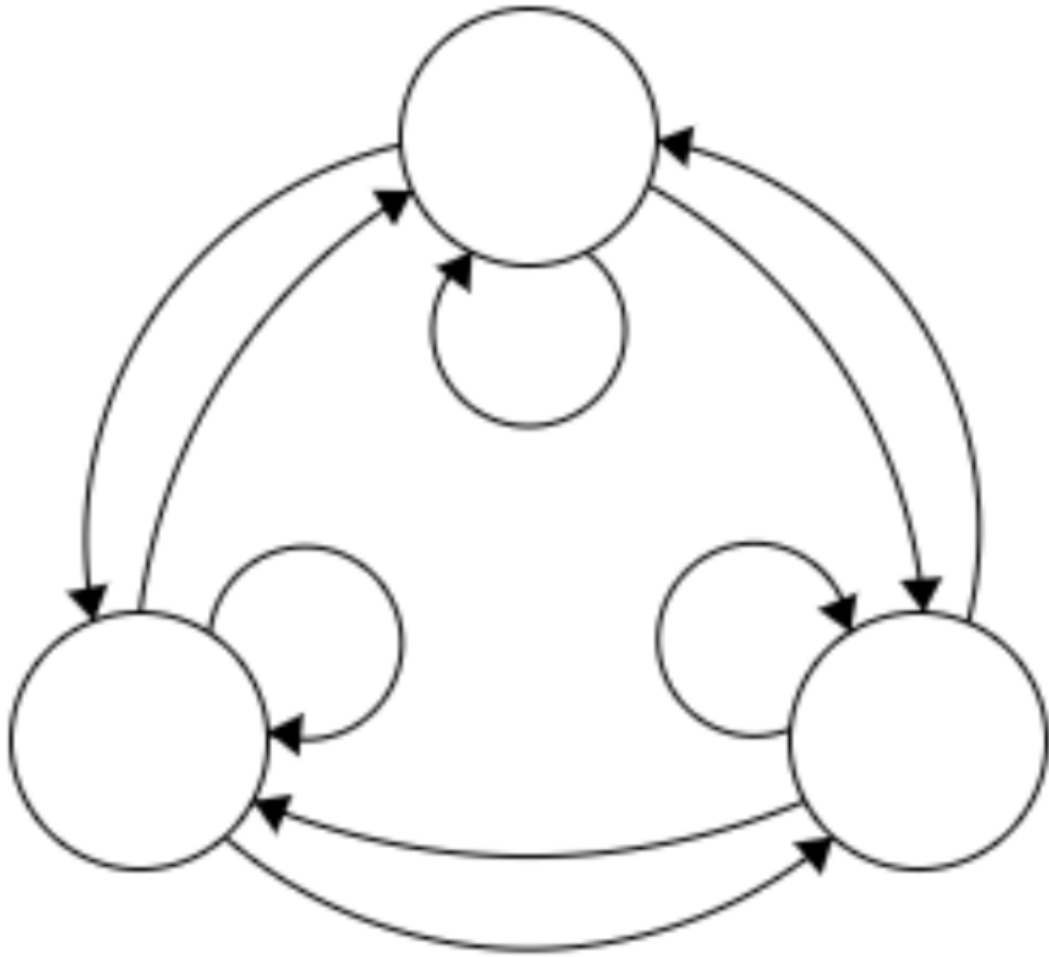


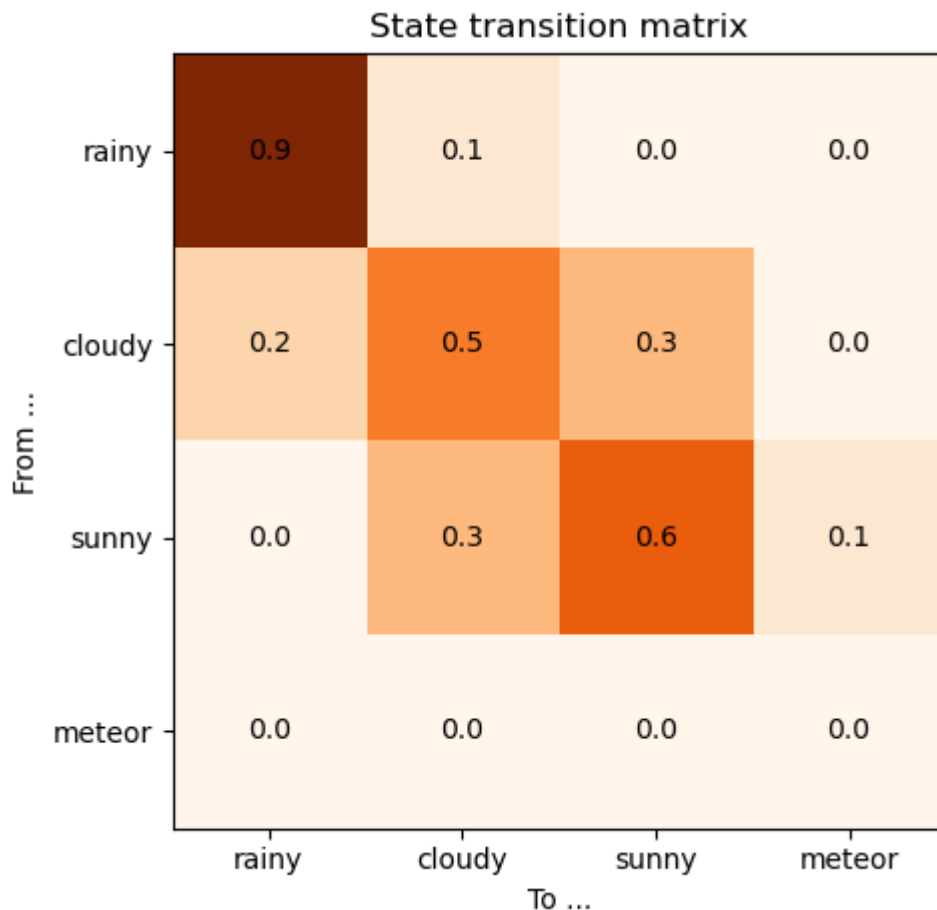
# Model-based prediction & control



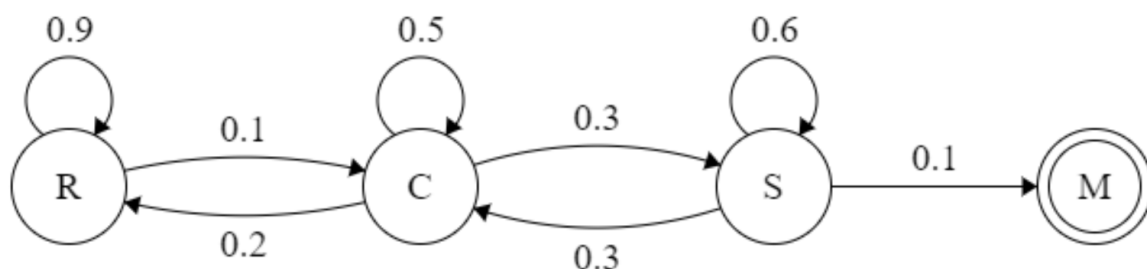
John Williams - 1790472

## A. Markov Chain

**Opdracht:** Teken een Markov Chain van 4 states aan de hand van deze state transition probability matrix  $\langle S, P \rangle$ ;  $S$  = states,  $P$  = probabilities. Er is geen specifieke beginstate. Er is wel een specifieke eindstate. Zorg ervoor dat visueel duidelijk is welke dit is.



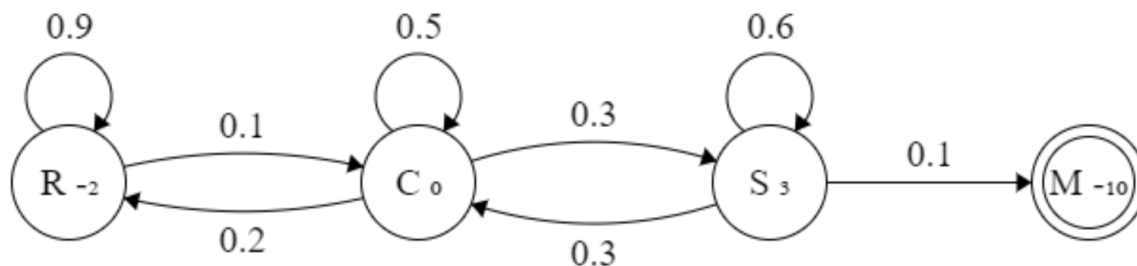
De state transition probability matrix in kwestie



De Markov Chain, vanuit de bovenstaande matrix. De state symbolen volgen uit de eerste letter van iedere state in de state transition probability matrix.

## B. Markov Reward Process

**Opdracht:** Maak van de Markov Process een Markov Reward Process. De states (van links naar rechts) hebben respectievelijk rewards -2, 0, 3 en -10. Zie het zo: naar de state 'rain' toe gaan geeft altijd een reward van -2, waar je ook vandaan komt. Het maakt voor de berekeningen niet uit of je zegt dat in een bepaalde state zijn of naar een bepaalde state toegaan de reward geeft. Dus doe wat voor jou intuïtief is.



De reward van iedere state is nu naast het symbool van de state geschreven.

## C. Sampling

**Opdracht:** Pak twee mogelijke samples van je MRP en leg uit wat de return was voor elke sample.

De formule voor het berekenen van de return:  $G_t = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1}$

De gekozen samples en de bijbehorende return zijn als volgt:

**CRCSS:**

$$\begin{aligned}
 G_t &= (1^1 \cdot 0) + (1^2 \cdot -2) + (1^3 \cdot 0) + (1^4 \cdot 3) + (1^5 \cdot 3) \\
 &= 0 + (-2) + 0 + 3 + 3 \\
 &= 4
 \end{aligned}$$

**RCCSM:**

$$\begin{aligned}
 G_t &= (1^1 \cdot -2) + (1^2 \cdot 0) + (1^3 \cdot 0) + (1^4 \cdot 3) + (1^5 \cdot -10) \\
 &= (-2) + 0 + 0 + 3 + (-10) \\
 &= -9
 \end{aligned}$$

Bij de bovenstaande berekeningen zijn de eerste regels de sommatie volledig uitgeschreven, waarbij ieder groep haakjes 1 iteratie is van de sommatie.

## D. Value Function

**Opdracht:** Bepaal nu voor alle states wat de value is na 2 iteraties. De value voor alle states worden geïnitieerd op 0. Gebruik hiervoor de Bellman expectation equation met  $\gamma = 1$ .

De Bellman expectation equation die gebruikt wordt is als volgt:

$$v_i = R + \gamma P v_{i-1}$$

Waarbij:

- $v_i$  = een vector met alle values van de states na  $i$  iteraties
- $R$  = een vector met alle rewards van de states
- $\gamma$  = een scalar, de discount (in dit geval gelijk aan 1)
- $P$  = een matrix, de state transition probability matrix

$$v_0 = \begin{bmatrix} 0 & \text{(rain)} \\ 0 & \text{(cloudy)} \\ 0 & \text{(sunny)} \\ 0 & \text{(meteor)} \end{bmatrix} \quad (\text{gegeven in de opdracht})$$

$$v_1 = \begin{bmatrix} -2 \\ 0 \\ 3 \\ -10 \end{bmatrix} + \gamma \begin{bmatrix} 0.9 & 0.1 & 0.0 & 0.0 \\ 0.2 & 0.5 & 0.3 & 0.0 \\ 0.0 & 0.3 & 0.6 & 0.1 \\ 0.0 & 0.0 & 0.0 & 0.0 \end{bmatrix} \cdot v_0$$

$$v_1 = \begin{bmatrix} -2 \\ 0 \\ 3 \\ -10 \end{bmatrix}$$

$$v_2 = \begin{bmatrix} -2 \\ 0 \\ 3 \\ -10 \end{bmatrix} + \gamma \begin{bmatrix} 0.9 & 0.1 & 0.0 & 0.0 \\ 0.2 & 0.5 & 0.3 & 0.0 \\ 0.0 & 0.3 & 0.6 & 0.1 \\ 0.0 & 0.0 & 0.0 & 0.0 \end{bmatrix} \cdot v_1$$

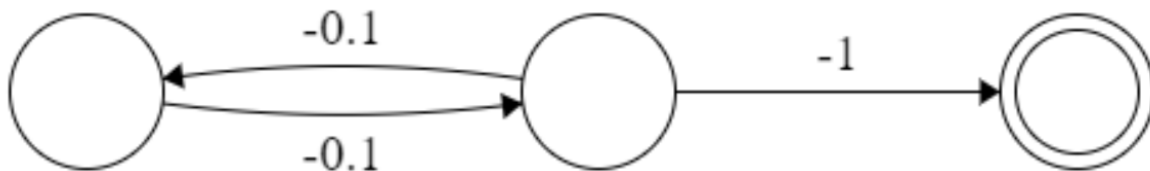
$$v_2 = \begin{bmatrix} -3.8 \\ 0.5 \\ 3.8 \\ -10.0 \end{bmatrix}$$

**Opdracht:** In werkelijkheid zien we vrijwel nooit een discount factor van  $\gamma = 1$ . Een discount factor van  $\gamma = 0.9$  is wel veelvoorkomend. Noem twee mogelijke problemen als  $\gamma = 1$ .

1. In het geval van oneindige processen met een discount factor van 1 betekent het dat de return oneindig optelt en dus oneindig groot wordt.
2. Als de discount altijd 1 blijft, dan heeft de reward van de huidige tijdstip even veel impact als de reward van een tijdstip ver in de toekomst. Dit is vaak ongewenst omdat in non-deterministische omgevingen de toekomst niet te voorspellen is, en dus niet evenveel waard is.

## E. Value Iteration

**Opdracht:** Bepaal de utility van elke state in de MDP. Je moet bepalen wanneer je stopt met itereren -- beargumenteer je keuze.



De MDP in kwestie

Er wordt alweer gebruik gemaakt van de Bellman expectation equation, zoals gebruikt bij onderdeel D hierboven. De rewards die je krijgt voor het gaan naar een state "s" worden in dit geval gebruikt als of het de reward van de state "s" zelf is. De discount  $\gamma$  is in dit geval alweer gelijk aan 1.

$$\begin{aligned}
 v_0 &= \begin{bmatrix} 0 & \text{(links)} \\ 0 & \text{(midden)} \\ 0 & \text{(rechts)} \end{bmatrix} \\
 v_1 &= \begin{bmatrix} -0.1 \\ -0.1 \\ -1.0 \end{bmatrix} + \gamma \begin{bmatrix} 0.0 & 1.0 & 0.0 \\ 0.5 & 0.0 & 0.5 \\ 0.0 & 0.0 & 0.0 \end{bmatrix} \cdot v_0 = \begin{bmatrix} -0.1 \\ -0.1 \\ -1.0 \end{bmatrix} \\
 v_2 &= \begin{bmatrix} -0.1 \\ -0.1 \\ -1.0 \end{bmatrix} + \gamma \begin{bmatrix} 0.0 & 1.0 & 0.0 \\ 0.5 & 0.0 & 0.5 \\ 0.0 & 0.0 & 0.0 \end{bmatrix} \cdot v_1 = \begin{bmatrix} -0.2 \\ -0.65 \\ -1.0 \end{bmatrix} \\
 &\vdots \\
 v_n &= \begin{bmatrix} -0.1 \\ -0.1 \\ -1.0 \end{bmatrix} + \gamma \begin{bmatrix} 0.0 & 1.0 & 0.0 \\ 0.5 & 0.0 & 0.5 \\ 0.0 & 0.0 & 0.0 \end{bmatrix} \cdot v_{n-1} = \begin{bmatrix} -1.4 \\ -1.3 \\ -1.0 \end{bmatrix}
 \end{aligned}$$

Het proces werd herhaaldelijk geïtereerd en na ongeveer 40 iteraties bleven de waardes, kijkend naar 5 significante cijfers, consistent.