Fitzhugh-Nagumo Bayesian Parameter Estimation Benchmarks

Vaibhav Dixit, Chris Rackauckas

March 10, 2019

```
using DiffEqBayes, BenchmarkTools

using OrdinaryDiffEq, RecursiveArrayTools, Distributions, ParameterizedFunctions, Mamba
using Plots

gr(fmt=:png)

Plots.GRBackend()
```

0.0.1 Defining the problem.

The FitzHugh-Nagumo model is a simplified version of Hodgkin-Huxley model and is used to describe an excitable system (e.g. neuron).

```
fitz = @ode_def FitzhughNagumo begin
  dv = v - v^3/3 -w + l
  dw = τinv*(v +  a - b*w)
end a b τinv l
```

```
(::Main.WeaveSandBox10.FitzhughNagumo{getfield(Main.WeaveSandBox10, Symbol(
"##1#5")),getfield(Main.WeaveSandBox10, Symbol("##2#6")),getfield(Main.Weav
eSandBox10, Symbol("##3#7")),Nothing,Nothing,getfield(Main.WeaveSandBox10,
Symbol("##4#8")),Expr,Expr}) (generic function with 2 methods)
```

```
prob_ode_fitzhughnagumo = ODEProblem(fitz,[1.0,1.0],(0.0,10.0),[0.7,0.8,1/12.5,0.5])
sol = solve(prob_ode_fitzhughnagumo, Tsit5())
```

```
retcode: Success
Interpolation: specialized 4th order "free" interpolation
t: 14-element Array{Float64,1}:
  0.0
  0.15079562872319327
  0.6663751602069676
  1.45491551203011
  2.634172592705079
  3.7872847149850593
  5.149289290296318
  6.764810021639485
  7.606013876691694
  8.324324391350054
  9.040746828687205
  9.552469682130221
```

```
  9.985006344186914
 10.0
u: 14-element Array{Array{Float64,1},1}:
 [1.0, 1.0]
 [1.02428, 1.01095]
 [1.09254, 1.04957]
 [1.14789, 1.11021]
 [1.13454, 1.19755]
 [1.04328, 1.27187]
 [0.844691, 1.3381]
 [0.313544, 1.36894]
 [-0.409826, 1.34276]
 [-1.40824, 1.27062]
 [-1.90978, 1.15634]
 [-1.96185, 1.06889]
 [-1.95443, 0.996705]
 [-1.95386, 0.994246]
```

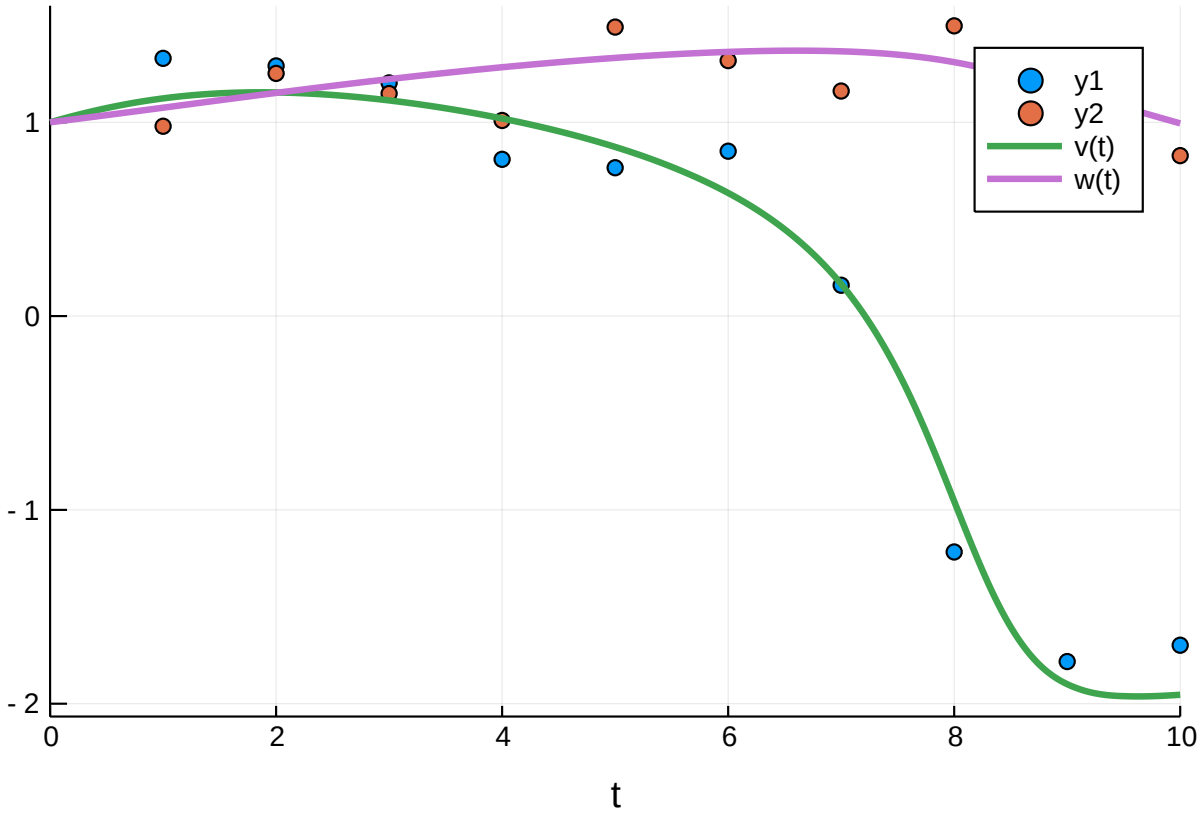
Data is genereated by adding noise to the solution obtained above.

```
t = collect(range(1,stop=10,length=10))
sig = 0.20
data = convert(Array, VectorOfArray([(sol(t[i]) + sig*randn(2)) for i in 1:length(t)]))
```

```
2×10 Array{Float64,2}:
 1.32985   1.29074  1.20305  0.808793  …  -1.21709  -1.7825   -1.69817
 0.979494  1.25211  1.14817  1.00919       1.49749   1.33255   0.827827
```

0.0.2 Plot of the data and the solution.

```
scatter(t, data[1,:])
scatter!(t, data[2,:])
plot!(sol)
```



0.0.3 Priors for the parameters which will be passed for the Bayesian Inference

```
priors =
    [Truncated(Normal(1.0,0.5),0,1.5),Truncated(Normal(1.0,0.5),0,1.5),Truncated(Normal(0.0,0.5),-0.5,
```

```
4-element Array{Distributions.Truncated{Distributions.Normal{Float64},Distr
ibutions.Continuous},1}:
 Truncated(Distributions.Normal{Float64}(μ=1.0, σ=0.5), range=(0.0, 1.5))
 Truncated(Distributions.Normal{Float64}(μ=1.0, σ=0.5), range=(0.0, 1.5))
 Truncated(Distributions.Normal{Float64}(μ=0.0, σ=0.5), range=(-0.5, 0.5))
 Truncated(Distributions.Normal{Float64}(μ=0.5, σ=0.5), range=(0.0, 1.0))
```

0.1 Parameter Estimation with Stan.jl backend

```
@time bayesian_result_stan =
    stan_inference(prob_ode_fitzhughnagumo,t,data,priors;reltol=1e-5,abstol=1e-5,vars
    =(StanODEData(),InverseGamma(3,2)))
```

```
make: `/Users/vaibhav/DiffEqBenchmarks.jl/tmp/parameter_estimation_model' i
s up to date.

Length of data array is not equal to nchains,
all chains will use the first data dictionary.

Calling /Users/vaibhav/Downloads/cmdstan-2.18.0/bin/stansummary to infer ac
ross chains.

Inference for Stan model: parameter_estimation_model_model
4 chains: each with iter=(1000,1000,1000,1000); warmup=(0,0,0,0); thin=(1,1
,1,1); 4000 iterations saved.

Warmup took (25, 28, 26, 24) seconds, 1.7 minutes total
```

```
Sampling took (21, 29, 28, 25) seconds, 1.7 minutes total

                    Mean     MCSE   StdDev     5%    50%    95%    N_Eff  N
_Eff/s    R_hat
lp__              8.3e+00  7.0e-02  2.0e+00    4.5    8.6     11  8.5e+02  8
.3e+00  1.0e+00
accept_stat__     9.1e-01  1.6e-02  1.6e-01   0.55   0.97   1.00  1.0e+02  1
.0e+00  1.0e+00
stepsize__        3.8e-02  3.6e-03  5.0e-03  0.034  0.038  0.046  2.0e+00  1
.9e-02  8.1e+13
treedepth__       6.2e+00  1.4e-01  7.5e-01    5.0    6.0    7.0  3.0e+01  2
.9e-01  1.1e+00
n_leapfrog__      9.5e+01  8.5e+00  3.8e+01     31    127    127  2.0e+01  2
.0e-01  1.1e+00
divergent__       2.5e-04      nan  1.6e-02   0.00   0.00   0.00      nan
   nan  1.0e+00
energy__         -5.3e+00  8.9e-02  2.7e+00   -9.1   -5.6  -0.38  9.1e+02  8
.8e+00  1.0e+00
sigma1[1]         2.8e-01  1.7e-03  7.6e-02   0.18   0.27   0.42  2.0e+03  1
.9e+01  1.0e+00
sigma1[2]         2.9e-01  2.1e-03  8.2e-02   0.19   0.28   0.44  1.5e+03  1
.4e+01  1.0e+00
theta1            9.2e-01  8.0e-03  3.2e-01   0.35   0.94    1.4  1.6e+03  1
.6e+01  1.0e+00
theta2            9.5e-01  5.9e-03  2.8e-01   0.46   0.97    1.4  2.2e+03  2
.2e+01  1.0e+00
theta3            9.6e-02  1.1e-03  3.8e-02  0.041  0.093   0.16  1.2e+03  1
.2e+01  1.0e+00
theta4            5.4e-01  2.3e-03  8.5e-02   0.41   0.53   0.69  1.4e+03  1
.3e+01  1.0e+00
theta[1]          9.2e-01  8.0e-03  3.2e-01   0.35   0.94    1.4  1.6e+03  1
.6e+01  1.0e+00
theta[2]          9.5e-01  5.9e-03  2.8e-01   0.46   0.97    1.4  2.2e+03  2
.2e+01  1.0e+00
theta[3]          9.6e-02  1.1e-03  3.8e-02  0.041  0.093   0.16  1.2e+03  1
.2e+01  1.0e+00
theta[4]          5.4e-01  2.3e-03  8.5e-02   0.41   0.53   0.69  1.4e+03  1
.3e+01  1.0e+00

Samples were drawn using hmc with nuts.
For each parameter, N_Eff is a crude measure of effective sample size,
and R_hat is the potential scale reduction factor on split chains (at
convergence, R_hat=1).

 57.295403 seconds (729.99 k allocations: 33.604 MiB, 0.10% gc time)
DiffEqBayes.StanModel{Int64,Mamba.Chains}(0, Object of type "Mamba.Chains"

Iterations = 1:1000
Thinning interval = 1
Chains = 1,2,3,4
Samples per chain = 1000

[10.6671 0.93648 ... 0.083311 0.475477; 8.93928 0.725224 ... 0.0673232 0.434814
; ... ; 5.49003 0.712586 ... 0.154265 0.510522; 7.26222 0.969502 ... 0.0600613 0.
567987]

[10.3335 0.993081 ... 0.136939 0.641814; 10.1996 0.715973 ... 0.035856 0.402193
; ... ; 2.52175 0.532444 ... 0.131175 0.606105; 8.2773 0.961207 ... 0.0495424 0.4
68416]
```

```
[9.46131 0.902956 ... 0.0957212 0.551446; 10.5499 0.987558 ... 0.0470144 0.4558
19; ... ; 7.53464 0.924342 ... 0.138289 0.569619; 8.49457 0.994005 ... 0.125617 0
.676039]

[7.95098 0.949766 ... 0.0843145 0.440079; 6.94445 0.996451 ... 0.0425567 0.5270
43; ... ; 10.0713 0.986234 ... 0.0630492 0.486561; 6.96996 0.994605 ... 0.0903822
 0.558057])
```

```
plot_chain(bayesian_result_stan)
```

```
Compose.SVG(203.2mm, 203.2mm, IOBuffer(data=UInt8[...], readable=true, writ
able=true, seekable=true, append=false, size=192646, maxsize=Inf, ptr=19264
7, mark=-1), nothing, "img-f0d2e2a9", 0, Compose.SVGPropertyFrame[], Dict{T
ype,Union{Nothing, Property}}(Property{LineWidthPrimitive}=>nothing,Propert
y{JSCallPrimitive}=>nothing,Property{FontSizePrimitive}=>nothing,Property{S
VGClassPrimitive}=>nothing,Property{JSIncludePrimitive}=>nothing,Property{F
ontPrimitive}=>nothing,Property{SVGAttributePrimitive}=>nothing,Property{St
rokeDashPrimitive}=>nothing,Property{StrokePrimitive}=>nothing,Property{Fil
lPrimitive}=>nothing...), OrderedCollections.OrderedDict{Compose.ClipPrimitiv
e,String}(ClipPrimitive{Tuple{Length{:mm,Float64},Length{:mm,Float64}}}(Tup
le{Measures.Length{:mm,Float64},Measures.Length{:mm,Float64}}[(117.885mm, 1
46.078mm), (198.2mm, 146.078mm), (198.2mm, 183.915mm), (117.885mm, 183.915m
m)])=>"img-f0d2e2a9-4",ClipPrimitive{Tuple{Length{:mm,Float64},Length{:mm,F
loat64}}}(Tuple{Measures.Length{:mm,Float64},Measures.Length{:mm,Float64}}[
(21.9783mm, 146.078mm), (96.6mm, 146.078mm), (96.6mm, 183.915mm), (21.9783m
m, 183.915mm)])=>"img-f0d2e2a9-35",ClipPrimitive{Tuple{Length{:mm,Float64},
Length{:mm,Float64}}}(Tuple{Measures.Length{:mm,Float64},Measures.Length{:m
m,Float64}}[(119.558mm, 78.345mm), (198.2mm, 78.345mm), (198.2mm, 116.182mm
), (119.558mm, 116.182mm)])=>"img-f0d2e2a9-73",ClipPrimitive{Tuple{Length{:
mm,Float64},Length{:mm,Float64}}}(Tuple{Measures.Length{:mm,Float64},Measur
es.Length{:mm,Float64}}[(18.6317mm, 78.345mm), (96.6mm, 78.345mm), (96.6mm,
 116.182mm), (18.6317mm, 116.182mm)])=>"img-f0d2e2a9-111",ClipPrimitive{Tup
le{Length{:mm,Float64},Length{:mm,Float64}}}(Tuple{Measures.Length{:mm,Floa
t64},Measures.Length{:mm,Float64}}[(121.905mm, 10.6117mm), (198.2mm, 10.611
7mm), (198.2mm, 48.4483mm), (121.905mm, 48.4483mm)])=>"img-f0d2e2a9-145",Cl
ipPrimitive{Tuple{Length{:mm,Float64},Length{:mm,Float64}}}(Tuple{Measures.
Length{:mm,Float64},Measures.Length{:mm,Float64}}[(17.9583mm, 10.6117mm), (
96.6mm, 10.6117mm), (96.6mm, 48.4483mm), (17.9583mm, 48.4483mm)])=>"img-f0d
2e2a9-185"), Tuple{Compose.FormPrimitive,String}[], Set(AbstractString[]),
true, false, nothing, true, "img-f0d2e2a9-214", false, 214, Set(AbstractStr
ing["/Users/vaibhav/.julia/packages/Gadfly/09PWZ/src/gadfly.js"]), Set(Tupl
e{AbstractString,AbstractString}[("Snap.svg", "Snap"), ("Gadfly", "Gadfly")
]), AbstractString["fig.select(\"#img-f0d2e2a9-5\")\n   .init_gadfly();", "
fig.select(\"#img-f0d2e2a9-36\")\n   .init_gadfly();", "fig.select(\"#img-f
0d2e2a9-74\")\n   .init_gadfly();", "fig.select(\"#img-f0d2e2a9-112\")\n
.init_gadfly();", "fig.select(\"#img-f0d2e2a9-146\")\n   .init_gadfly();",
"fig.select(\"#img-f0d2e2a9-186\")\n   .init_gadfly();"], false, :none, (BB
ox{l,t,r,b,w,h = 17.95833333333333mm,10.611666666666665mm, 96.6mm,48.448333
33333332mm, 78.64166666666667mm,37.83666666666666mm}, Compose.UnitBox{Float
64,Float64,Float64,Float64}(-26.794685720665406, 16.182149541917052, 1053.5
89371441331, -22.364299083834105, 0.0mm, 0.0mm, 0.0mm, 0.0mm), Compose.Iden
tityTransform()))
Press ENTER to draw next plot
Compose.SVG(203.2mm, 203.2mm, IOBuffer(data=UInt8[...], readable=true, writ
able=true, seekable=true, append=false, size=203608, maxsize=Inf, ptr=20360
9, mark=-1), nothing, "img-40f07e0c", 0, Compose.SVGPropertyFrame[], Dict{T
ype,Union{Nothing, Property}}(Property{LineWidthPrimitive}=>nothing,Propert
y{JSCallPrimitive}=>nothing,Property{FontSizePrimitive}=>nothing,Property{S
VGClassPrimitive}=>nothing,Property{JSIncludePrimitive}=>nothing,Property{F
```

```
ontPrimitive}=>nothing,Property{SVGAttributePrimitive}=>nothing,Property{St
rokeDashPrimitive}=>nothing,Property{StrokePrimitive}=>nothing,Property{Fil
lPrimitive}=>nothing...), OrderedCollections.OrderedDict{Compose.ClipPrimitiv
e,String}(ClipPrimitive{Tuple{Length{:mm,Float64},Length{:mm,Float64}}}(Tup
le{Measures.Length{:mm,Float64},Measures.Length{:mm,Float64}}[(120.232mm, 1
46.078mm), (198.2mm, 146.078mm), (198.2mm, 183.915mm), (120.232mm, 183.915m
m)])=>"img-40f07e0c-4",ClipPrimitive{Tuple{Length{:mm,Float64},Length{:mm,F
loat64}}}(Tuple{Measures.Length{:mm,Float64},Measures.Length{:mm,Float64}}[
(18.6317mm, 146.078mm), (96.6mm, 146.078mm), (96.6mm, 183.915mm), (18.6317m
m, 183.915mm)])=>"img-40f07e0c-43",ClipPrimitive{Tuple{Length{:mm,Float64},
Length{:mm,Float64}}}(Tuple{Measures.Length{:mm,Float64},Measures.Length{:m
m,Float64}}[(121.905mm, 78.345mm), (198.2mm, 78.345mm), (198.2mm, 116.182mm
), (121.905mm, 116.182mm)])=>"img-40f07e0c-79",ClipPrimitive{Tuple{Length{:
mm,Float64},Length{:mm,Float64}}}(Tuple{Measures.Length{:mm,Float64},Measur
es.Length{:mm,Float64}}[(19.6317mm, 78.345mm), (96.6mm, 78.345mm), (96.6mm,
 116.182mm), (19.6317mm, 116.182mm)])=>"img-40f07e0c-119",ClipPrimitive{Tup
le{Length{:mm,Float64},Length{:mm,Float64}}}(Tuple{Measures.Length{:mm,Floa
t64},Measures.Length{:mm,Float64}}[(117.885mm, 10.6117mm), (198.2mm, 10.611
7mm), (198.2mm, 48.4483mm), (117.885mm, 48.4483mm)])=>"img-40f07e0c-157",Cl
ipPrimitive{Tuple{Length{:mm,Float64},Length{:mm,Float64}}}(Tuple{Measures.
Length{:mm,Float64},Measures.Length{:mm,Float64}}[(16.285mm, 10.6117mm), (9
6.6mm, 10.6117mm), (96.6mm, 48.4483mm), (16.285mm, 48.4483mm)])=>"img-40f07
e0c-195"), Tuple{Compose.FormPrimitive,String}[], Set(AbstractString[]), tr
ue, false, nothing, true, "img-40f07e0c-224", false, 224, Set(AbstractStrin
g["/Users/vaibhav/.julia/packages/Gadfly/09PWZ/src/gadfly.js"]), Set(Tuple{
AbstractString,AbstractString}[("Snap.svg", "Snap"), ("Gadfly", "Gadfly")])
, AbstractString["fig.select(\"#img-40f07e0c-5\")\n   .init_gadfly();", "fi
g.select(\"#img-40f07e0c-44\")\n   .init_gadfly();", "fig.select(\"#img-40f
07e0c-80\")\n   .init_gadfly();", "fig.select(\"#img-40f07e0c-120\")\n   .i
nit_gadfly();", "fig.select(\"#img-40f07e0c-158\")\n   .init_gadfly();", "f
ig.select(\"#img-40f07e0c-196\")\n   .init_gadfly();"], false, :none, (BBox
{l,t,r,b,w,h = 16.284999999999997mm,10.611666666666665mm, 96.6mm,48.4483333
3333332mm, 80.315mm,37.83666666666666mm}, Compose.UnitBox{Float64,Float64,F
loat64,Float64}(-26.20716766035511, 8.472859816766821, 1052.4143353207103,
-8.945719633533642, 0.0mm, 0.0mm, 0.0mm, 0.0mm), Compose.IdentityTransform(
)))
Press ENTER to draw next plot
Compose.SVG(203.2mm, 203.2mm, IOBuffer(data=UInt8[...], readable=true, writ
able=true, seekable=true, append=false, size=213972, maxsize=Inf, ptr=21397
3, mark=-1), nothing, "img-530e50b0", 0, Compose.SVGPropertyFrame[], Dict{T
ype,Union{Nothing, Property}}(Property{LineWidthPrimitive}=>nothing,Propert
y{JSCallPrimitive}=>nothing,Property{FontSizePrimitive}=>nothing,Property{S
VGClassPrimitive}=>nothing,Property{JSIncludePrimitive}=>nothing,Property{F
ontPrimitive}=>nothing,Property{SVGAttributePrimitive}=>nothing,Property{St
rokeDashPrimitive}=>nothing,Property{StrokePrimitive}=>nothing,Property{Fil
lPrimitive}=>nothing...), OrderedCollections.OrderedDict{Compose.ClipPrimitiv
e,String}(ClipPrimitive{Tuple{Length{:mm,Float64},Length{:mm,Float64}}}(Tup
le{Measures.Length{:mm,Float64},Measures.Length{:mm,Float64}}[(117.885mm, 1
46.078mm), (198.2mm, 146.078mm), (198.2mm, 183.915mm), (117.885mm, 183.915m
m)])=>"img-530e50b0-4",ClipPrimitive{Tuple{Length{:mm,Float64},Length{:mm,F
loat64}}}(Tuple{Measures.Length{:mm,Float64},Measures.Length{:mm,Float64}}[
(18.6317mm, 146.078mm), (96.6mm, 146.078mm), (96.6mm, 183.915mm), (18.6317m
m, 183.915mm)])=>"img-530e50b0-45",ClipPrimitive{Tuple{Length{:mm,Float64},
Length{:mm,Float64}}}(Tuple{Measures.Length{:mm,Float64},Measures.Length{:m
m,Float64}}[(117.885mm, 78.345mm), (198.2mm, 78.345mm), (198.2mm, 116.182mm
), (117.885mm, 116.182mm)])=>"img-530e50b0-88",ClipPrimitive{Tuple{Length{:
mm,Float64},Length{:mm,Float64}}}(Tuple{Measures.Length{:mm,Float64},Measur
es.Length{:mm,Float64}}[(18.6317mm, 78.345mm), (96.6mm, 78.345mm), (96.6mm,
 116.182mm), (18.6317mm, 116.182mm)])=>"img-530e50b0-129",ClipPrimitive{Tup
```

le{Length{:mm,Float64},Length{:mm,Float64}}}(Tuple{Measures.Length{:mm,Floa
t64},Measures.Length{:mm,Float64}}[(121.905mm, 10.6117mm), (198.2mm, 10.611
7mm), (198.2mm, 48.4483mm), (121.905mm, 48.4483mm)])=>"img-530e50b0-168",Cl
ipPrimitive{Tuple{Length{:mm,Float64},Length{:mm,Float64}}}(Tuple{Measures.
Length{:mm,Float64},Measures.Length{:mm,Float64}}[(19.0717mm, 10.6117mm), (
96.6mm, 10.6117mm), (96.6mm, 48.4483mm), (19.0717mm, 48.4483mm)])=>"img-530
e50b0-207"), Tuple{Compose.FormPrimitive,String}[], Set(AbstractString[]),
true, false, nothing, true, "img-530e50b0-238", false, 238, Set(AbstractStr
ing["/Users/vaibhav/.julia/packages/Gadfly/09PWZ/src/gadfly.js"]), Set(Tupl
e{AbstractString,AbstractString}[("Snap.svg", "Snap"), ("Gadfly", "Gadfly")
]), AbstractString["fig.select(\"#img-530e50b0-5\")\n   .init_gadfly();", "
fig.select(\"#img-530e50b0-46\")\n   .init_gadfly();", "fig.select(\"#img-5
30e50b0-89\")\n   .init_gadfly();", "fig.select(\"#img-530e50b0-130\")\n
.init_gadfly();", "fig.select(\"#img-530e50b0-169\")\n   .init_gadfly();",
"fig.select(\"#img-530e50b0-208\")\n   .init_gadfly();"], false, :none, (BB
ox{l,t,r,b,w,h = 19.07166666666666mm,10.611666666666665mm, 96.6mm,48.448333
33333332mm, 77.52833333333334mm,37.83666666666666mm}, Compose.UnitBox{Float
64,Float64,Float64,Float64}(-27.200398939184442, 11.477686927396316, 1054.4
007978783688, -27.955373854792633, 0.0mm, 0.0mm, 0.0mm, 0.0mm), Compose.Ide
ntityTransform()))
Press ENTER to draw next plot
Compose.SVG(203.2mm, 203.2mm, IOBuffer(data=UInt8[...], readable=true, writ
able=true, seekable=true, append=false, size=213540, maxsize=Inf, ptr=21354
1, mark=-1), nothing, "img-8ce514dd", 0, Compose.SVGPropertyFrame[], Dict{T
ype,Union{Nothing, Property}}(Property{LineWidthPrimitive}=>nothing,Propert
y{JSCallPrimitive}=>nothing,Property{FontSizePrimitive}=>nothing,Property{S
VGClassPrimitive}=>nothing,Property{JSIncludePrimitive}=>nothing,Property{F
ontPrimitive}=>nothing,Property{SVGAttributePrimitive}=>nothing,Property{St
rokeDashPrimitive}=>nothing,Property{StrokePrimitive}=>nothing,Property{Fil
lPrimitive}=>nothing...), OrderedCollections.OrderedDict{Compose.ClipPrimitiv
e,String}(ClipPrimitive{Tuple{Length{:mm,Float64},Length{:mm,Float64}}}(Tup
le{Measures.Length{:mm,Float64},Measures.Length{:mm,Float64}}[(119.558mm, 1
46.078mm), (198.2mm, 146.078mm), (198.2mm, 183.915mm), (119.558mm, 183.915m
m)])=>"img-8ce514dd-4",ClipPrimitive{Tuple{Length{:mm,Float64},Length{:mm,F
loat64}}}(Tuple{Measures.Length{:mm,Float64},Measures.Length{:mm,Float64}}[
(18.6317mm, 146.078mm), (96.6mm, 146.078mm), (96.6mm, 183.915mm), (18.6317m
m, 183.915mm)])=>"img-8ce514dd-42",ClipPrimitive{Tuple{Length{:mm,Float64},
Length{:mm,Float64}}}(Tuple{Measures.Length{:mm,Float64},Measures.Length{:m
m,Float64}}[(120.232mm, 78.345mm), (198.2mm, 78.345mm), (198.2mm, 116.182mm
), (120.232mm, 116.182mm)])=>"img-8ce514dd-79",ClipPrimitive{Tuple{Length{:
mm,Float64},Length{:mm,Float64}}}(Tuple{Measures.Length{:mm,Float64},Measur
es.Length{:mm,Float64}}[(18.6317mm, 78.345mm), (96.6mm, 78.345mm), (96.6mm,
 116.182mm), (18.6317mm, 116.182mm)])=>"img-8ce514dd-116",ClipPrimitive{Tup
le{Length{:mm,Float64},Length{:mm,Float64}}}(Tuple{Measures.Length{:mm,Floa
t64},Measures.Length{:mm,Float64}}[(120.232mm, 10.6117mm), (198.2mm, 10.611
7mm), (198.2mm, 48.4483mm), (120.232mm, 48.4483mm)])=>"img-8ce514dd-154",Cl
ipPrimitive{Tuple{Length{:mm,Float64},Length{:mm,Float64}}}(Tuple{Measures.
Length{:mm,Float64},Measures.Length{:mm,Float64}}[(18.6317mm, 10.6117mm), (
96.6mm, 10.6117mm), (96.6mm, 48.4483mm), (18.6317mm, 48.4483mm)])=>"img-8ce
514dd-192"), Tuple{Compose.FormPrimitive,String}[], Set(AbstractString[]),
true, false, nothing, true, "img-8ce514dd-219", false, 219, Set(AbstractStr
ing["/Users/vaibhav/.julia/packages/Gadfly/09PWZ/src/gadfly.js"]), Set(Tupl
e{AbstractString,AbstractString}[("Snap.svg", "Snap"), ("Gadfly", "Gadfly")
]), AbstractString["fig.select(\"#img-8ce514dd-5\")\n   .init_gadfly();", "
fig.select(\"#img-8ce514dd-43\")\n   .init_gadfly();", "fig.select(\"#img-8
ce514dd-80\")\n   .init_gadfly();", "fig.select(\"#img-8ce514dd-117\")\n
.init_gadfly();", "fig.select(\"#img-8ce514dd-155\")\n   .init_gadfly();",
"fig.select(\"#img-8ce514dd-193\")\n   .init_gadfly();"], false, :none, (BB
ox{l,t,r,b,w,h = 18.63166666666666mm,10.611666666666665mm, 96.6mm,48.448333

33333332mm, 77.96833333333333mm,37.83666666666666mm}, Compose.UnitBox{Float
64,Float64,Float64,Float64}(-27.038597598071252, 1.588661215643779, 1054.07
71951961426, -1.6773224312875579, 0.0mm, 0.0mm, 0.0mm, 0.0mm), Compose.Iden
tityTransform()))
Press ENTER to draw next plot
Compose.SVG(203.2mm, 203.2mm, IOBuffer(data=UInt8[...], readable=true, writ
able=true, seekable=true, append=false, size=215376, maxsize=Inf, ptr=21537
7, mark=-1), nothing, "img-d5fd3818", 0, Compose.SVGPropertyFrame[], Dict{T
ype,Union{Nothing, Property}}(Property{LineWidthPrimitive}=>nothing,Propert
y{JSCallPrimitive}=>nothing,Property{FontSizePrimitive}=>nothing,Property{S
VGClassPrimitive}=>nothing,Property{JSIncludePrimitive}=>nothing,Property{F
ontPrimitive}=>nothing,Property{SVGAttributePrimitive}=>nothing,Property{St
rokeDashPrimitive}=>nothing,Property{StrokePrimitive}=>nothing,Property{Fil
lPrimitive}=>nothing...), OrderedCollections.OrderedDict{Compose.ClipPrimitiv
e,String}(ClipPrimitive{Tuple{Length{:mm,Float64},Length{:mm,Float64}}}(Tup
le{Measures.Length{:mm,Float64},Measures.Length{:mm,Float64}}[(120.232mm, 1
46.078mm), (198.2mm, 146.078mm), (198.2mm, 183.915mm), (120.232mm, 183.915m
m)])=>"img-d5fd3818-4",ClipPrimitive{Tuple{Length{:mm,Float64},Length{:mm,F
loat64}}}(Tuple{Measures.Length{:mm,Float64},Measures.Length{:mm,Float64}}[
(18.6317mm, 146.078mm), (96.6mm, 146.078mm), (96.6mm, 183.915mm), (18.6317m
m, 183.915mm)])=>"img-d5fd3818-41",ClipPrimitive{Tuple{Length{:mm,Float64},
Length{:mm,Float64}}}(Tuple{Measures.Length{:mm,Float64},Measures.Length{:m
m,Float64}}[(120.232mm, 78.345mm), (198.2mm, 78.345mm), (198.2mm, 116.182mm
), (120.232mm, 116.182mm)])=>"img-d5fd3818-79",ClipPrimitive{Tuple{Length{:
mm,Float64},Length{:mm,Float64}}}(Tuple{Measures.Length{:mm,Float64},Measur
es.Length{:mm,Float64}}[(18.6317mm, 78.345mm), (96.6mm, 78.345mm), (96.6mm,
 116.182mm), (18.6317mm, 116.182mm)])=>"img-d5fd3818-117",ClipPrimitive{Tup
le{Length{:mm,Float64},Length{:mm,Float64}}}(Tuple{Measures.Length{:mm,Floa
t64},Measures.Length{:mm,Float64}}[(117.885mm, 10.6117mm), (198.2mm, 10.611
7mm), (198.2mm, 48.4483mm), (117.885mm, 48.4483mm)])=>"img-d5fd3818-156",Cl
ipPrimitive{Tuple{Length{:mm,Float64},Length{:mm,Float64}}}(Tuple{Measures.
Length{:mm,Float64},Measures.Length{:mm,Float64}}[(18.6317mm, 10.6117mm), (
96.6mm, 10.6117mm), (96.6mm, 48.4483mm), (18.6317mm, 48.4483mm)])=>"img-d5f
d3818-201"), Tuple{Compose.FormPrimitive,String}[], Set(AbstractString[]),
true, false, nothing, true, "img-d5fd3818-232", false, 232, Set(AbstractStr
ing["/Users/vaibhav/.julia/packages/Gadfly/09PWZ/src/gadfly.js"]), Set(Tupl
e{AbstractString,AbstractString}[("Snap.svg", "Snap"), ("Gadfly", "Gadfly")
]), AbstractString["fig.select(\"#img-d5fd3818-5\")\n   .init_gadfly();", "
fig.select(\"#img-d5fd3818-42\")\n   .init_gadfly();", "fig.select(\"#img-d
5fd3818-80\")\n   .init_gadfly();", "fig.select(\"#img-d5fd3818-118\")\n
.init_gadfly();", "fig.select(\"#img-d5fd3818-157\")\n   .init_gadfly();",
"fig.select(\"#img-d5fd3818-202\")\n   .init_gadfly();"], false, :none, (BB
ox{l,t,r,b,w,h = 18.63166666666666mm,10.611666666666665mm, 96.6mm,48.448333
33333332mm, 77.96833333333333mm,37.83666666666666mm}, Compose.UnitBox{Float
64,Float64,Float64,Float64}(-27.038597598071252, 1.0591074770958526, 1054.0
771951961426, -1.1182149541917052, 0.0mm, 0.0mm, 0.0mm, 0.0mm), Compose.Ide
ntityTransform()))
Press ENTER to draw next plot
Compose.SVG(203.2mm, 203.2mm, IOBuffer(data=UInt8[...], readable=true, writ
able=true, seekable=true, append=false, size=143481, maxsize=Inf, ptr=14348
2, mark=-1), nothing, "img-0158d053", 0, Compose.SVGPropertyFrame[], Dict{T
ype,Union{Nothing, Property}}(Property{LineWidthPrimitive}=>nothing,Propert
y{JSCallPrimitive}=>nothing,Property{FontSizePrimitive}=>nothing,Property{S
VGClassPrimitive}=>nothing,Property{JSIncludePrimitive}=>nothing,Property{F
ontPrimitive}=>nothing,Property{SVGAttributePrimitive}=>nothing,Property{St
rokeDashPrimitive}=>nothing,Property{StrokePrimitive}=>nothing,Property{Fil
lPrimitive}=>nothing...), OrderedCollections.OrderedDict{Compose.ClipPrimitiv
e,String}(ClipPrimitive{Tuple{Length{:mm,Float64},Length{:mm,Float64}}}(Tup
le{Measures.Length{:mm,Float64},Measures.Length{:mm,Float64}}[(117.885mm, 7

```
8.345mm), (198.2mm, 78.345mm), (198.2mm, 116.182mm), (117.885mm, 116.182mm)
])=>"img-0158d053-4",ClipPrimitive{Tuple{Length{:mm,Float64},Length{:mm,Flo
at64}}}(Tuple{Measures.Length{:mm,Float64},Measures.Length{:mm,Float64}}[(1
8.6317mm, 78.345mm), (96.6mm, 78.345mm), (96.6mm, 116.182mm), (18.6317mm, 1
16.182mm)])=>"img-0158d053-49",ClipPrimitive{Tuple{Length{:mm,Float64},Leng
th{:mm,Float64}}}(Tuple{Measures.Length{:mm,Float64},Measures.Length{:mm,Fl
oat64}}[(119.558mm, 10.6117mm), (198.2mm, 10.6117mm), (198.2mm, 48.4483mm),
 (119.558mm, 48.4483mm)])=>"img-0158d053-91",ClipPrimitive{Tuple{Length{:mm
,Float64},Length{:mm,Float64}}}(Tuple{Measures.Length{:mm,Float64},Measures
.Length{:mm,Float64}}[(18.6317mm, 10.6117mm), (96.6mm, 10.6117mm), (96.6mm,
 48.4483mm), (18.6317mm, 48.4483mm)])=>"img-0158d053-129"), Tuple{Compose.F
ormPrimitive,String}[], Set(AbstractString[]), true, false, nothing, true,
"img-0158d053-156", false, 156, Set(AbstractString["/Users/vaibhav/.julia/p
ackages/Gadfly/09PWZ/src/gadfly.js"]), Set(Tuple{AbstractString,AbstractStr
ing}[("Snap.svg", "Snap"), ("Gadfly", "Gadfly")]), AbstractString["fig.sele
ct(\"#img-0158d053-5\")\n    .init_gadfly();", "fig.select(\"#img-0158d053-5
0\")\n    .init_gadfly();", "fig.select(\"#img-0158d053-92\")\n    .init_gadf
ly();", "fig.select(\"#img-0158d053-130\")\n    .init_gadfly();"], false, :n
one, (BBox{l,t,r,b,w,h = 18.63166666666666mm,10.611666666666665mm, 96.6mm,4
8.44833333333332mm, 77.96833333333333mm,37.83666666666666mm}, Compose.UnitB
ox{Float64,Float64,Float64,Float64}(-27.038597598071252, 0.3177322431287558
, 1054.0771951961426, -0.33546448625751163, 0.0mm, 0.0mm, 0.0mm, 0.0mm), Co
mpose.IdentityTransform()))
```

0.2 Turing.jl backend

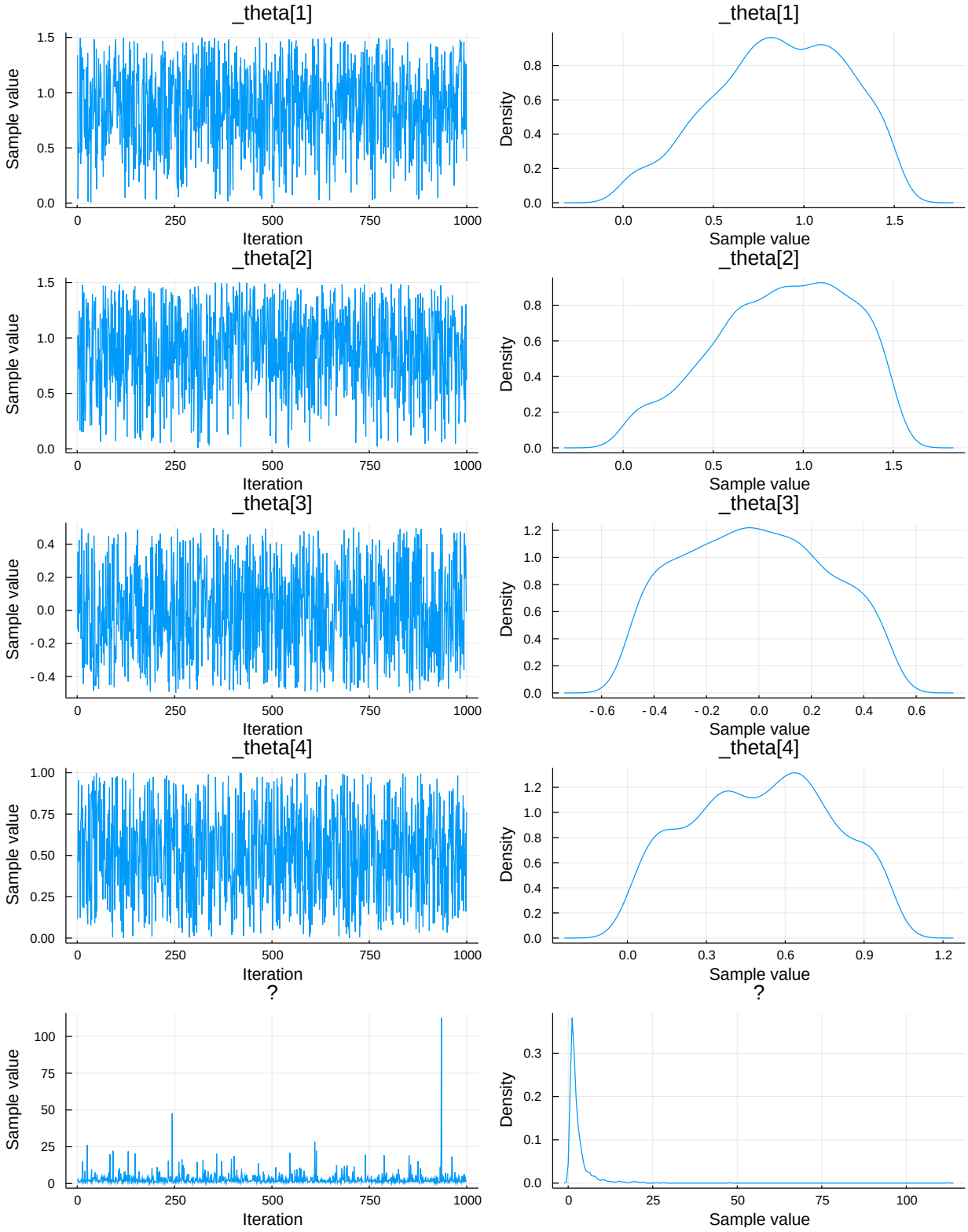
```
@time bayesian_result_turing =
    turing_inference(prob_ode_fitzhughnagumo,Tsit5(),t,data,priors)
```

```
0.549080 seconds (1.51 M allocations: 107.968 MiB, 12.47% gc time)
Object of type Chains, with data of type 1000×6×1 Array{Union{Missing, Floa
t64},3}

Log evidence      = -15.458125651168196
Iterations        = 1:1000
Thinning interval = 1
Chains            = 1
Samples per chain = 1000
internals         = lp
parameters        = _theta[2], σ, _theta[1], _theta[4], _theta[3]

parameters
            Mean    SD   Naive SE  MCSE     ESS
_theta[1]  0.8649 0.3625   0.0115 0.0116  977.7876
_theta[2]  0.8830 0.3703   0.0117 0.0175  445.5005
_theta[3] -0.0165 0.2681   0.0085 0.0061 1000.0000
_theta[4]  0.5079 0.2658   0.0084 0.0054 1000.0000
        σ  3.0367 5.0253   0.1589 0.1112 1000.0000
```

```
plot_chain(bayesian_result_turing)
```



1 Conclusion

In the FitzHugh-Nagumo model the parameters to be estimated were `[0.7,0.8,0.08,0.5]`. We use default number of samples and warmup to get a better estimate of the default performance of the samplers.

Individually, Stan.jl backend takes 1.7 minutes for warmup and 1.6 seconds for sampling, giving `[0.98,0.83,0.079,0.56]`. Higher accuracy can be obtained with tighter priors,

increase in warmup samples and adjusting the tolerance values.

Turing.jl took just over 0.58 seconds and gave `[0.88,0.88,0.017,0.49]` as the result. The the trace plots indicate some non-convergance, this can be handled by increasing the sampling size for longer iterations.

Overall we observe some non-convergance in both the backends and to avoid it longer iterations would be required at the cost of effiency the choice of which depends on the user.

```
using DiffEqBenchmarks
DiffEqBenchmarks.bench_footer(WEAVE_ARGS[:folder],WEAVE_ARGS[:file])
```

1.1 Appendix

These benchmarks are a part of the DiffEqBenchmarks.jl repository, found at: https://github.com/JuliaDi

To locally run this tutorial, do the following commands:

```
using DiffEqBenchmarks
DiffEqBenchmarks.weave_file("ParameterEstimation/","DiffEqBayesFitzHughNagumo.jmd")
```

Computer Information:

```
Julia Version 1.1.0
Commit 80516ca202 (2019-01-21 21:24 UTC)
Platform Info:
  OS: macOS (x86_64-apple-darwin14.5.0)
  CPU: Intel(R) Core(TM) i7-6700HQ CPU @ 2.60GHz
  WORD_SIZE: 64
  LIBM: libopenlibm
  LLVM: libLLVM-6.0.1 (ORCJIT, skylake)
```

Package Information:

```
Status: `/Users/vaibhav/.julia/environments/v1.1/Project.toml`
[f5f396d3-230c-5e07-80e6-9fadf06146cc] ApproxBayes 0.3.0
[6e4b80f9-dd63-53aa-95a3-0cdb28fa8baf] BenchmarkTools 0.4.2
[159f3aea-2a34-519c-b102-8c37f9878175] Cairo 0.5.6
[ebbdde9d-f333-5424-9be2-dbf1e9acfb5e] DiffEqBayes 1.1.0+
[31c91b34-3c75-11e9-0341-95557aab0344] DiffEqBenchmarks 0.1.0
[0c46a032-eb83-5123-abaf-570d42b7fbaa] DifferentialEquations 6.3.0
[b4f34e82-e78d-54a5-968a-f98e89d6e8f7] Distances 0.8.0
[31c24e10-a181-5473-b8eb-7969acd0382f] Distributions 0.16.4
[f6369f11-7733-5829-9624-2563aa707210] ForwardDiff 0.10.3
[c91e804a-d5a3-530f-b6f0-dfbca275c004] Gadfly 1.0.1
[6fdf6af0-433a-55f7-b3ed-c6c6e0b8df7c] LogDensityProblems 0.6.0
[5424a776-8be3-5c5b-a13f-3551f69ba0e6] Mamba 0.12.0
[eff96d63-e80a-5855-80a2-b1b0885c5ab7] Measurements 2.0.0
```

```
[1dea7af3-3e70-54e6-95c3-0bf5283fa5ed] OrdinaryDiffEq 5.3.0
[65888b18-ceab-5e60-b2b9-181511a3b968] ParameterizedFunctions 4.1.1
[91a5bcdd-55d7-5caf-9e0b-520d859cae80] Plots 0.23.1
[71ad9d73-34c4-5ce9-b7b1-f7bd31ac38ba] PuMaS 0.0.0
[731186ca-8d62-57ce-b412-fbd966d074cd] RecursiveArrayTools 0.20.0
[f3b207a7-027a-5e70-b257-86293d7955fd] StatsPlots 0.10.2
[09ab397b-f2b6-538f-b94a-2f83cf4a842a] StructArrays 0.2.1
[fce5fe82-541a-59a6-adf8-730c64b5f9a0] Turing 0.6.11
[44d3d7a6-8a23-5bf8-98c5-b353f8df5ec9] Weave 0.8.1
```