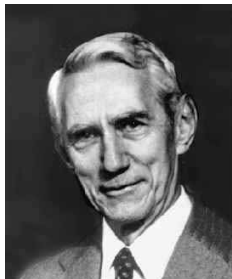
We look for a number and considering the arithmetic properties of this number we choose the one with the less properties.

Even-Odd against the machine

8 states automaton



Claude Shannon (1916-2001)



Claude Elwood Shannon (30 avril 1916 à Gaylord, Michigan - 24 février 2001), ingénieur électrique, est l'un des pères, si ce n'est le père fondateur, de la théorie de l'information. Son nom est attaché à un célèbre "schéma de Shannon" très utilisé en sciences humaines, qu'il a constamment désavoué.

Il étudia le génie électrique et les mathématiques à l'Université du Michigan en 1932. Il utilisa notamment l'algèbre booléenne pour sa maîtrise soutenue en 1938 au MIT. Il y expliqua comment construire des machines à relais en utilisant l'algèbre de Boole pour décrire l'état des relais (1 : fermé, 0 : ouvert). Shannon travailla 20 ans au MIT, de 1958 à 1978. Parallèlement à ses activités académiques, il travailla aussi aux laboratoires Bell de 1941 à 1972. Claude Shannon était connu non seulement pour ses travaux dans les télécommunications, mais aussi pour l'étendue et l'originalité de ses hobbies, comme la jonglerie, la pratique du monocycle et l'invention de machines farfelues : une souris mécanique sachant trouver son chemin dans un labyrinthe, un robot jongleur, un joueur d'échecs (roi tour contre roi)... Souffrant de la maladie d'Alzheimer dans les dernières années de sa vie, Claude Shannon est mort à 84 ans le 24 février 2001.

To go further...

Historical books

Laplace, P.-S. (1812), *Théorie analytique des probabilités*, Mme Ve Courcier, Paris.

Laplace, P.-S. (1825), *Essai philosophique sur les probabilités*, Gauthier-Villard et Cie, Paris.

Poincaré, H. (1912), *Calcul des probabilités*, Gauthier-Villars, Paris.

Kolmogorov, A. (1933), *Foundations of the theory of probability*, Chelsea Publishing Company.

Borel, E. (1922), *Principes et formules classiques du calcul des probabilités*, Gauthier-Villars, Paris.

Shannon, C. (1948), *A mathematical theory of communications*, Bells Systems Technical Journal.

Li, M. et Vitányi, P. (1990), *Kolmogorov Complexity and its Applications*, Elsevier Science Publisher, chapter 4, pp.

To go further (2) ...

Synthesis (in french)

Ekeland, I. (1991), Au hasard, Le Seuil.

Dacunha-Castelle, D. (1996) Chemins de l'aléatoire, Flammarion.

Le hasard (1996), Dossier Pour la science.

Pagès, G. et Bouzitat, C. (1999) En passant par hasard... Vuibert
Paris

Delahaye, J.-P. (1993), Le désordre total existe-t-il ?, Pour la Science
(193), 152 ?156.

Delahaye, J.-P. (1994a), Information, complexité et hasard, Hermes.

Delahaye, J.-P. (1994b), Le complexe surgit-il du simple ?, Pour la
Science (203), 102-107.

To go further (3) ...

Web sites

For biographies

- en.wikipedia.org

- <http://www-groups.dcs.st-and.ac.uk/history/>

Chaitin's web page :

<http://www.cs.auckland.ac.nz/chaitin>

Levin's web page :

<http://www.cs.bu.edu/Ind>

Delahaye's web page

<http://www2.lifl.fr/delahaye/>