

Computation of π

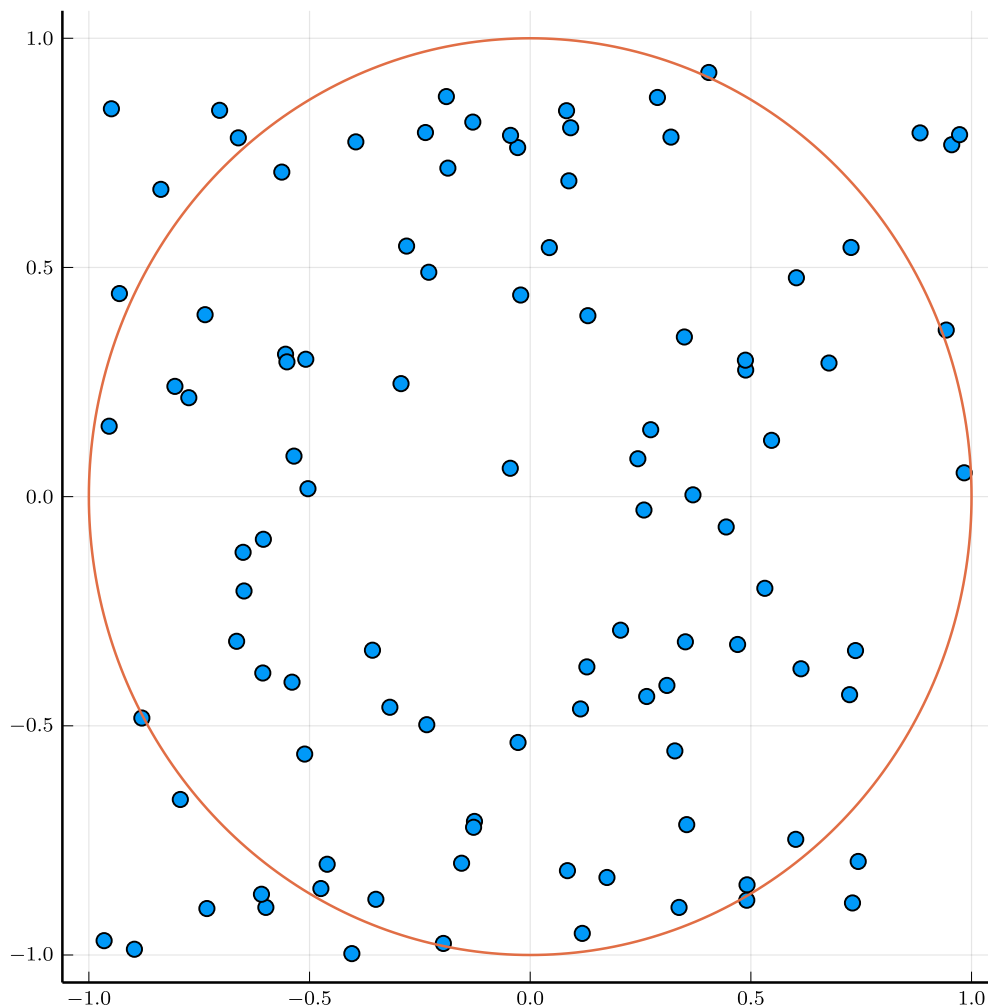
Alberto ramos <alberto.ramos@maths.tcd.ie>

Wed Apr 6 18:37:02 2022

Contents

Code (Julia)

```
1 using Random, Statistics, BenchmarkTools, Plots
2 pgfplotsx();
3
4 isincircle(x) = (x[1]^2 + x[2]^2) < 1.0
5
6 samp = rand(100, 2)
7 samp .*= 2.0 .* samp .- 1.0
8 pl = plot(samp[:,1], samp[:,2], label="", seriestype=:scatter, size=(600,600))
9 pl = plot!([cos(2*pi*t/360) for t in 0:360], [sin(2*pi*t/360) for t in 0:360], label="")
10 display(pl)
```



```

1 function getpi(nb)
2
3     eta = Vector{Float64}(undef, 2)
4     avg = 0.0
5     M2 = 0.0
6     for i in 1:nb
7         rand!(eta)
8         eta .= 2 .* eta .- 1.0
9         inside = isincircle(eta)
10        A = inside ? 4.0 - avg : -avg
11        avg = avg + A/i
12        M2 = inside ? M2 + A*(4.0-avg) : M2 - A*avg
13    end
14
15    return avg, sqrt(M2)/nb
16 end
17
18 nb = 10000
19
20 Random.seed!(182)
21 piv = Vector{Float64}()
22 # Repeat estimation many times,
23 # check histogram of obtained values of pi
24 for i in 1:10000
25     piest, err = getpi(nb)
26     push!(piv, piest)
27 end
28 # print one example
29 println("Estiate of pi with $nb samples: ", piest, " +/- ", err)
30 println(" # Deviation: ", (pi-piest)/err, " sigmas")
31
32 h = 0.0004
33 pl = histogram(piv, bins = :auto, normalize=:pdf, label="")
34 pl = plot!([h*i + 3.1415926 for i in -100:100],
35           [exp(-(h*i)^2/(2*err^2))/(err*sqrt(2*pi)) for i in -100:100],
36           label="Not a fit!", linewidth=3)
37 display(pl)

```

```

Estiate of pi with 10000 samples: 3.1627999999999896 +/- 0.01627235741987007
# Deviation: -1.303274372790039 sigmas

```

