```julia
using Comrade
using Distributions
using Pathfinder
using AdvancedHMC
using Plots
# load eht-imaging we use this to load eht data
load_ehtim()
# To download the data visit https://doi.org/10.25739/g85n-f134
obs = ehtim.obsdata.load_uvfits("SR1_M87_2017_096_lo_hops_netcal_StokesI.uvfits")
obs.add_scans()
# kill 0-baselines since we don't care about
# large scale flux and make scan-average data
obs = obs.flag_uvdist(uv_min=0.1e9).avg_coherent(0.0, scan_avg=true)
# extract log closure amplitudes and closure phases
dlcamp = extract_lcamp(obs; count="min")
dcphase = extract_cphase(obs, count="min")
# form the likelihood
lklhd = RadioLikelihood(dlcamp, dcphase)
# build the model here we fit a ring with a azimuthal
#brightness variation and a Gaussian
function model(θ)
  (;radius, width, α, β, f, σG, τG, ξG, xG, yG) = θ
  ring = f*smoothed(stretched(MRing((α,), (β,)), radius, radius), width)
  g = (1-f)*shifted(rotated(stretched(Gaussian(), σG, σG*(1+τG)), ξG), xG, yG)
  return ring + g
end
# define the priors
prior = (
          radius = Uniform(μas2rad(10.0), μas2rad(30.0)),
          width = Uniform(μas2rad(1.0), μas2rad(10.0)),
          α = Uniform(-0.5, 0.5), β = Uniform(-0.5, 0.5),
          f = Uniform(0.0, 1.0),
          σG = Uniform(μas2rad(1.0), μas2rad(40.0)),
          τG = Uniform(0.0, 0.75),
          ξG = Uniform(0.0, 1π),
          xG = Uniform(-μas2rad(80.0), μas2rad(80.0)),
          yG = Uniform(-μas2rad(80.0), μas2rad(80.0))
        )
# Now form the posterior
post = Posterior(lklhd, prior, model)
# We will use HMC to sample the posterior.
# First to reduce burn in we use pathfinder
q, phi, _ = multipathfinder(post, 100)
# now we sample using hmc
metric = DiagEuclideanMetric(dimension(post))
chain, stats = sample(post, HMC(;metric), 2000; nadapts=1000, init_params=phi[1])
# plot a draw from the posterior
plot(model(chain[end]))
```