Lecture 6 Discrete probabilistic modeling Discrete-time Markov chains

Mathematical Modeling

Prof. Dr. Jingzhi Li

Department of Mathematics,
Southern University of Science and Technology

2025 Spring

Probabilistic modeling

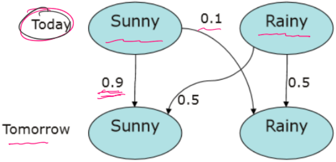
- So far in the course: deterministic models
  - precise sequences of actions, precise effects
  - only a matter of putting everything together and reasoning about the outcome
- What if at every stage of the modeling there are several options of how to continue?
  - evaluate which options are available
  - what is the probability of each option
  - choose one according to the probability distribution
  - the system ceases having only one future; several evolutions possible
  - the probability of each option might change as the system advances

Example

- A simple weather model
  - We only have two states for the model: *sunny* and *rainy*
  - The weather for tomorrow depends on today' s condition as follows:
  - if today is sunny, then there is a 90% chance of tomorrow being sunny, 10% being rainy
  - if today is rainy, then there is a 50% chance of tomorrow being sunny, 50% of being rainy
  - Summarize the probabilities in the following transition matrix:

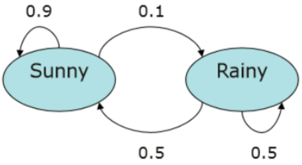
Tomorrow

		Sunny	Rainy
Today	Sunny	0.9	0.1
	Rainy	0.5	0.5



Example (continued)

- The model has only two states: *"sunny"* and *"rainy"*
- Transition matrix: gives the probabilities of changing the state
- The transition probabilities only depend on the current state, not on the past states
- If today is sunny, what is the prediction for:
  - tomorrow
  - 10 days from now
  - long-term



Example(continued

- Today is sunny: $x^{(0)} = \begin{bmatrix} 1 & 0 \end{bmatrix}$
- Weather prediction for tomorrow:

$$x^{(1)} = x^{(0)}P = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} 0.9 & 0.1 \\ 0.5 & 0.5 \end{bmatrix} = \begin{bmatrix} 0.9 & 0.1 \end{bmatrix}$$

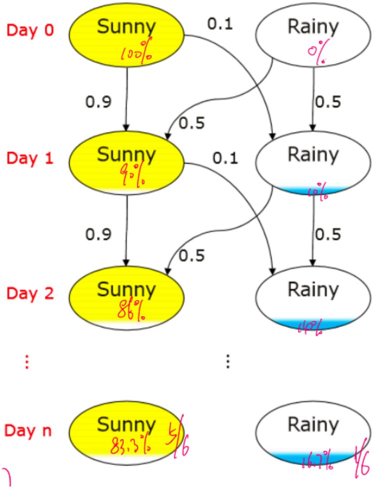
- Day after tomorrow:

$$x^{(2)} = x^{(1)}P = x^{(0)}P^2 = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} 0.9 & 0.1 \\ 0.5 & 0.5 \end{bmatrix}^2 = \begin{bmatrix} 0.86 & 0.14 \end{bmatrix}$$

- In general: $x^{(n)} = x^{(n-1)}P$
  - i.e.: $x^{(n)} = x^{(0)}P^n$
- In the long-run (steady state): $qP = q$
  - i.e.: $\begin{bmatrix} q_1 & q_2 \end{bmatrix} = \begin{bmatrix} 0.833 & 0.167 \end{bmatrix}$
- **Conclusion:** in the long term, 83.3% of the days are sunny

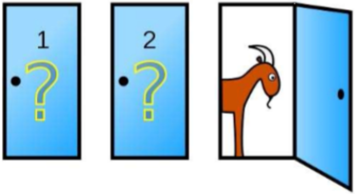
Example (continued)

π_∞



The probabilistic world

- Algorithms
  - deterministic
  - nondeterministic
  - stochastic (probabilistic)
- Probabilistic reasoning can be counter-intuitive
  - The Monty Hall problem
  - The birthday paradox
  - Primality testing: the Rabin-Miller test
    - the idea of a witness who could give a wrong ‘yes’, but its ‘no’ is always correct



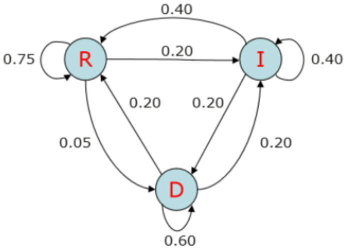
Picture: Wikipedia

Example: voting tendencies

- Assume 3 parties: Republicans, Democrats, Independents.
- Problem: identify the long-term behavior of voters in a presidential election
- Assumptions:
  - data collected over the last 10 years shows the following average trends in voting:

Next election

Last election	Republican	Democrat	Independent
Republican	0.75	0.05	0.20
Democrat	0.20	0.60	0.20
Independent	0.40	0.20	0.40



$P =$

dtmc(P)

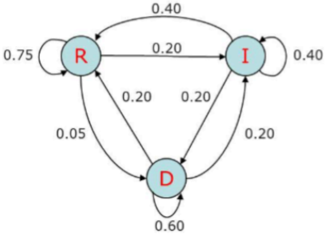
8

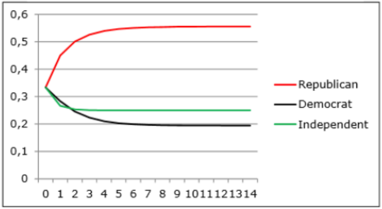
Example (continued)

- Model formulation
  - R_n = percentage of voters to vote Republican in election n
  - D_n = percentage of voters to vote Democratic in election n
  - I_n = percentage of voters to vote Independent in election n

$$
\begin{aligned}
R_{n+1} &= 0.75R_n + 0.20D_n + 0.40I_n \\
D_{n+1} &= 0.05R_n + 0.60D_n + 0.20I_n \\
I_{n+1} &= 0.20R_n + 0.20D_n + 0.40I_n
\end{aligned}
$$

- Numerical solution: start from an initial distribution of voters and calculate the model predictions for future elections and for long-term(asymptoic) behavior


0.40
0.75
R
0.20
I
0.40
0.20
0.20
0.05
0.20
D
0.60


0,6
0,5
0,4
0,3
0,2
0,1
0
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14
Republican
Democrat
Independent

Discrete-time Markov chains

- Discrete-time Markov chain
  - A discrete stochastic process
    - the state-space is discrete
  - satisfies the Markov property: the next state is only dependant on the present state, not on the past states
    - in other words: the process has no memory
- Formally: a sequence of random variables $X_1, X_2, \ldots$ taking values in a countable set S, such that
  - $\Pr(X_{n+1} = x \mid X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n) = \Pr(X_{n+1} = x \mid X_n = x_n)$
  - $\Pr(X_{n+1} = x \mid X_n = x_n)$ is called the (1-step) transition probability from state x_n to state x
  - S is called the state space of the chain
  - the Markov chain can be described by a labeled directed graph with the elements of S as the vertices and the transition probabilities as the labels of the edges

- Note:
  - discrete time
  - discrete state space
- We only consider in this course time-homogenous Markov chains (the transition probability matrix does not change in time)

Discrete-time Markov chains

- The transition probability matrix P
  - $P(r,s) = \Pr(X_{n+1} = s \mid X_n = r)$
  - Probability to go (in one step) from state r to state s
- P is a *stochastic matrix*, i.e.:
  - quadratic cardinality
  - $0 \leq P(r,s) \leq 1$, for all states r, s
  - $\sum_s P(r,s) = 1$, for any state r
- Easy to see that P^n is a *stochastic matrix*, for any $n \geq 1$

Discrete-time Markov chains

- A different perspective: Markov chains as transition systems
  - We are given a countable set of states S
  - $P : S \times S \to [0, 1]$ is a transition function such that $\sum_s P(r, s) = 1$, for any state
- The Markov chain is now seen as a computing device
- Starts in an initial state, advances according to its transition table
- It is a probabilistic machine
- Unlike typical machines where the computation is expected to end with a final output, a Markov chain is rather expected to run an infinite computation

Multi-step transitions

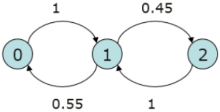
- Question: what is the probability to move from state q to state s in exactly $n \geq 0$ steps?
  - Denote it $p_n(q, s)$
- Answer:
  - Consider all possible intermediate steps and reason in terms of dynamic programming
  - The following equation holds for all $0 \leq m \leq n$

$$p_n(q, s) = \sum_{r \in S} p_m(q, r) p_{n-m}(r, s)$$

  - Denote by P_n the probability matrix for n-step transitions
  - Rewrite the equation above as $P_n = P_m P_{n-m}$
- By definition, $P_0 = I$ (identity matrix), $P_1 = P$
- For $m = 1$ and $n \geq 1$ we obtain that $P_n = P_1 P_{n-1} = P P_{n-1}$
- Iterate the scheme and conclude: $P_n = P^n$

Example: protein phosphorylation

- A protein with two phosphorylation sites
  - each site can be either phosphorylated or not
  - it can have 0, 1, or 2 phosphorylated sites
  - Follow the events in the system
- Phosphorylation rates as transition probabilities
- If you start with a 0-phosphorylated protein, what is your prediction for the next few steps
  - $x_0 = (1\ 0\ 0)$
  - $x_n = x_0 P^n$

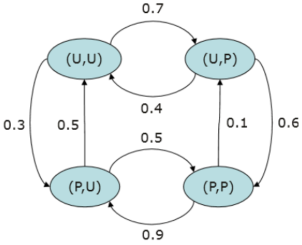


Current state \ Next state	0	1	2
0	0	1	0
1	0.55	0	0.45
2	0	1	0

The transition probability matrix P

Example (continued)

- A different approach: model explicitly the phosphorylation status of the two sites
  - the state of the system will be (i, j), with i, j being P or U
  - the first (second, resp.) component gives the phosphorylation status of the first (second, resp.) site
- Validation:
  - For any state s, the sum of the transition probabilities out of s has to be 1
  - Question: how about the sum of the transition probabilities into s?



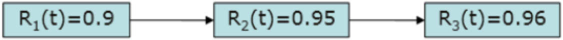
Next state

Current state	(U,U)	(U,P)	(P,U)	(P,P)
(U,U)	0	0.7	0,3	0
(U,P)	0.4	0	0	0.6
(P,U)	0.5	0	0	0.5
(P,P)	0	0.1	0.9	0

The transition probability matrix P

Example: modeling component and system reliability

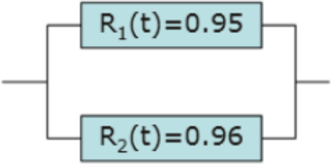
- We say that a device (say a computer or a network) is reliable if it performs well for a reasonably long time
  - reliability of a device: the probability that it will not fail over a specific time period
- Series systems: a sequence of devices that functions if and only if all of its devices are functioning



  - The system's reliability: $R(t) = R_1(t)R_2(t)R_3(t) = 0.8208$
  - less than each component' s reliability!

Example (continued)

- Parallel systems: it functions well as long as at least one of its components functions well
  - in other words: it fails if and only if ALL of its devices fail



  - calculating the reliability $R(t)$
  - easier to reason in terms of failing $1 - R(t)$

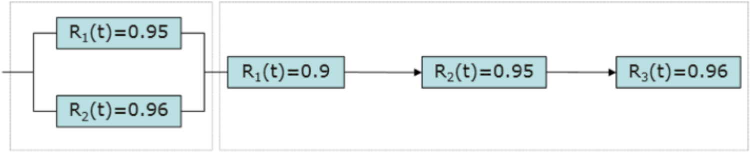
$$1 - R(t) = (1 - R_1(t))(1 - R_2(t))$$

$$R(t) = R_1(t) + R_2(t) - R_1(t)R_2(t) = 0.998$$

  - higher than any of the components' reliability

Example (continued)

- A mix of series and parallel combinations:



- Consider it as a series combination of the two systems on the previous slides
  - $R(t) = R'(t)R''(t) = 0.8192$

Conclusions: discrete-time Markov chains

- Discrete state-space
  - The system is in a certain state at each step, the state changes randomly in the next step
  - Countable state-space
- Transitions
  - In each state there is a possible transition to any other state (including to itself)
  - The probability of each transition depends only on the current state, not on the sequence of events that led to the current step
  - Only one transition takes place in the current state, moving the system to its new state
    - in the new state, there will be a new probability distribution for the state transitions

- Discrete time
  - Steps can be defined in terms of time points, but also in terms of distance, number of events, or some other discrete measurement
  - Time advances in discrete "ticks" –the state is only updated at discrete time points

Conclusions: discrete-time Markov chains

- Graph representation
  - A Markov chain can be represented as a complete graph
  - The state-space is the set of nodes
  - Edges are marked with the probability of the corresponding transition
- State machine representation
  - A Markov chain can be seen as a computing device
  - Starts in an initial state, advances according to its transition table
  - It is a probabilistic machine
  - Unlike typical machines where the computation is expected to end with a final output, a Markov chain is rather expected to run an infinite computation

- Model checking
  - Various (qualitative and quantitative) questions may be asked
  - Reachability: is a certain state reachable from the initial state?
  - Is the probability of eventually reaching state s from the initial state 1?
  - What is the probability of a given property (say, reliability) after 100 steps?
  - ...