Algorithm steps

1. Initialization Step
2. Check:
   1. g(x) is continuous and differentiable everywhere in D
   2. h(x) = log(g(x)) is concave everywhere in D
   3. Note: D is the domain
3. Initialize T\_k (a vector with k elements): :
   1. Tk =
   2. If D unbounded on the left, chose x1 s.t. h\_prime(x1) > 0
   3. If D unbounded on the right, chose xk s.t. h\_prime(xk) < 0
4. Evaluate h(x) and h0(x) on Tk and store these as two length k vectors, say h\_x and h\_prime\_x:
   1. g(x) is continuous and differentiable everywhere in D
   2. h(x) = log(g(x)) is concave everywhere in D
5. Calculate z (vector with k + 1 elements)::
   1. z0 = lower bound of D (or −∞ if D is not bounded below)
   2. zj =
   3. zk = upper bound of D (or +∞ if D is not bounded above)
6. Calculate u\_k :
   1. uk(x) =
   2. note: this is the piecewise linear upper hull formed by tangents to h(x) at Tk
7. Calculate s\_k:
   1. Write s\_k here:
8. Calculate l\_k ():
   1. g(x) is continuous and differentiable everywhere in D
   2. For x < x1 or x > xk, define lk(x) = −∞
   3. Note: this is the piecewise linear lower hull formed by connecting adjacent points on h(x) where Tk is evaluated
9. Sampling Step
10. Sample a value x\_star from s\_k(x)
11. Sample a value w independently from Unif(0, 1)
12. Perform the test:

1. Updating Step

If h(x∗) and h\_prime(x∗) were evaluated in Sampling Step:

Include x∗in Tk to form Tk+1

Relabel the xi in Tk in ascending order

Construct new functions uk+1(x), sk+1(x), and lk+1(x)

Increment k

Return to Sampling Step if n points have not been sampled yet

Else:

No need to update, repeat Sampling Step until n points are sampled.