EN SIMULERING AV MONTY HALL

Skapare: Melker Engblom, Stefan Emet, Simon Lindqvist, Öjvind Sandsjoe

Skapad hösten 2023.

Målgrupp:
Målgruppen är främst gymnasiestuderande i kurs MAB5, MAA8 eller MAA11. Vissa delar av materialet kan också tänkas att de går att använda i högstadiet

Lektionens längd: Beroende på gruppens förhandskunskaper och vilka delar som väljs kan en eller två 75 minuters lektioner användas.

Inlärningsmål: Efter lektionen förstår den studerande
- Monty Hall-problemet.
- principerna för sannolikhetsräkning och hur man beräknar ut sannolikheten i Monty Hall problemet
- hur man kan simulera Monty Hall problemet och kan experimentera med en simulering

Innehåll: Idén är att läraren kan plocka delar ur följande material för att skapa en lektion som passar den egna gruppen. Lektionen handlar om Monty Hall -problemet och simulation av det. Om man vill göra alla delar kan materialet delas upp på flera lektioner.

(1) Presentation av Monty Hall problemet
(2) Grundläggande information om simulering
(3) Om sannolikhet
(4) Simulering utan dator
(5) Analys av simulationsprogram

1. Presentation av Monty Hall problemet

Du är en deltagare i en TV-frågesport och har möjligheten att vinna en dyrbar bil. Programledaren presenterar dig 3 stängda dörrar. Du vet att bakom en av dörrarna finns en bil och bakom de två andra dörrarna finns det en get. Programledaren ger dig i uppgift att välja en av dörrarna och när du gjort ditt val så hålls din valda dörr stängd men programledaren öppnar en av de två andra dörrarna och bakom den dörren finns en get. Du har nu framför dig två stängda dörrar och en öppen dörr med en get. Du vet att bakom en av de två stängda dörrarna finns det en bil och bakom den andra en get. Nu får du den avgörande frågan: Vill du hålla dig vid det alternativet du valde först eller vill du istället välja den andra stängda dörren. - Väljer du ditt ursprungliga val eller vill du byta dörr? -Har du större sannolikhet att vinna bilen om du byter dörr eller inte?

2. Om sannolikheterna

Enligt klassiska sannolikhetsdefinitionen är sannolikheten för en händelse A

$$P(A) = \frac{\text{antalet gynnsamma fall}}{\text{totala antalet fall}}.$$

T.ex. sannolikheten för en sexa vid ett tärningskast är 1/6. Med simulering kan man lätt prova sig fram och kolla hur många sexor får man med 10 kast, 20 kast, 50 kast, 100 kast o.s.v. Antag att man fått 18 stycken sexor vid 100 kast. Relativa frekvensen är då 18/100 = 0,18. Ökar man antalet tärningskast kommer relativa frekvensen att närma sig exakta sannolikheten 1/6. Detta kallas stora talens lag i sannolikhetsläran (sv.wikipedia.org/wiki/De_stora_talens_lag).

I Monty Hall-problemet är sannolikheten för att bilen finns bakom den valda dörren $\frac{1}{3}$, då man väljer att inte byta dörr. Då man väljer att byta dörr är sannolikheten att man vinner spelet $\frac{2}{3}$.

3. Grundläggande information om simulering

En simulation är en process som innehåller slumpmässighet så, att resultatet av processen varierar när det upprepas. Simulationer vill ofta efterlikna verkligheten och hör ihop med till exempel sannolikhetsberäkningar.

Simulering används i många sammanhang för att på ett enkelt sätt få fram en approximativ lösning. Simulering där man använder slumptal kallas även Monte Carlo-metoden (sv.wikipedia.org/wiki/Monte_Carlo-metod).

4. Simulering utan dator

Eleverna får i grupper simulera Monty Hall med hjälp av tre koppar och något föremål, till exempel en leksaksbil. Spelet spelas så att bilens position och valet av första koppen görs slumpmässigt med hjälp av en sexsidig tärning. En i gruppen väljs till spelledare och utför det som tärningskastet avgör. Först kastas tärningen och bilen placeras enligt ögontalet: 1-2 i första koppen, 3-4 i andra och 5-6 i tredje. Sedan kastas tärningen igen för att bestämma vilken kopp som väljs först, enligt samma system. Spelledaren öppnar till näst en tom kopp enligt spelets regler och väljer att byta kopp eller hålla samma kopp som tärningen valde i början. Till slut skriver gruppen upp om det blev en vinst eller en förlust.

Processen görs först 15 gånger så att gruppen **alltid** väljer att hålla samma kopp och bokför om det resulterar i vinst eller förlust. Till exempel kan man för varje försök dra streck under vinster eller förluster i en tabell.

Vinster	Förluster
II	IIII

Tabell 1. Exempel på hur tabellen kan se ut efter de sex första försöken när gruppen väljer att alltid hålla samma kopp som tärningen valde i början

Därefter görs samma sak på nytt, men nu så att gruppen **alltid** väljer att byta.

Totala antalet vinster från de olika fallen skrivs sedan upp i en ny tabell. Den relativa frekvensen, med andra ord vinstprocenten, räknas också ut genom att dividera antalet vinster med antalet försök.

Vinster	Förluster
IIII	II

Tabell 2. Exempel på hur tabellen kan se ut efter de sex första försöken när gruppen väljer att alltid byta kopp

Håller samma	Byter
Antal vinster	*Antal vinster*
Relativ frekvens	*Relativ frekvens*

Tabell 3. Exempel på hur den slutliga bokföringen kan se ut

Resultatet analyseras; lönar det sig att hålla samma kopp eller byta? Om gruppen är snabb kan flera försök göras för att komma närmare relativa frekvenserna 1/3 och 2/3. I slutet kan läraren med hjälp av den färdiga koden visa hur det skulle se ut med flera försök, till exempel 100, 1000 eller mera, för att övertyga eleverna om att det nog lönar sig att byta. Det är värt att nämna att summan av de relativa frekvenserna inte blir 1 eftersom två helt skilda försök görs.

5. Analys av simulationsprogram

Det här materialet innehåller python koden till ett simulationsprogram som simulerar Monty Hall-spelet med hjälp av slumptalsgeneratorer. Idén med materialet är att undersöker koden, gör ändringar och kör programmet för att svara på frågor. Övningarna lämpar sig i gymnasiets studieavsnitt MAA11.

Exempelövningar

1. Undersök hur de olika funktionerna `spel_utan_byte()` och `spel_med_byte()` fungerar.
2. Kalla på vardera funktion 20 gånger, och ha programmet att skriva ut det funktionen returnerar. Kan man ur utskriften avgöra om någondera strategi är bättre?
3. Använd en slinga för att simulera 1000 spel där man väljer att inte byta dörr, och 1000 spel där man väljer att byta dörr. Skapa en hjälpvariabel som håller reda på hur många gånger man vinner med vardera strategi. Lönar det sig att välja någondera strategi?
4. Använd en slinga för att simulera 10 000 spel där man väljer att inte byta dörr, och 10 000 spel där man väljer att byta dörr. Ha programmet att beräkna och skriva ut hur stor andel av gångerna man vinner spelet med vardera strategi. Hur mycket skiljer sig resultatet från de exakta sannolikheterna?

```python
import random

"""
Funktionen spel_utan_byte simulerar ett spel
där man väljer att inte byta dörr.
Funktionen returnerar 1 om vid vinst, och
annars 0.
"""
def spel_utan_byte():
    dorrar = ["bil", "get1", "get2"]
    random.shuffle(dorrar)
    val = random.choice(dorrar)
    if val == "bil":
        return 1
    return 0

"""
Funktionen spel_med_byte simulerar ett spel
där man väljer att byta dörr.
Funktionen returnerar 1 om vid vinst, och
annars 0.
"""
def spel_med_byte():
    dörrar = ["bil", "get1", "get2"]
    random.shuffle(dörrar)
    val = random.choice(dörrar)
    if val == "bil":
        oppnad_dorr = random.choice(["get1", "get2"])
    elif val == "get1":
        oppnad_dorr = "get2"
    else:
        oppnad_dorr = "get1"
    dörrar.remove(val)
    dörrar.remove(oppnad_dorr)
    val = dörrar[0]
    if val == "bil":
        return 1
    return 0

"""Skriv din kod nedanför för att simulera med
hjälp av funktionerna."""

#Exempel:
#for i in range(10):
#    print(spel_utan_byte())
```