Markov Chain Monte Carlo (MCMC)

Sampling methods

Sampling

- Selection of a subset of individuals from within a statistical population to estimate characteristics of the whole population
- A **probability sample** is a sample in which every unit in the population has a chance (greater than zero) of being selected in the sample, and this probability can be accurately determined
- **Nonprobability sampling** is any sampling method where some elements of the population have *no* chance of selection → does not allow to estimate sampling errors!
- Replacement of samples
  - Sampling **with** replacement
  - Sampling **without** replacement

    - (Wikipedia)

Sampling methods

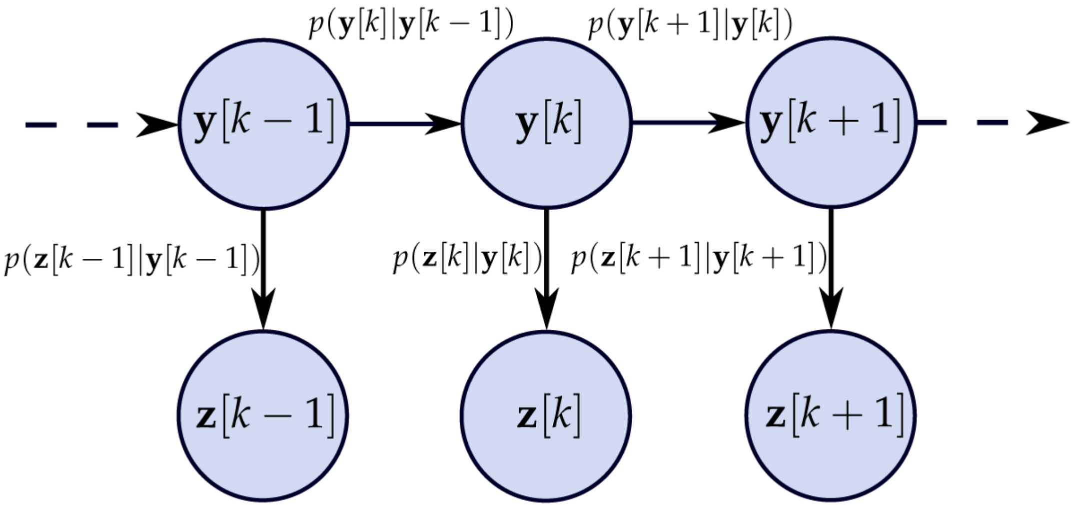
- In a **simple random sample** (SRS) of a given size, all such subsets of the frame are given an equal probability
  - Minimizes biases and simplifies analysis
    - Within sample variance is approximation of population variance
- **Systematic sampling** (also known as interval sampling) relies on arranging the study population according to some ordering scheme and then selecting elements at regular intervals through that ordered list
  - Often more accurate than SRS but difficult to *quantify* accuracy
- Where the population embraces a number of distinct categories, the frame can be organized by these categories into separate "strata“ (**stratified sampling**). Each stratum is then sampled as an independent sub-population, out of which individual elements can be randomly selected
- **Probability-proportional-to-size sampling**
- Sometimes it is more cost-effective to select respondents in groups (**cluster sampling**)
- In **quota sampling**, the population is first segmented into mutually exclusive sub-groups, then subjects are selected from each group based on a specified proportion
- …

Errors

- Sampling errors
  - **Selection bias**: When the true selection probabilities differ from those assumed in calculating the results.
  - **Random sampling error**: Random variation in the results due to the elements in the sample being selected at random
- Non-sampling error
  - Non-sampling errors are other errors which can impact the final survey estimates, caused by problems in data collection, processing, or sample design
    - Over-coverage
    - Under-coverage
    - Measurement error
    - Processing error
    - Non-response

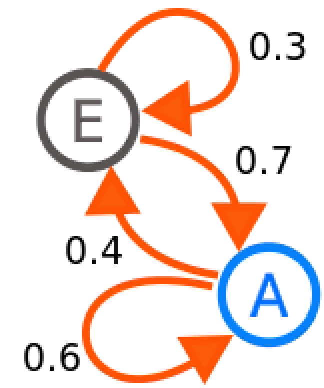
Markov chain Monte Carlo?

- a class of algorithms for sampling from a probability distribution based on constructing a Markov chain that has the desired distribution as its equilibrium distribution
- Markov chain = a random process that undergoes transitions from one state to another on a state space
  - NO memory!!!
  - Example: hidden MC



Markov chain

- **Markov chain** = sequence of random variables a process moves through, with the Markov property defining serial dependence only between adjacent periods (as in a "chain")



$$\begin{pmatrix} E \\ A \end{pmatrix}_{k+1} = \underbrace{\begin{pmatrix} 0.3 & 0.4 \\ 0.7 & 0.6 \end{pmatrix}}_{P} \begin{pmatrix} E \\ A \end{pmatrix}_{k} + noise$$

  - Systems that follow a chain of linked events, where what happens next depends only on the current state of the system
  - Example: random walk
- Without external inputs, this typically converges! → stationary

$$P^{k} \begin{pmatrix} E \\ A \end{pmatrix}_{k} \approx P^{k+1} \begin{pmatrix} E \\ A \end{pmatrix}_{k+1}$$

Monte Carlo?

- Monte Carlo methods (or Monte Carlo experiments) are a broad class of computational algorithms that rely on repeated random sampling to obtain numerical results
  - E.g. through generating draws from a sequence of probability distributions
- Monte Carlo methods tend to follow a particular pattern:
  - Define a domain of possible inputs.
  - Generate inputs randomly from a probability distribution over the domain.
  - Perform a deterministic computation on the inputs.
  - Aggregate the results.

Monte Carlo casino: Monaco

Markov chain Monte Carlo!

- Markov chain Monte Carlo (MCMC) methods have revolutionized the practicability of Bayesian inference methods, allowing a wide range of posterior distributions to be simulated and their parameters found numerically
  - MCMC methods are primarily used for calculating numerical approximations of multi-dimensional integrals, for example in Bayesian statistics, computational physics, and computational biology
  - Random samples used in a conventional Monte Carlo are statistically independent, those used in MCMC are correlated
  - For approximating a multi-dimensional integral, an ensemble of "walkers" move around randomly. At each point where a walker steps, the integrand value is counted towards the integral. The walker then may make a number of tentative steps around the area, looking for a place with a reasonably high contribution to the integral to move into next
  - A Markov chain is constructed in such a way as to have the integrand as its equilibrium distribution

Problem formulation

- Given observed data (D) and model parameters / missing data θ, we can use Bayes theorem to infer the posterior distribution of θ conditional on D:

$$P(\theta|D) = \frac{P(\theta)P(D|\theta)}{\int P(\theta)P(D|\theta)d\theta}$$

- Any feature of the posterior can be expressed as posterior expectations of functions of θ, i.e. f(θ) as:

$$E[f(\theta)|D] = \frac{\int f(\theta)P(\theta)P(D|\theta)d\theta}{\int P(\theta)P(D|\theta)d\theta} \Longleftrightarrow E[f(X)] = \frac{\int f(x)\pi(x)dx}{\int \pi(x)dx}$$

posterior

- These integrals are hard! MCMC provides a way to approximate them!

Using Monte Carlo

- For time-invariant Markov chains, the state will eventually converge to a stationary distribution $\phi(.)$
  - As k increases samples $\{X_k\}$ will increasingly look like dependent samples from $\phi(.)$
  - Initial convergence = *burn-in* (m iterations)
- We can now use the output from the Markov chain to estimate $E[f(X)]$, where X has distribution $\phi(.)$:

$$\bar{f} = \frac{1}{n-m} \sum_{t=m+1}^{n} f(X_t)$$

  - This is called an *ergodic average*

MCMC

- Problem: we have assumed the following
  - $\phi(.)$ approximates $\pi(.)$
  - But for that to be true, we need to construct a Markov chain in a way that it's stationary description is exactly the distribution of interest.
- This is surprisingly easy!!!
- **Hastings-Metropolis algorithm** (Metropolis et al., 1953; Hastings 1970)

Hastings-Metropolis algorithm

- Iterative procedure
  - At time t, sample a point Y from proposal distribution $q(.|X_t)$
  - Accept candidate point with probability α:

$$\alpha(X, Y) = \min\left(1, \frac{\pi(Y)q(X|Y)}{\pi(X)q(Y|X)}\right)$$

Initialize X_0; **set** $t = 0$.
Repeat {
  Sample a point Y **from** $q(.|X_t)$
  Sample a Uniform(0,1) random variable U
  If $U \leq \alpha(X_t, Y)$ **set** $X_{t+1} = Y$
    otherwise set $X_{t+1} = X_t$
  Increment t
}.

Hastings-Metropolis algorithm

- Remarkably, the proposal distribution q(.|.) can have any form!
  - The stationary distribution of the chain will be π(.), i.e. one can prove that $P^{(t)}(X_t|X_0)$ will converge to π(.)
- How do we chose the proposal distribution q(.|.)?
  - Convergence will depend on relationship between q and π
  - Exploratory analysis to determine rough form of π necessary
  - Practically, q should be chosen so that it can be easily sampled and evaluated
  - There are strategies for choosing q… (see literature on Blohm lab wiki!)

Applications of MCMC

- Estimate parameters of known (or assumed) distributions
  - E.g. Normal distribution $\pi=N(\mu,\sigma|data)$
- Sample randomly from a given distribution
  - Hastings-Metropolis random method will result in random samples of a given distribution
- Nested sampling for model comparison (Skilling, 2006)
- Identify most plausible cause of something among many possible causes (e.g. cryptography, decoding)
- Time series predictions: estimation of Markov transition probability matrix
- …