The Wilsonian Flux

Issue 3 / March 2025 - Maths

Welcome to Issue 3 of the Wilsonian Flux and a warm welcome to all the mathematicians. The Wilsonian Flux aims to introduce content that you may find interesting, and that will hopefully encourage you to learn more and challenge yourself. The maths this week focuses on linear algebra, as well as Markov chains, which are stochastic models useful in probability.

Contents

Figure 1: Portrait of Carl Friedrich Gauss

1 Solving systems of linear equations using matrices

Written by **Adhrit, Year 12**

1.1 Introduction

Solving linear sets of simultaneous equations is an area of maths that appears everywhere in maths. Therefore, it is important to get a strong grasp and understanding of how their solutions can be computed in different ways. Matrices provide a clear, concise way of solving these systems of equations, and we'll explore two methods in this article.

1.2 Gauss-Jordan Elimination

Perhaps the more commonly known method, the essence of this method is to use row eliminations to find the solutions to a system of linear equations. Let me demonstrate; take the system:

$$\begin{aligned} a_1x_1 + a_2x_2 + a_3x_3 &= b_1 \\ a_4x_1 + a_5x_2 + a_6x_3 &= b_2 \\ a_7x_1 + a_8x_2 + a_9x_3 &= b_1 \end{aligned}$$

where a_n and b_n are constants and x_n are variables. This can be written in the form of an augmented matrix:

$$\left[\begin{array}{ccc|c} a_1 & a_2 & a_3 & b_1 \\ a_4 & a_5 & a_6 & b_2 \\ a_7 & a_8 & a_9 & b_3 \end{array}\right]$$

where the solid line in the middle can also be dotted or omitted completely.

The end goal of this method is to to end up with a matrix of the form:

$$\left[\begin{array}{ccc|c} 1 & 0 & 0 & y_1 \\ 0 & 1 & 0 & y_2 \\ 0 & 0 & 1 & y_3 \end{array}\right]$$

where y_n will be the solution for x_n. This form of the matrix is called the row-echelon form. How do we get there? Through row eliminations. Simply put, we want to add or subtract multiples of one row from another to replace that row with something different, and keep going until we achieve row-echelon form. There are 3 types of row operations:

- Swapping the position of two rows
- Multiplying a row by a scalar value
- Add or subtract the scalar multiple of one row from another

Take the system of equations:

$$\begin{aligned} 2x + 3y &= 13 \\ 4x + 5y &= 23 \end{aligned}$$

This can be written as

$$\left[\begin{array}{cc|c} 2 & 3 & 13 \\ 4 & 5 & 23 \end{array}\right]$$

We can now subtract 2 lots of row one ($2R_1$) from row two (R_2), to get:

$$\left[\begin{array}{cc|c} 2 & 3 & 13 \\ 0 & -1 & -3 \end{array}\right]$$

Now multiply the second row by -1:

$$\left[\begin{array}{cc|c} 2 & 3 & 13 \\ 0 & 1 & 3 \end{array}\right]$$

Followed by $R_1 - 3R_2$:

$$\left[\begin{array}{cc|c} 2 & 0 & 4 \\ 0 & 1 & 3 \end{array}\right]$$

And finally dividing R_1 by 2 to achieve row echelon form and obtaining our solutions:

$$\left[\begin{array}{cc|c} 1 & 0 & 2 \\ 0 & 1 & 3 \end{array}\right]$$

So we see that $x = 2$ and $y = 3$.

We can apply a similar method to solve a system with 3 variables and 3 equations:

$$\begin{aligned} x + 2y + 3z &= 11 \\ 3x + 7y + 11z &= 8 \\ 2x + 4y + 7z &= 2 \end{aligned}$$

which is represented as

$$\left[\begin{array}{ccc|c} 1 & 2 & 3 & 1 \\ 3 & 7 & 11 & 8 \\ 2 & 4 & 7 & 2 \end{array}\right]$$

Subtracting R_3 then R_1 from R_2:

$$\left[\begin{array}{ccc|c} 1 & 2 & 3 & 1 \\ 0 & 1 & 1 & 5 \\ 2 & 4 & 7 & 2 \end{array}\right]$$

Then $R_3 - 2R_1$:

$$\left[\begin{array}{ccc|c} 1 & 2 & 3 & 1 \\ 0 & 1 & 1 & 5 \\ 0 & 0 & 1 & 0 \end{array}\right]$$

Followed by $R_2 - R_3$:

$$\left[\begin{array}{ccc|c} 1 & 2 & 3 & 1 \\ 0 & 1 & 0 & 5 \\ 0 & 0 & 1 & 0 \end{array}\right]$$

And finally $R_1 - 2R_2 - 3R_3$:

$$\left[\begin{array}{ccc|c} 1 & 0 & 0 & -9 \\ 0 & 1 & 0 & 5 \\ 0 & 0 & 1 & 0 \end{array}\right]$$

Giving us $x = -9$, $y = 5$ and $z = 0$.

Admittedly, the operations required may at first seem rather random, but developing an intuition for how to most efficiently put a matrix into row echelon form will come with time. This method can be extended to a system of n equations containing n variables, it will just take longer.

A little more on the row echelon form - consider the general row echelon matrix:

$$\left[\begin{array}{ccccc|c} 1 & 0 & 0 & \cdots & 0 & a_1 \\ 0 & 1 & 0 & \cdots & 0 & a_2 \\ 0 & 0 & 1 & \cdots & 0 & a_3 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & a_n \end{array}\right]$$

The rank of this system is the number of leading 1s down the main diagonal. If the rank is equal to the number of columns in the matrix, then there is one solution to the system of equations. However, it is not always possible to have 1s on the whole leading diagonal. When this is the case, where the rank is less than the number of columns in the matrix, then there are either zero solutions to this system, or an infinite amount.

1.3 Cramer's Method

For this method, you should be comfortable with working out the discriminant of matrices. That is:

$$A = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \implies |A| = ad - bc$$

and for a 3x3 matrix:

$$A = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} \implies |A| = a\begin{vmatrix} e & f \\ h & i \end{vmatrix} - b\begin{vmatrix} d & f \\ g & i \end{vmatrix} + c\begin{vmatrix} d & e \\ g & h \end{vmatrix}$$

Now let's consider our set of equations from earlier:

$$\begin{aligned} a_1x_1 + a_2x_2 + a_3x_3 &= b_1 \\ a_4x_1 + a_5x_2 + a_6x_3 &= b_2 \\ a_7x_1 + a_8x_2 + a_9x_3 &= b_1 \end{aligned}$$

Cramer's rule tells us that:

$$x_n = \frac{|A_n|}{|A|}$$

where A is the coefficient matrix:

$$A = \begin{bmatrix} a_1 & a_2 & a_3 \\ a_4 & a_5 & a_6 \\ a_7 & a_8 & a_9 \end{bmatrix}$$

and A_n is the coefficient matrix A but with the nth column replaced with the constants $b_1 \dots b_n$, such that:

$$A_1 = \begin{bmatrix} b_1 & a_2 & a_3 \\ b_2 & a_5 & a_6 \\ b_3 & a_8 & a_9 \end{bmatrix}, \; A_2 = \begin{bmatrix} a_1 & b_1 & a_3 \\ a_4 & b_2 & a_6 \\ a_7 & b_3 & a_9 \end{bmatrix} \text{ and } A_3 = \begin{bmatrix} a_1 & a_2 & b_1 \\ a_4 & a_5 & b_2 \\ a_7 & a_8 & b_3 \end{bmatrix}$$

Let's demonstrate.

$$\begin{aligned} x_1 + 3x_2 &= 5 \\ 2x_1 + 2x_2 &= 6 \end{aligned}$$

has a coefficient matrix:

$$\begin{aligned} A &= \begin{bmatrix} 1 & 3 \\ 2 & 2 \end{bmatrix} \\ |A| &= (1)(2) - (3)(2) = -4 \end{aligned}$$

So now to find x_1 we compute the determinant of A_1:

$$\begin{aligned} A_1 &= \begin{bmatrix} 5 & 3 \\ 6 & 2 \end{bmatrix} \\ |A_1| &= (5)(2) - (3)(6) = -8 \\ x_1 &= \frac{|A_1|}{|A|} \\ &= \frac{-8}{-4} = \boxed{2} \end{aligned}$$

A similar process with A_2 yields that $x_2 = 1$, and so we have found the solutions to our system of equations. We could easily apply this to a 3x3 matrix system as well.

Cramer's method easily highlights when a system of equations lacks solutions or has infinitely many. As the solution for x_n is $|A_n| \div |A|$, when $|A| = 0$ - the coefficient matrix is singular - it will be impossible to find a solution.

There are many other ways of solving systems of linear equations; these are just two methods where matrices are used to do so. Some methods will be more efficient than others in certain scenarios, so just use whichever is preferred by you.

2 A brief introduction to Markov chains

Written by **Hritesh, Year 12**

2.1 Bayes' Theorem

Bayes' theorem describes how to update probabilities based on new evidence:

$$P(A \mid B) = \frac{P(B \mid A)P(A)}{P(B)}$$

where:

- $P(A \mid B)$ is the probability of event A given that B has occurred (posterior probability).
- $P(B \mid A)$ is the probability of observing B if A is true (likelihood).
- $P(A)$ is the prior probability of A.
- $P(B)$ is the total probability of B, computed as:

$$P(B) = P(B \mid A)P(A) + P(B \mid A^c)P(A^c)$$

Example

Suppose a disease affects 1% of a population, and a test correctly identifies a sick person 95% of the time but falsely identifies a healthy person as sick 5% of the time. What is the probability that a person who tested positive actually has the disease?

Using Bayes' Theorem:

$$P(D \mid T) = \frac{P(T \mid D)P(D)}{P(T)}$$

where:

- $P(D) = 0.01$ (prior probability of disease),
- $P(T \mid D) = 0.95$ (test sensitivity),
- $P(T \mid \neg D) = 0.05$ (false positive rate),
- $P(T) = P(T \mid D)P(D) + P(T \mid \neg D)P(\neg D)$
$$= (0.95)(0.01) + (0.05)(0.99) = 0.059.$$

Thus:

$$P(D \mid T) = \frac{0.95 \times 0.01}{0.059} \approx 0.161.$$

So, despite testing positive, there is only a 16.1% chance the person actually has the disease.

2.2 Markov Chains

To understand Markov chains, let us begin with an example. Suppose we have a restaurant that serves a Burger, a Pizza, or a Sandwich. On any given day, the restaurant serves only one of these items, and it depends on what they served the previous day. In other words, you can predict what the restaurant will serve tomorrow, given that you know what they served today.

Figure 2 shows weighted arrows going from the current state (the meal today) to a future state (the meal tomorrow). These are known as transitions. The numbers are the probabilities

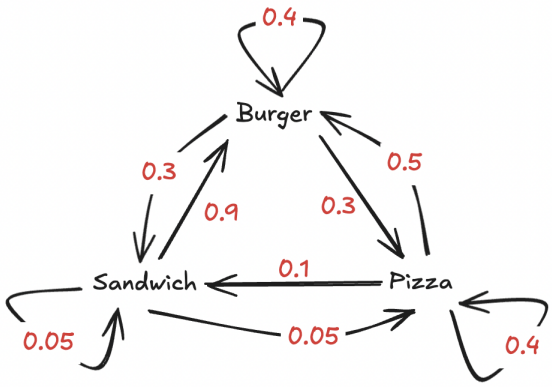

Figure 2: Transition diagram for the restaurant's menu choices.

of transitioning from one state to another. The diagram representing our example is a Markov Chain: a random process that moves from one state to another in a sequence of steps, where the probability of moving to the next state depends only on the current state and not on how you got there.

We can represent this mathematically as:

$$P(X_{n+1} = x_{n+1} \mid X_n = x_n, X_{n-1} = x_{n-1}, \ldots, X_0 = x_0) = P(X_{n+1} = x_{n+1} \mid X_n = x_n)$$

where:

- X_i is the random variable representing the state at step i
- x_i is a specific state that the process can occupy at step i

In simple terms:

$$P(Future\ event \mid Past + Present) = P(Future\ Event \mid Present),$$

Now, when dealing with multiple states, representing the information using weighted arrows becomes messy and complicated. We can therefore use a Transition Matrix to represent the probabilities of moving between states. For our restaurant example, a transition matrix would look like this:

$$P = \begin{pmatrix} 0.4 & 0.3 & 0.3 \\ 0.5 & 0.4 & 0.1 \\ 0.9 & 0.05 & 0.05 \end{pmatrix}$$

Each row of the matrix corresponds to the current state (in our example, it goes Burger, then Pizza, then Sandwich), and each column corresponds to the next state (again, Burger, then Pizza, then Sandwich). The entry p_{ij} represents the probability of transitioning from state i (row) to state j (column). For example, $p_{12} = 0.3$ indicates that if the restaurant served a Burger today, there is a 30% chance it will serve a Pizza tomorrow.

Initial and Final States in Markov Chains

An important aspect of understanding Markov chains is recognizing how the process is influenced by its starting point and how it behaves in the long run. In many practical applications, two concepts often arise: the **initial state** (or initial distribution) and the **final state** (or steady-state distribution).

Initial State

The *initial state* is essentially the starting configuration (or starting point) of the system. It tells us the probability of being in each state at time $t = 0$. For instance, if we are absolutely sure that the system begins in the Burger state, we might represent the initial state as:

$$\pi^{(0)} = \begin{pmatrix} 1 & 0 & 0 \end{pmatrix}$$

where:

- $\pi^{(0)}$ represents the inital state (not 3.14159...),

indicating a 100% chance of starting at Burger and 0% at Pizza or Sandwich. In cases where there is uncertainty in where we are ,when $t = 0$, the initial state is described by a probability vector that sums to 1 (e.g., $\pi^{(0)} = \begin{pmatrix} 0.5 & 0.3 & 0.2 \end{pmatrix}$). As the system evolves, the transition matrix is repeatedly applied to this initial vector to update the state probabilities.

Example: Evolving the Initial State Vector

Suppose the initial state is given by

$$\pi^{(0)} = \begin{pmatrix} 0.5 & 0.3 & 0.2 \end{pmatrix}$$

where:

- 0.5 indicates a 50% chance that the restaurant is serving a **Burger**,
- 0.3 indicates a 30% chance that the restaurant is serving a **Pizza**,
- 0.2 indicates a 20% chance that the restaurant is serving a **Sandwich**.

Now, consider the same transition matrix as before:

$$P = \begin{pmatrix} 0.4 & 0.3 & 0.3 \\ 0.5 & 0.4 & 0.1 \\ 0.9 & 0.05 & 0.05 \end{pmatrix}$$

The state distribution after one transition, denoted $\pi^{(1)}$, is obtained by multiplying the initial state vector by the transition matrix:

$$\pi^{(1)} = \pi^{(0)} P$$

We calculate each component of $\pi^{(1)}$ as follows:

$$\begin{pmatrix} 0.5 & 0.3 & 0.2 \end{pmatrix} \cdot \begin{pmatrix} 0.4 & 0.3 & 0.3 \\ 0.5 & 0.4 & 0.1 \\ 0.9 & 0.05 & 0.05 \end{pmatrix} = \begin{pmatrix} 0.5 \times 0.4 + 0.3 \times 0.5 + 0.2 \times 0.9 \\ 0.5 \times 0.3 + 0.3 \times 0.4 + 0.2 \times 0.05 \\ 0.5 \times 0.3 + 0.3 \times 0.1 + 0.2 \times 0.05 \end{pmatrix}$$

$$\begin{pmatrix} 0.2 + 0.15 + 0.18 \\ 0.15 + 0.12 + 0.01 \\ 0.15 + 0.03 + 0.01 \end{pmatrix} = \begin{pmatrix} 0.53, & 0.28, & 0.19 \end{pmatrix}$$

Thus, the updated state vector is

$$\pi^{(1)} = \begin{pmatrix} 0.53 & 0.28 & 0.19 \end{pmatrix}$$

Now, we compute $\pi^{(2)}$, which represents the state probabilities on the second day:

$$\pi^{(2)} = \pi^{(1)} \cdot P$$

Substituting $\pi^{(1)}$ from the previous calculation:

$$\begin{pmatrix} 0.53 & 0.28 & 0.19 \end{pmatrix} \cdot \begin{pmatrix} 0.4 & 0.3 & 0.3 \\ 0.5 & 0.4 & 0.1 \\ 0.9 & 0.05 & 0.05 \end{pmatrix} = \begin{pmatrix} 0.53 \times 0.4 + 0.28 \times 0.5 + 0.19 \times 0.9 \\ 0.53 \times 0.3 + 0.28 \times 0.4 + 0.19 \times 0.05 \\ 0.53 \times 0.3 + 0.28 \times 0.1 + 0.19 \times 0.05 \end{pmatrix}$$

$$\begin{pmatrix} 0.212 + 0.14 + 0.171 \\ 0.159 + 0.112 + 0.0095 \\ 0.159 + 0.028 + 0.0095 \end{pmatrix} = \begin{pmatrix} 0.523, & 0.2805, & 0.1965 \end{pmatrix}$$

From this pattern, we can generalize the probability vector at step n:

$$\pi^{(n)} = \pi^{(0)} P^n$$

where:

- $\pi^{(n)}$ represents the state probabilities after n days,
- P^n is the transition matrix raised to the power of n,
- $\pi^{(0)}$ is the initial state probability vector.

In the long run, as $n \to \infty$, $\pi^{(n)}$ will converge to a steady-state distribution π^*, which satisfies:

$$\pi^* = \pi^* P$$

This means that at equilibrium, applying the transition matrix does not change the probabilities anymore. The steady state can be found by solving:

$$\pi^*(I - P) = 0$$

subject to the constraint that the probabilities sum to 1:

$$\sum_i \pi_i^* = 1$$

2.3 Does Every Markov Chain Have a Steady State?

For a Markov chain to have a steady-state distribution, it must satisfy the following conditions:

1. **Irreducibility:** The chain must be able to reach any state from any other state, possibly in multiple steps. If there are isolated groups of states that the process can never leave, then there may not be a single steady-state distribution.
2. **Aperiodicity:** The system should not be stuck in cycles. If a state is only revisited at fixed intervals (like every 2 or 3 steps), then the probabilities won't settle into a single steady state.
3. **Finite or Well-Behaved Infinite State Space:** If the number of states is infinite, we need additional checks to ensure probabilities don't spread out too much over time.

If these conditions are met, the Markov chain will have a unique steady-state distribution.

How to Check for a Steady State

To determine whether a Markov chain has a steady state, follow these tests:

- **Check if the transition matrix is irreducible:** Can every state be reached from any other state in some number of steps? If yes, it's irreducible.
- **Check for periodicity:** If there's a fixed cycle in the state transitions, then the chain is periodic. To test this, pick a state and find the greatest common divisor (gcd) of all possible step lengths that return to that state. If gcd $= 1$, the chain is aperiodic.
- **Compute the steady-state probabilities:** Solve the equation $\pi^* P = \pi^*$ and check if a valid probability distribution (summing to 1) exists (which we are about to do with the example).

If all these tests pass, the Markov chain will converge to a steady state over time. Otherwise, it might oscillate between states or never settle into a stable pattern.

How to Calculate the Steady State

Let us return to our previous example to calculate the steady state.

We are given the transition matrix:

$$P = \begin{pmatrix} 0.4 & 0.3 & 0.3 \\ 0.5 & 0.4 & 0.1 \\ 0.9 & 0.05 & 0.05 \end{pmatrix}$$

To find the steady-state distribution π^*, we need to solve the equation:

$$\pi^* P = \pi^*$$

This results in the following system of linear equations:

$$\pi_1^* = 0.4\pi_1^* + 0.5\pi_2^* + 0.9\pi_3^*$$

$$\pi_2^* = 0.3\pi_1^* + 0.4\pi_2^* + 0.05\pi_3^*$$

$$\pi_3^* = 0.3\pi_1^* + 0.1\pi_2^* + 0.05\pi_3^*$$

In addition, the sum of the probabilities must be 1:

$$\pi_1^* + \pi_2^* + \pi_3^* = 1$$

We can solve this system of linear equations using either matrix methods or substitution. I will leave it up to the reader to try it out for themselves.

In the end, solving the system of equations, you should obtain the steady-state probabilities:

$$\pi_1^* \approx 0.5256, \quad \pi_2^* \approx 0.2791, \quad \pi_3^* \approx 0.1953.$$

Thus, the steady-state distribution is approximately:

$$\pi^* = \begin{pmatrix} 0.5256 & 0.2791 & 0.1953 \end{pmatrix}$$

This means that, in the long run, the system will spend about 52.56% of the time in state 1 (Burger), 27.91% of the time in state 2 (Pizza), and 19.53% of the time in state 3 (Sandwich).

2.4 Hidden Markov Chain Models

In many real-world cases, the important parts of a process (the states in a Markov Chain) are hidden from us. Take speech recognition as an example: when someone speaks, we don't directly hear the individual sounds (called phonemes) that make up the words. Instead, we only get the overall sound that comes out of the speaker's mouth—the acoustic signal. Hidden Markov Models help us work backwards from that sound to figure out which phonemes (and ultimately, which words) were intended. In these cases, we can use Hidden Markov Models (HMMs) which are statistical models in which the system being modeled is assumed to be a Markov process with unobservable (hidden) states. In an HMM, the actual state of the system is not directly visible; instead, an observer only has access to a set of observations that provide informations about the probabilities of these hidden states. An HMM is defined by:

- **States:** A finite set of hidden states $S = \{s_1, s_2, \ldots, s_N\}$.
- **Transition Probabilities:** A matrix $A = [a_{ij}]$ where $a_{ij} = P(s_j \mid s_i)$ represents the probability of transitioning from state s_i to state s_j.
- **Observation Probabilities:** A set of probabilities $B = \{b_j(o)\}$ where $b_j(o) = P(o \mid s_j)$ is the probability of observing o given that the system is in state s_j.
- **Initial Distribution:** A vector $\pi = (\pi_1, \pi_2, \ldots, \pi_N)$ representing the probability of the system starting in each state.

The key difference between a standard Markov chain and an HMM is that in an HMM the states are hidden; only outputs (observations) that are probabilistically related to the states are visible. Common tasks when working with HMMs include:

- *Decoding:* Determining the most likely sequence of hidden states given a sequence of observations.
- *Evaluation:* Computing the likelihood of an observed sequence

  These can be done with different algorithms such as the Forward algorithm and the Viterbi algorithm, which I am going to leave to the reader to research further.

Applications of Markov Chains

At first glance, the theory of Markov chains may seem unimpressive due to its simplicity and reliance on the current state to predict future states. However, this very simplicity makes Markov chains a powerful and useful model across various fields.

1. **Google PageRank Algorithm:** Markov chains are fundamental to Google's PageRank algorithm, which ranks web pages based on their link structures. The algorithm treats web pages as states and the links between them as transitions, utilizing the Markov property to determine the probability of a user visiting a particular page.
2. **Weather Prediction:** In meteorology, Markov chains model weather patterns by representing different weather states (e.g., sunny, rainy) and the probabilities of transitioning from one state to another. This approach helps in forecasting future weather conditions based on current observations.
3. **Financial Modeling:** Markov chains are used in finance to model stock prices and market behaviors. By analyzing the probabilities of different market states, investors can assess risks and make informed decisions.

4. **Machine Learning:** Markov chains are vital in machine learning for modeling sequential data and building probabilistic models. Hidden Markov Models (HMMs) use the Markov property to predict future states and are applied in speech recognition, natural language processing, and bioinformatics.

This is by far no means a comprehensive guide to Markov models and in the future, I may talk about Monte Carlo Simulations and expand upon the theory here to cover Markov Chain Monte Carlo simulations and Bayesian statistics

3 Problems

Solving systems of linear equations using matrices

Any system of linear equations that you can practice solving. Here's one:

$$\begin{aligned} 2\alpha + 7\beta - \gamma &= -\frac{31}{2} \\ 3\alpha + 2\beta - 3\gamma + 4\delta &= -71 \\ -\alpha + 6\beta + 2\delta &= -19 \\ 5\alpha - 4\beta + \gamma - 2\delta &= 42 \end{aligned}$$

A brief introduction to Markov Chains

1. Consider the following transition matrix for a simple weather model where the states are "Sunny" (S) and "Rainy" (R):

$$P = \begin{pmatrix} 0.8 & 0.2 \\ 0.4 & 0.6 \end{pmatrix}$$

   where the first row represents the transition probabilities from "Sunny" to "Sunny" (0.8) and from "Sunny" to "Rainy" (0.2) and the second row represents the transition probabilities from "Rainy" to "Sunny" (0.4) and from "Rainy" to "Rainy" (0.6).

   (a) If the weather is "Sunny" on day 0, calculate the weather probabilities for day 1.

   (b) What will the weather probabilities be on day 2, given that the weather on day 0 was "Sunny"?

   (c) Determine the steady-state distribution for this weather model.

2. You are given the following transition matrix for a system that describes a simple game between two players, where they can either win (W) or lose (L) each round:

$$P = \begin{pmatrix} 0.6 & 0.4 \\ 0.3 & 0.7 \end{pmatrix}.$$

   (a) If the game starts with player 1 winning, calculate the probabilities of each player winning in the following rounds.

   (b) What is the long-term probability of player 1 winning the game?

Solutions

Solutions to the problems from this issue will feature in the next issue!

Credits

Edited by **Vivaan**.

A special thanks to Mr Carew-Robinson and Mr Hudson for their help and support in running the publication.