Markov Chain

- *Stochastic process* (discrete time):

$$\{X_1, X_2, ..., \}$$

- *Markov chain*
  - Consider a discrete time stochastic process with discrete space. $X_n \in \{0, 1, 2, ...\}$.
  - Markovian property

$$\begin{aligned} & P\{X_{n+1} = j \mid X_n = i, X_{n-1} = i_{n-1}, ..., X_0 = i_0\} \\ = \; & P\{X_{n+1} = j \mid X_n = i\} = P_{i,j} \end{aligned}$$

  - $P_{i,j}$ is the *transition probability*: the probability of making a transition from i to j.
  - *Transition probability matrix*

$$\mathbf{P} = \begin{pmatrix} P_{0,0} & P_{0,1} & P_{0,2} & \cdots \\ P_{1,0} & P_{1,1} & P_{1,2} & \cdots \\ \vdots & \vdots & \vdots & \vdots \\ P_{i,0} & P_{i,1} & P_{i,2} & \cdots \\ \vdots & \vdots & \vdots & \vdots \end{pmatrix}$$

Example

- Suppose whether it will rain tomorrow depends on past weather condition ony through whether it rains today. Consider the stochastic process $\{X_n, n = 1, 2, ...\}$

$$X_n = \begin{cases} 0 & \text{rain on day } n \\ 1 & \text{not rain on day } n \end{cases}$$

$$P(X_{n+1}|X_n, X_{n-1}, ..., X_1) = P(X_{n+1} \mid X_n)$$

- State space $\{0, 1\}$.
- Transition matrix:

$$\begin{pmatrix} P_{0,0} & P_{0,1} \\ P_{1,0} & P_{1,1} \end{pmatrix}$$

- $P_{0,0} = P(\text{tomorrow rain}|\text{today rain}) = \alpha$. Then $P_{0,1} = 1 - \alpha$.
- $P_{1,0} = P(\text{tomorrow rain}|\text{today not rain}) = \beta$. Then $P_{1,1} = 1 - \beta$.
- Transition matrix:

$$\begin{pmatrix} \alpha & 1 - \alpha \\ \beta & 1 - \beta \end{pmatrix}$$

Random Walk

- A Markov chain with state space $i = 0, \pm 1, \pm 2, ...$
- Transition probability: $P_{i,i+1} = p = 1 - P_{i,i-1}$.
  - At every step, move either 1 step forward or 1 step backward.
- Example: a gambler either wins a dollar or loses a dollar at every game. X_n is the number of dollars he has when starting the nth game.

Chapman-Kolmogorov Equations

- Transition after nth steps:
  $P_{i,j}^{n} = P(X_{n+m} = j \mid X_m = i)$.
- **Chapman-Kolmogorov Equations**:

$$P_{i,j}^{n+m} = \sum_{k=0}^{\infty} P_{i,k}^{n} P_{k,j}^{m}, \quad n, m \geq 0 \text{ for all } i, j.$$

- Proof (by Total probability formula):

$$\begin{aligned} P_{i,j}^{n+m} &= P(X_{n+m} = j | X_0 = i) \\ &= \sum_{k=0}^{\infty} P(X_{n+m} = j, X_n = k | X_0 = i) \\ &= \sum_{k=0}^{\infty} P(X_n = k | X_0 = i) \cdot \\ &\quad P(X_{n+m} = j | X_n = k, X_0 = i) \\ &= \sum_{k=0}^{\infty} P_{i,k}^{n} P_{k,j}^{m} \end{aligned}$$

- n-step transition matrix:

$$\mathbf{P^{(n)}} = \begin{pmatrix} P^n_{0,0} & P^n_{0,1} & P^n_{0,2} & \cdots \\ P^n_{1,0} & P^n_{1,1} & P^n_{1,2} & \cdots \\ \vdots & \vdots & \vdots & \vdots \\ P^n_{i,0} & P^n_{i,1} & P^n_{i,2} & \cdots \\ \vdots & \vdots & \vdots & \vdots \end{pmatrix}$$

- Chapman-Kolmogorov Equations:

$$\mathbf{P^{(n+m)}} = \mathbf{P^{(n)}} \cdot \mathbf{P^{(m)}}, \qquad \mathbf{P^{(n)}} = \mathbf{P^n}.$$

- Weather example:

$$\mathbf{P} = \begin{pmatrix} \alpha & 1-\alpha \\ \beta & 1-\beta \end{pmatrix} = \begin{pmatrix} 0.7 & 0.3 \\ 0.4 & 0.6 \end{pmatrix}$$

Find $P(\text{rain on Tuesday} \mid \text{rain on Sunday})$ and $P(\text{rain on Tuesday and rain on Wednesday} \mid \text{rain on Sunday})$.

Solution:

$$P(\text{rain on Tuesday} \mid \text{rain on Sunday}) = P^2_{0,0}$$

$$\begin{aligned} \mathbf{P^{(2)}} &= \mathbf{P} \cdot \mathbf{P} = \begin{pmatrix} 0.7 & 0.3 \\ 0.4 & 0.6 \end{pmatrix} \times \begin{pmatrix} 0.7 & 0.3 \\ 0.4 & 0.6 \end{pmatrix} \\ &= \begin{pmatrix} 0.61 & 0.39 \\ 0.52 & 0.48 \end{pmatrix} \end{aligned}$$

$$P(\text{rain on Tuesday} \mid \text{rain on Sunday}) = 0.61$$

$$
\begin{aligned}
& P(\text{rain on Tuesday and rain on Wednesday} \mid \text{rain on Sunday}) \\
=\ & P(X_n = 0, X_{n+1} = 0 \mid X_{n-2} = 0) \\
=\ & P(X_n = 0 | X_{n-2} = 0) P(X_{n+1} = 0 | X_n = 0, X_{n-2} = 0) \\
=\ & P(X_n = 0 | X_{n-2} = 0) P(X_{n+1} = 0 | X_n = 0) \\
=\ & P^2_{0,0} P_{0,0} \\
=\ & 0.61 \times 0.7 = 0.427
\end{aligned}
$$

Classification of States

- *Accessible*: State j is accessible from state i if $P^n_{i,j} > 0$ for some $n \geq 0$.
  - $i \to j$.
  - Equivalent to: $P(\text{ever enter } j|\text{start in } i) > 0$.

$$
\begin{aligned}
& P(\text{ever enter } j|\text{start in } i) \\
= \; & P(\cup_{n=0}^{\infty}\{X_n = j\}|X_0 = i) \\
\leq \; & \sum_{n=0}^{\infty} P(X_n = j \mid X_0 = i) \\
= \; & \sum_{n=0}^{\infty} P^n_{i,j}
\end{aligned}
$$

  Hence if $P^n_{i,j} = 0$ for all n, $P(\text{ever enter } j|\text{start in } i) = 0$. On the other hand,

$$
\begin{aligned}
& P(\text{ever enter } j|\text{start in } i) \\
= \; & P(\cup_{n=0}^{\infty}\{X_n = j\}|X_0 = i) \\
\geq \; & P(\{X_n = j\}|X_0 = i) \quad \text{for any } n \\
= \; & P^n_{i,j} \, .
\end{aligned}
$$

  If $P^n_{i,j} > 0$ for some n, $P(\text{ever enter } j|\text{start in } i) \geq P^n_{i,j} > 0$.
  - Examples

- *Communicate*: State i and j communicate if they are accessible from each other.
  - $i \leftrightarrow j$.
  - Properties:
    1. $P^0_{i,i} = P(X_0 = i | X_0 = i) = 1$. Any state i communicates with itself.
    2. If $i \leftrightarrow j$, then $j \leftrightarrow i$.
    3. If $i \leftrightarrow j$ and $j \leftrightarrow k$, then $i \leftrightarrow k$.
       Proof:

$$
\begin{aligned}
i \leftrightarrow j &\implies P^n_{i,j} > 0 \text{ and } P^{n'}_{j,i} > 0 \\
j \leftrightarrow k &\implies P^m_{j,k} > 0 \text{ and } P^{m'}_{k,j} > 0 \\
P^{n+m}_{i,k} &= \sum_{l=0}^{\infty} P^n_{i,l} P^m_{l,k} \quad \text{Chapman-Kolmogorov Eq.} \\
&\geq P^n_{i,j} \cdot P^m_{j,k} \\
&> 0
\end{aligned}
$$

Similarly, we can show $P^{n'+m'}_{k,i} > 0$. Hence $i \leftrightarrow k$.

- *Class*: Two states that communciate are said to be in the same class. A class is a subset of states that communicate with each other.
  - Different classes do NOT overlap.
  - Classes form a partition of states.
- *Irreducible*: A Markov chain is irreducible if there is only one class.
  - Consider the Markov chain with transition probability matrix:

$$\mathbf{P} = \begin{pmatrix} \frac{1}{2} & \frac{1}{2} & 0 \\ \frac{1}{2} & \frac{1}{4} & \frac{1}{4} \\ 0 & \frac{1}{3} & \frac{2}{3} \end{pmatrix}$$

    The MC is irreducible.
  - MC with transition probability matrix:

$$\mathbf{P} = \begin{pmatrix} \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

    Three classes: $\{0, 1\}, \{2\}, \{3\}$.

Recurrent and Transient States

- f_i: probability that starting in state i, the MC will ever reenter state i.
- *Recurrent*: If $f_i = 1$, state i is recurrent.
  - A recurrent states will be visited infinitely many times by the process starting from i.
- *Transient*: If $f_i < 1$, state i is transient.
  - Starting from i, the MC will be in state i for exactly n times (including the starting state) is

    $$f_i^{n-1}(1 - f_i)\ ,\ n = 1, 2, ...$$

    This is a geometric distribution with parameter $1-f_i$. The expected number of times spent in state i is $1/(1 - f_i)$.

- A state is recurrent *if and only if* the expected number of time periods that the process is in state i, starting from state i, is infinite.

$$\text{Recurrent} \Longleftrightarrow E(\text{number of visits to } i | X_0 = i) = \infty$$
$$\text{Transient} \Longleftrightarrow E(\text{number of visits to } i | X_0 = i) < \infty$$

- Compute $E(\text{number of visits to } i | X_0 = i)$. Define

$$I_n = \begin{cases} 1 & \text{if } X_n = i \\ 0 & \text{if } X_n \neq i \end{cases}$$

Then the number of visits to i is $\sum_{n=0}^{\infty} I_n$.

$$\begin{aligned} & E(\text{number of visits to } i | X_0 = i) \\ = & \sum_{n=0}^{\infty} E(I_n | X_0 = i) \\ = & \sum_{n=0}^{\infty} P(I_n = 1 | X_0 = i) \\ = & \sum_{n=0}^{\infty} P(X_n = i | X_0 = i) \\ = & \sum_{n=0}^{\infty} P_{i,i}^{n} \end{aligned}$$

- **Proposition 4.1**: State i is recurrent if $\sum_{n=0}^{\infty} P_{i,i}^{n} = \infty$, and transient if $\sum_{n=0}^{\infty} P_{i,i}^{n} < \infty$.
- **Corollary 4.2**: If state i is recurrent and state i communicates with state j, then state j is recurrent.
- **Corollary 4.3**: A finite state Markov chain cannot have all transient states.
  - For any irreducible and finite-state Markov chain, all states are recurrent.

- Consider a MC with

$$\mathbf{P} = \begin{pmatrix} 0 & 0 & \frac{1}{2} & \frac{1}{2} \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

The MC is irreducible and finite state, hence all states are recurrent.

- Consider a MC with

$$\mathbf{P} = \begin{pmatrix} \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} & 0 \\ \frac{1}{4} & \frac{1}{4} & 0 & 0 & \frac{1}{2} \end{pmatrix}$$

Three classes: $\{0, 1\}$, $\{2, 3\}$, $\{4\}$. State $0, 1, 2, 3$ are recurrent and state 4 is transient.

Random Walk

- A Markov chain with state space $i = 0, \pm 1, \pm 2, \ldots.$
- Transition probability: $P_{i,i+1} = p = 1 - P_{i,i-1}$.
  - At every step, move either 1 step forward or 1 step backward.
- Example: a gambler either wins a dollar or loses a dollar at every game. X_n is the number of dollars he has when starting the nth game.
- For any $i < j$, $P_{i,j}^{j-i} = p^{j-i} > 0$, $P_{j,i}^{j-i} = (1-p)^{j-i} > 0$. The MC is irreducible.
- Hence, either all the states are transient or all the states are recurrent.

- Under which condition are the states transient or recurrent?
  - Consider State 0.

$$\sum_{n=1}^{\infty} P_{0,0}^{n} = \begin{cases} \infty & \text{recurrent} \\ \text{finite} & \text{transient} \end{cases}$$

  - Only for even m, $P_{0,0}^{m} > 0$.

$$P_{0,0}^{2n} = \binom{2n}{n} p^n(1-p)^n = \frac{(2n)!}{n!n!}(p(1-p))^n \qquad n = 1, 2, 3, \ldots$$

  - By Stirling's approximation

$$n! \sim n^{n+1/2}e^{-n}\sqrt{2\pi}$$

  - $P_{0,0}^{2n} \sim \frac{(4p(1-p))^n}{\sqrt{\pi n}}$.
  - When $p = 1/2$, $4p(1-p) = 1$.

$$\sum_{n=0}^{\infty} \frac{(4p(1-p))^n}{\sqrt{\pi n}} = \sum_{n=0}^{\infty} \frac{1}{\sqrt{\pi n}},$$

    The summation diverges. Hence, all the states are recurrent.
  - When $p \neq 1/2$, $4p(1-p) < 1$. $\sum_{n=0}^{\infty} \frac{(4p(1-p))^n}{\sqrt{\pi n}}$ converges. All the states are transient.

Limiting Probabilities

- Weather example

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

-

$$\mathbf{P}^{(4)} = \mathbf{P}^4 = \begin{pmatrix} 0.5749 & 0.4251 \\ 0.5668 & 0.4332 \end{pmatrix}$$

$$\mathbf{P}^{(8)} = \mathbf{P}^{(4)}\mathbf{P}^{(4)} = \begin{pmatrix} 0.572 & 0.428 \\ 0.570 & 0.430 \end{pmatrix}$$

- $\mathbf{P}^{(4)}$ and $\mathbf{P}^{(8)}$ are close. The rows in $\mathbf{P}^{(8)}$ are close.
- Limiting probabilities?

- **Theorem 4.1**: For an irreducible ergodic Markov chain, $\lim_{n\to\infty} P_{i,j}^n$ exists and is independent of i. Let $\pi_j = \lim_{n\to\infty} P_{i,j}^n$, $j \geq 0$, then π_j is the unique nonnegative solution of

$$\boxed{\pi_j = \sum_{i=0}^{\infty} \pi_i P_{i,j} \quad j \geq 0}$$

$$\sum_{j=0}^{\infty} \pi_j = 1 \,.$$

- The Weather Example:

$$\mathbf{P} = \begin{pmatrix} \alpha & 1-\alpha \\ \beta & 1-\beta \end{pmatrix}$$

The MC is irreducible and ergodic.

$$\begin{aligned} \pi_0 + \pi_1 &= 1 \\ \pi_0 &= \pi_0 P_{0,0} + \pi_1 P_{1,0} = \pi_0 \alpha + \pi_1 \beta \\ \pi_1 &= \pi_0 P_{0,1} + \pi_1 P_{1,1} = \pi_0(1-\alpha) + \pi_1(1-\beta) \end{aligned}$$

Solve the linear equations, $\pi_0 = \frac{\beta}{1+\beta-\alpha}$, $\pi_1 = \frac{1-\alpha}{1+\beta-\alpha}$.

- *Period d*: For state i, if $P_{i,i}^n = 0$ whenever n is not divisible by d and d is the largest integer with this property, d is the period of state i.
  - Period d is the greatest common divisor of all the m such that $P_{i,i}^m > 0$.
- *Aperiodic*: State i is aperiodic if its period is 1.
- *Positive recurrent*: If a state i is recurrent and the expected time until the process returns to state i is finite.
  - If $i \leftrightarrow j$ and i is positive recurrent, then j is positive recurrent.
  - For a finite-state MC, a recurrent state is also positive recurrent.
  - A finite-state irreducible MC contains all positive recurrent states.
- *Ergodic*: A positive recurrent and aperiodic state is an ergodic state.
- A Markov chain is ergodic if all its states are ergodic.

application n. 1

Already discussed. Now you can see it as a Markov Chain.

Markov Chain Interpretation of Google Page Rank

Suppose there are N Web pages in total. Let the page rank of page i, $i = 1, ..., N$ be $PR(i)$. The page ranks are determined by the following linear equations:

$$PR(i) = (1 - d) + \sum_{j:\ i \text{ linked to } j} PR(j) \frac{d}{C(j)}\ ,\ i = 1, ..., N$$

where $C(j)$ is the total number of links contained in page j.

Model the network of Web pages by a Markov chain. Regard the network as a huge finite state machine, where every state is a page. The conclusion we will arrive at is that the page ranks are proportional to the stationary probabilities of the states in the Markov chain. That is, if you wander around the Web pages randomly according to this Markov chain, after a long time, the probability of visiting any page at any time converges, and this probability is not affected by how you start your navigation. The higher the probability is, the higher the rank of the page will be. The scaling factor between the page rank and the probability is N/d, where $0 < d < 1$ is a chosen constant related to how likely you will restart your navigation by not following links in pages.

Set up the Markov chain as follows.

1. Every page is a state, $i = 1, ..., N$.
2. Add an imaginary state, referred to as the *Restart* page, and label it as state 0.
3. The transition probabilities between the states are defined as follows. Note that the transition probability $p_{j,i}$ is the probability of entering state i given the current state is j. Valid transition probabilities have to satisfy $\sum_i p_{j,i} = 1$ for any j.
   (a) Every state i, $0 \leq i \leq N$ has probability of $1-d$ transiting to the restart state 0, that is, $p_{i,0} = 1 - d$ for all the states i. Heuristically, this means that no matter which page you are currently in, you always have a fixed probability of restarting instead of hopping around via links.
   (b) The probability of going from state 0 to state i, $i \neq 0$, is $p_{0,i} = d/N$. That is, from the restart state, the probability of going to any real page is equal, and the total probability of going to a real page is d (the rest probability, $1 - d$, is assigned to restart again).
   (c) The probability of going from state j, $j \neq 0$, to state i, $i \neq 0$, is

$$p_{j,i} = \frac{d}{C(j)} \cdot I(j \text{ links to } i) \ ,$$

   where $I(\cdot)$ is the indicator function that equals 1 if the argument is true, 0 otherwise. This means that for every page, besides the probability of $1-d$ going to restart, the rest probability d is evenly divided among the $C(j)$ links contained in it. If a page i is not linked to j, $p_{j,i} = 0$.

Let the stationary probabilities (i.e., limiting probabilities) of state i be π_i. By a theorem on Markov chain (assuming the MC is irreducible and ergodic, easily satisfied by a connected finite graph without cycling patterns), these probabilities satisfy the following set of linear equations:

$$\begin{aligned} \pi_i &= \sum_{j=0}^{N} \pi_j p_{j,i} \quad i = 0, 1, ..., N \\ \sum_{i=0}^{N} \pi_i &= 1 \end{aligned}$$

Specific to the Markov chain set up above:

$$\begin{aligned} \pi_0 &= \sum_{j=0}^{N} \pi_j p_{j,0} = \sum_{j=0}^{N} \pi_j (1-d) = (1-d) \sum_{j=0}^{N} \pi_j = 1 - d \\ \pi_i &= \pi_0 p_{0,i} + \sum_{j=1}^{N} \pi_j p_{j,i} \\ &= (1-d) \cdot \frac{d}{N} + \sum_{j:\ i \text{ linked to } j} \pi_j \frac{d}{C(j)} \end{aligned}$$

Multiple the last equation by $\frac{N}{d}$:

$$\frac{N}{d} \cdot \pi_i = (1-d) + \sum_{j:\ i \text{ linked to } j} \left(\frac{N}{d} \cdot \pi_j \right) \frac{d}{C(j)}$$

Define the page rank as $PR(i) = \frac{N}{d}\pi_i$, we get:

$$PR(i) = (1-d) + \sum_{j:\ i \text{ linked to } j} PR(j) \frac{d}{C(j)} \ , \quad i = 1, 2, ..., N$$

which is precisely the page rank equation of the early Google.

Simulated Annealing

This is a new alagorithm!

- General method to solve combinatorial optimization problems

Principle:

- Start with initial configuration
- Repeatedly search by a Markov Chain Step in the neighborhood and select a neighbor as candidate
- Evaluate some cost function (or fitness function) and accept candidate if "better"; if not, select another neighbor
- Stop if quality is sufficiently high, if no improvement can be found or after some fixed time

Needed are:

- A method to generate initial configuration
- A transition or generation function to find a neighbor as next candidate
- A cost function
- An Evaluation Criterion
- A Stop Criterion

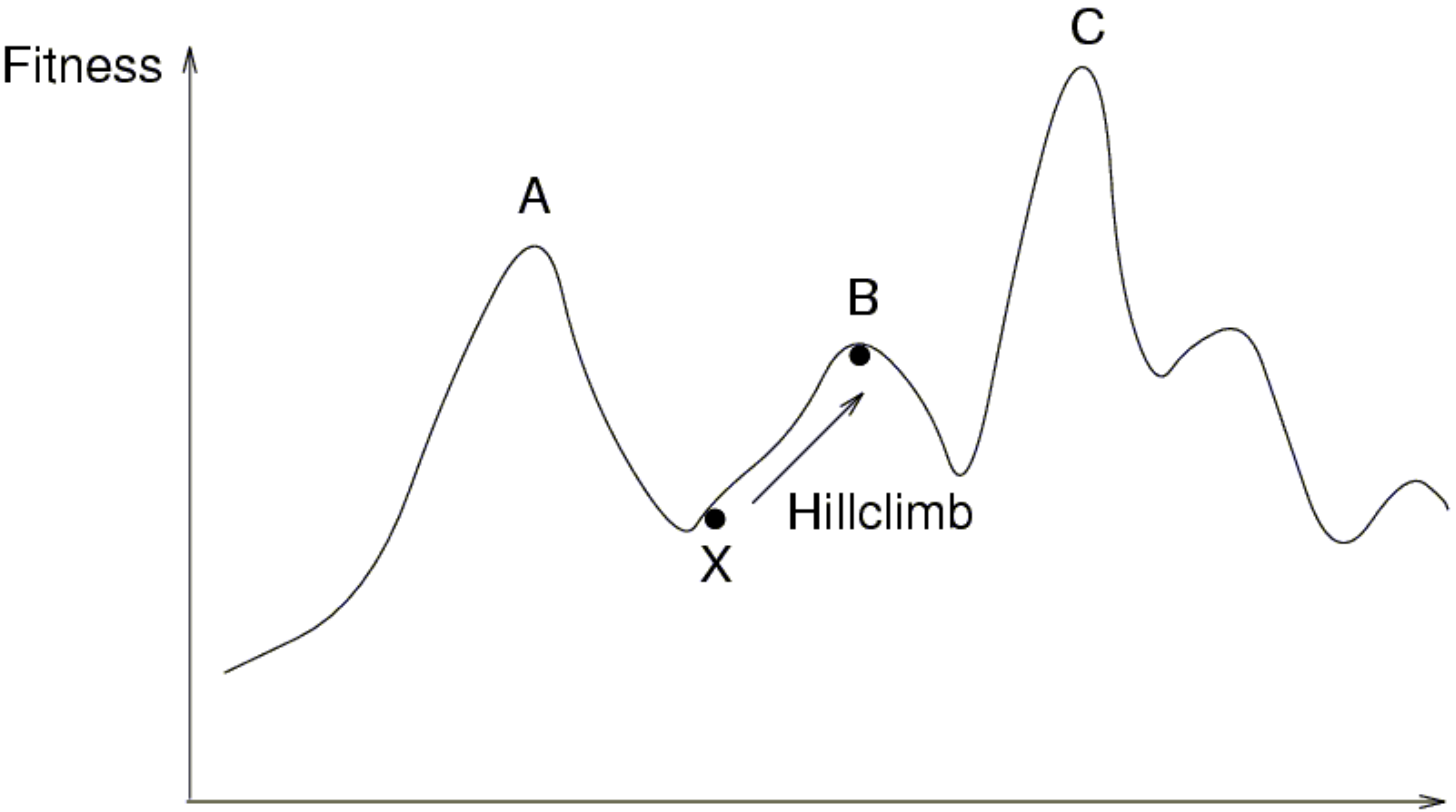
Simple Iterative Improvement or Hill Climbing:

- Candidate is always and only accepted if cost is lower (or fitness is higher) than current configuration
- Stop when no neighbor with lower cost (higher fitness) can be found

Disadvantages:

- Local optimum as best result
- Local optimum depends on initial configuration
- Generally no upper bound on iteration length

Hill climbing



- Repeat algorithm many times with different initial configurations
- Use information gathered in previous runs
- Use a more complex Generation Function to jump out of local optimum
- Use a more complex Evaluation Criterion that accepts sometimes (randomly) also solutions away from the (local) optimum

Use a more complex Evaluation Function:

- Do sometimes accept candidates with higher cost to escape from local optimum
- Adapt the parameters of this Evaluation Function during execution
- Based upon the analogy with the simulation of the annealing of solids

Other Names

- Monte Carlo Annealing
- Statistical Cooling
- Probabilistic Hill Climbing
- Stochastic Relaxation
- Probabilistic Exchange Algorithm

Analogy

- Slowly cool down a heated solid, so that all particles arrange in the ground energy state
- At each temperature wait until the solid reaches its thermal equilibrium
- Probability of being in a state with energy E :

$$\mathrm{Prob}(E = \mathcal{E}) = \frac{e^{-\frac{1}{T}\mathcal{E}}}{Z(T)}$$

E	Energy/Cost
T	Temperature
Z(T)	Normalization factor (temperature dependant)

Simulation of cooling (Metropolis 1953)

- At a fixed temperature T :
- Perturb (randomly) the current state to a new state
- ΔE is the difference in energy between current and new state
- If $\Delta E < 0$ (new state is lower), accept new state as current state
- If $\Delta E \geq 0$, accept new state with probability

  $$Pr\ (\text{accepted}) = \exp\ (- \Delta E / T)$$

- Eventually the systems evolves into thermal equilibrium at temperature T ; then the formula mentioned before holds
- When equilibrium is reached, temperature T can be lowered and the process can be repeated

Simulated Annealing

- Same algorithm can be used for combinatorial optimization problems:
- Energy E corresponds to the Cost function C
- Temperature T corresponds to control parameter c

$$\mathrm{Prob}(\mathrm{Configuration} = i) = \frac{e^{-\frac{1}{c}\mathcal{C}(i)}}{Q(c)}$$

C	Cost
c	Control parameter (same as T)
Q(c)	Normalization factor (same as Z)

Algorithm

```
initialize;
REPEAT
  REPEAT
    perturb ( config.i → config.j, ΔCij);
    IF ΔCij < 0 THEN accept
    ELSE IF exp(-ΔCij/c) > random[0,1) THEN accept;
    IF accept THEN update(config.j);
  UNTIL equilibrium is approached sufficient closely;
  c := next_lower(c);
UNTIL system is frozen or stop criterion is reached
```

Inhomogeneous Algorithm

- Previous algorithm is the homogeneous variant:

  c is kept constant in the inner loop and is only decreased in the outer loop

- Alternative is the inhomogeneous variant:

  There is only one loop; c is decreased each time in the loop, but only very slightly

Parameters

- Choose the start value of c so that in the beginning nearly all perturbations are accepted (exploration), but not too big to avoid long run times
- The function next_lower in the homogeneous variant is generally a simple function to decrease c, e.g. a fixed part (80%) of current c
- At the end c is so small that only a very small number of the perturbations is accepted (exploitation)

- If possible, always try to remember explicitly the best solution found so far; the algorithm itself can leave its best solution and not find it again

This algorithm is a Markov Chain

Markov Chain:

Sequence of trials where the outcome of each trial depends only on the outcome of the previous one

- Markov Chain is a set of conditional probabilities:

$$P_{ij}(k-1,k)$$

Probability that the outcome of the k-th trial is j, when trial k-1 is i

- Markov Chain is homogeneous when the probabilities do not depend on k

- When c is kept constant (homogeneous variant), the probabilities do not depend on k and for each c there is one homogeneous Markov Chain

- When c is not constant (inhomogeneous variant), the probabilities do depend on k and there is one inhomogeneous Markov Chain

Performance

- SA is a general solution method that is easily applicable to a large number of problems
- "Tuning" of the parameters (initial c, decrement of c, stop criterion) is relatively easy
- Generally the quality of the results of SA is good, although it can take a lot of time
- Results are generally not reproducible: another run can give a different result
- SA can leave an optimal solution and not find it again (so try to remember the best solution found so far)
- Proven to find the optimum under certain conditions; one of these conditions is that you must run forever

Video on TSP