

# Documentação do Projeto: Modelo Oculto de Markov (O Mundo da Chuva)

## 1. Visão Geral do Projeto

Este projeto implementa um **Modelo Oculto de Markov (HMM - Hidden Markov Model)** para resolver um problema clássico de raciocínio temporal e probabilístico: o "Mundo da Chuva".

O cenário é o seguinte: você é um segurança em um prédio sem janelas. Você não sabe se está chovendo ou fazendo sol lá fora (o estado **oculto**). A única informação que você tem é observar se o diretor traz ou não um guarda-chuva quando chega ao trabalho (a **observação**).

O objetivo do sistema é inferir a probabilidade do clima real (Chuva ou Sol) e reconstruir a sequência climática mais provável com base apenas na sequência de observações dos guarda-chuvas ao longo dos dias.

## 2. Como Foi Desenvolvido

O projeto foi desenvolvido em **Python**, utilizando a biblioteca **NumPy** para realizar operações matriciais eficientes, essenciais para os algoritmos de HMM.

- **Linguagem:** Python.
- **Biblioteca Principal:** `numpy` (para manipulação de vetores de probabilidade e multiplicação de matrizes).
- **Estrutura do Modelo:**
  - **Estados Ocultos (X):** `Rainy` (Chuva) e `Sunny` (Sol).
  - **Observações (E):** `Yes` (Trouxe guarda-chuva) e `No` (Não trouxe).
  - **Modelo de Transição P(Xt | X{t-1}):** A probabilidade do clima mudar de um dia para o outro (ex: se choveu ontem, há 70% de chance de chover hoje).
  - **Modelo de Sensor (P(Et | X\_)):** A probabilidade de observar um guarda-chuva dado o clima atual (ex: se chove, há 90% de chance do diretor trazer guarda-chuva).

## 3. Algoritmos Utilizados

O código implementa quatro tarefas fundamentais de inferência em HMMs:

### 1. Filtragem (*Filtering*) - Algoritmo Forward

Responde à pergunta: "Qual a chance de estar chovendo **hoje**, dadas todas as observações até agora?"

- O algoritmo calcula  $P(X_t | e_{1:t})$  recursivamente.
- Ele combina a estimativa do dia anterior (predição) com a evidência do dia atual (atualização).

## 2. Suavização (*Smoothing*) - Algoritmo **Forward-Backward**

Responde à pergunta: "Qual era a chance de ter chovido **ontem** (ou dias atrás), sabendo tudo o que sei hoje?"

- Utiliza dados futuros para melhorar a estimativa do passado.
- Combina o resultado do **Forward** (passado até  $t$ ) com um passo **Backward** (futuro até  $t$ ), resultando em uma estimativa muito mais precisa do que a filtragem sozinha.

## 3. Caminho Mais Provável - Algoritmo de **Viterbi**

Responde à pergunta: "Qual a sequência completa de dias de chuva e sol que melhor explica essa sequência de guarda-chuvas?"

- Não olha apenas para cada dia isoladamente, mas busca a **sequência global** mais coerente (Caminho de Viterbi).
- Utiliza programação dinâmica para maximizar  $P(x_{1:t} | e_{1:t})$ .

## 4. Referências Bibliográficas

Russell, S. J., & Norvig, P. (2021). *Artificial Intelligence: A Modern Approach*. 4<sup>a</sup> ed. (Capítulo sobre Probabilistic Reasoning over Time).