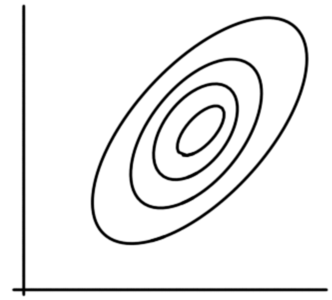
Multivariate MCMC

- Multivariate version of Metropolis-Hastings algorithm.

Example: Target distribution exists in 2 dimensions.



Target PDF: $P(\vec{x})$

We also have some function $f(\vec{x}) \propto P(\vec{x})$
proportional to PDF

$\vec{x}$ is a vector in 2D $\mathbb{R}^2$

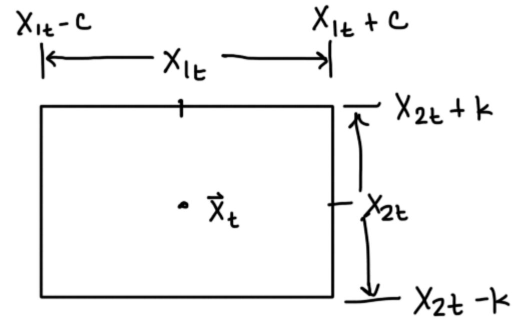
MH algorithm:

- Start location arbitrary $\vec{x}_t$
- Propose a new location $\vec{x}_{proposed}$, where $\vec{x}_{proposed} \sim g(\vec{x} \mid \vec{x}_t)$ — g is proposal distribution in 2D.
- we move to propsed location with $p_{move} = \min\left(1, \dfrac{f(\vec{x}_{proposed})}{f(\vec{x}_t)} \cdot \dfrac{g(\vec{x}_t \mid \vec{x}_{proposed})}{g(\vec{x}_{proposed} \mid \vec{x}_t)}\right)$
- If $u < p_{move}$: accept. $\vec{x}_{t+1} = \vec{x}_{proposed}$.
  else: reject $\vec{x}_{t+1} = \vec{x}_t$

possible proposal distribution: 2D Uniform distribution.

$\vec{X}_t = [X_{1t}, X_{2t}]$

$\vec{X}_{proposed} = [X_{1p}, X_{2p}]$



$X_{1p} \sim \text{Unif}(X_{1t} - c, X_{1t} + c)$

$X_{2p} \sim \text{Unif}(X_{2t} - k, X_{2t} + k)$

c may or may not be equal to k.

any location inside the rectangle is equally likely to be proposed.

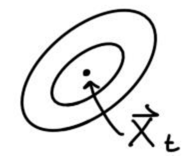
$$\text{proposal pdf} = \frac{1}{2c} \cdot \frac{1}{2k} = \frac{1}{4ck}$$

X_{1p} is proposed independently of X_{2p}

Another possible proposal:

Multivariate Gaussian / MV Normal.

$$\vec{X}_{proposed} \sim N_2\left(\vec{\mu} = \vec{X}_t,\ \Sigma\right)$$



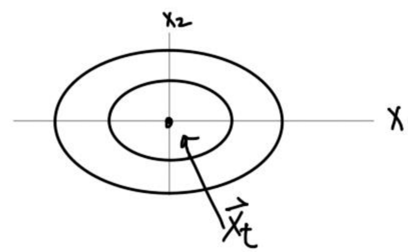
proposal distribution is centered at $\vec{X}_t$
with an arbitrary var-cov matrix Σ

R has library mvtnorm, and function rmvnorm( )

You can also sample X_1 independently of X_2.

$$X_{1p} \sim \text{Norm}\left(\mu = X_{1t},\ \sigma_1\right)$$

$$X_{2p} \sim \text{Norm}\left(\mu = X_{2t},\ \sigma_2\right)$$



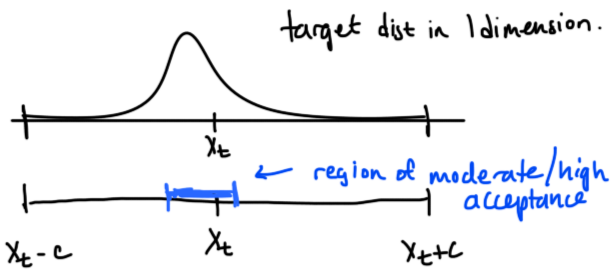
$$\frac{g(\vec{X}_t \mid \vec{X}_{proposed})}{g(\vec{X}_{proposed} \mid \vec{X}_t)} = 1$$

$$\vec{X}_{proposed} = (X_{1p},\ X_{2p})$$

this is Symmetric so, MH algorithm $\rightarrow$ Metropolis algorithm.

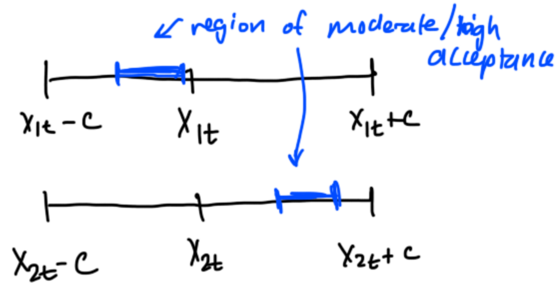
We do run into problems trying to use the metropolis algorithm in very high dimensions.

As the number of dimensions increase, the prob of a successful proposal shrinks.



propose uniform $(x_t - c, x_t + c)$

In 1 dimension, we might accept ~10% of proposed values.



~10% of proposed x_1 values

~10% of proposed x_2 values

fall in region of high acceptance

→ ~1% of values are in the region of high acceptance.

with 3D

~10% in each dim

only ~0.1% will be in the region of high acceptance.

4D → 0.01%

shrinks exponentially.

Gibbs sampler — MCMC method for multivariate distributions.

Contrast to Metropolis-Hastings:

In MH, we propose all coordinates of the next location at the same time. let's say our target dist exists in 17 dimensions. Our proposal location requires 17 coordinates. Even is 16 of the coordinates proposed are in good locations, if one coordinate ends up in a "bad" low prob location, $P(\vec{X}_{\text{proposed}})$ will be small and P_{move} will be small, leading to its rejection. MH algorithm moves slowly because it is hard to get a good proposal.

Gibbs sampler: We factor the multivariate target distribution into the product of several univariate (possibly bivariate) conditional distributions. (could be analytically challenging).

example: $P(\vec{x})$ exists in d dimensions.

$$\vec{x} = (x_1, x_2, x_3 \dots x_d)$$

Next location is selected in d separate steps:

$$x_{1(t+1)} \sim P(x_1 \mid x_{2t}, x_{3t}, \dots x_{dt})$$

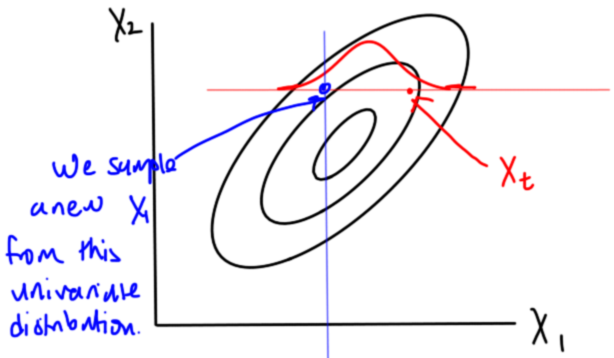
we use the newly sampled value in the next step:

$$x_{2(t+1)} \sim P(x_2 \mid x_{1(t+1)}, x_{3t}, \dots x_{dt})$$

$$x_{3(t+1)} \sim P(x_3 \mid x_{1(t+1)}, x_{2(t+1)}, x_{4t}, \dots x_{dt})$$

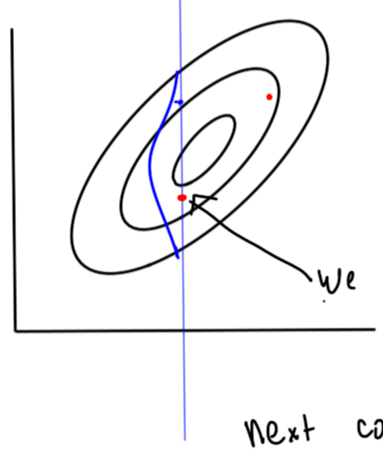
$$\vdots$$

$$x_{i(t+1)} \sim P(x_i \mid x_{1(t+1)} \dots x_{i-1(t+1)}, x_{i+1(t)}, \dots, x_{dt})$$



We draw

$$X_{1(t+1)} \sim P(X_1 \mid X_{2t}) \rightarrow \text{a univariate distribution.}$$



We sample a new value for X_2.

We lock in X_1.
We draw

$$X_{2(t+1)} \sim P(X_2 \mid X_{1(t+1)}) \rightarrow \text{a univariate dist.}$$

we use the value we just drew.

next coordinate $X_{(t+1)} = [X_{1(t+1)}, X_{2(t+1)}]$