

SIMULASI MONTE CARLO

INTRODUCTION TO COMPUTATIONAL MODELLING

Ikang Fadhli

Nutrifood Indonesia

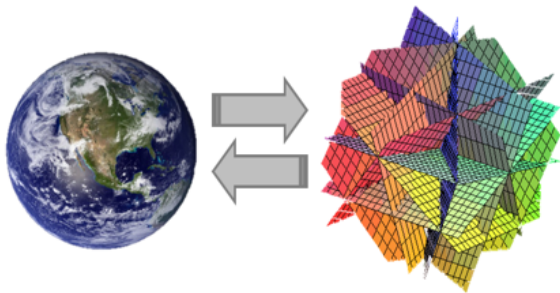
17 December 2021

- 1 MODELLING
- 2 SIMULASI
- 3 *USE CASES I*
- 4 *USE CASES II*

Section 1

MODELLING

Model



A model is a tool to help us understand the complexities of the universe, and never a substitute for the universe itself. Nate Silver.

Mencari Solusi Model

Suatu permasalahan bisa dituliskan menjadi model matematis (termasuk statistik). Biasanya *goals* dari model tersebut adalah untuk:

- 1 Mencari akar (*roots*).
- 2 Memaksimalkan atau meminimalkan (*optimize*).

Mencari Solusi Model

Untuk menyelesaikan dan mencari solusi dari suatu model, setidaknya ada dua pendekatan yang bisa dilakukan:

- 1 Metode *exact*: menggunakan prinsip matematika (kalkulus dan aljabar) untuk mencari solusi.
 - *Advantages*: **pasti memberikan solusi terbaik.**
 - *Disadvantages*: Lama dan tidak semua masalah di dunia bisa dicari solusi eksaknya.
- 2 Metode *numeric*: menggunakan aproksimasi dengan prinsip-prinsip komputasi (*conditional*, *looping*, dan *sequence*).
 - *Advantages*: **belum tentu memberikan solusi terbaik.**
 - *Disadvantages*: Relatif cepat dan bisa untuk semua masalah.

Mencari Solusi Model (lanjutan)

Sebenarnya ada satu lagi pendekatan yang bisa dilakukan, yakni:

Metode *heuristic*: menggunakan aproksimasi dengan prinsip komputasi **namun** cara kerjanya terinspirasi dari kejadian-kejadian alam yang bersifat acak.

- *Advantages*: **belum tentu memberikan solusi terbaik.**
- *Disadvantages*: Relatif cepat. Apakah bisa untuk semua masalah? *No free lunch theorem.*

Contoh Metode *Heuristic*

- *Artificial bee colony (ABC) algorithm*. Terinspirasi dari koloni lebah yang selalu berkumpul mencari madu. Lebah akan dibagi menjadi lebah pekerja dan lebah *scout*.
- *Simulated annealing algorithm*. Terinspirasi dari pendinginan metal yang dilebur. Pada metode ini, kita akan menggunakan istilah temperatur, cooling rate, dan epoch.
- *Spiral optimization algorithm*. Terinspirasi dari bentuk-bentuk spiral yang ada di alam (orbit planet, kerang, dll). Pada metode ini, kita akan menggunakan istilah rotasi dan kontraksi.
- *Genetic algorithm*. Terinspirasi dari mutasi DNA akibat perkawinan silang. Pada metode ini, kita akan menggunakan istilah populasi, kromosom, gen, allele, genotype, fenotype, dll.

Metode *Heuristic*

Saya akan bahas di tahun depan.

Section 2

SIMULASI

Definition

- *Simulation is the operation of a model, which is a representation of that system.*
- *A simulation uses a model to emulate the dynamic characteristics of a system.*
- *The operation of the model can be studied, and then properties concerning the behavior of the actual system can be inferred.*

Dalam Bahasa Manusia

Kita mungkin tidak bisa memiliki data dari suatu sistem. Namun jika kita tahu sifat-sifat dan karakteristik dari sistem tersebut, kita bisa membuat model yang bekerja layaknya sistem tersebut.

- *Mathematicians* → tidak selalu membutuhkan data.
- *Statisticians* → membutuhkan data.

Monte Carlo



Diambil dari suatu nama daerah di Monaco yang memiliki banyak kasino dan tempat perjudian.

Simulasi Monte Carlo

Prinsip

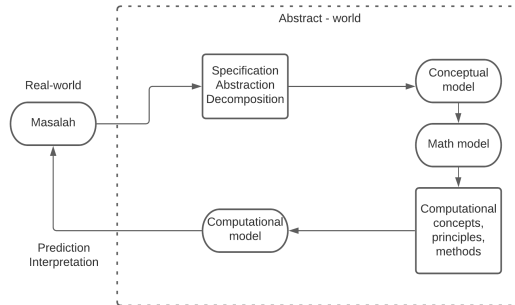
*Melakukan sebanyak-banyaknya pengulangan model **secara acak** untuk mencapai kondisi di mana law of large number terpenuhi.*

Remember, Dr. Strange watched the "Endgame"
fourteen million six hundred and five times.



And he didn't spoil any of it.
Be like Dr. Strange.

Flow chart



Further reading

Simulasi Monte Carlo

Advantages

- ❶ Tidak perlu menuliskan dan menurunkan formulasi matematika dari permasalahan.
- ❷ Cukup menuliskan algoritma *random number generator* sesuai kebutuhan.
- ❸ Bisa digunakan untuk berbagai macam masalah *real*.

Disdvantages

- ❶ Untuk permasalahan yang kompleks, butuh waktu yang lebih lama.
- ❷ Tidak ada jaminan solusi yang didapatkan adalah yang paling optimal.

Section 3

USE CASES I

Monty Hall Problem



Monty Hall Problem

Tirai 1



Tirai 2



Tirai 3



Algoritma *Monty Hall* dalam *Single Round*

Berikut adalah algoritma *Monty Hall* dalam satu babak:

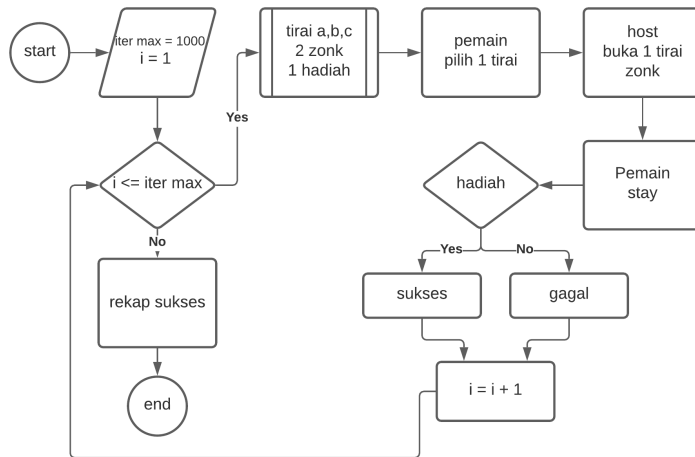
```
pintu = 1:3
hadiah = sample(pintu,1)
pilih = sample(pintu,1)
buka = pintu[c(-pilih,-hadiah)]
  if(length(buka)>1){buka = sample(buka,1)}
decision = ifelse(pilih == hadiah,"stay","switch")
```

Prinsip Simulasi untuk Menyelesaikan *Monty Hall*

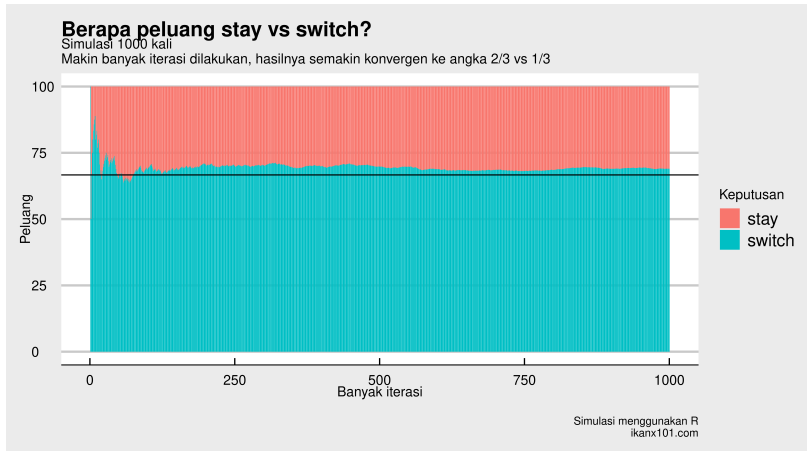
Prinsip Simulasi

Dari algoritma *single round* di *slide* sebelumnya, kita akan ulang ribuan kali untuk melihat seberapa mana decision terbaik dari sekian banyak kemungkinan *event* yang terjadi.

Flowchart Simulasi Monte Carlo untuk Monty Hall



Hasil Simulasi Monte Carlo untuk *Monty Hall*



Section 4

USE CASES II

Resampling Data

Salah satu kegunaan dari simulasi *Monte Carlo* adalah kita bisa melakukan *resampling* memanfaatkan prinsip ini.

Sebagai contoh, tim *market research* sering mendapatkan *request* untuk menyamakan proporsi SES dari berbagai *Annual Research* atau Riset *Media Habit*.

Analogi

Di suatu toko buah, ada 4 jenis buah yang dijual yaitu: apel, jeruk, mangga, dan salak. Dari informasi *supplier*, diketahui:

1. Sebanyak 35 buah dari total 100 buah apel berasa masam.
2. Sebanyak 25 buah dari total 55 buah jeruk berasa masam.
3. Sebanyak 40 buah dari total 60 buah mangga berasa masam.
4. Sebanyak 15 buah dari total 75 buah salak berasa masam.

Setiap jenis buah ditempatkan di baknya masing-masing (terpisah) dan tidak bisa dibedakan secara kasat mata mana yang masam dan manis.

Analogi (lanjutan)

Lalu seorang pelanggan datang dan mengambil:

- 4 buah apel,
- 8 buah jeruk,
- 3 buah mangga,
- 5 buah salak.

Pertanyaannya

Berapa banyak buah masam yang bisa diperoleh pelanggan tersebut?

Analogi (lanjutan)

Ada tiga kemungkinan yang bisa terjadi. Yakni:

- 1 Si pelanggan mengambil **semua buah masam** (20 buah masam).
- 2 Si pelanggan **tidak mengambil buah yang masam** sama sekali (0 buah masam).
- 3 Si pelanggan mengambil $1 \leq n \leq 19$ buah yang masam.

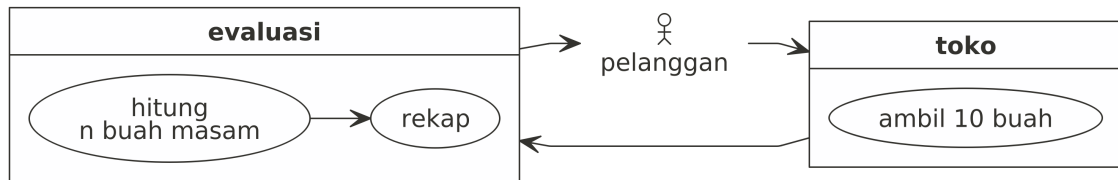
Masalah Utama

Jadi masalahnya adalah menghitung kira-kira berapa buah masam yang berpeluang tinggi bisa diambil pelanggan (*expected value*).

Prinsip Simulasi

Prinsip dari simulasi adalah dengan membuat seolah-olah ada banyak pelanggan datang, mengambil buah, lalu menghitung berapa banyak yang masam. Semua pelanggan akan direkap, rata-rata berapa banyak buah masam akan dihitung.

Ilustrasi Simulasi



Simulasi akan mengulang-ulang proses di atas hingga ribuan bahkan jutaan kali.

Algoritma Simulasi I

```
# probability
p_apel = 35/100
p_jeruk = 25/55
p_mangga = 40/60
p_salak = 15/75

# yang diambil pelanggan
n_apel = 4
n_jeruk = 8
n_mangga = 3
n_salak = 5
```


Algoritma Simulasi II

```
# simulasi
busuk = c()
n_simu = 50000
# iterasi
for(i in 1:n_simu){
  apel = sample(c(1,0),n_apel,replace = T,prob = c(p_apel,1-p_apel))
  jeruk = sample(c(1,0),n_jeruk,replace = T,prob = c(p_jeruk,1-p_jeruk))
  mangga = sample(c(1,0),n_mangga,replace = T,prob = c(p_mangga,1-p_mangga))
  salak = sample(c(1,0),n_salak,replace = T,prob = c(p_salak,1-p_salak))
  buah_busuk = sum(apel,jeruk,mangga,salak)
  busuk = c(buah_busuk,busuk)
}
# rekap hasil simulasi
rekap_simulasi = data.frame(iter = 1:n_simu,busuk)
```