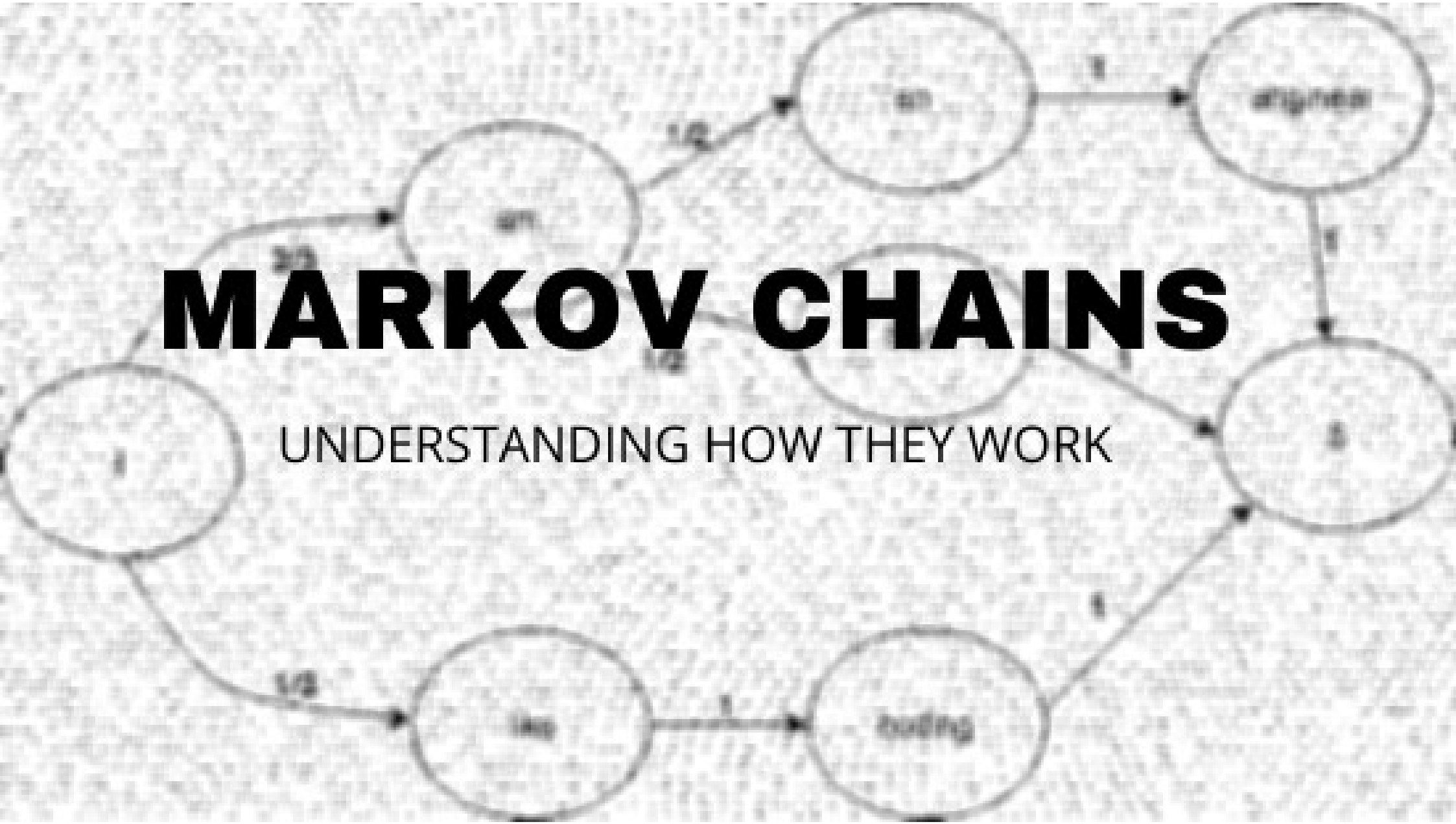
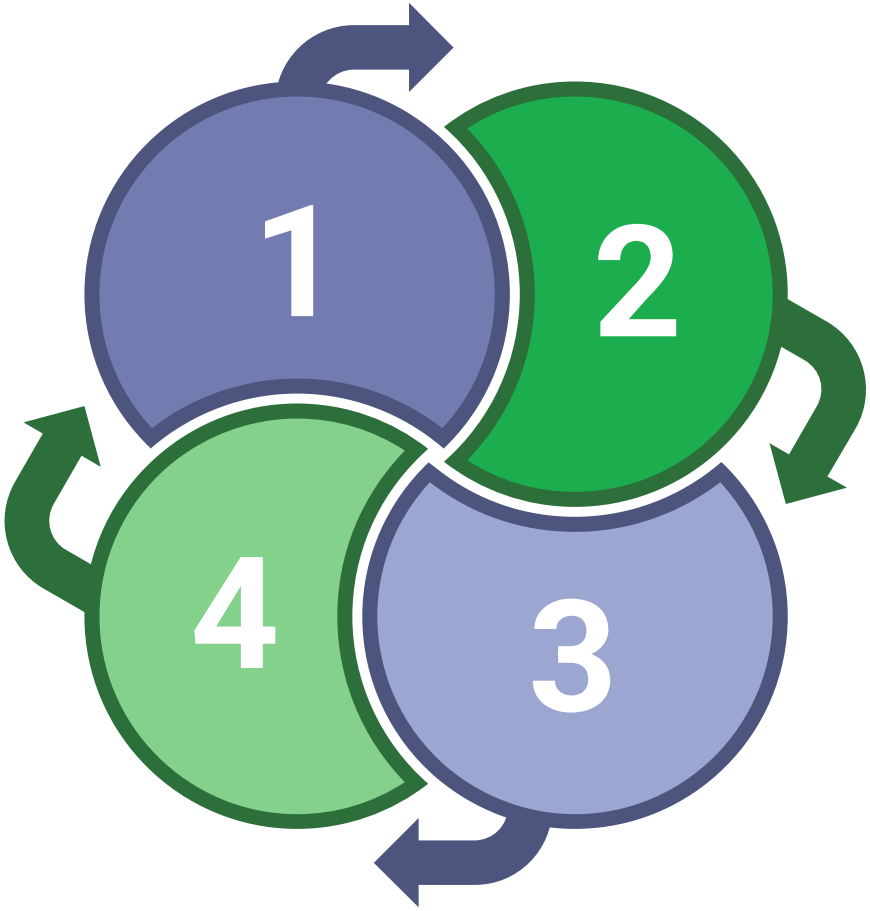

MARKOV CHAINS
UNDERSTANDING HOW THEY WORK

Contents

A brief overview

Formal definition

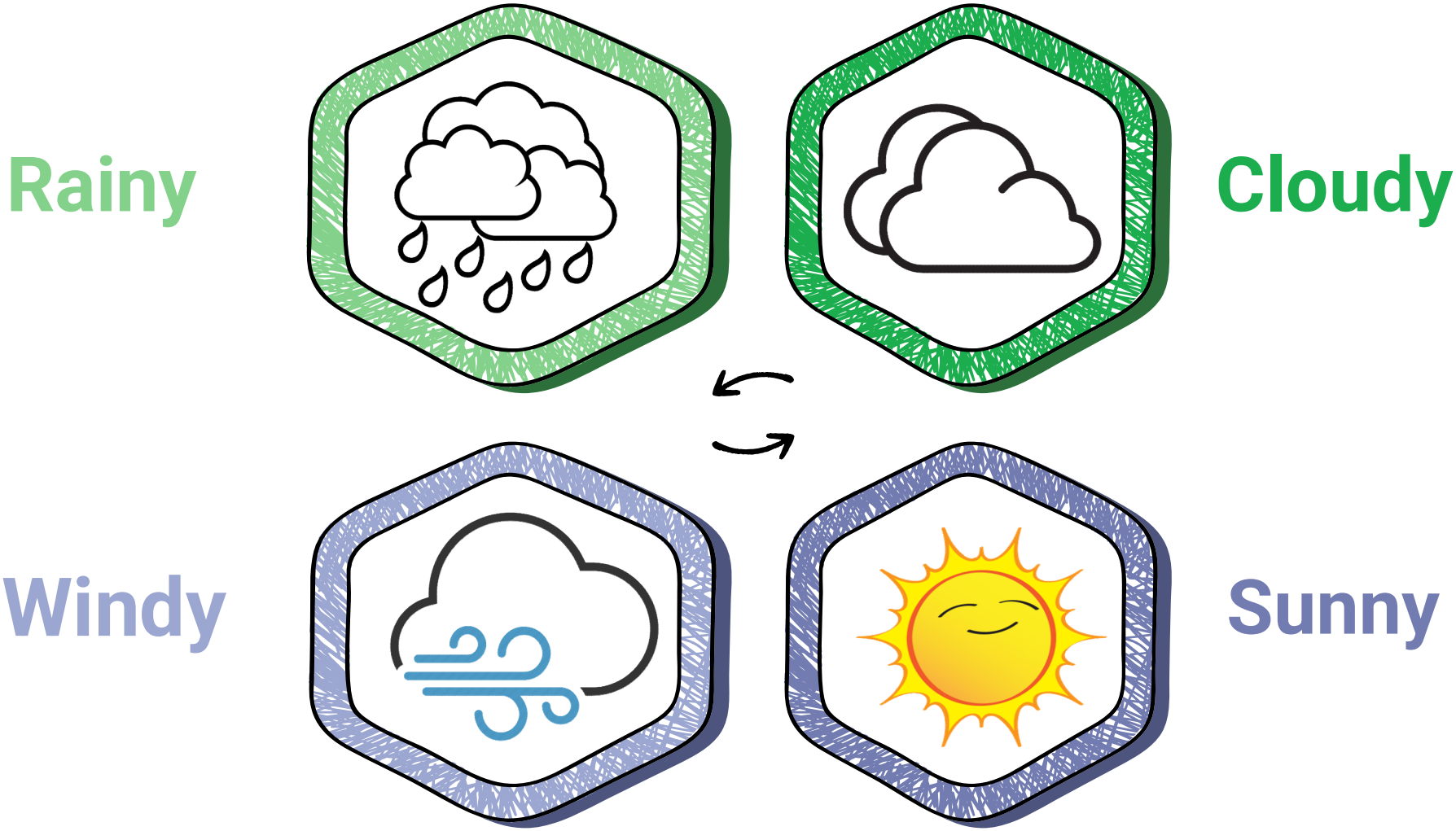


Applications in AI

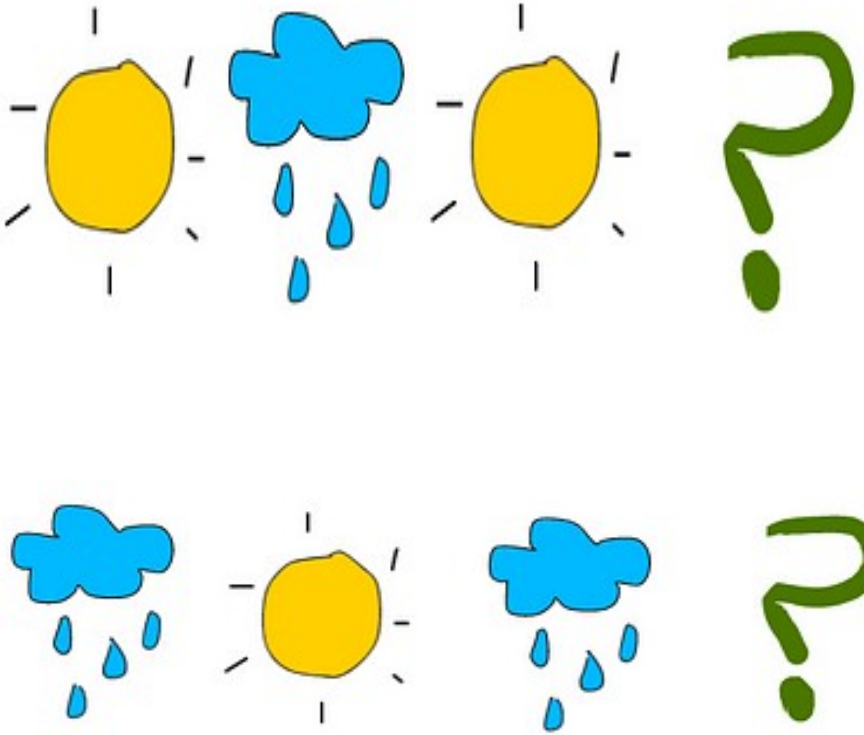
Transition Matrices

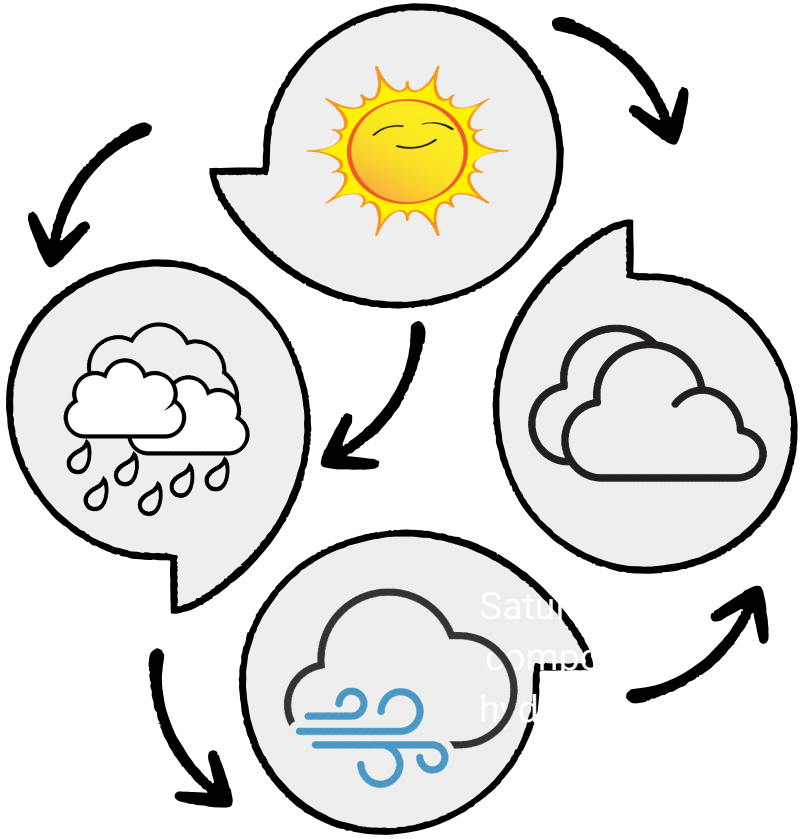
I. A brief overview

Weather states

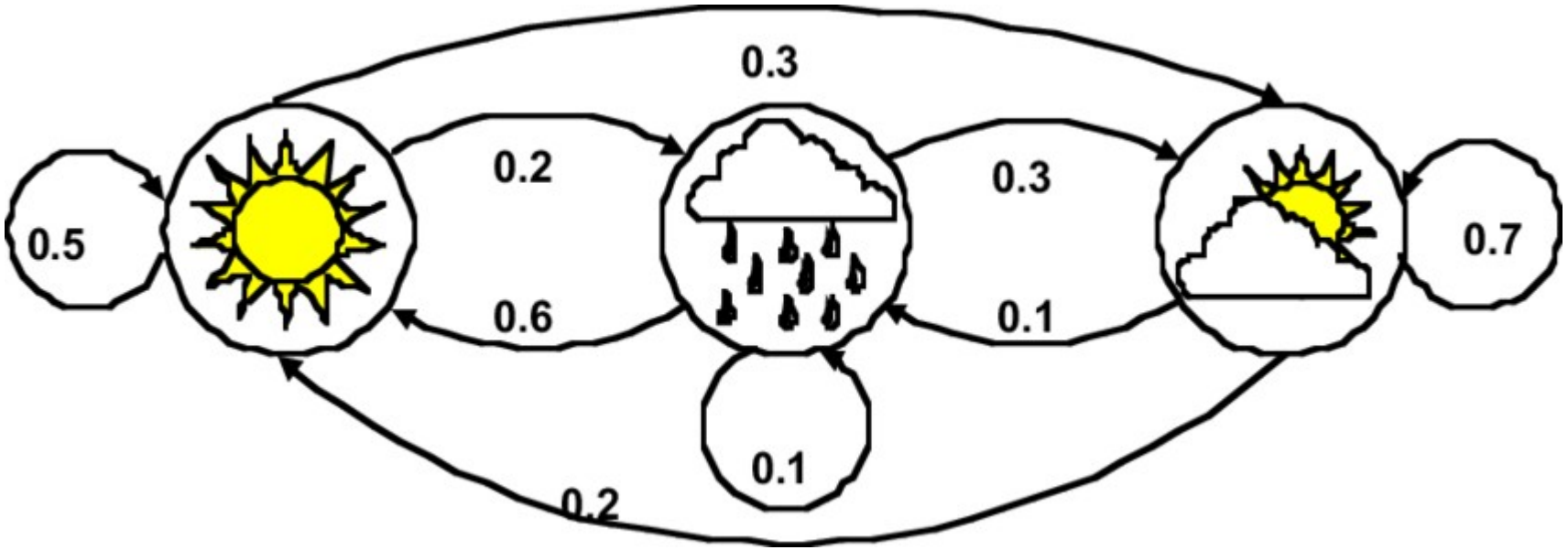


Weather forecasting? How it works?



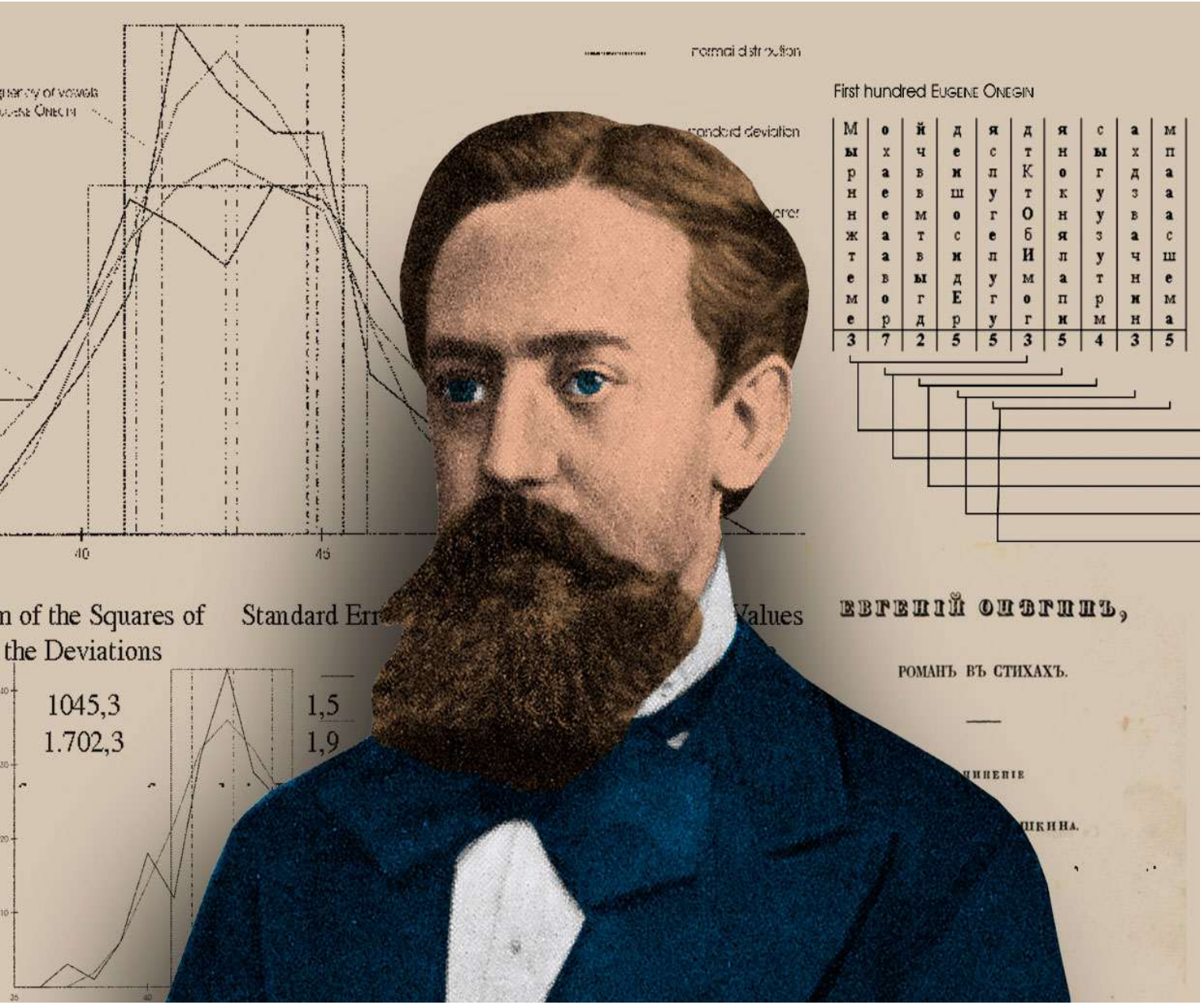


Weather forecasting? How it works?



Everything existing in the universe is the fruit of **chance**.

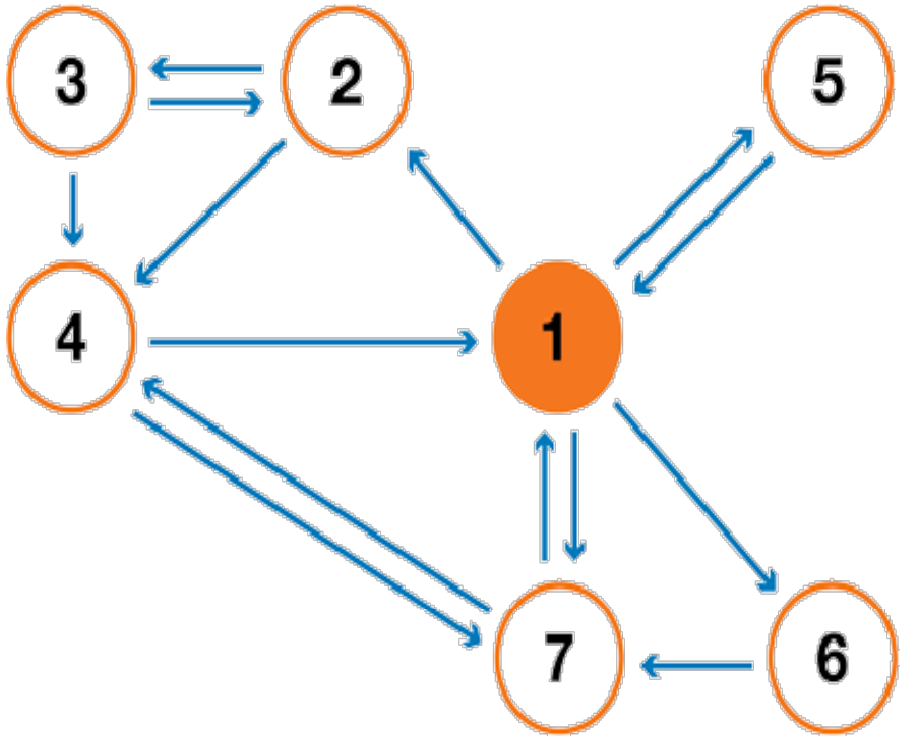
-- **Democritus**



II. Formal definition

Definition

A **Markov Chain** is a stochastic model describing a sequence of possible events in which the probability of each event depends only on the state attained in the previous event.

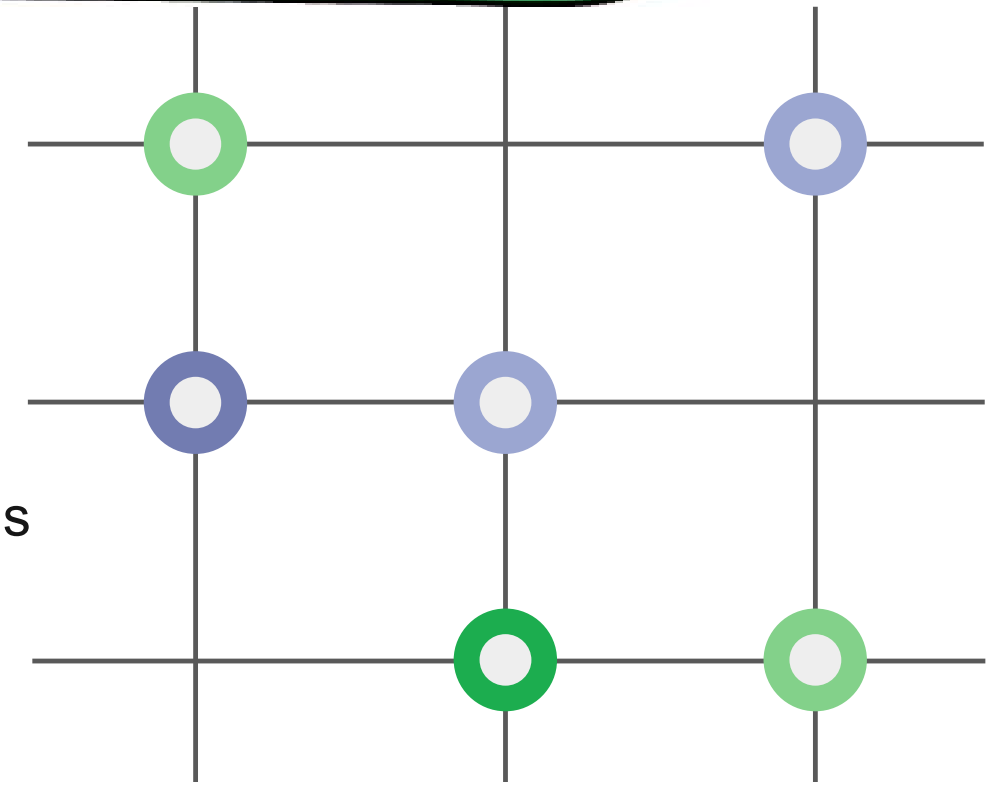


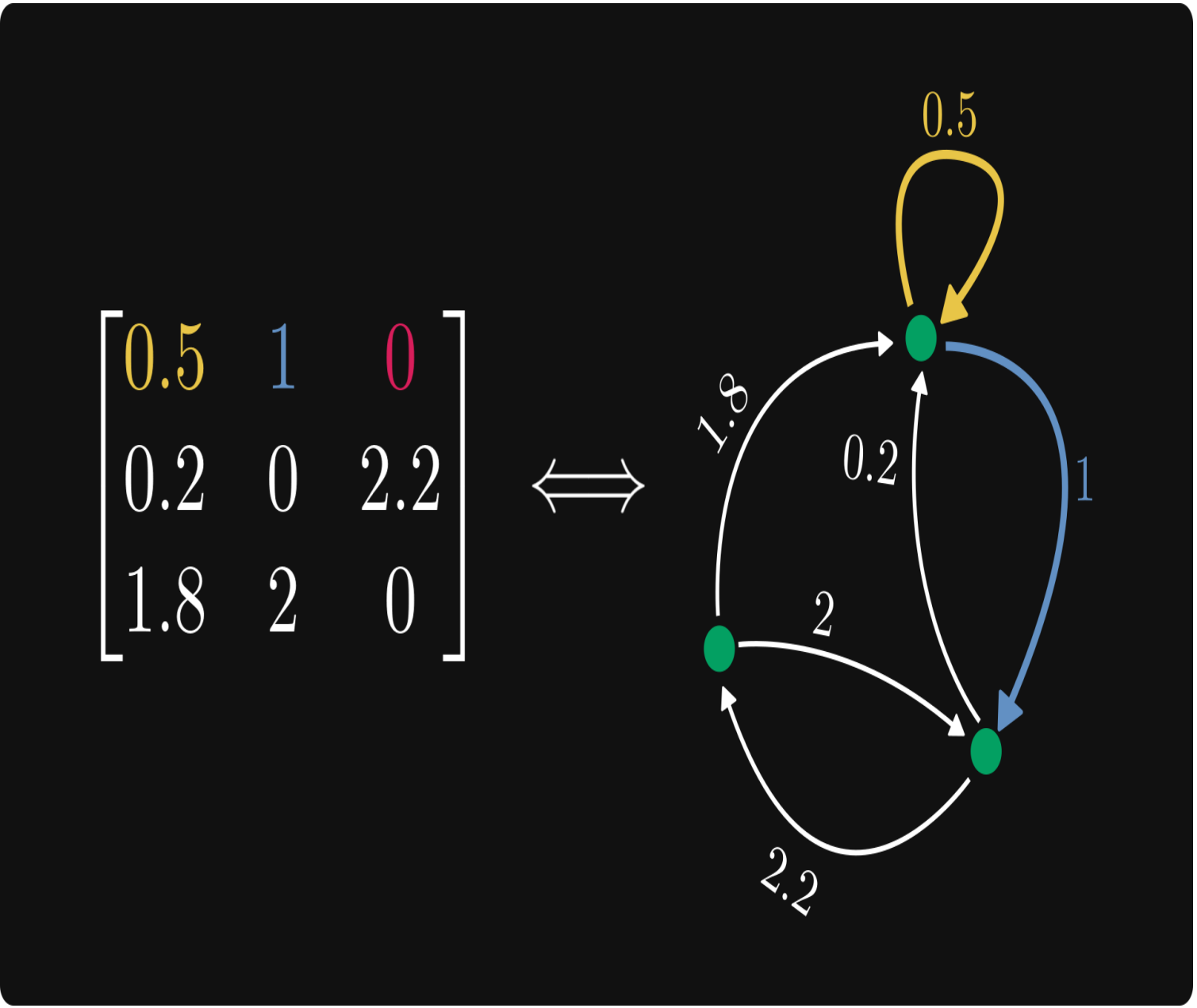
Trajectory : 1

The Markov Property: "memorylessness"

$$P(X_n=i_n \mid X_0=i_0, X_1=i_1, \ldots, X_{n-1}=i_{n-1}) = P(X_n=i_n \mid X_{n-1}=i_{n-1})$$

- For any positive **integer n** and possible **states** $\mathbf{i_0}, i_1, \ldots, i_n$ of the random variables
- Knowledge of the **previous state** is all that is necessary to determine the probability distribution of the **current state.**



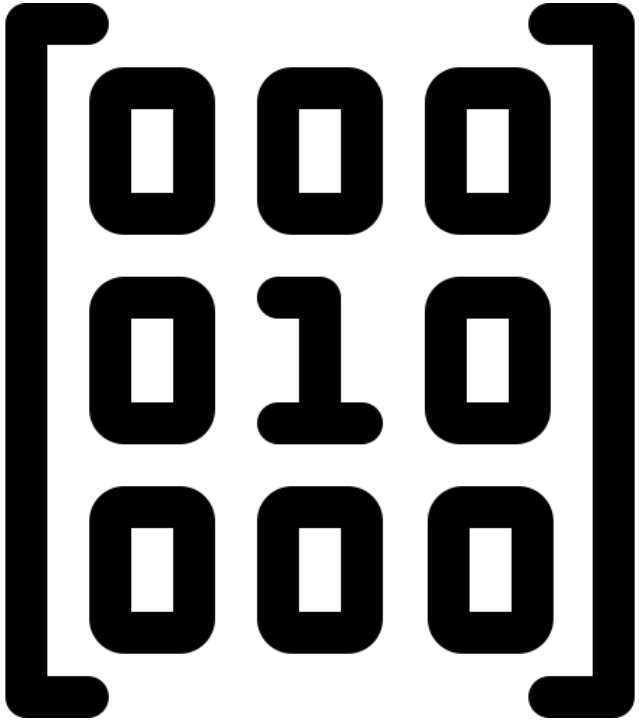


III. Transition matrices

Transition Matrices

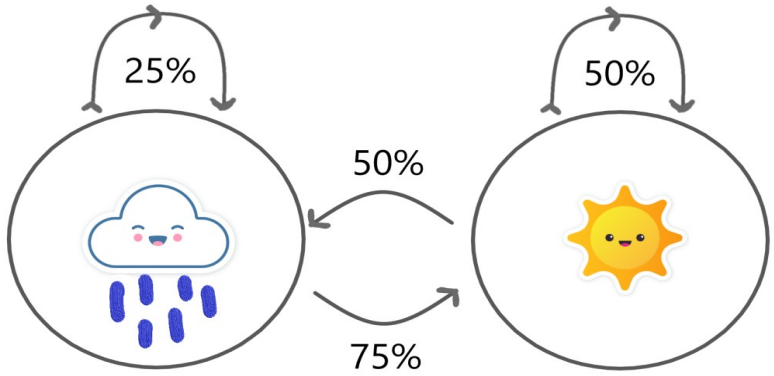
$$(P_t)_{i,j} = P(X_{t+1}=j \mid X_t=i).$$

- Contain information on the **probability of transitioning** between states.
- Each row of the matrix is a **probability vector**, and the sum of its entries is **1.**

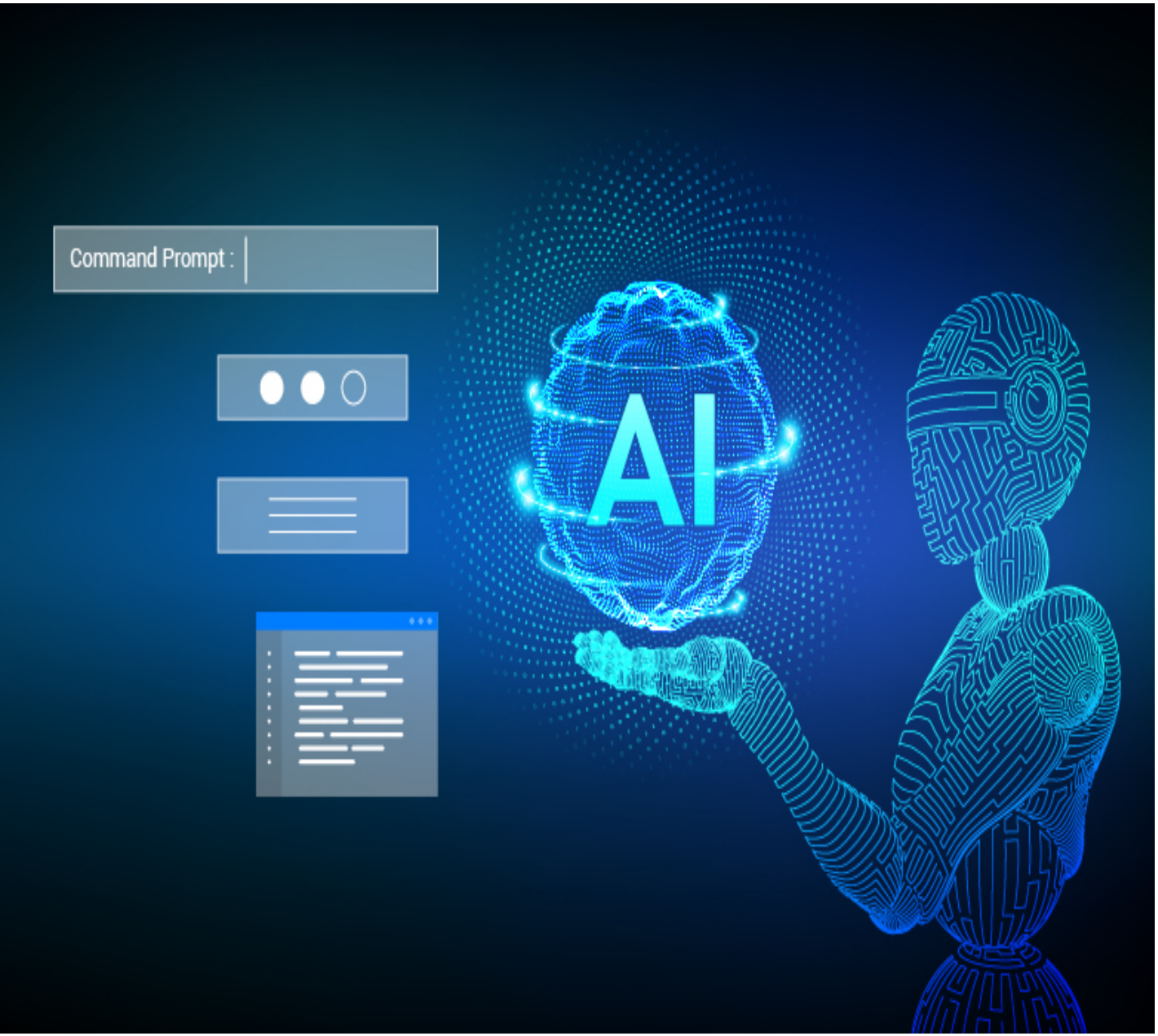


Transition matrice example

	Rainy	Sunny
Rainy	0.25	0.75
Sunny	0.5	0.5



Visit: https://setosa.io/ev/markov-chains/



IV. Applications

AI text generation

- Markov Chains can be used to **generate text** by **modeling the probability** of a word or character following another.

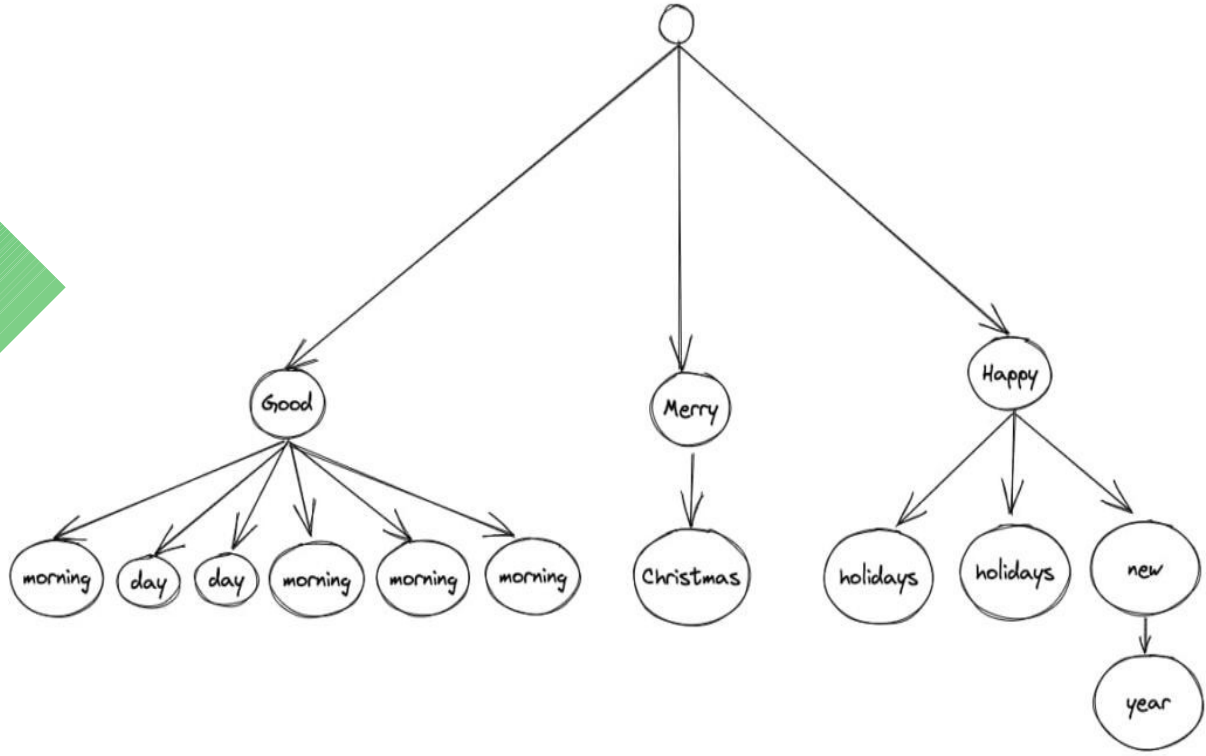
- By training the model on a large corpus of text, it can learn the probabilities of word or character sequences and **generate new text that resembles the training data.**

AI Text Generation

Imagine we have a list of standard greetings:

1. Good morning
2. Good day
3. Merry Christmas
4. Good morning
5. Good morning
6. Good day
7. Happy holidays
8. Happy New Year
9. Happy holidays
10. Good morning

We can **visualize** the them by connecting each word together in a **graph**,



The creative possibilities

Going through that exercise we can make a few conclusions:

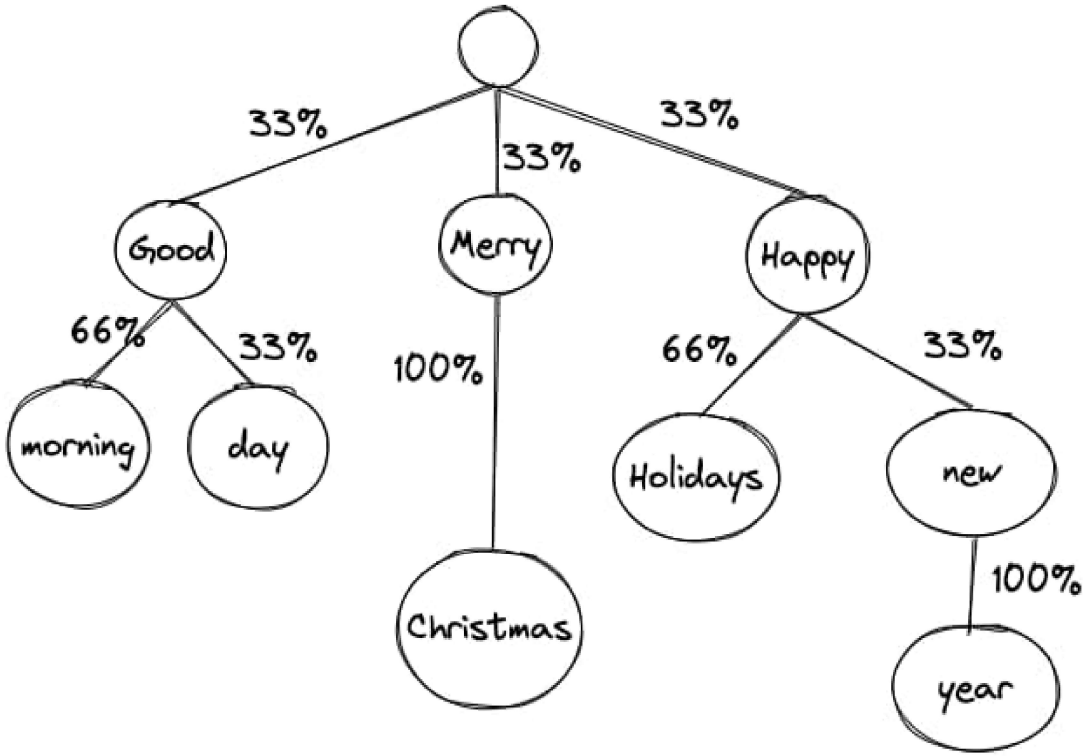
•There is a **1/3 chance** to start with “good”, “merry”, or “happy” respectively.

•If we say “good”, there is a **1/3 chance** we should say “day” and a **2/3 chance** we should say “morning”.

•If we say “merry”, we should say “Christmas”.

•If we say “happy”, there is a **2/3 chance** we should say “holidays” and a **1/3 chance** we should say “New Year”.

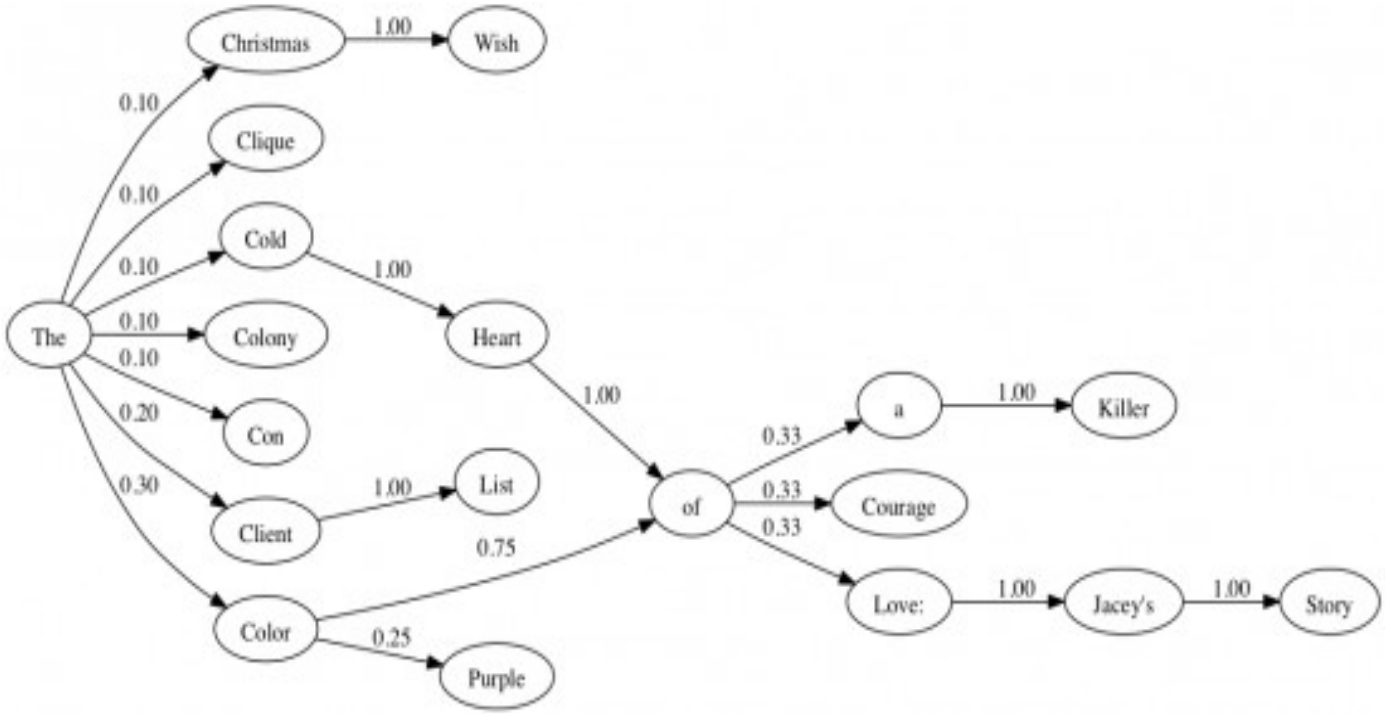
We redraw that graph to display those percentages, we get a new graph.



AI Text Generation

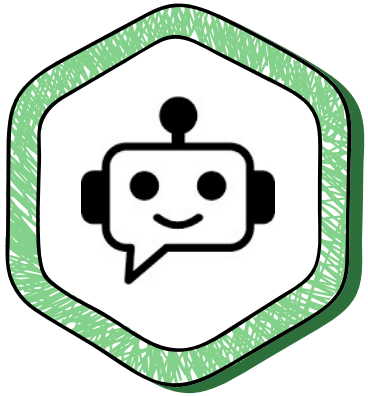
To build a **Markov chain algorithm** that can generate random text, we can follow these steps:

- Find a text **source**
- **Tokenize** the text
- Build the **Markov chain**
- **Traverse** the Markov chain

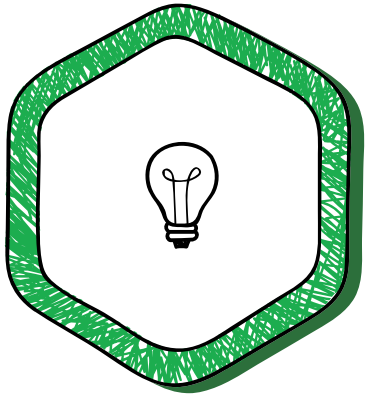


Visit: https://projects.haykranen.nl/markov/demo/

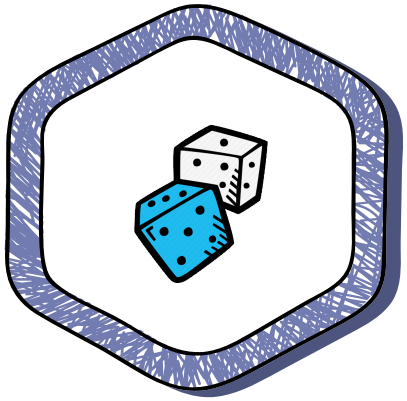
Common applications of Text Generation with Markov Chains



ChatBots

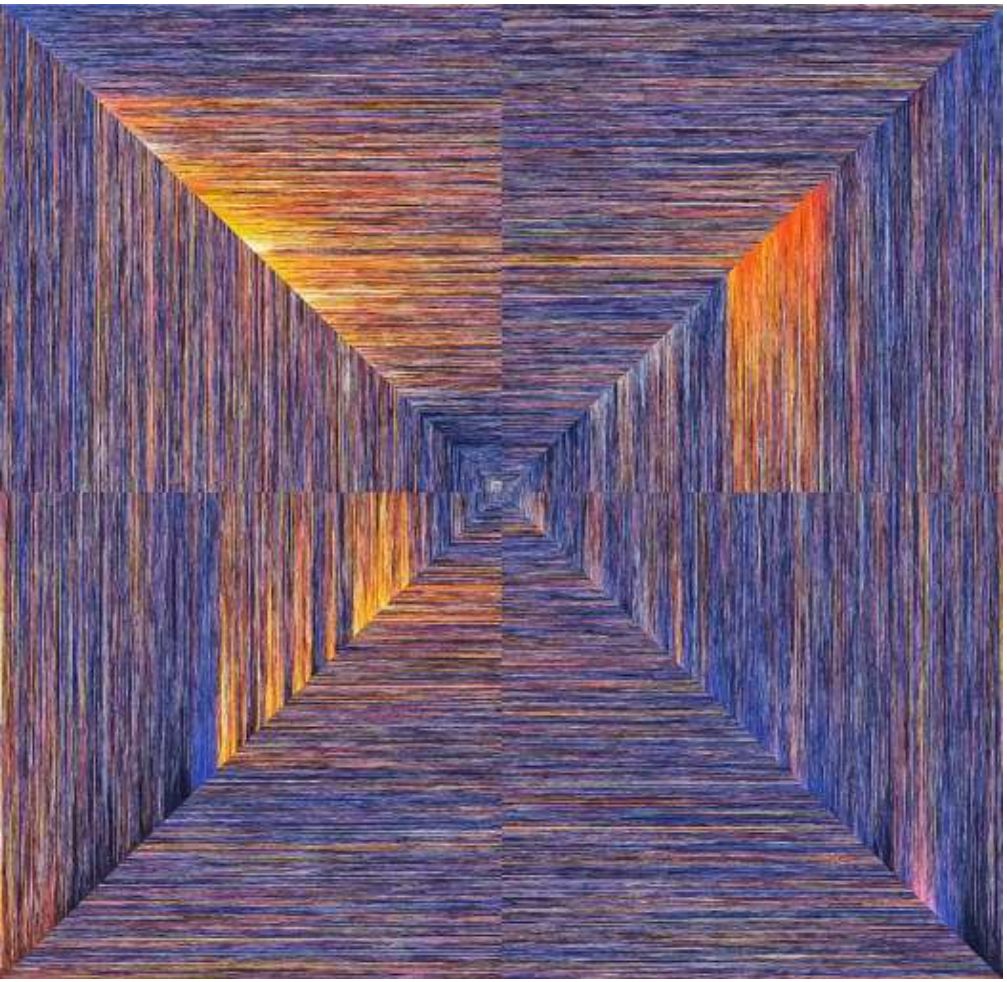


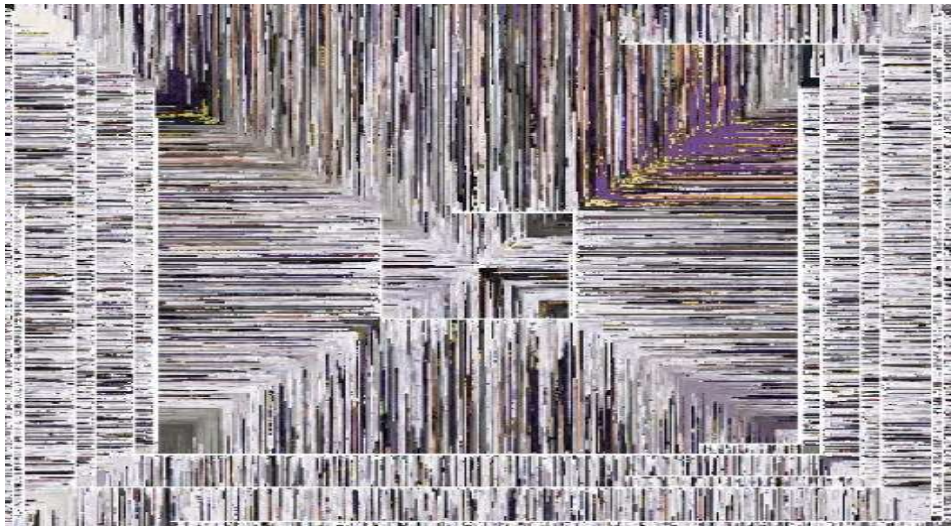
Creative story writing



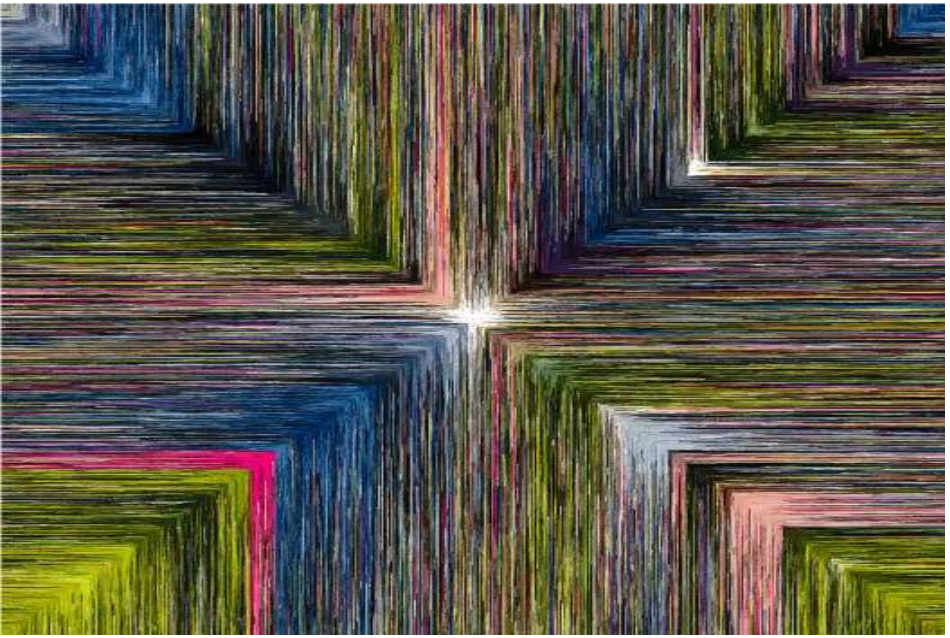
Just for fun !

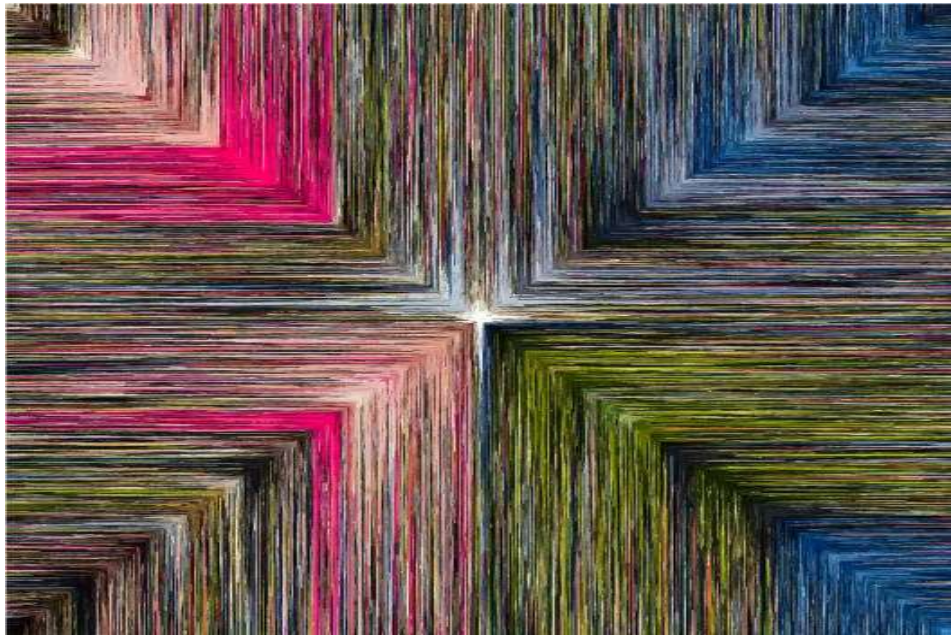
Mathematical Art





Visit: https://www.timswast.com/blog/2013/08/07/markov-chains-tutorial/





Conclusion