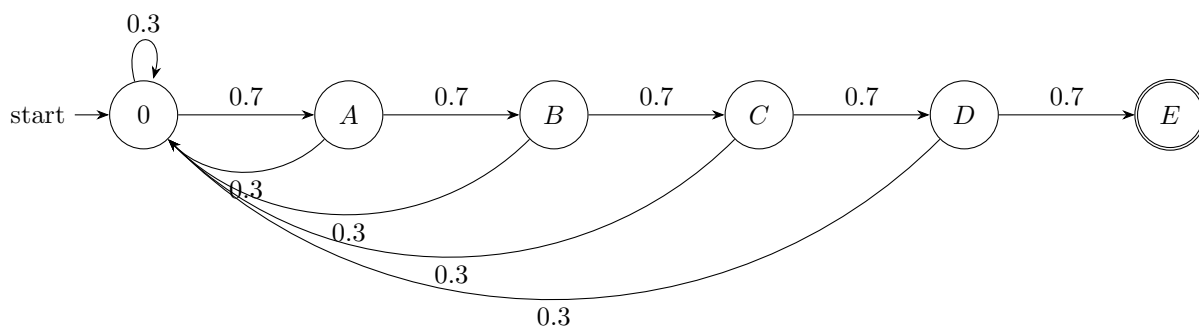


November Problem of the Month

Ben DeJong

Problem. Frank Ilett, a dedicated Manchester United (MU) fan, promised not to get a haircut until his team wins five consecutive games. MU has recently improved and now has a 70% (probability = 0.7) of winning any individual game. We want to answer the following question: on average how many games will Frank need to watch before MU wins five games in a row and he can get his haircut?

We will use a Markov chain to represent the states, moving towards E as we see more wins.



"Unrolling" this Markov chain, we get

$$\begin{aligned}P(0) &= 1 + 0.7P(A) + 0.3P(0) \\P(A) &= 1 + 0.7P(B) + 0.3P(0) \\P(B) &= 1 + 0.7P(C) + 0.3P(0) \\P(C) &= 1 + 0.7P(D) + 0.3P(0) \\P(D) &= 1 + 0.7P(E) + 0.3P(0) \\P(E) &= 0\end{aligned}$$

Solving, we get

$$\begin{aligned}P(E) &= 0 \\P(D) &\approx 5.949 \\P(C) &\approx 10.115 \\P(B) &\approx 13.030 \\P(A) &\approx 15.071 \\P(0) &\approx 16.499\end{aligned}$$

Therefore, MU would have to play roughly 17 games to win 5 in a row.

```
import random
def simulate_games_until_consecutive_wins(win_probability):
    total_games_played = 0
    consecutive_wins = 0
    required_consecutive_wins = 5

    while consecutive_wins < required_consecutive_wins:
        total_games_played += 1
        if random.random() <= win_probability:
            consecutive_wins += 1
        else:
            consecutive_wins = 0
    return total_games_played

win_probability = 0.7 # Probability of MU winning a single game
num_trials = 1000000 # Number of simulations to run

results = []

for _ in range(num_trials):
    games_played = simulate_games_until_consecutive_wins(win_probability)
    results.append(games_played)

average_games = sum(results) / len(results)
print(f"Average number of games to achieve 5 consecutive wins: {average_games:.2f}")
```