Linear Algebra-Markov Chains

What is a Markov Chain?

A Markov Chain is a mathematical system that undergoes transitions from one state to another based on a set of probabilities. It's a sequence of states, where the probability of transitioning from one state to another is dependent only on the current state and not on any other factors.

Linear Algebra Applications:

Markov Chains are closely related to Linear Algebra, particularly in the context of:

1. **Stochastic Matrices:** A Markov Chain can be represented by a stochastic matrix (also known as a probability matrix), where each row represents a probability distribution over all possible states.
2. **Eigenvalues and Eigenvectors:** The stationary distribution of a Markov Chain, which is the long-term behavior of the system, is related to the eigenvalues and eigenvectors of the transition matrix.

Example: Google's PageRank Algorithm

One famous example of Markov Chains in Linear Algebra is Google's PageRank algorithm. Suppose we have a web graph with N nodes (web pages), where each node has an outgoing link to another node. We can represent this as a stochastic matrix A, where:

- `a_ij = 1` if node i links to node j
- `a_ij = 0` otherwise

The PageRank of each node is then calculated using the formula:

`PR_i = (1 - d) / N + d * Σ (PR_j / C_j)`

where PR_i is the PageRank of node i, d is a damping factor (usually set to 0.85), and `C_j` is the number of outgoing links from node j.

The transition matrix A has some interesting properties:

- It's stochastic: each row sums up to 1.
- It's irreducible: it's possible to reach any state from another state, either directly or indirectly.
- It has a unique stationary distribution, which represents the PageRank of each node.

Linear Algebra Operations on Markov Chains

Some common Linear Algebra operations used on Markov Chains include:

- Matrix multiplication: `A * B` combines the transition matrices of two Markov Chains.
- Eigenvalue decomposition: this is used to find the stationary distribution of a Markov Chain.
- Singular Value Decomposition (SVD): this can be used to analyze the structure of a Markov Chain.

Example Python Code

Here's an example of using NumPy and SciPy to compute the PageRank of a small web graph:

```
 import numpy as np

# Define the web graph as an adjacency matrix
A = np.array([[0, 1, 0],
              [1, 0, 1],
              [0, 1, 0]])

# Define the damping factor (d)
d = 0.85

# Compute the PageRank using eigenvalue decomposition
PR = np.linalg.eigvals(A.T).real[0] * np.ones((3,))
PR /= np.sum(PR)

print("PageRank:", PR)
```

This example uses the `numpy` library to compute the PageRank of a small web graph. Note that this is a simplified version of the actual Google PageRank algorithm, which involves many more steps and considerations.

I hope this summary provides a good overview of Markov Chains in Linear Algebra!

Curated by Brajesh Kumar