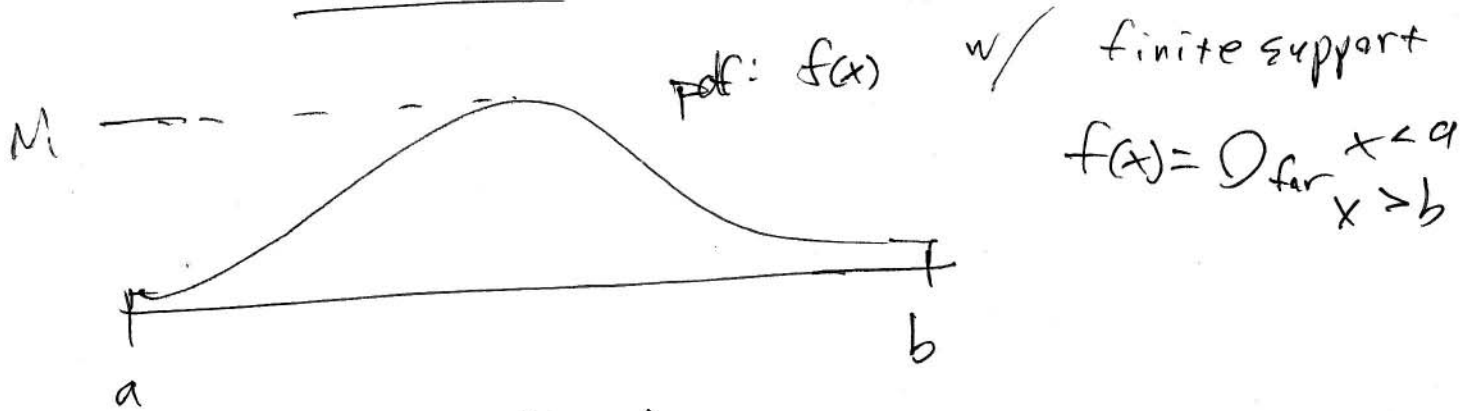


Sampling:

(1)

Rejection sampling:

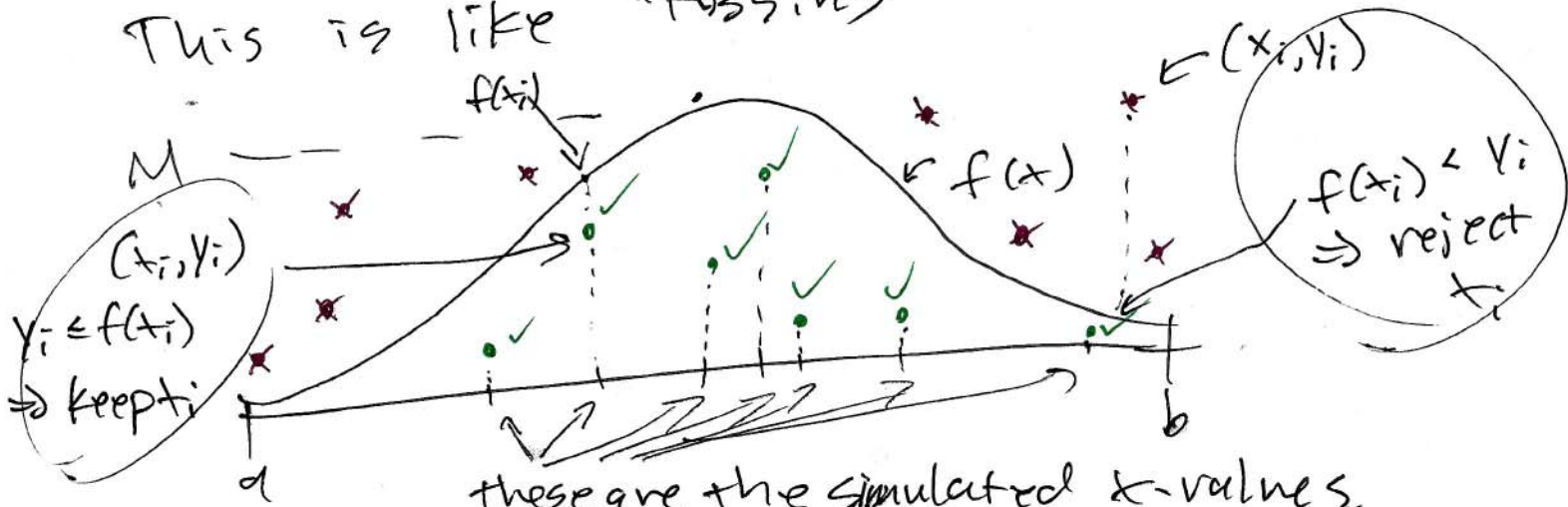


$$M = \max_{[a,b]} \{f(x)\}$$

If we want a sample of size n ,
we can generate n unif. x & y coords. indep.

$$\left. \begin{aligned} X &= \text{runif}(n, a, b) \\ Y &= \text{runif}(n, 0, M) \end{aligned} \right\} \text{plot as } (x, Y)$$

This is like "tossing darts" uniformly



\Rightarrow Keep all x_i with $f(x_i) \geq y_i$

\Rightarrow don't get a sample of size n , but can, keep sampling until get desired sample size.

Rejection sampling.

(2)

Approx. w/ pdf support is infinite



pick a & b s.t. $P(X < a)$, $P(X > b)$
very small

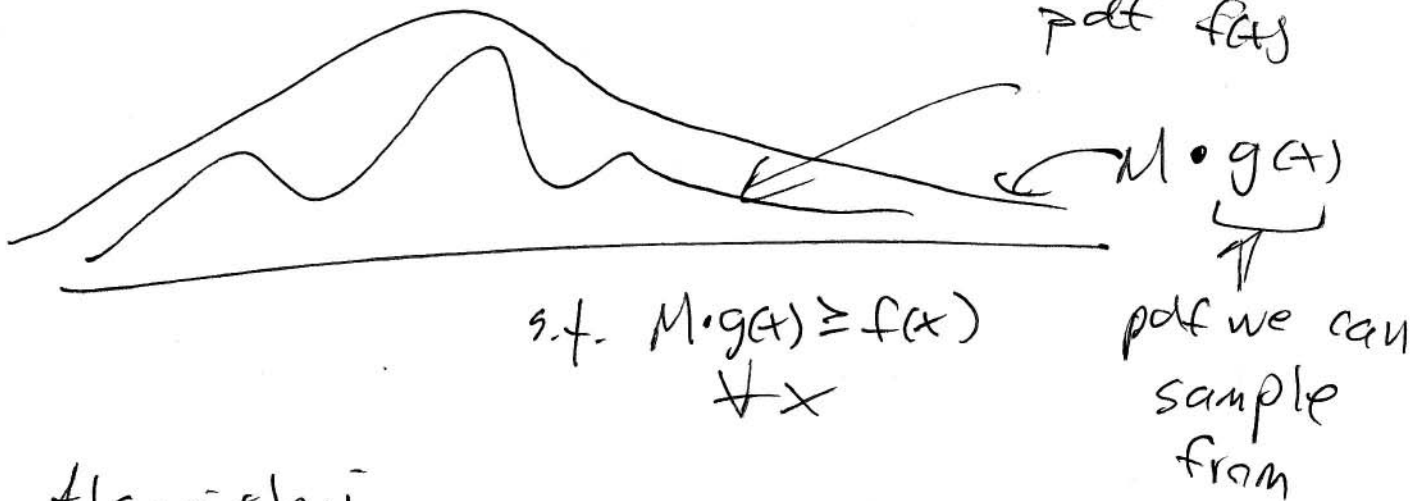
e.g. $f(x) = e^{-x}$, let $a = 0$, $b = 100$ or so.

Then sample as before.

This will be biased since $x > b$ not allowed,
but will still be a good fit.

Better more general rejection sampling

3



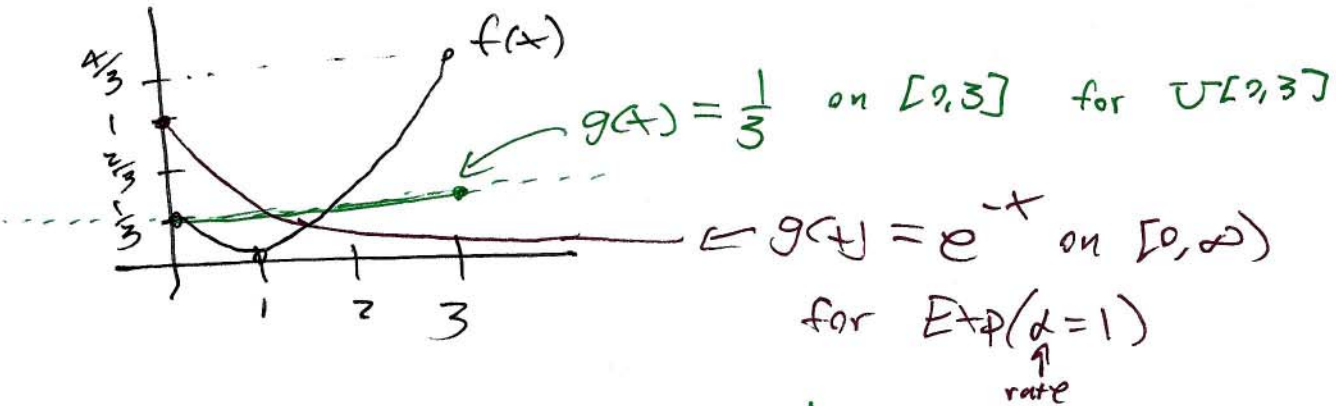
Algorithm.

- 1) simulate x from $g(x)$
- 2) simulate $u \sim U[0,1]$
- 3) If $u \leq \frac{f(x)}{M \cdot g(x)}$ keep x
 $u > \frac{f(x)}{M \cdot g(x)}$ reject x
- 4) continue until desired sample size reached.

ex. $f(x) = \frac{1}{3}(x-1)^2 \quad x \in [0, 3]$

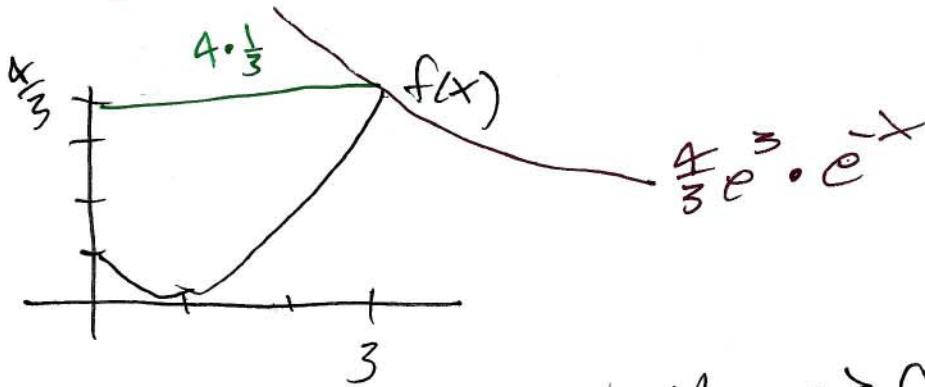
4.

*check L.O.T.P. $\int_0^3 f(x) dx = 1$



1) $g(x) = \frac{1}{3}$ scale up by $M = 4$

2) $g(x) = e^{-x}$ scale up by $M = \frac{4}{3} \cdot e^3$



Now we have 2 options w/ $M \cdot g(x) \geq f(x) \quad \forall x \in [0, 3]$

Algorithm: 1) $x = \text{runif}(n, 0, 3)$ \leftarrow $\begin{matrix} \text{sample from} \\ g(x) = \frac{1}{3} \\ U[0, 3] \end{matrix}$

2) $x = \text{rexp}(n, 1)$
 $u = \text{runif}(n, 0, 1)$

keep only x w/
 $u < \frac{f(x)}{M \cdot g(x)} = \frac{\frac{1}{3}(x-1)^2}{4 \cdot \frac{1}{3}}$

keep only x
 $\forall x \leq 3$
 $\& u < \frac{f(x)}{M \cdot g(x)} = \frac{\frac{1}{3}(x-1)^2}{\frac{4}{3} e^3 \cdot e^{-x}}$

$x_{\text{keep}} = x \left[u < \frac{\frac{1}{3}(x-1)^2}{4 \cdot \frac{1}{3}} \right]$

$x \left[u < \frac{f(x)}{M \cdot g(x)} \right]$