RANDOM VARIABLE GENERATION (PART 1)

And now for something completely different...

Prof. Chris Jermaine

cmj4@cs.rice.edu

Goal In This Class

- Eclectic mix of algorithms and programming
- Remember when I claimed:
  - CS is fundamentally about algorithmics?
  - Programming is relevant only to the extent that it allows computers to run our algs
  - The fact we have to write code is evidence of the epic fail of PL researchers!
- So we won't ever leave algorithms behind
  - And over the next 1.5 classes, will make a significant detour away from PL
  - Will study RV generation
  - Vital to getting our ML algorithm over text to work

What's a Random Variable?

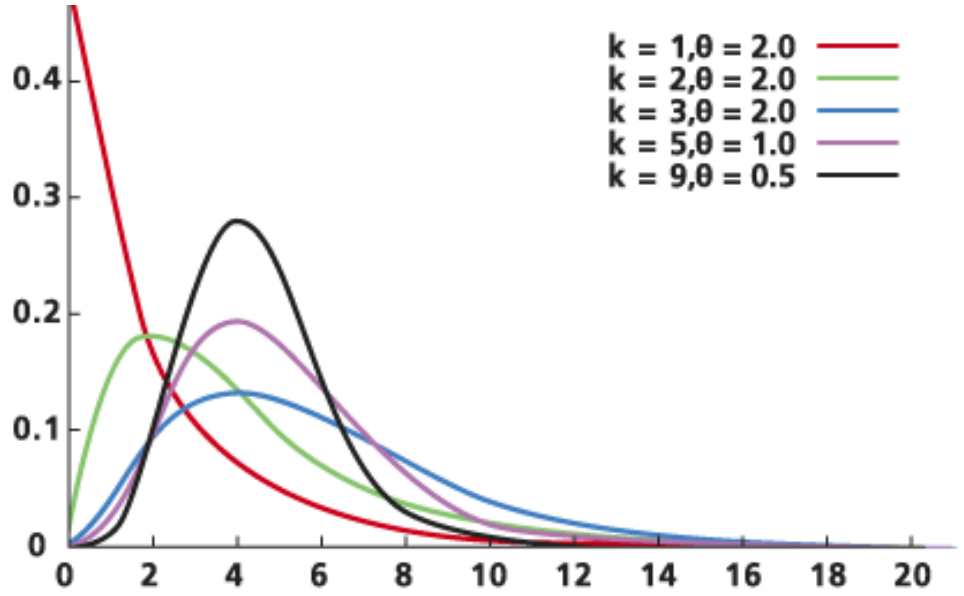
- Think of it like a little machine
- You press a button
- It spits out a random value
- Often, that value is a real number
- But can be anything. You commonly see:
  - — A randomly chosen integer
  - — A randomly chosen item from a set
  - — A random vector of reals
  - — A random sequence of items chosen from a set
  - — And so on...

RVs Whose Domain is Real Numbers

- Meaning you press a button, get out a real
- Are often characterized via a “PDF”
  - — “Probability Density Function”
  - — $f(x)$ denotes the “density” of the RVs distribution at x
- Then $\int_{x_1}^{x_2} f(x)dx$ is...
  - — the probability the machine spits out a value from x_1 to x_2.
  - — Clearly, the total area under the curve f must be one

One Particularly Important Distribution

- Is the “gamma distribution”---will be fundamental to our ML algo
- Takes two parameters:
  - — The shape *k*
  - — The scale θ
- Here’s the PDF for various parameter combos



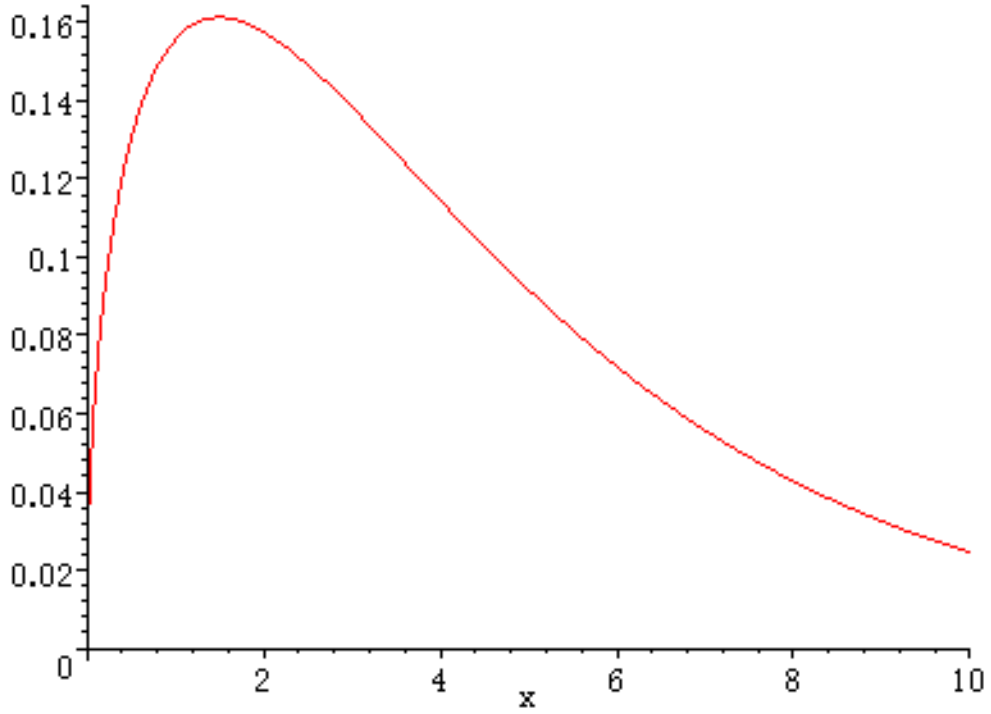
One Particularly Important Distribution

- The mathematical form of the gamma PDF is

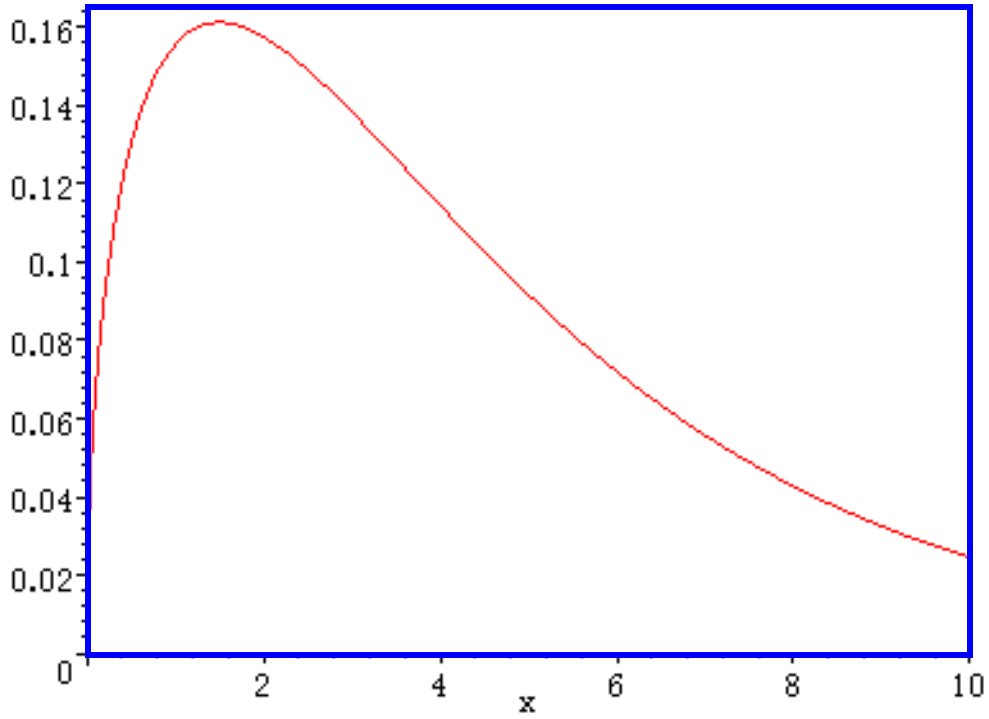
$$f(x|k,\theta) = \frac{x^{k-1}e^{\left(-\frac{x}{\theta}\right)}}{\Gamma(k)\theta^{k}}$$

- An age-old question in CS/comp. statistics:
  - — Given a particular PDF...
  - — how to simulate a RV having exactly that PDF?
  - — In our case, how to simulate a gamma-distributed RV, given the PDF?

The Easiest Way Is “Rejection Sampling”

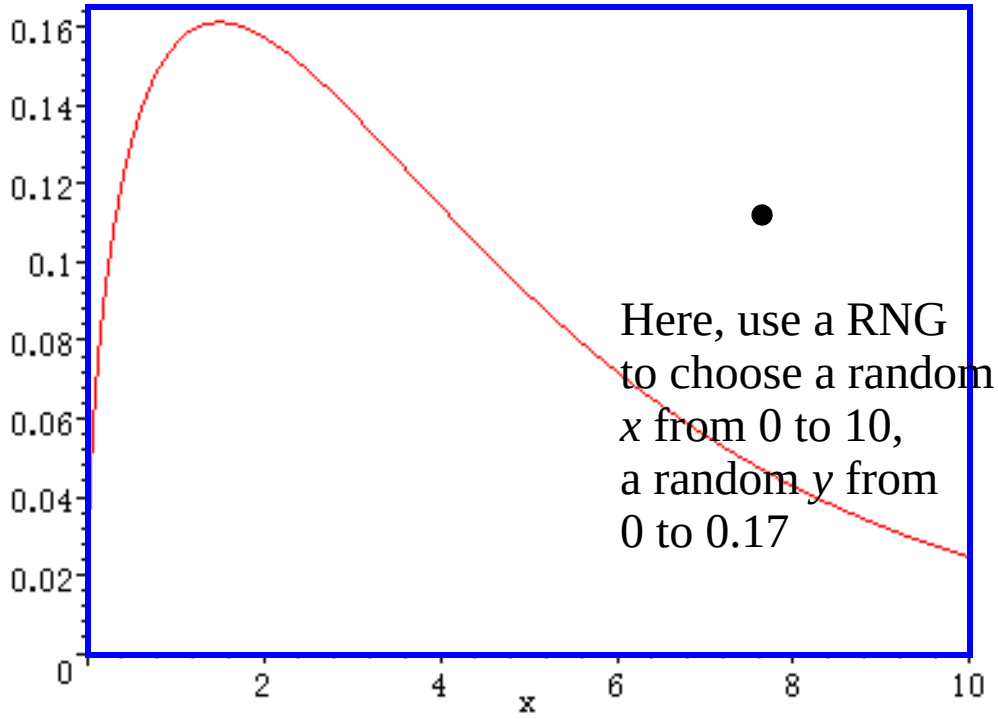


The Easiest Way Is “Rejection Sampling”



— Basic idea is to draw a box (“envelope”) around the PDF

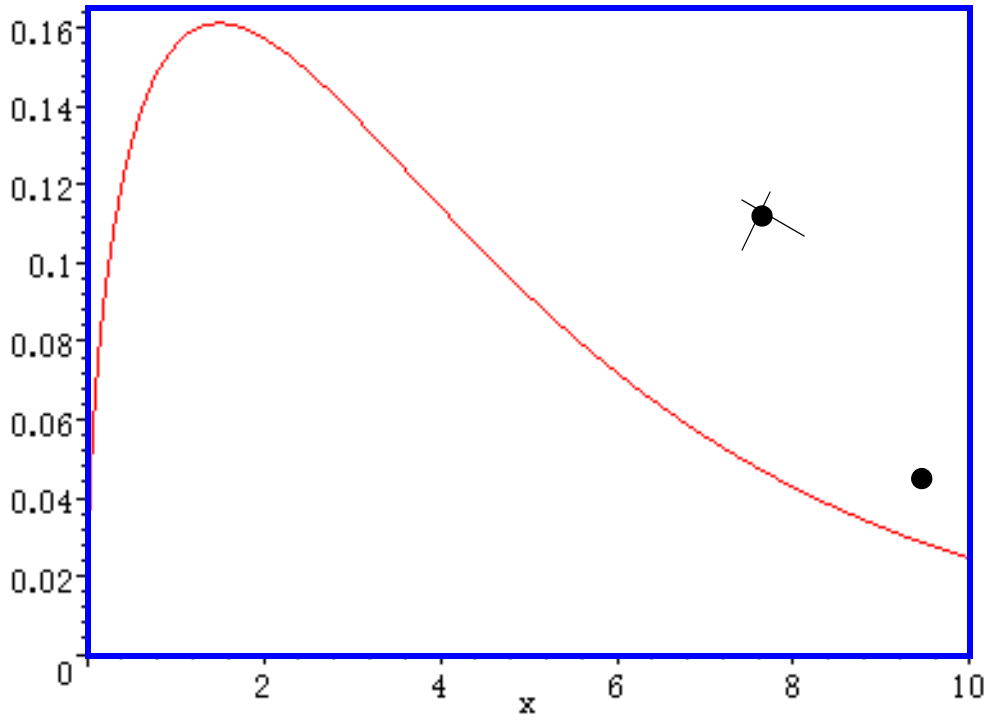
The Easiest Way Is “Rejection Sampling”



— Basic idea is to draw a box (“envelope”) around the PDF

— Then throw a dart randomly into the box

The Easiest Way Is “Rejection Sampling”

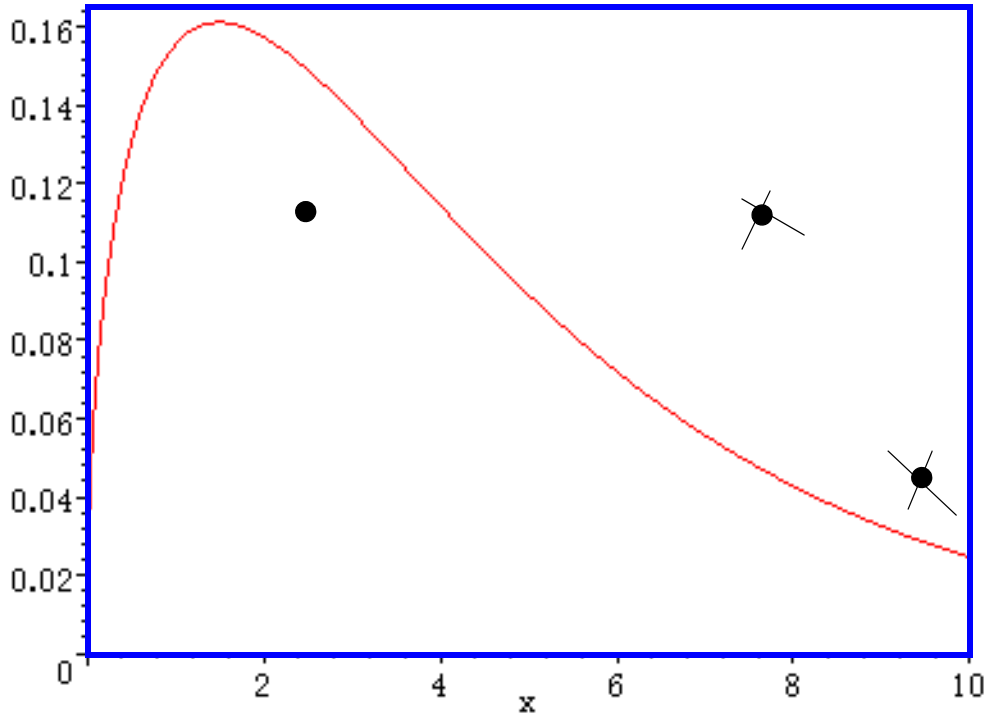


— Basic idea is to draw a box (“envelope”) around the PDF

— Then throw a dart randomly into the box

— If above the PDF, you reject it, and try again

The Easiest Way Is “Rejection Sampling”

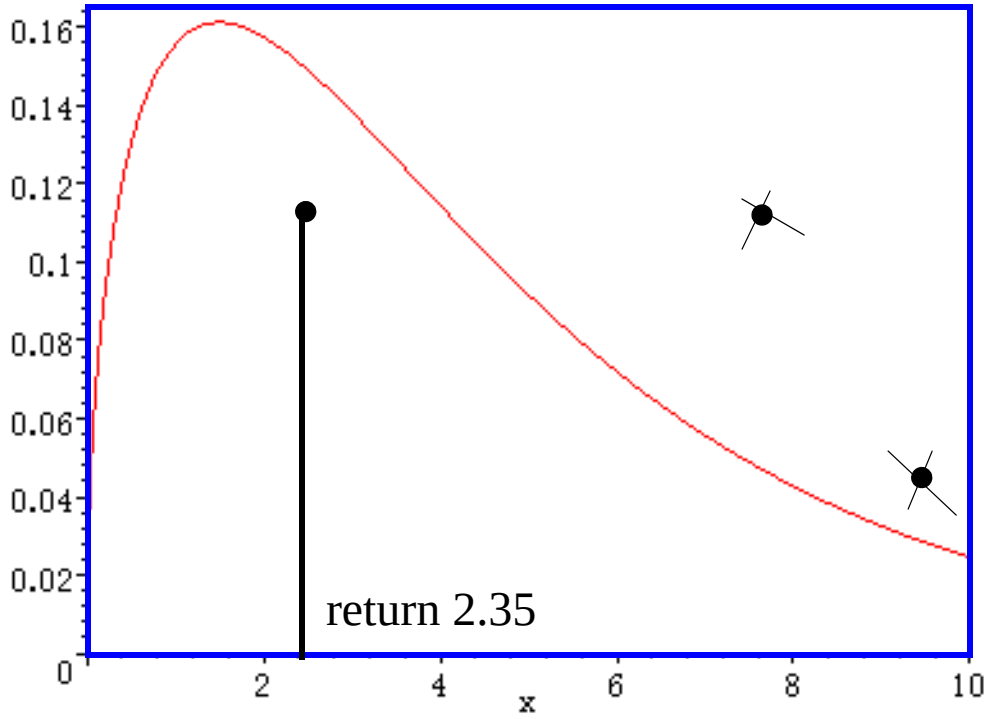


— Basic idea is to draw a box (“envelope”) around the PDF

— Then throw a dart randomly into the box

— If above the PDF, you reject it, and try again, and again

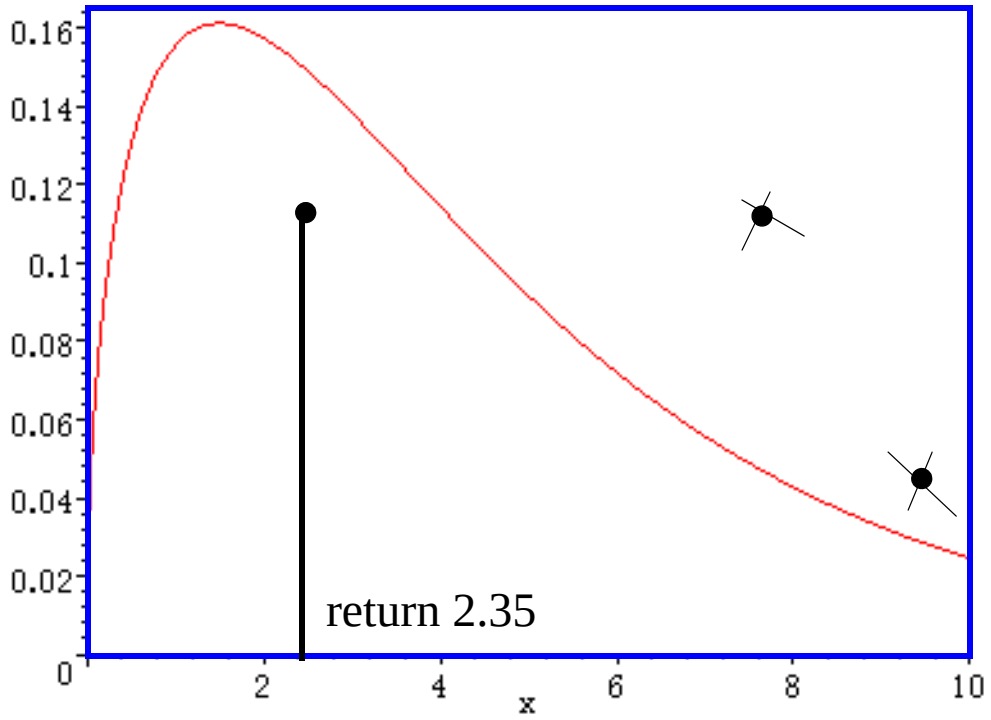
The Easiest Way Is “Rejection Sampling”



— Got one! So the *x* you return is the *x*-val of the first dart under the line

— Can prove this process does in fact sample from a RV whose PDF is the red line

One Super-Cool Thing About RS



— Constant multiplicative factors in PDF can be ignored so for Gamma, kill denom

$$f(x|k,\theta) = \frac{x^{k-1}e^{\left(-\frac{x}{\theta}\right)}}{\Gamma(k)\theta^{k}}$$

Why?

Envelope Can Be Any Shape

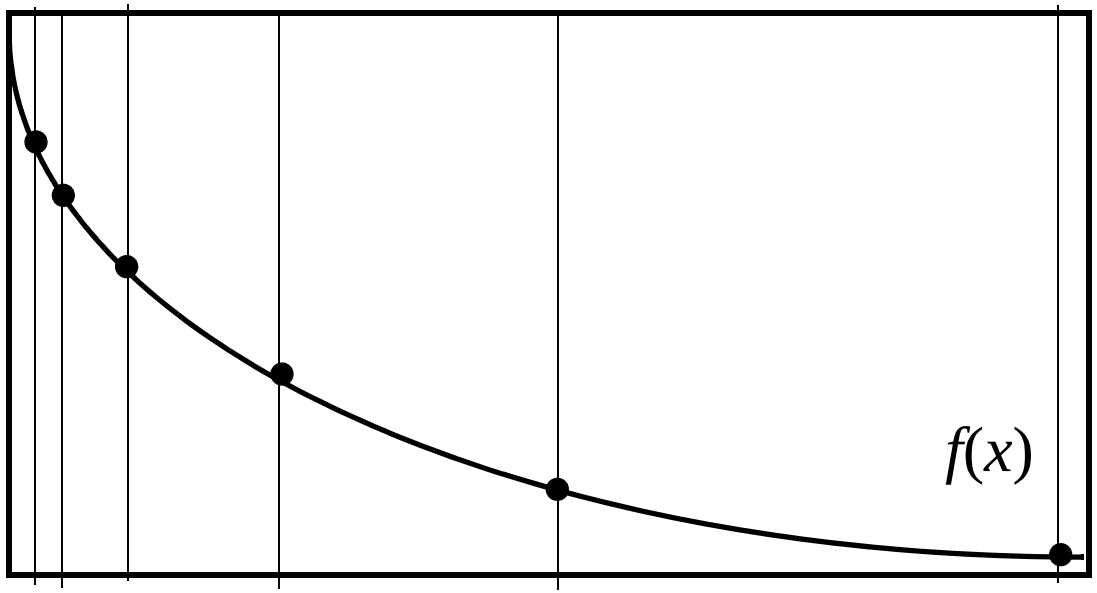
- As long as it fully encloses the PDF
- And as long as your darts hit uniformly in the envelope
- How to do this (somewhat efficiently) for the gamma PDF?
- First, assume the shape is at most one, so concave, decreasing
  - More on what to do if shape exceeds one in a minute
- And assume the scale is one so the PDF becomes

$$f(x|k, \theta) = x^{k-1} e^{(-x)}$$

- Also assume we have l and $h = 2^i l$ for some i, which are the lowest and highest values we can ever sample

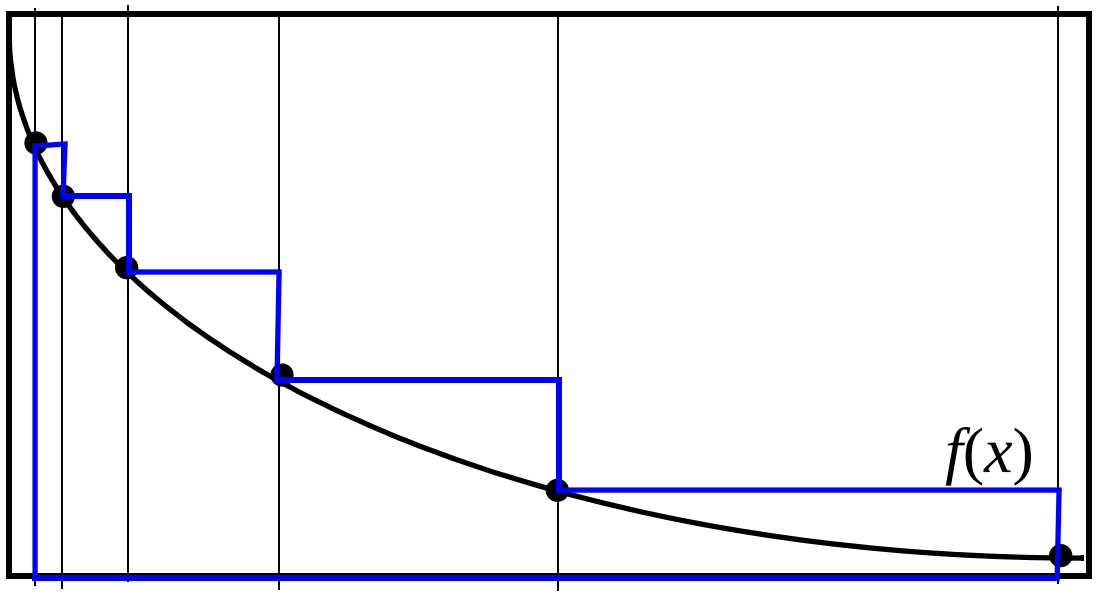
Now What Do We Do?

- First, compute $f(x)$ at l, $2l$, $4l$, $8l$, $16l$, ..., h



Now What Do We Do?

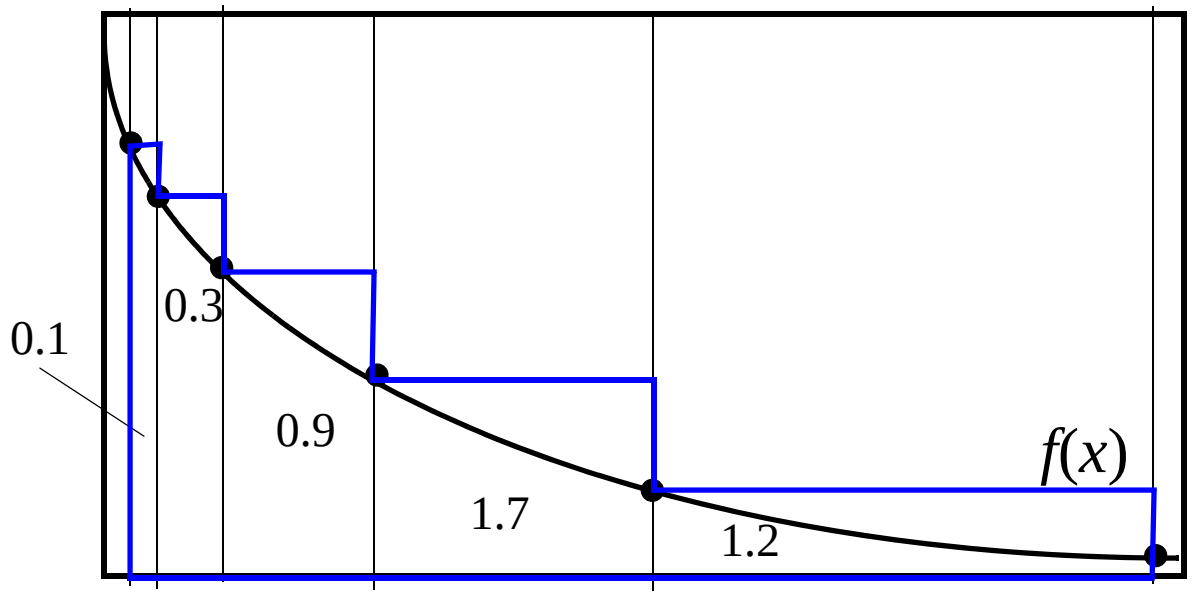
- Your envelope becomes the shape enclosing these dots



— Note: this is *i* steps!

Now What Do We Do?

- Your envelope becomes the shape enclosing these dots

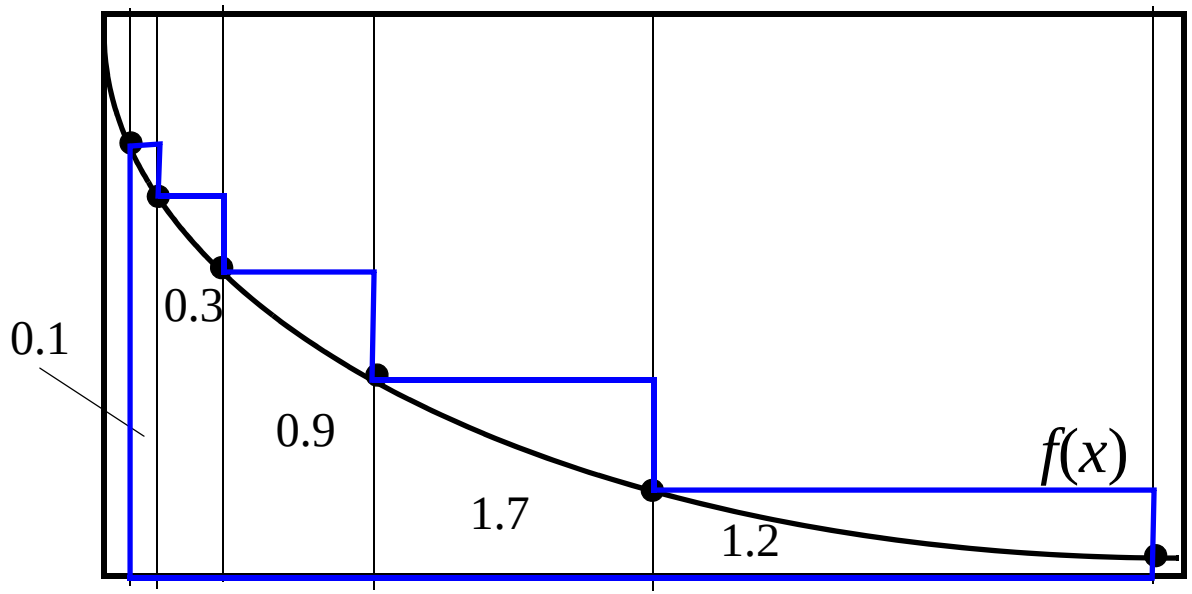


— Note: this is *i* steps!

— Compute the area of each step

Now What Do We Do?

- Your envelope becomes the shape enclosing these dots



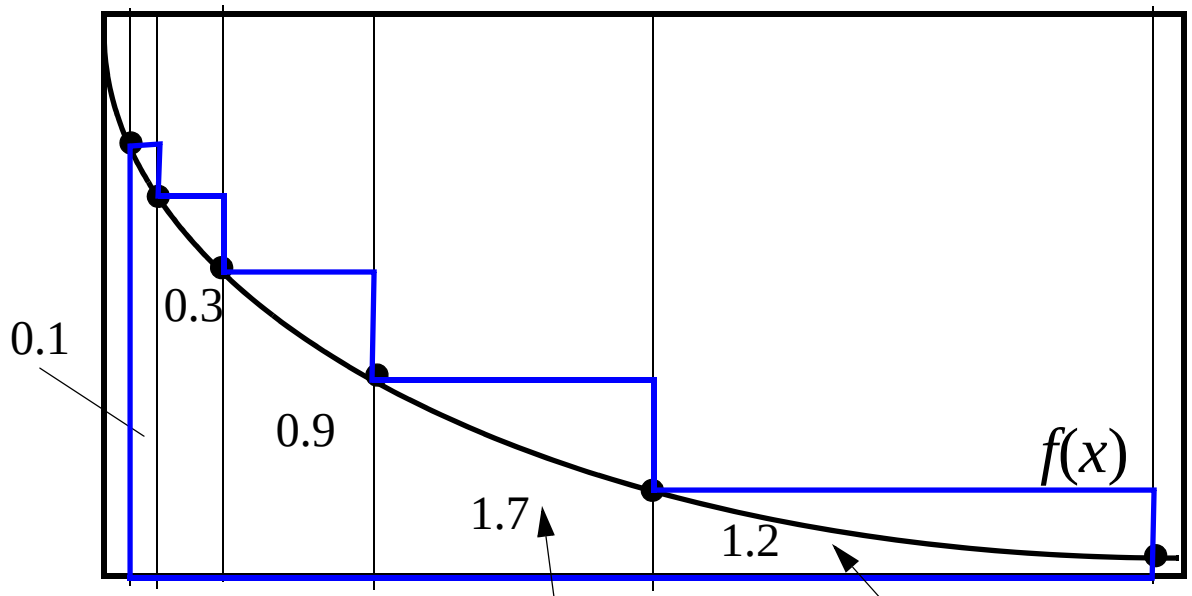
— When it's time to thow a dart, get a random number *r* from 0 to total area

— Choose step *t* if sum of all areas to left of *t* is less than *r*

— But sum of all areas to left of *t*, plus area of *t*, is at least *r*

Now What Do We Do?

- Your envelope becomes the shape enclosing these dots



— When it's time to throw a dart, get a random number *r* from 0 to total area

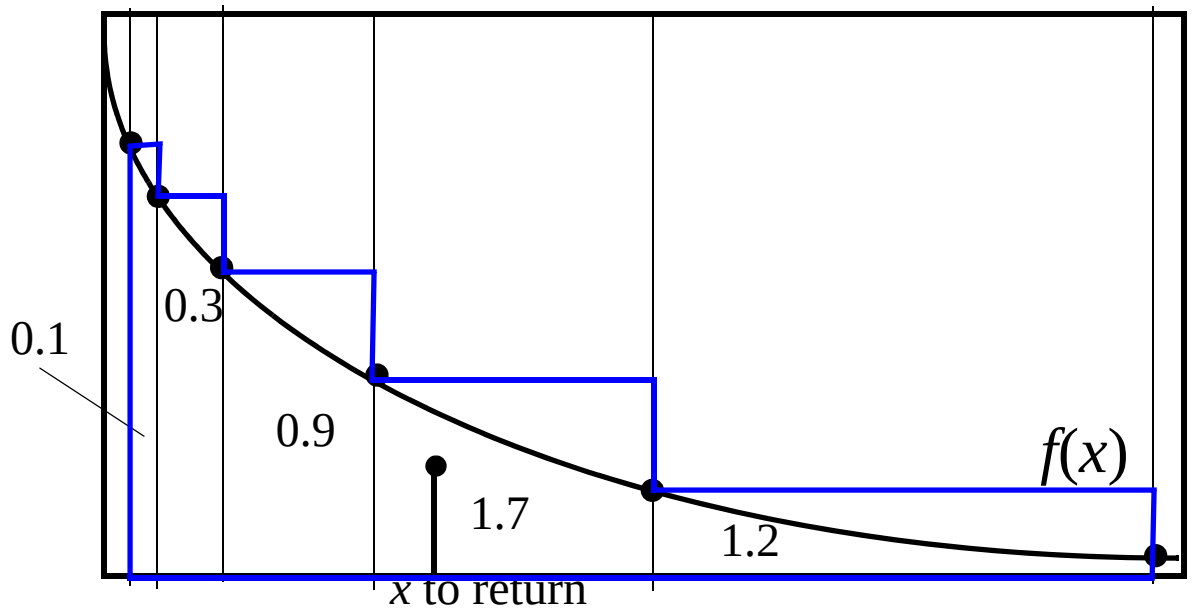
— Choose step *t* if sum of all areas to left of *t* is less than *r*

— But sum of all areas to left of *t*, plus area of *t*, is at least *r*

— Example: *r* = 2.9, choose this one; r = 4.1, choose this one

Now What Do We Do?

- Your envelope becomes the shape enclosing these dots

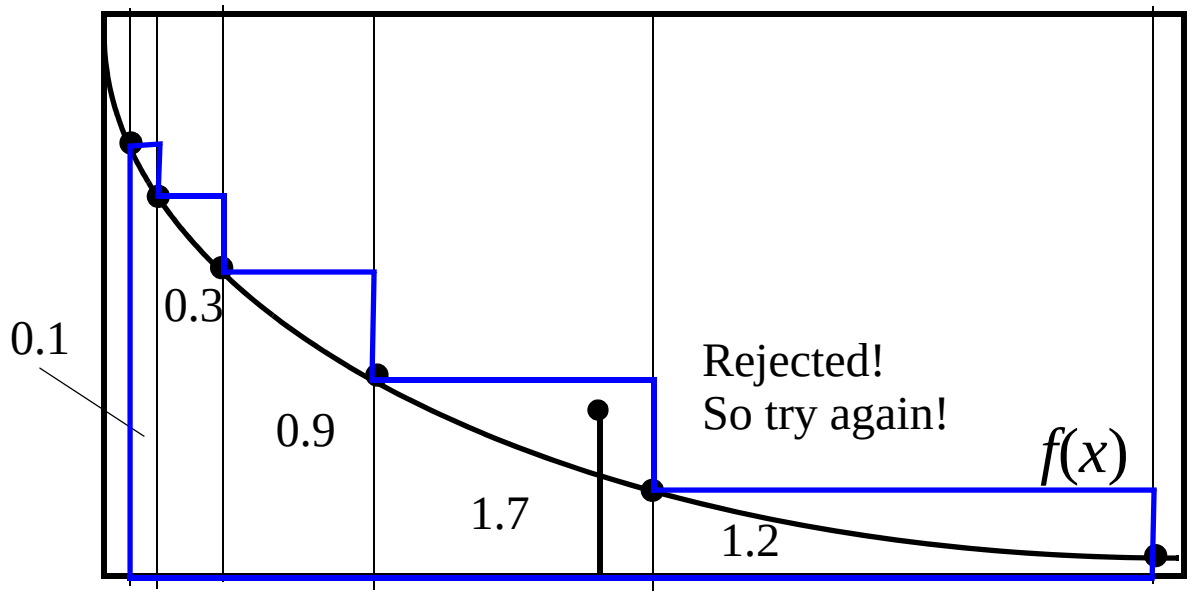


— Whatever step you choose, “throw” a dart uniformly within the step

— If it is under the PDF, then return the *x* of the dart that you chose

Now What Do We Do?

- Your envelope becomes the shape enclosing these dots



— Whatever step you choose, “throw” a dart uniformly within the step

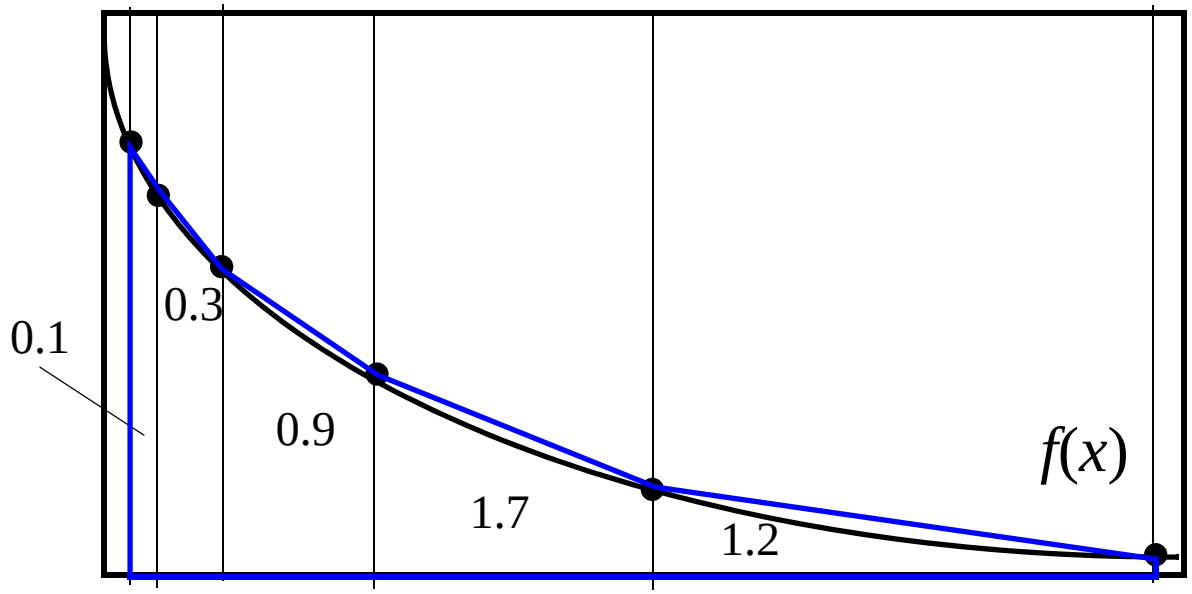
— If it is under the PDF, then return the *x* of the dart that you chose

— If it is not under the PDF, then choose a step using same process, and try again

— Keep going until you get one under the PDF!

Want To Be Really Fancy?

- Use trapezoids instead of steps!
- Will result in less rejections
- But remember, your dart must fall uniformly within the trapezoid



Two More Issues

- What if scale you want is not one?
  - — Can just generate scale one RV, then multiply by theta to get a scale-theta RV
- What if shape you want exceeds one?
  - — Turns out if X_1 and X_2 are gamma RV with same scale
  - — And X_1, has shape k_1, X_2 has shape k_2
  - — Then $X_1 + X_2$ is a gamma RV with shape $k_1 + k_2$
  - — Sooooo...
  - — To generate gamma RV with $k > 1$...
  - — Let j = floor(k)
  - — Generate and sum j gamma RVs with shape 1, and one more with shape (k - j)

Questions?