Mathematics of Transition Matrices and Expected Steps in YAM

Contents

1 Introduction **1**

2 Model Framework **1**
2.1 Directed Graph Representation . . . 1
2.2 Repertoire and Learnability . . . 2
2.3 Learning Strategies . . . 2

3 Transition Matrices **2**
3.1 Transition Probabilities . . . 2
3.1.1 Explanation . . . 3
3.2 Construction of the Transition Matrix . . . 3

4 Markov Chain Analysis **3**
4.1 Transient and Absorbing States . . . 3
4.2 Canonical Form of the Transition Matrix . . . 3
4.3 Fundamental Matrix . . . 4

5 Expected Steps Until Absorption **4**
5.1 Calculation of Expected Steps . . . 4
5.2 Derivation . . . 4
5.3 Initial State . . . 4

1 Introduction

This document presents the mathematical framework used to calculate transition matrices and the expected number of steps until absorption in YAM (Yet Another Model). The model considers how agents learn traits by moving through a state space defined by their repertoire of known traits. The focus is on the construction of the transition matrices and the computation of the expected steps until the agent learns all available traits.

2 Model Framework

2.1 Directed Graph Representation

We consider a directed graph $G = (V, E)$, where:

- $V = \{0, 1, \ldots, n-1\}$ is a set of n traits.
- $E \subseteq V \times V$ is a set of directed edges representing prerequisite relationships between traits.

The adjacency matrix $A \in \{0,1\}^{n\times n}$ represents the graph structure:

$$A_{ij} = \begin{cases} 1 & \text{if there is a directed edge from trait } i \text{ to trait } j, \\ 0 & \text{otherwise.} \end{cases}$$

The set of parent traits (prerequisites) for trait j is:

$$P_j = \{i \in V : A_{ij} = 1\}.$$

2.2 Repertoire and Learnability

An agent's repertoire is represented by a vector $r \in \{0,1\}^n$, where:

$$r_j = \begin{cases} 1 & \text{if trait } j \text{ is known to the agent,} \\ 0 & \text{otherwise.} \end{cases}$$

The set of unlearned traits is:

$$U(r) = \{j \in V : r_j = 0\}.$$

A trait j is *learnable* at state r if all its parent traits are known:

$$L_j(r, A) = (1 - r_j) \cdot \prod_{i \in P_j} r_i.$$

Thus, $L_j(r, A) = 1$ if trait j is unlearned and all its prerequisites are learned.

2.3 Learning Strategies

An agent follows a learning strategy s, which determines the probability weights assigned to each unlearned trait.

Base Weights The base weight vector $\mathbf{w}^*(s) \in \mathbb{R}^n$ depends on the strategy s:

$$w_j^*(s) = \begin{cases} 1 & \text{for Random Learning,} \\ p_j & \text{for Payoff-Based Learning,} \end{cases}$$

where $p_j \geq 0$ represents the payoff associated with trait j.

Adjusted Weights At state r, the adjusted weights assigned to unlearned traits are:

$$w_j(s, r) = \begin{cases} \dfrac{w_j^*(s)}{\sum_{k \in U(r)} w_k^*(s)} & \text{if } j \in U(r), \\ 0 & \text{otherwise.} \end{cases}$$

3 Transition Matrices

3.1 Transition Probabilities

From state r, the probability of transitioning to state r' under strategy s is defined as:

$$P(r \to r' | s) = \begin{cases} w_j(s, r), & \text{if } r' = r + e_j \text{ and } L_j(r, A) = 1, \\ 1 - \sum\limits_{k \in L(r,A)=1} w_k(s, r), & \text{if } r' = r, \\ 0, & \text{otherwise,} \end{cases}$$

where e_j is the unit vector with 1 at position j and 0 elsewhere.

3.1.1 Explanation

At each time step, the agent attempts to learn one unlearned trait according to the weights $w(s,r)$. The possible outcomes are:

1. If the agent attempts a learnable trait j (i.e., $L_j(r, A) = 1$), it successfully learns it and transitions to $r' = r + e_j$.
2. If the agent attempts an unlearnable trait, the attempt fails, and the state remains r.
3. No other transitions are possible in one time step.

3.2 Construction of the Transition Matrix

The set of all possible states (repertoires) is finite and can be enumerated. Let the state space be represented as $\mathcal{S} = \{r^{(1)}, r^{(2)}, \ldots, r^{(m)}\}$, where $m = 2^n$.

The transition matrix $P \in \mathbb{R}^{m \times m}$ is constructed by computing $P_{ij} = P(r^{(i)} \to r^{(j)}|s)$ for all pairs of states.

Example For each state $r^{(i)}$, we:

1. Compute the learnability vector $L(r^{(i)}, A)$.
2. Determine the adjusted weights $w(s, r^{(i)})$.
3. For each trait j such that $L_j(r^{(i)}, A) = 1$, compute the transition to $r' = r^{(i)} + e_j$ with probability $w_j(s, r^{(i)})$.
4. The probability of staying in state $r^{(i)}$ is the sum of probabilities of unsuccessful attempts: $1 - \sum_k P(r^{(i)} \to r^{(i)} + e_k|s)$.

By repeating this process for all states, we fill the transition matrix P.

4 Markov Chain Analysis

The transition matrix P defines a Markov chain over the state space $\mathcal{S}$.

4.1 Transient and Absorbing States

A state r is *absorbing* if $P(r \to r|s) = 1$, i.e., the agent remains in that state with probability 1. In our model, the absorbing states correspond to repertoires where all traits are learned (i.e., $r_j = 1$ for all j).

States that are not absorbing are called *transient* states.

4.2 Canonical Form of the Transition Matrix

By reordering the states, we can write P in canonical form:

$$P = \begin{pmatrix} Q & R \\ \mathbf{0} & I \end{pmatrix},$$

where:

- Q is the submatrix of transition probabilities between transient states.

- R is the submatrix of transition probabilities from transient to absorbing states.
- I is the identity matrix corresponding to absorbing states.
- $\mathbf{0}$ is a zero matrix.

4.3 Fundamental Matrix

The fundamental matrix N is defined as:

$$N = (I - Q)^{-1}.$$

The entry N_{ij} represents the expected number of times the process is in transient state j starting from transient state i.

5 Expected Steps Until Absorption

5.1 Calculation of Expected Steps

Let $\mathbf{t}$ be the column vector where t_i is the expected number of steps to absorption starting from transient state i. Then,

$$\mathbf{t} = N\mathbf{1},$$

where $\mathbf{1}$ is a column vector of ones.

5.2 Derivation

Since each transient state contributes one step towards the expected total steps, the expected total steps starting from state i is the sum over all transient states:

$$t_i = \sum_j N_{ij} = (N\mathbf{1})_i \, .$$

5.3 Initial State

Assuming the agent starts from an initial repertoire $r^{(0)}$, we identify its index in the reordered transient state list and compute t_0, the expected number of steps until absorption from the initial state.