Title: Numerical Methods Lecture - 230201

Speakers: Erik Schnetter

Collection: Numerical Methods (2022/2023)

Date: February 01, 2023 - 3:30 PM

URL: https://pirsa.org/23020003

Safari File Edit View History Bookmarks Develop Window Help

symmetry.pi.local

JupyterLab

File Edit View Run Kernel Git Tabs Settings Help

Mem:1.05 GB

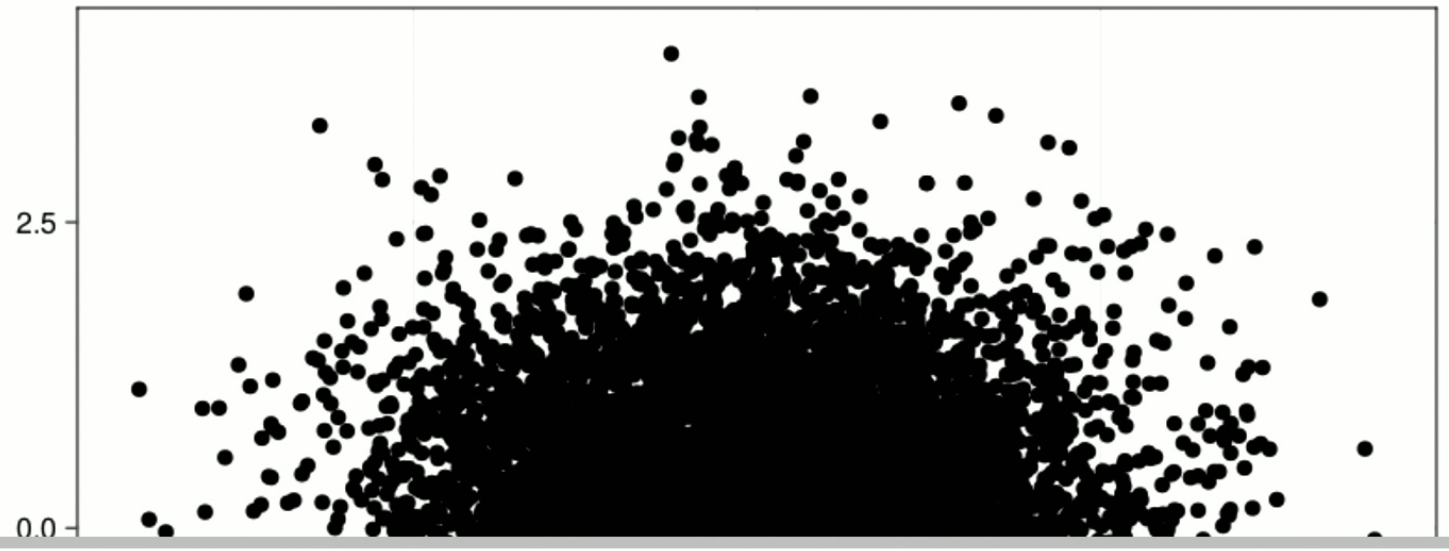
Untitled10.ipynb

Code git Julia 1.8.5

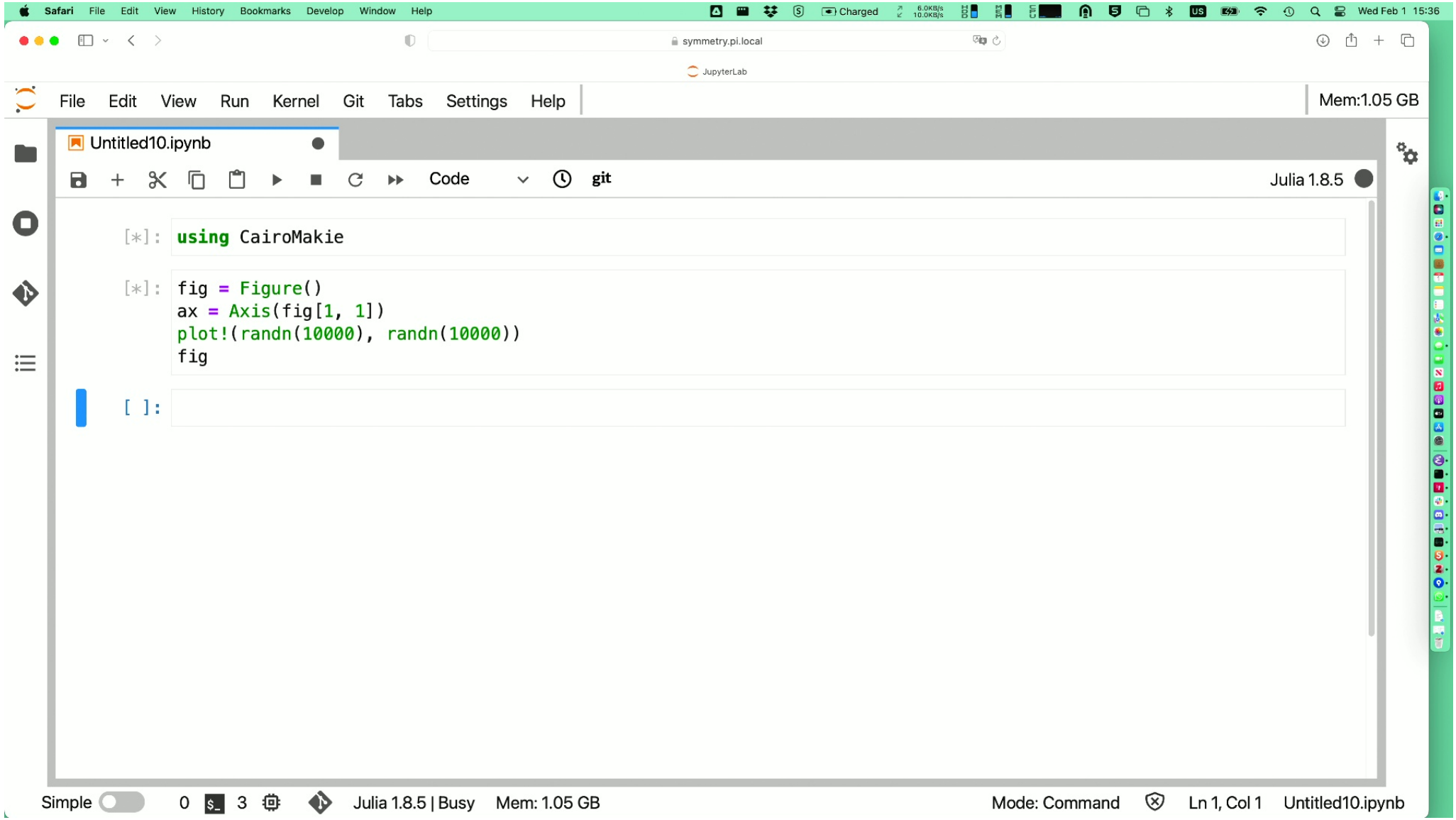
[1]:
```
using CairoMakie
```

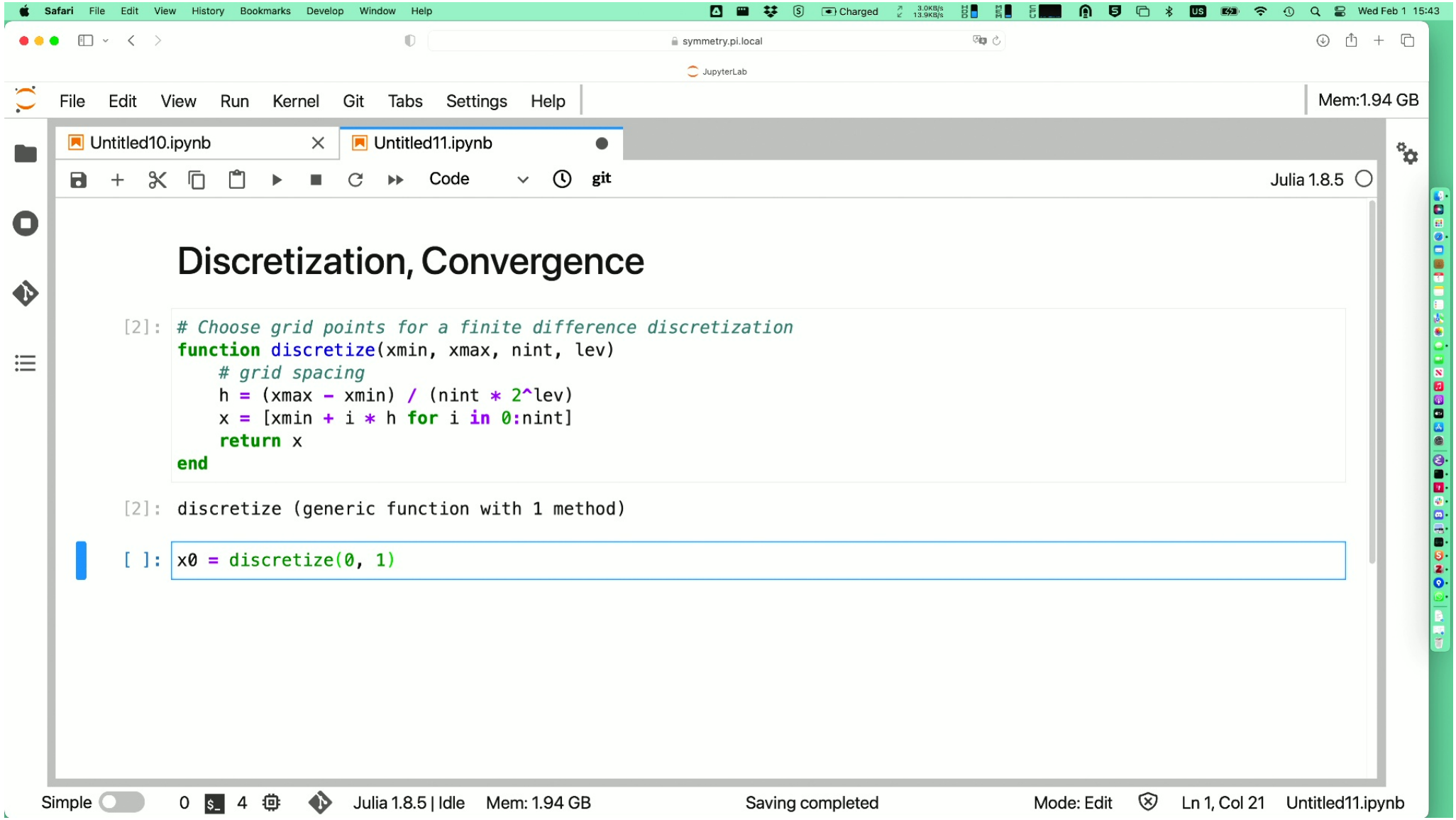
[5]:
```
fig = Figure()
ax = Axis(fig[1, 1])
plot!(randn(10000), randn(10000))
fig
```

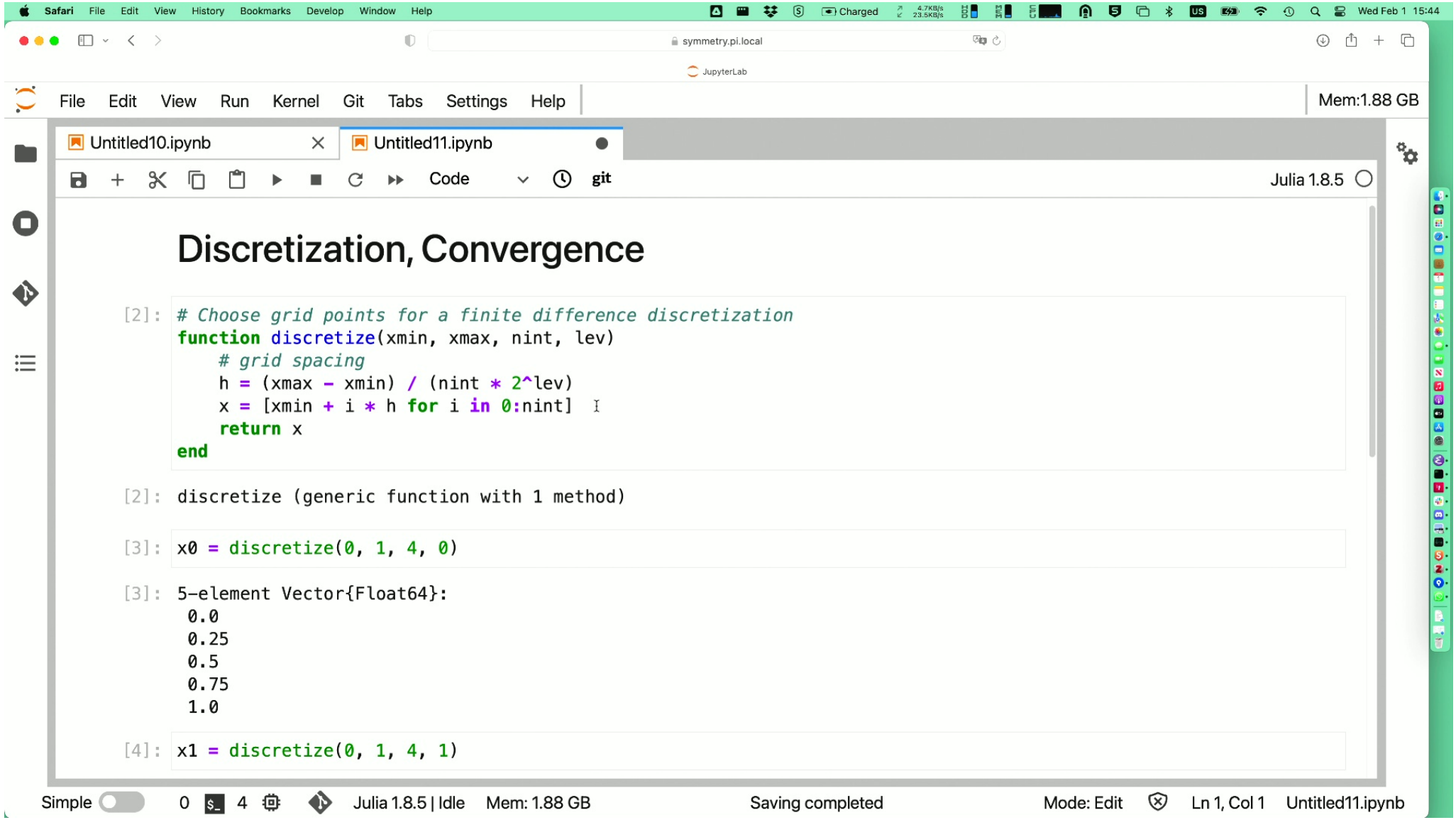
[5]:

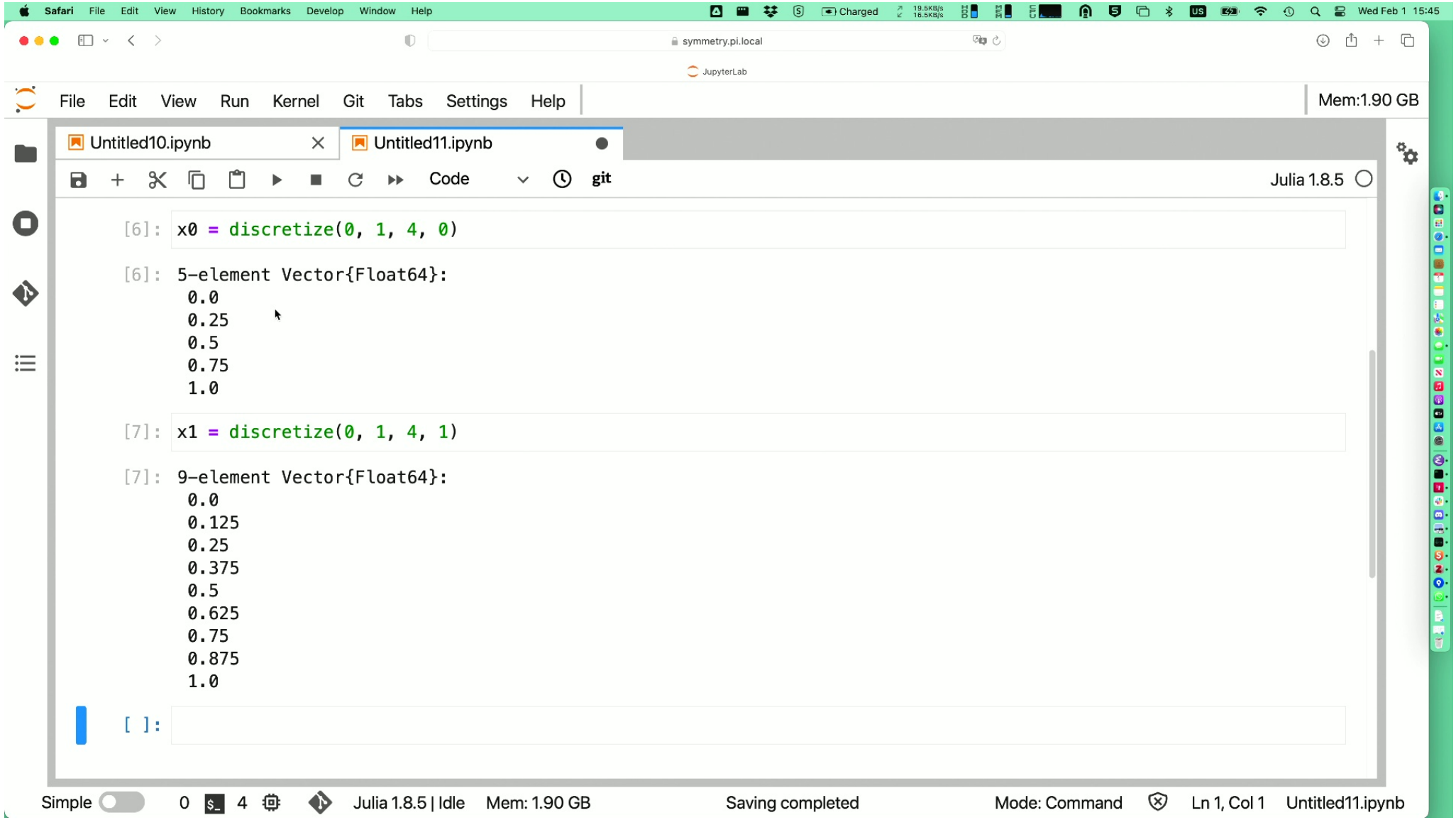


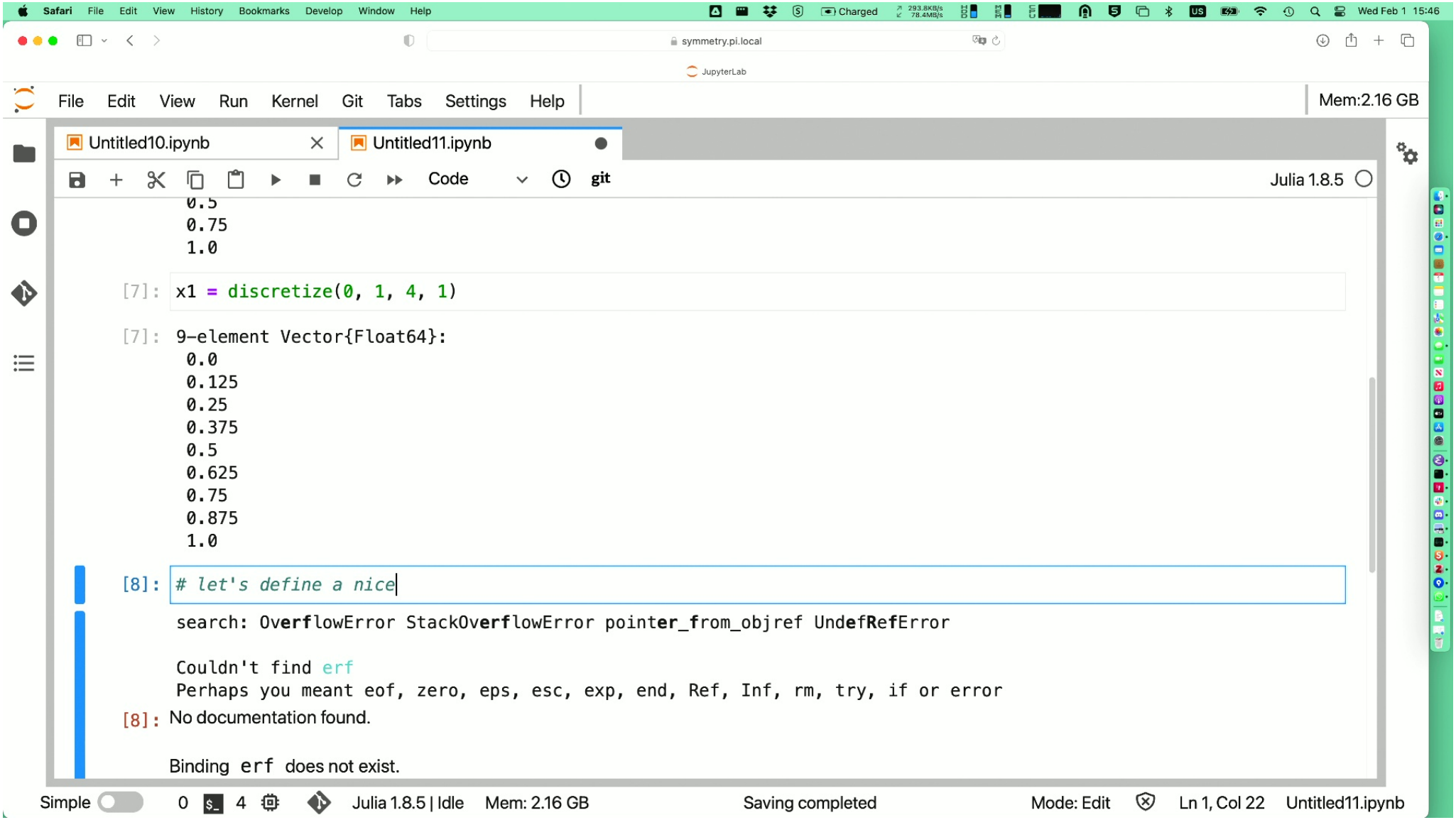
Simple 0 3 Julia 1.8.5 | Idle Mem: 1.05 GB Mode: Command Ln 1, Col 1 Untitled10.ipynb

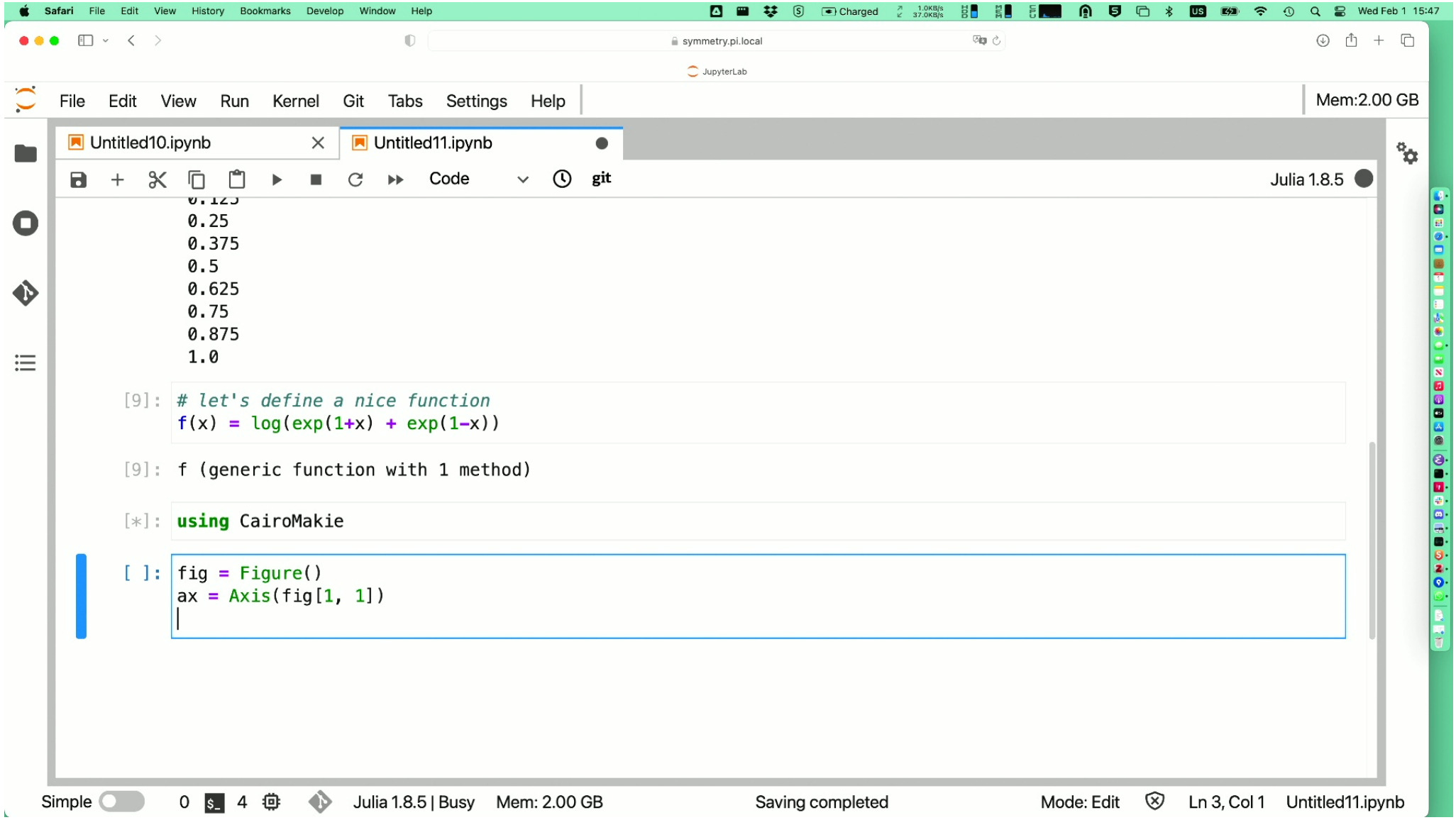

Safari File Edit View History Bookmarks Develop Window Help
Wed Feb 1 15:36
symmetry.pi.local
JupyterLab
File Edit View Run Kernel Git Tabs Settings Help
Mem:1.05 GB
Untitled10.ipynb
Code
git
Julia 1.8.5
[*]: using CairoMakie
[*]: fig = Figure()
ax = Axis(fig[1, 1])
plot!(randn(10000), randn(10000))
fig
[ ]:
Simple
0
3
Julia 1.8.5 | Busy
Mem: 1.05 GB
Mode: Command
Ln 1, Col 1
Untitled10.ipynb

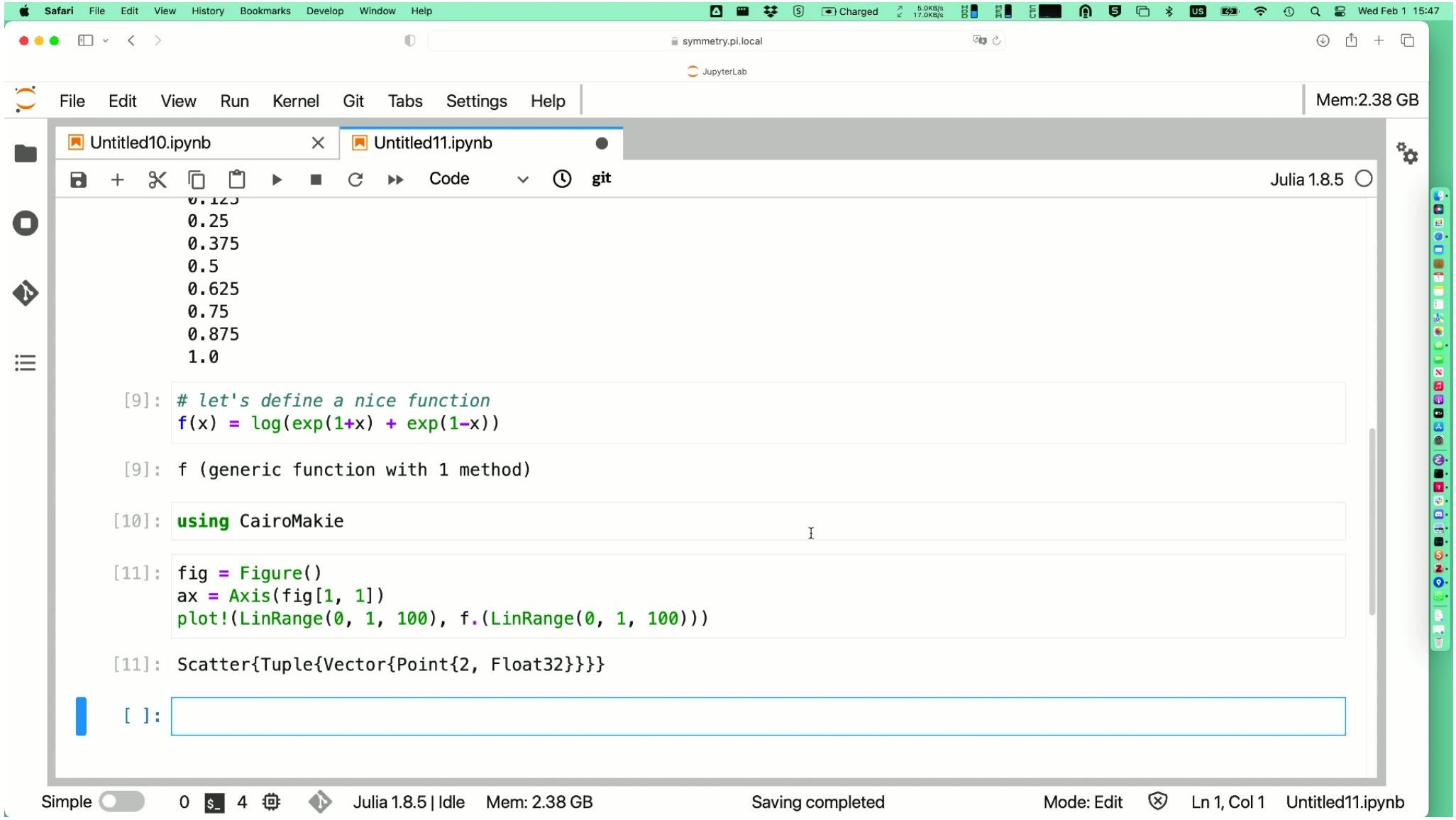

Safari File Edit View History Bookmarks Develop Window Help
Wed Feb 1 15:43
symmetry.pi.local
JupyterLab
File Edit View Run Kernel Git Tabs Settings Help
Mem:1.94 GB
Untitled10.ipynb
Untitled11.ipynb
Code
git
Julia 1.8.5
Discretization, Convergence
[2]:
Choose grid points for a finite difference discretization
function discretize(xmin, xmax, nint, lev)
# grid spacing
h = (xmax - xmin) / (nint * 2^lev)
x = [xmin + i * h for i in 0:nint]
return x
end
[2]:
discretize (generic function with 1 method)
[ ]:
x0 = discretize(0, 1)
Simple
0
4
Julia 1.8.5 | Idle
Mem: 1.94 GB
Saving completed
Mode: Edit
Ln 1, Col 21
Untitled11.ipynb


Safari File Edit View History Bookmarks Develop Window Help
Wed Feb 1 15:44
symmetry.pi.local
JupyterLab
File Edit View Run Kernel Git Tabs Settings Help
Mem:1.88 GB
Untitled10.ipynb
Untitled11.ipynb
Code
git
Julia 1.8.5
Discretization, Convergence
[2]:
Choose grid points for a finite difference discretization
function discretize(xmin, xmax, nint, lev)
# grid spacing
h = (xmax - xmin) / (nint * 2^lev)
x = [xmin + i * h for i in 0:nint]
return x
end
[2]:
discretize (generic function with 1 method)
[3]:
x0 = discretize(0, 1, 4, 0)
[3]:
5-element Vector{Float64}:
0.0
0.25
0.5
0.75
1.0
[4]:
x1 = discretize(0, 1, 4, 1)
Simple
0
4
Julia 1.8.5 | Idle
Mem: 1.88 GB
Saving completed
Mode: Edit
Ln 1, Col 1
Untitled11.ipynb


Safari File Edit View History Bookmarks Develop Window Help
Wed Feb 1 15:45
symmetry.pi.local
JupyterLab
File Edit View Run Kernel Git Tabs Settings Help
Mem:1.90 GB
Untitled10.ipynb
Untitled11.ipynb
Code
git
Julia 1.8.5
[6]: x0 = discretize(0, 1, 4, 0)
[6]: 5-element Vector{Float64}:
0.0
0.25
0.5
0.75
1.0
[7]: x1 = discretize(0, 1, 4, 1)
[7]: 9-element Vector{Float64}:
0.0
0.125
0.25
0.375
0.5
0.625
0.75
0.875
1.0
[ ]:
Simple
0
4
Julia 1.8.5 | Idle
Mem: 1.90 GB
Saving completed
Mode: Command
Ln 1, Col 1
Untitled11.ipynb


Safari File Edit View History Bookmarks Develop Window Help
Wed Feb 1 15:46
symmetry.pi.local
JupyterLab
File Edit View Run Kernel Git Tabs Settings Help
Mem:2.16 GB
Untitled10.ipynb
Untitled11.ipynb
Code
git
Julia 1.8.5
0.5
0.75
1.0
[7]: x1 = discretize(0, 1, 4, 1)
[7]: 9-element Vector{Float64}:
0.0
0.125
0.25
0.375
0.5
0.625
0.75
0.875
1.0
[8]: # let's define a nice
search: OverflowError StackOverflowError pointer_from_objref UndefRefError
Couldn't find erf
Perhaps you meant eof, zero, eps, esc, exp, end, Ref, Inf, rm, try, if or error
[8]: No documentation found.
Binding erf does not exist.
Simple
0
4
Julia 1.8.5 | Idle
Mem: 2.16 GB
Saving completed
Mode: Edit
Ln 1, Col 22
Untitled11.ipynb


Safari File Edit View History Bookmarks Develop Window Help
Wed Feb 1 15:47
symmetry.pi.local
JupyterLab
File Edit View Run Kernel Git Tabs Settings Help
Mem:2.00 GB
Untitled10.ipynb
Untitled11.ipynb
Code
git
Julia 1.8.5
0.25
0.375
0.5
0.625
0.75
0.875
1.0
[9]: # let's define a nice function
f(x) = log(exp(1+x) + exp(1-x))
[9]: f (generic function with 1 method)
[*]: using CairoMakie
[ ]: fig = Figure()
ax = Axis(fig[1, 1])
Simple
0
4
Julia 1.8.5 | Busy
Mem: 2.00 GB
Saving completed
Mode: Edit
Ln 3, Col 1
Untitled11.ipynb


Safari File Edit View History Bookmarks Develop Window Help
Wed Feb 1 15:47
symmetry.pi.local
JupyterLab
File Edit View Run Kernel Git Tabs Settings Help
Mem:2.38 GB
Untitled10.ipynb
Untitled11.ipynb
Code
git
Julia 1.8.5
0.25
0.375
0.5
0.625
0.75
0.875
1.0
[9]: # let's define a nice function
f(x) = log(exp(1+x) + exp(1-x))
[9]: f (generic function with 1 method)
[10]: using CairoMakie
[11]: fig = Figure()
ax = Axis(fig[1, 1])
plot!(LinRange(0, 1, 100), f.(LinRange(0, 1, 100)))
[11]: Scatter{Tuple{Vector{Point{2, Float32}}}}
[ ]:
Simple
0
4
Julia 1.8.5 | Idle
Mem: 2.38 GB
Saving completed
Mode: Edit
Ln 1, Col 1
Untitled11.ipynb

Safari File Edit View History Bookmarks Develop Window Help — Wed Feb 1 15:48

symmetry.pi.local

JupyterLab

File Edit View Run Kernel Git Tabs Settings Help — Mem:2.27 GB

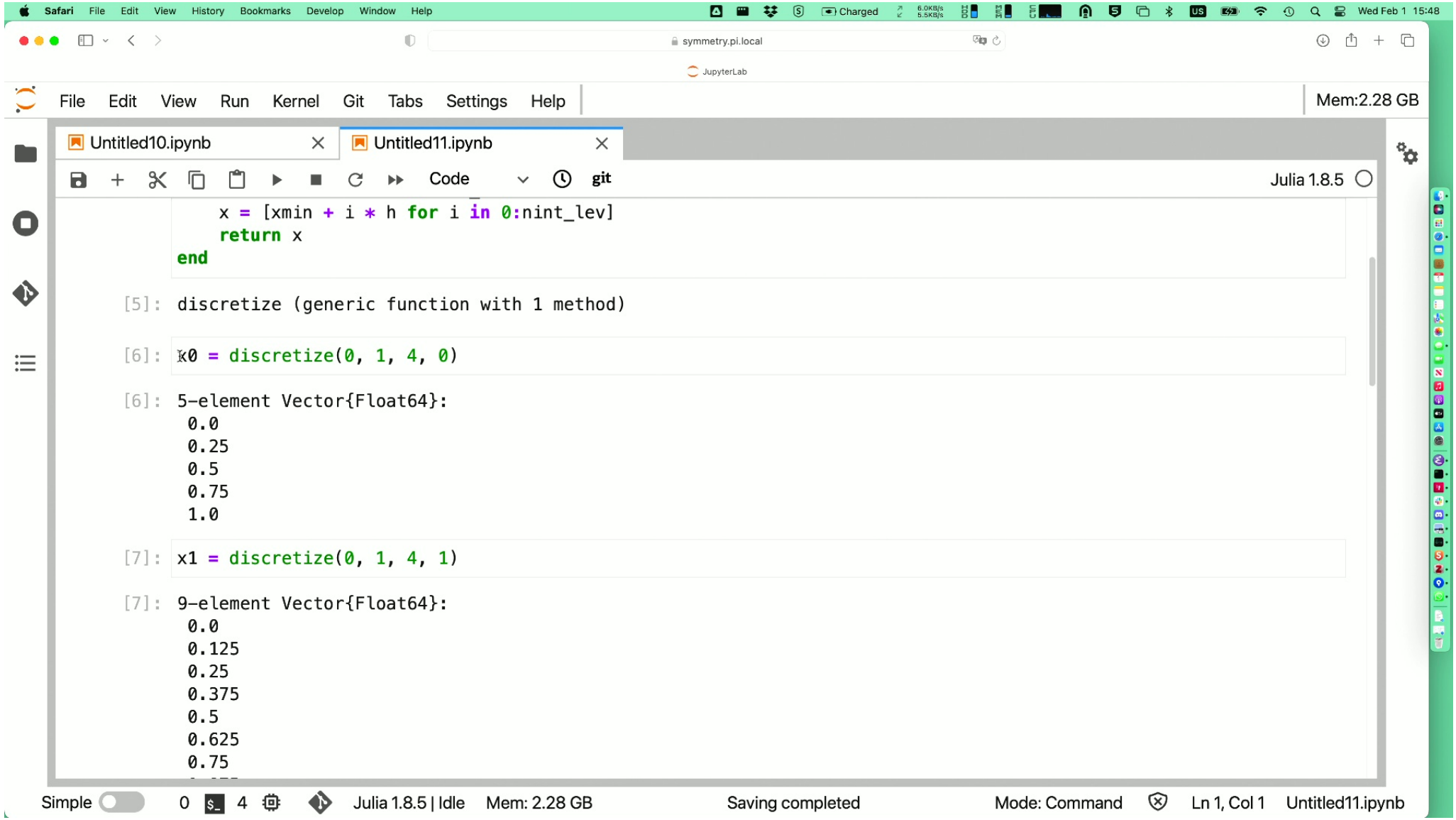
Untitled10.ipynb | Untitled11.ipynb

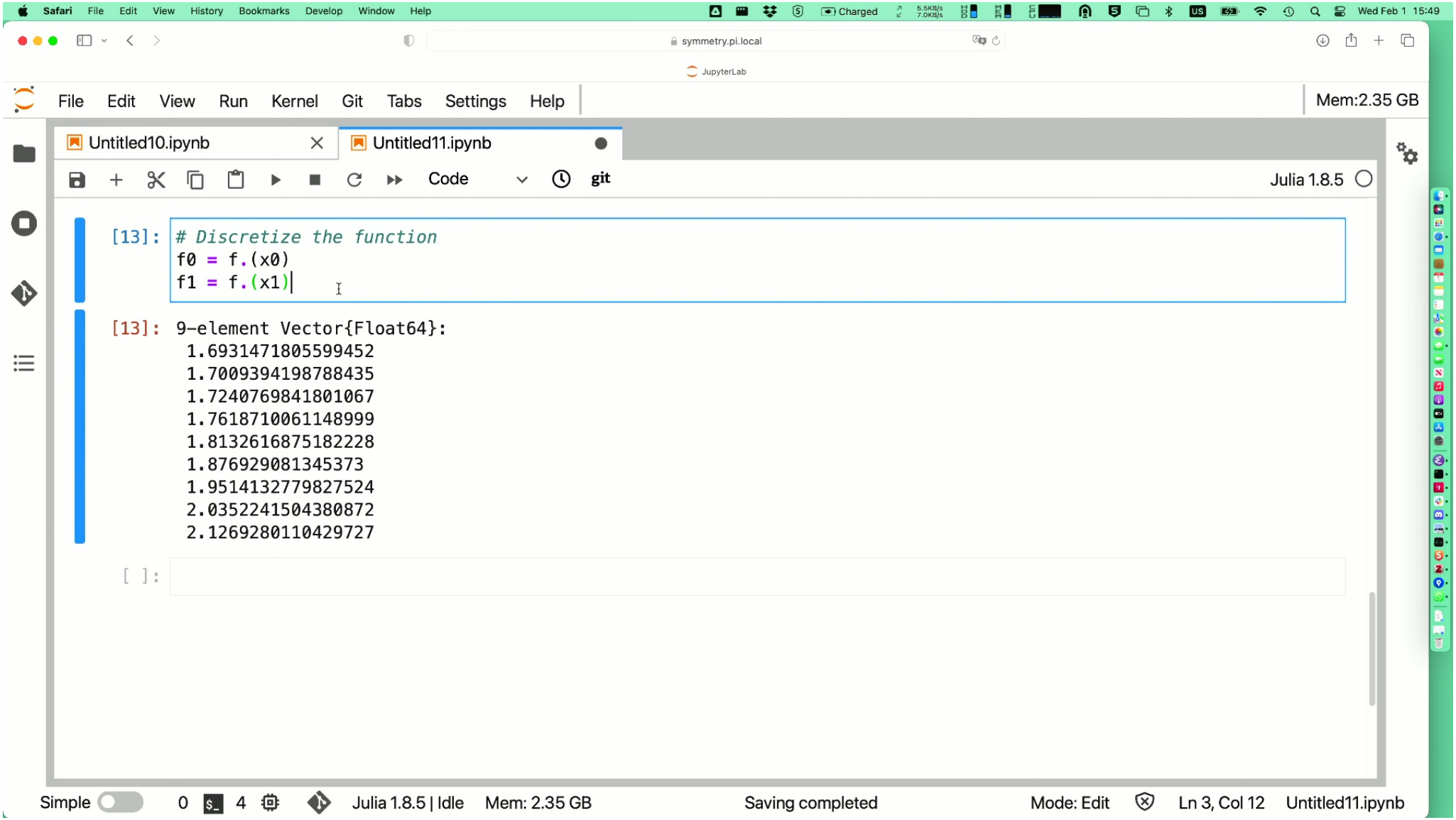
Code — git — Julia 1.8.5

```
[9]: f (generic function with 1 method)

[10]: using CairoMakie

[12]: fig = Figure()
      ax = Axis(fig[1, 1])
      plot!(LinRange(0, 1, 100), f.(LinRange(0, 1, 100)))
      fig
```

[12]:

Simple 0 4 Julia 1.8.5 | Idle Mem: 2.27 GB Saving completed Mode: Command Ln 1, Col 1 Untitled11.ipynb


Safari File Edit View History Bookmarks Develop Window Help
Wed Feb 1 15:48
symmetry.pi.local
JupyterLab
File Edit View Run Kernel Git Tabs Settings Help
Mem:2.28 GB
Untitled10.ipynb
Untitled11.ipynb
Code
git
Julia 1.8.5
x = [xmin + i * h for i in 0:nint_lev]
return x
end
[5]: discretize (generic function with 1 method)
[6]: x0 = discretize(0, 1, 4, 0)
[6]: 5-element Vector{Float64}:
0.0
0.25
0.5
0.75
1.0
[7]: x1 = discretize(0, 1, 4, 1)
[7]: 9-element Vector{Float64}:
0.0
0.125
0.25
0.375
0.5
0.625
0.75
Simple
0
4
Julia 1.8.5 | Idle
Mem: 2.28 GB
Saving completed
Mode: Command
Ln 1, Col 1
Untitled11.ipynb


Safari File Edit View History Bookmarks Develop Window Help
Wed Feb 1 15:49
symmetry.pi.local
JupyterLab
File Edit View Run Kernel Git Tabs Settings Help
Mem:2.35 GB
Untitled10.ipynb
Untitled11.ipynb
Code
git
Julia 1.8.5
[13]:
# Discretize the function
f0 = f.(x0)
f1 = f.(x1)
[13]:
9-element Vector{Float64}:
1.6931471805599452
1.7009394198788435
1.7240769841801067
1.7618710061148999
1.8132616875182228
1.876929081345373
1.9514132779827524
2.0352241504380872
2.1269280110429727
[ ]:
Simple
0
4
Julia 1.8.5 | Idle
Mem: 2.35 GB
Saving completed
Mode: Edit
Ln 3, Col 12
Untitled11.ipynb

Safari File Edit View History Bookmarks Develop Window Help — Wed Feb 1 15:50

symmetry.pi.local

JupyterLab

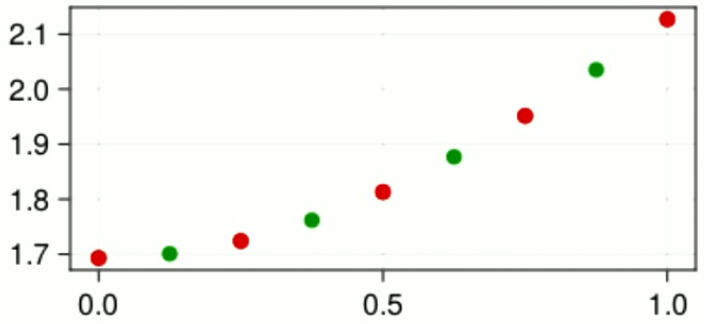
File Edit View Run Kernel Git Tabs Settings Help — Mem:2.29 GB

Untitled10.ipynb | Untitled11.ipynb

Code — git — Julia 1.8.5

[15]:
```
# Discretize the function
f0 = f.(x0)
f1 = f.(x1)
fig = Figure(; resolution=(400, 200))
ax = Axis(fig[1, 1])
plot!(x1, f1; color=:green)
plot!(x0, f0; color=:red)
fig
```

[15]:



[ ]:

Simple | 0 | 4 | Julia 1.8.5 | Idle | Mem: 2.45 GB | Saving completed | Mode: Command | Ln 1, Col 1 | Untitled11.ipynb

Safari File Edit View History Bookmarks Develop Window Help — Wed Feb 1 15:51

symmetry.pi.local

JupyterLab

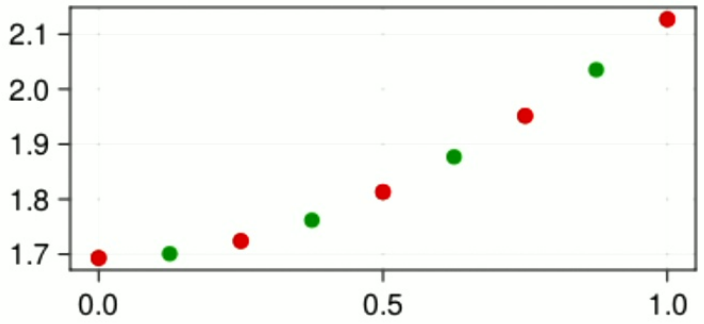
File Edit View Run Kernel Git Tabs Settings Help — Mem:2.42 GB

Untitled10.ipynb | Untitled11.ipynb

Code — git — Julia 1.8.5

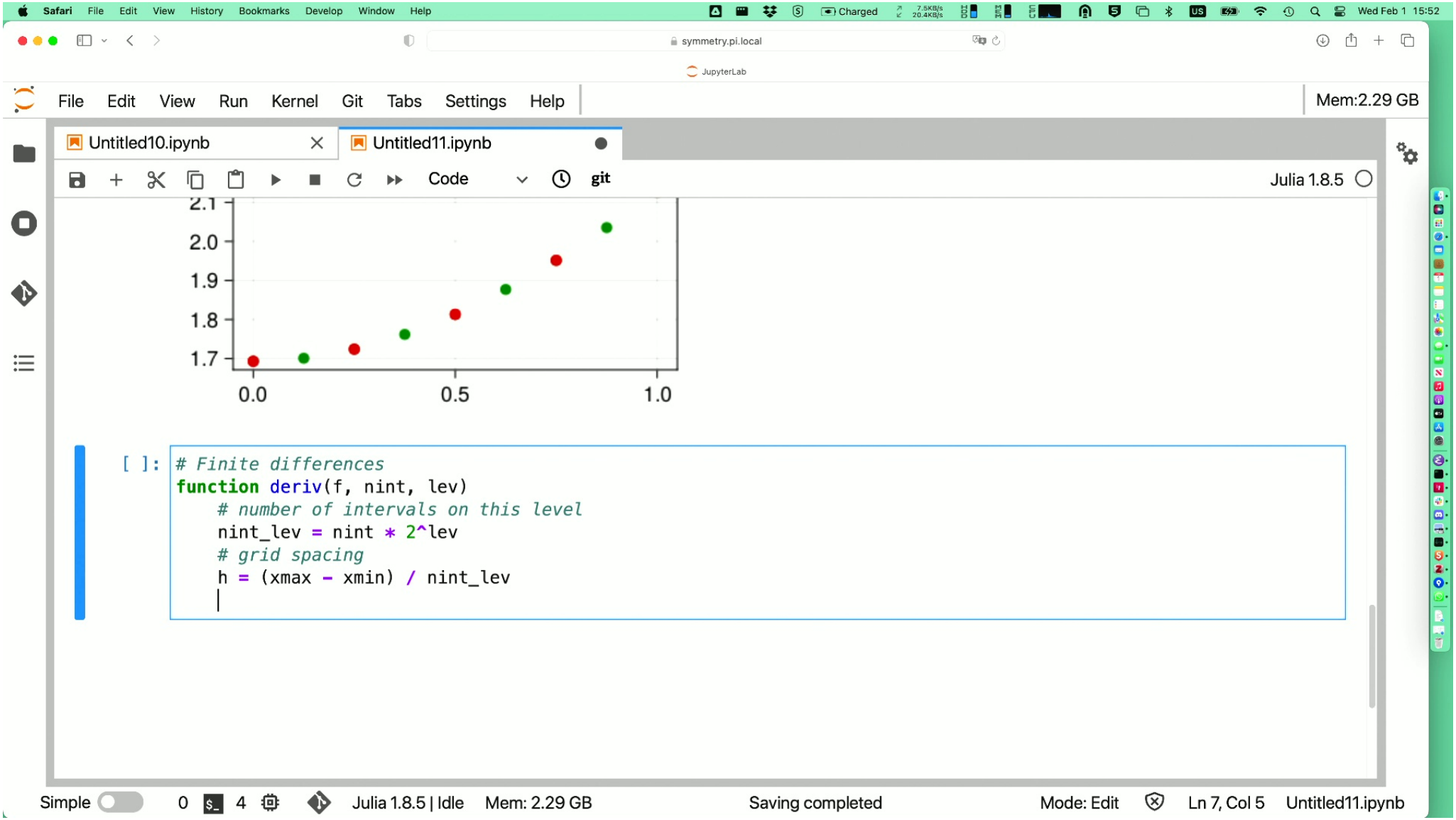
[16]:
```
# Discretize the function
f0 = f.(x0)
f1 = f.(x1)
fig = Figure(resolution=(400, 200))
ax = Axis(fig[1, 1])
plot!(x1, f1; color=:green)
plot!(x0, f0; color=:red)
fig
```

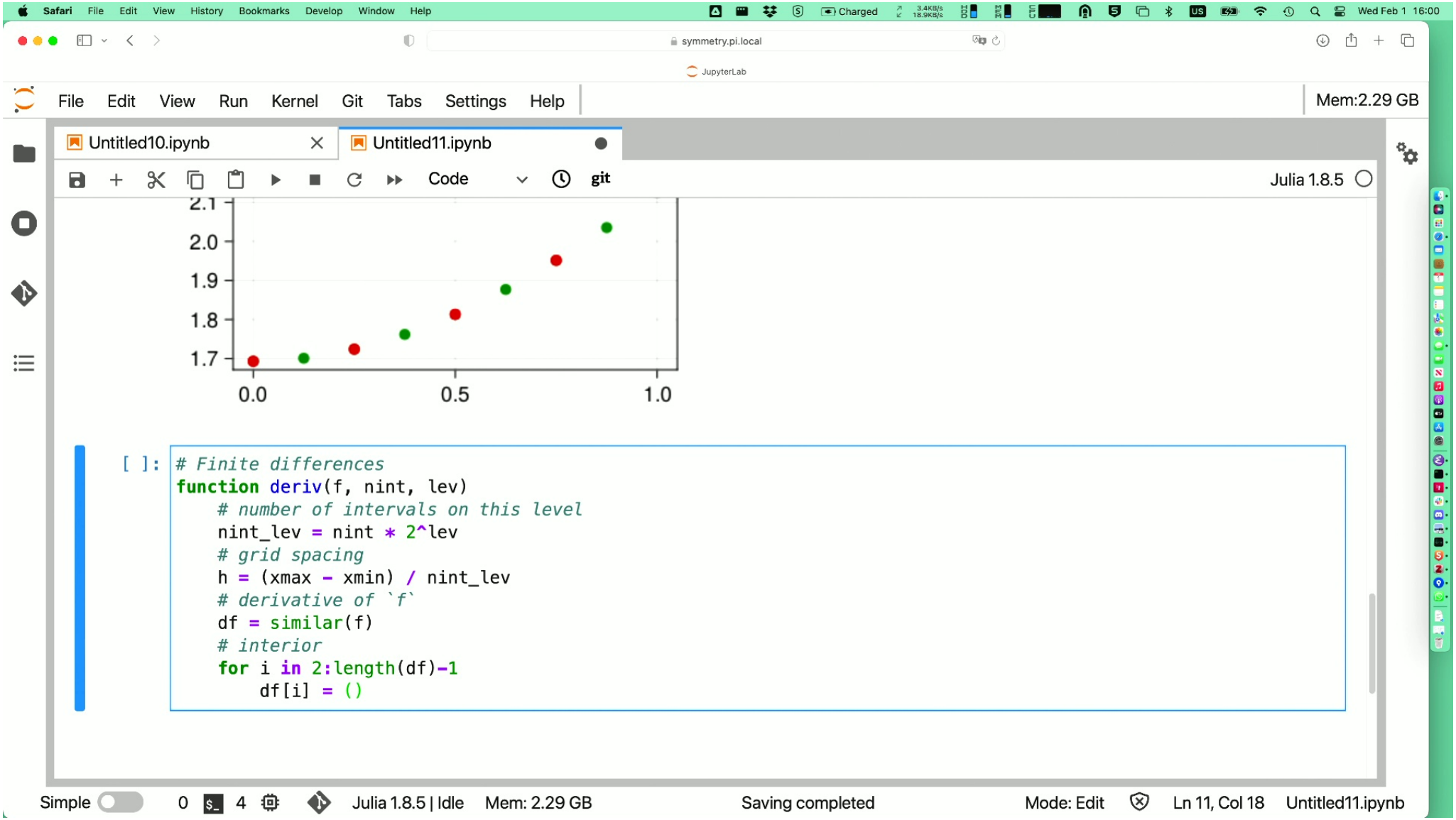
[16]:

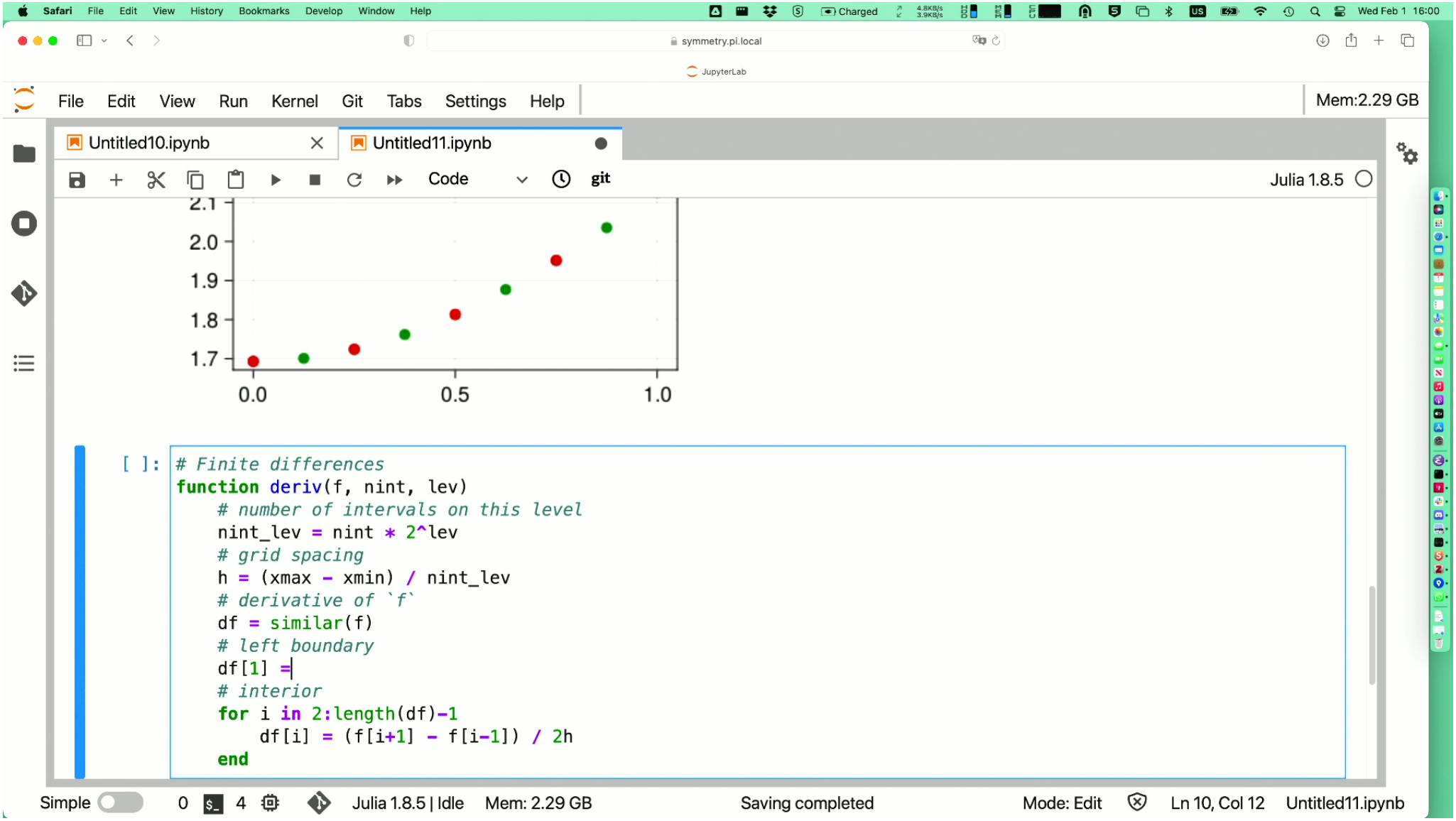


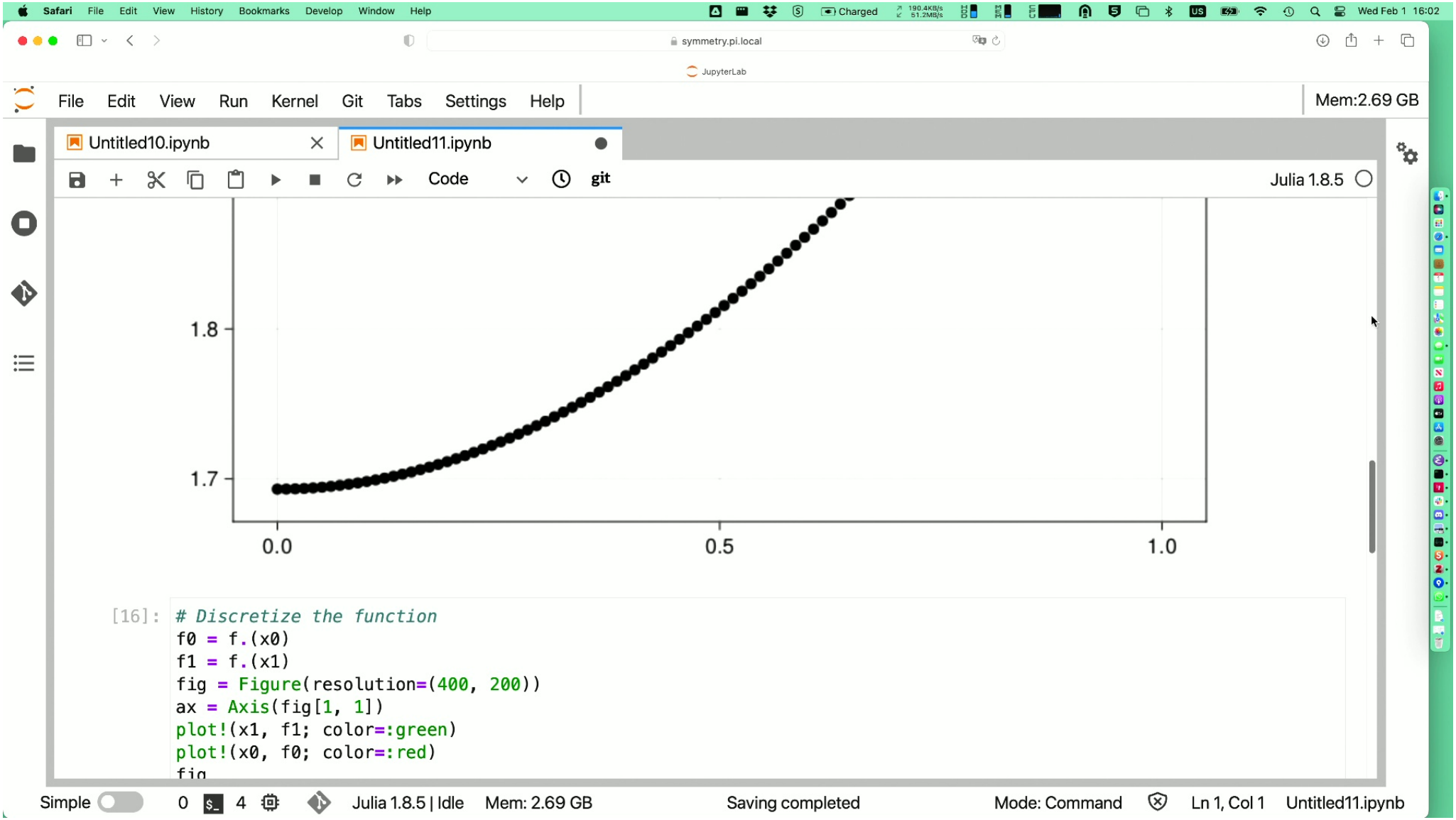
[ ]:
```
# Finite differences
function deriv(f, )
```

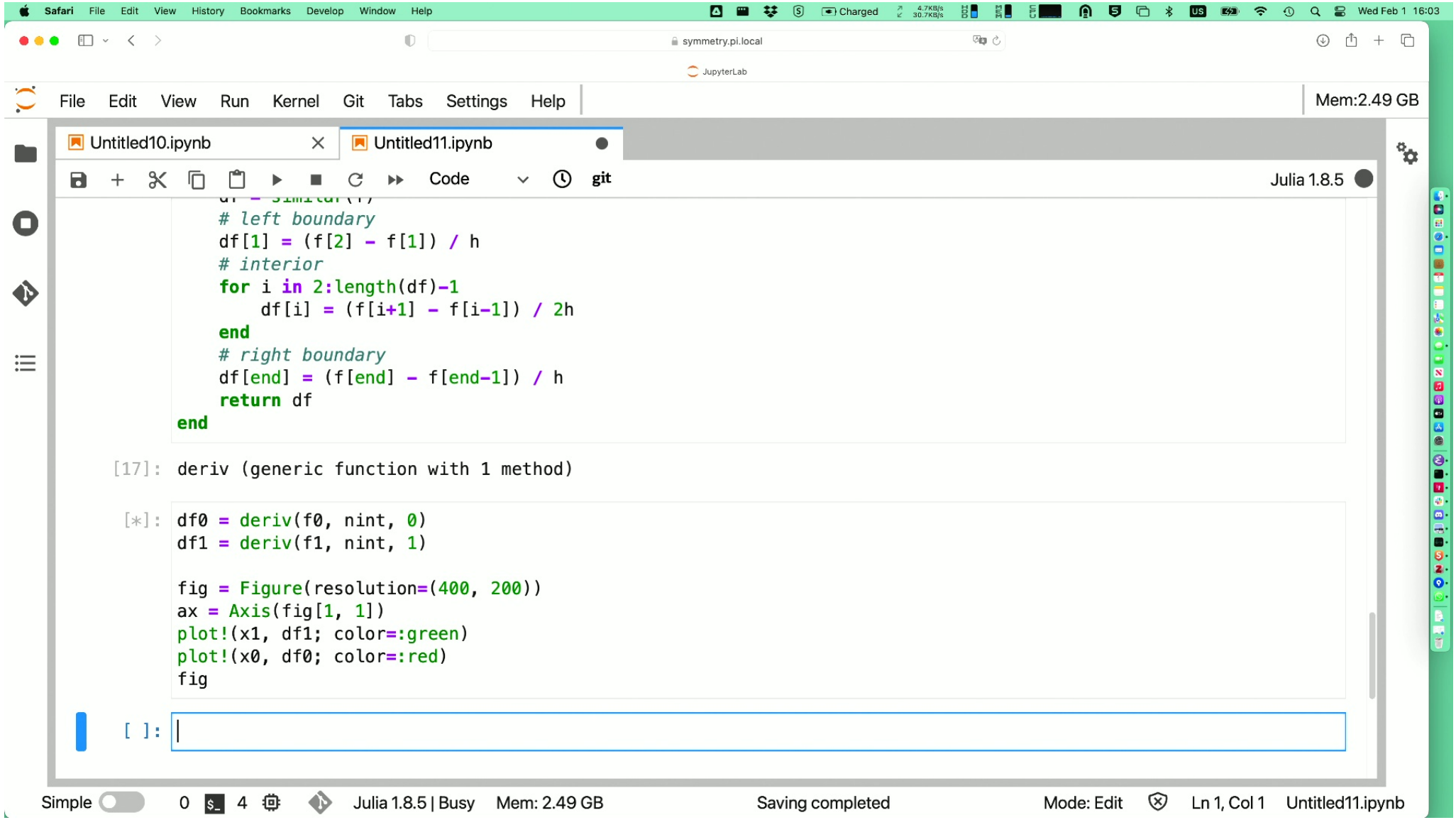
Simple 0 4 Julia 1.8.5 | Idle Mem: 2.42 GB Saving completed Mode: Edit Ln 2, Col 19 Untitled11.ipynb


Safari File Edit View History Bookmarks Develop Window Help
Wed Feb 1 15:52
symmetry.pi.local
JupyterLab
File Edit View Run Kernel Git Tabs Settings Help
Mem:2.29 GB
Untitled10.ipynb
Untitled11.ipynb
Code
git
Julia 1.8.5
2.1
2.0
1.9
1.8
1.7
0.0
0.5
1.0
[ ]:
Finite differences
function deriv(f, nint, lev)
# number of intervals on this level
nint_lev = nint * 2^lev
# grid spacing
h = (xmax - xmin) / nint_lev
Simple
0
4
Julia 1.8.5 | Idle
Mem: 2.29 GB
Saving completed
Mode: Edit
Ln 7, Col 5
Untitled11.ipynb


Safari File Edit View History Bookmarks Develop Window Help
Wed Feb 1 16:00
symmetry.pi.local
JupyterLab
File Edit View Run Kernel Git Tabs Settings Help
Mem:2.29 GB
Untitled10.ipynb
Untitled11.ipynb
Code
git
Julia 1.8.5
2.1
2.0
1.9
1.8
1.7
0.0
0.5
1.0
[ ]:
Finite differences
function deriv(f, nint, lev)
# number of intervals on this level
nint_lev = nint * 2^lev
# grid spacing
h = (xmax - xmin) / nint_lev
# derivative of `f`
df = similar(f)
# interior
for i in 2:length(df)-1
df[i] = ()
Simple
0
4
Julia 1.8.5 | Idle
Mem: 2.29 GB
Saving completed
Mode: Edit
Ln 11, Col 18
Untitled11.ipynb


Safari File Edit View History Bookmarks Develop Window Help
Wed Feb 1 16:00
symmetry.pi.local
JupyterLab
File Edit View Run Kernel Git Tabs Settings Help
Mem:2.29 GB
Untitled10.ipynb
Untitled11.ipynb
Code
git
Julia 1.8.5
2.1
2.0
1.9
1.8
1.7
0.0
0.5
1.0
[ ]:
Finite differences
function deriv(f, nint, lev)
# number of intervals on this level
nint_lev = nint * 2^lev
# grid spacing
h = (xmax - xmin) / nint_lev
# derivative of `f`
df = similar(f)
# left boundary
df[1] =
interior
for i in 2:length(df)-1
df[i] = (f[i+1] - f[i-1]) / 2h
end
Simple
0
4
Julia 1.8.5 | Idle
Mem: 2.29 GB
Saving completed
Mode: Edit
Ln 10, Col 12
Untitled11.ipynb


Safari File Edit View History Bookmarks Develop Window Help
Wed Feb 1 16:02
symmetry.pi.local
JupyterLab
File Edit View Run Kernel Git Tabs Settings Help
Mem:2.69 GB
Untitled10.ipynb
Untitled11.ipynb
Code
git
Julia 1.8.5
1.8
1.7
0.0
0.5
1.0
[16]:
# Discretize the function
f0 = f.(x0)
f1 = f.(x1)
fig = Figure(resolution=(400, 200))
ax = Axis(fig[1, 1])
plot!(x1, f1; color=:green)
plot!(x0, f0; color=:red)
fig
Simple
0
4
Julia 1.8.5 | Idle
Mem: 2.69 GB
Saving completed
Mode: Command
Ln 1, Col 1
Untitled11.ipynb


Safari File Edit View History Bookmarks Develop Window Help
Wed Feb 1 16:03
symmetry.pi.local
JupyterLab
File Edit View Run Kernel Git Tabs Settings Help
Mem:2.49 GB
Untitled10.ipynb
Untitled11.ipynb
Code
git
Julia 1.8.5
# left boundary
df[1] = (f[2] - f[1]) / h
# interior
for i in 2:length(df)-1
df[i] = (f[i+1] - f[i-1]) / 2h
end
# right boundary
df[end] = (f[end] - f[end-1]) / h
return df
end
[17]: deriv (generic function with 1 method)
[*]: df0 = deriv(f0, nint, 0)
df1 = deriv(f1, nint, 1)
fig = Figure(resolution=(400, 200))
ax = Axis(fig[1, 1])
plot!(x1, df1; color=:green)
plot!(x0, df0; color=:red)
fig
[ ]:
Simple
0 4
Julia 1.8.5 | Busy
Mem: 2.49 GB
Saving completed
Mode: Edit
Ln 1, Col 1
Untitled11.ipynb

Safari File Edit View History Bookmarks Develop Window Help

symmetry.pi.local

JupyterLab

File Edit View Run Kernel Git Tabs Settings Help

Mem:2.41 GB

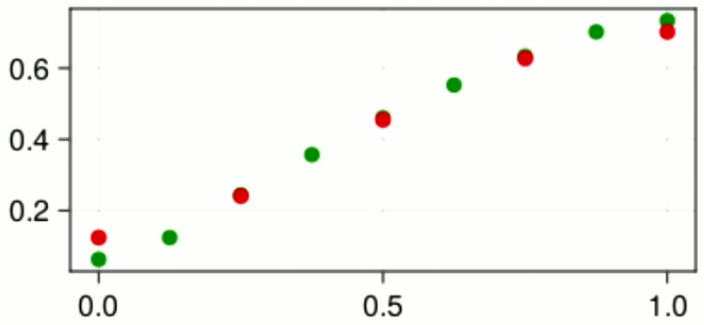
Untitled10.ipynb | Untitled11.ipynb

Code git Julia 1.8.5

```
[21]: deriv (generic function with 2 methods)
```

```
[22]: df0 = deriv(f0, 1/4)
      df1 = deriv(f1, 1/8)

      fig = Figure(resolution=(400, 200))
      ax = Axis(fig[1, 1])
      plot!(x1, df1; color=:green)
      plot!(x0, df0; color=:red)
      fig
```

[22]:

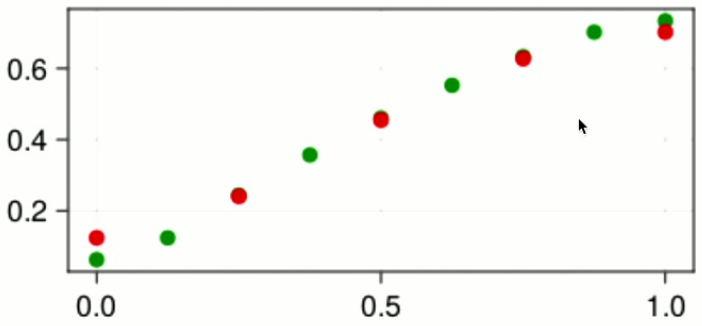



```
[ ]:
```

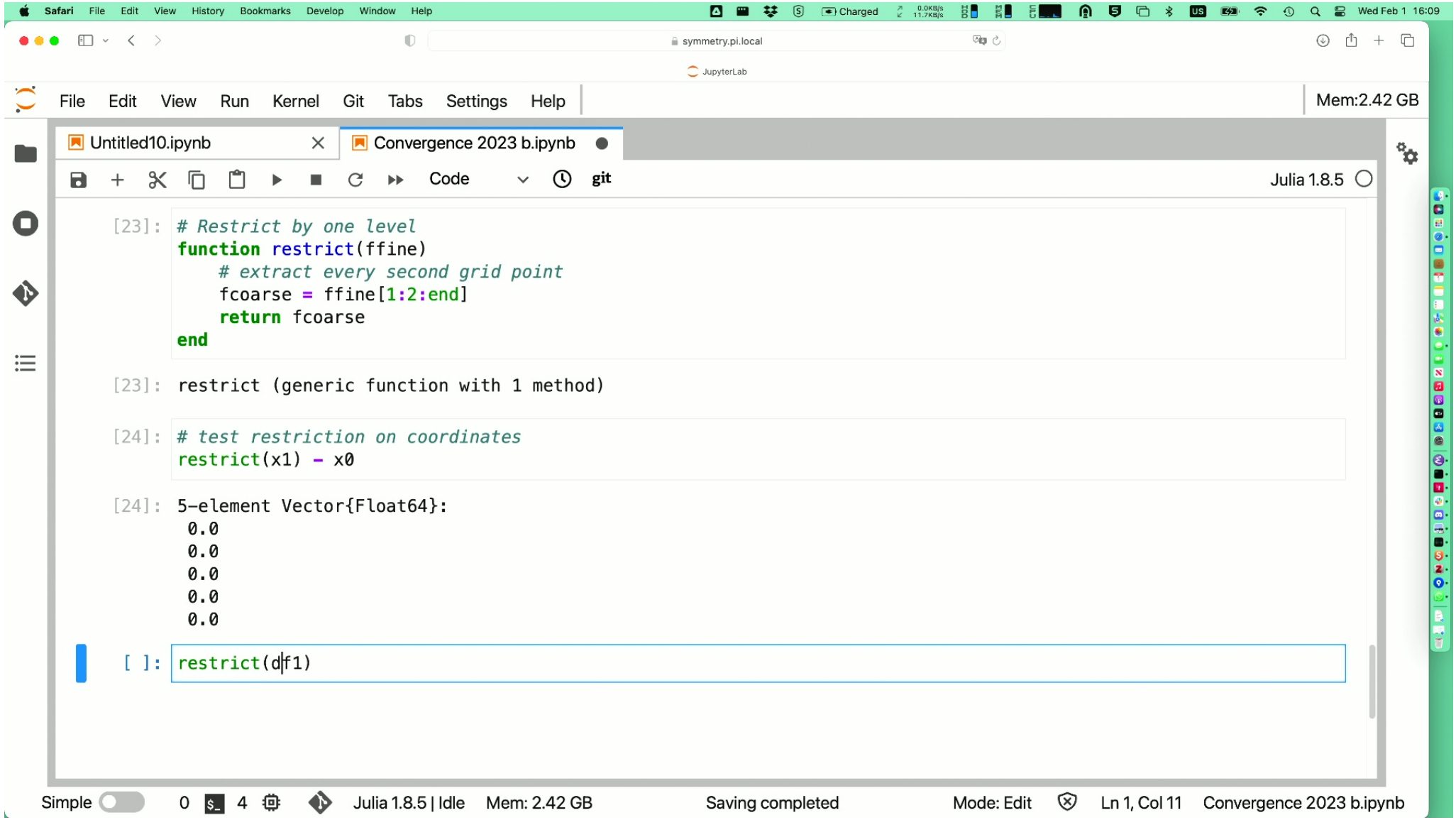
Simple 0 4 Julia 1.8.5 | Idle Mem: 2.41 GB Saving completed Mode: Command Ln 1, Col 1 Untitled11.ipynb

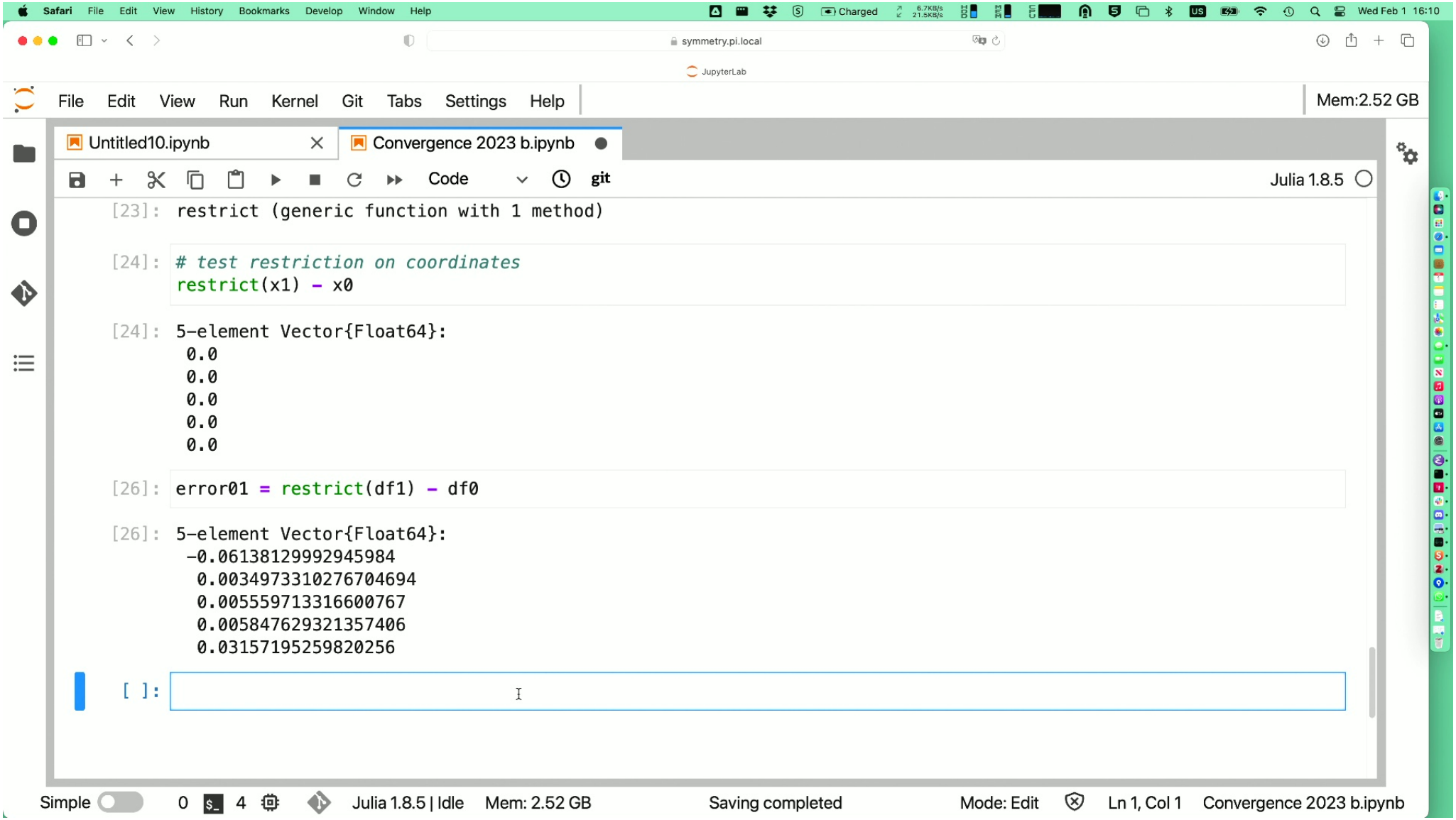
```
df1 = deriv(f1, 1/8)

fig = Figure(resolution=(400, 200))
ax = Axis(fig[1, 1])
plot!(x1, df1; color=:green)
plot!(x0, df0; color=:red)
fig
```

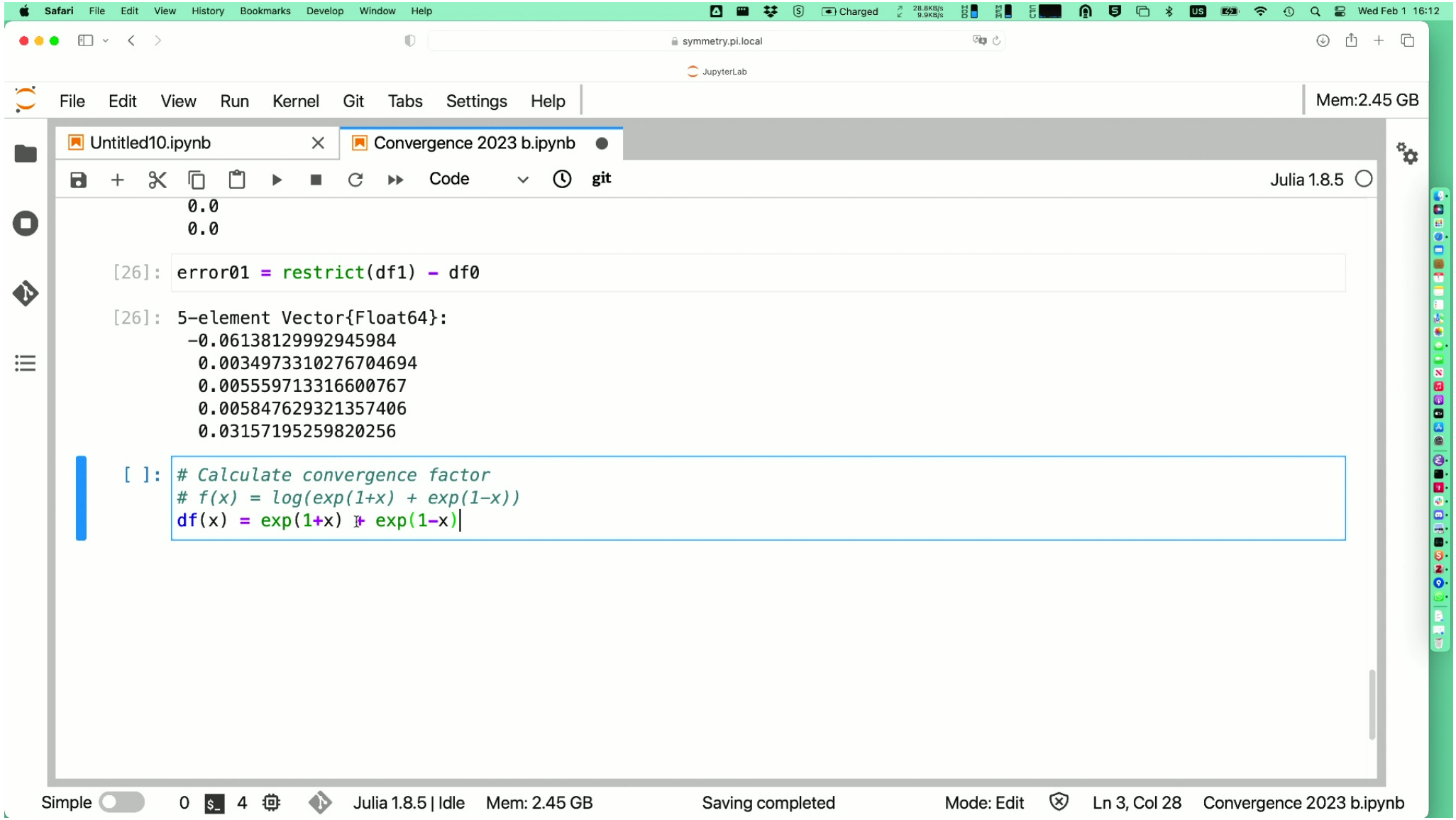
[22]:

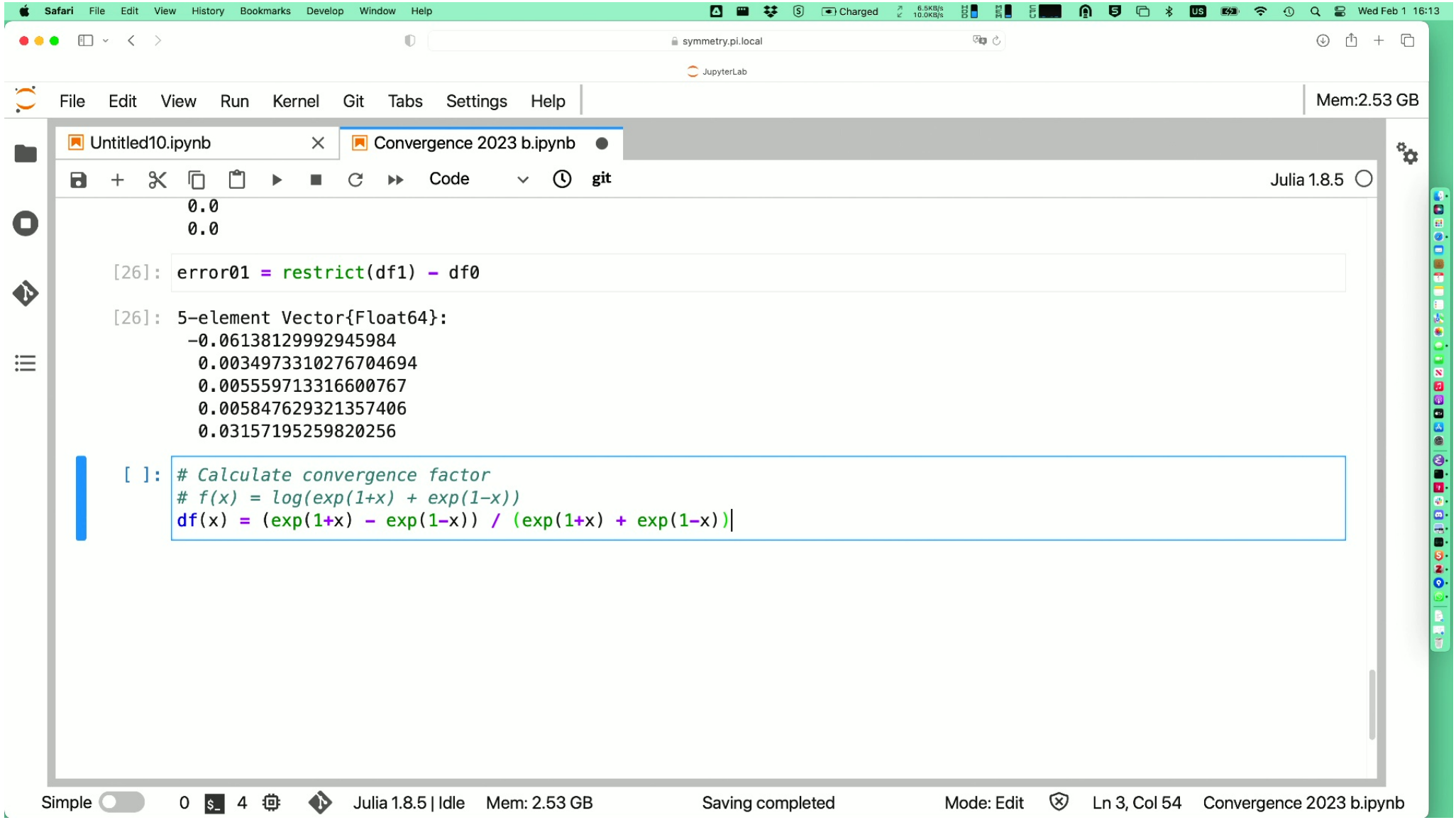


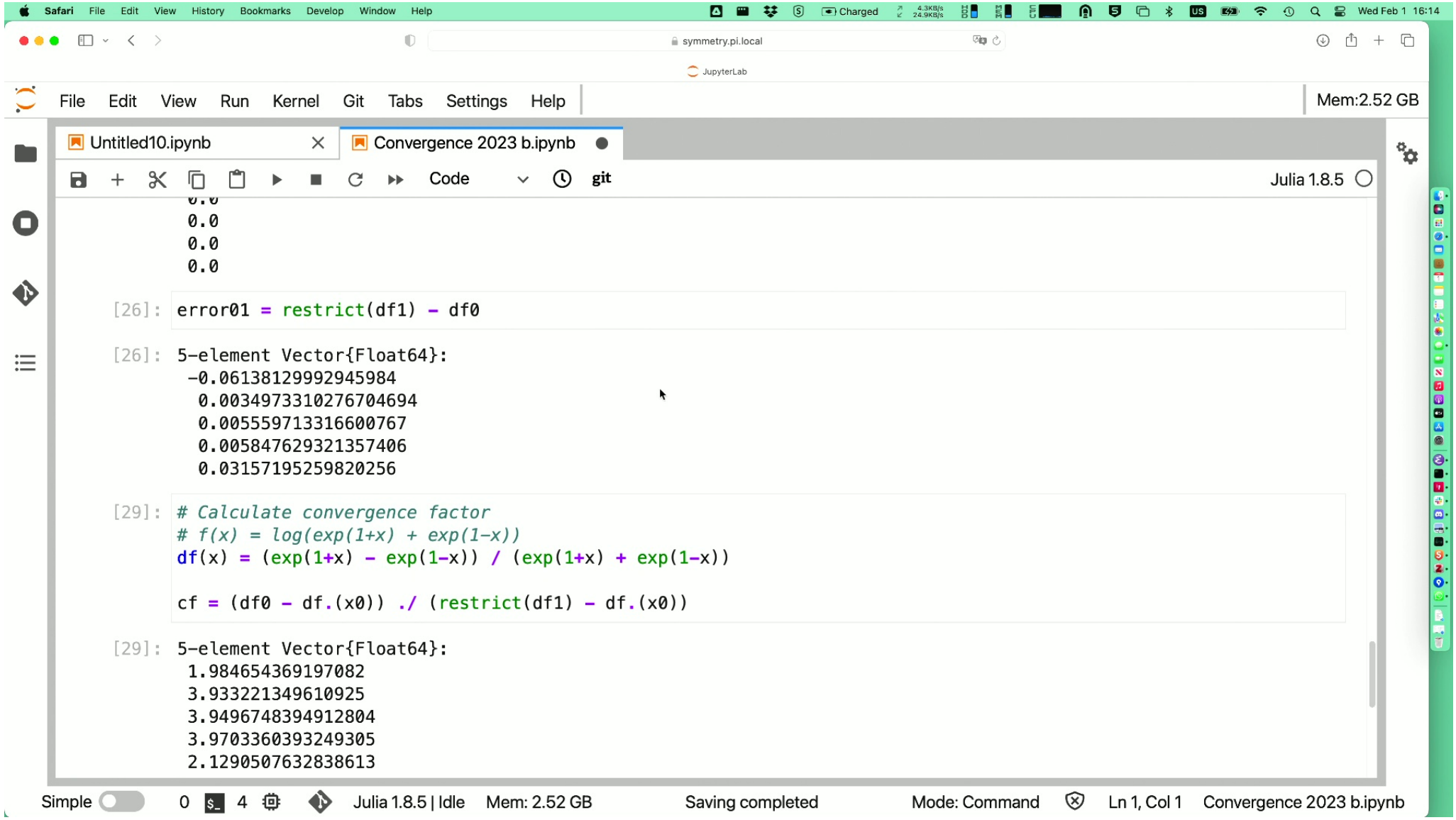
[ ]:


Safari File Edit View History Bookmarks Develop Window Help
Wed Feb 1 16:09
symmetry.pi.local
JupyterLab
File Edit View Run Kernel Git Tabs Settings Help
Mem:2.42 GB
Untitled10.ipynb
Convergence 2023 b.ipynb
Code
git
Julia 1.8.5
[23]:
Restrict by one level
function restrict(ffine)
# extract every second grid point
fcoarse = ffine[1:2:end]
return fcoarse
end
[23]:
restrict (generic function with 1 method)
[24]:
test restriction on coordinates
restrict(x1) - x0
[24]:
5-element Vector{Float64}:
0.0
0.0
0.0
0.0
0.0
[ ]:
restrict(df1)
Simple
0
4
Julia 1.8.5 | Idle
Mem: 2.42 GB
Saving completed
Mode: Edit
Ln 1, Col 11
Convergence 2023 b.ipynb


Safari File Edit View History Bookmarks Develop Window Help
Wed Feb 1 16:10
symmetry.pi.local
JupyterLab
File Edit View Run Kernel Git Tabs Settings Help
Mem:2.52 GB
Untitled10.ipynb
Convergence 2023 b.ipynb
Code
git
Julia 1.8.5
[23]: restrict (generic function with 1 method)
[24]: # test restriction on coordinates
restrict(x1) - x0
[24]: 5-element Vector{Float64}:
0.0
0.0
0.0
0.0
0.0
[26]: error01 = restrict(df1) - df0
[26]: 5-element Vector{Float64}:
-0.06138129992945984
0.0034973310276704694
0.005559713316600767
0.005847629321357406
0.03157195259820256
[ ]:
Simple
0
4
Julia 1.8.5 | Idle
Mem: 2.52 GB
Saving completed
Mode: Edit
Ln 1, Col 1
Convergence 2023 b.ipynb


Safari File Edit View History Bookmarks Develop Window Help
Wed Feb 1 16:12
symmetry.pi.local
JupyterLab
File Edit View Run Kernel Git Tabs Settings Help
Mem:2.45 GB
Untitled10.ipynb
Convergence 2023 b.ipynb
Code
git
Julia 1.8.5
0.0
0.0
[26]: error01 = restrict(df1) - df0
[26]: 5-element Vector{Float64}:
-0.06138129992945984
0.0034973310276704694
0.005559713316600767
0.005847629321357406
0.03157195259820256
[ ]: # Calculate convergence factor
# f(x) = log(exp(1+x) + exp(1-x))
df(x) = exp(1+x) + exp(1-x)
Simple
0
4
Julia 1.8.5 | Idle
Mem: 2.45 GB
Saving completed
Mode: Edit
Ln 3, Col 28
Convergence 2023 b.ipynb


Safari File Edit View History Bookmarks Develop Window Help
Wed Feb 1 16:13
symmetry.pi.local
JupyterLab
File Edit View Run Kernel Git Tabs Settings Help
Mem:2.53 GB
Untitled10.ipynb
Convergence 2023 b.ipynb
Code
git
Julia 1.8.5
0.0
0.0
[26]: error01 = restrict(df1) - df0
[26]: 5-element Vector{Float64}:
-0.06138129992945984
0.0034973310276704694
0.005559713316600767
0.005847629321357406
0.03157195259820256
[ ]: # Calculate convergence factor
# f(x) = log(exp(1+x) + exp(1-x))
df(x) = (exp(1+x) - exp(1-x)) / (exp(1+x) + exp(1-x))
Simple
0
4
Julia 1.8.5 | Idle
Mem: 2.53 GB
Saving completed
Mode: Edit
Ln 3, Col 54
Convergence 2023 b.ipynb


Safari File Edit View History Bookmarks Develop Window Help
Wed Feb 1 16:14
symmetry.pi.local
JupyterLab
File Edit View Run Kernel Git Tabs Settings Help
Mem:2.52 GB
Untitled10.ipynb
Convergence 2023 b.ipynb
Code
git
Julia 1.8.5
0.0
0.0
0.0
0.0
[26]: error01 = restrict(df1) - df0
[26]: 5-element Vector{Float64}:
-0.06138129992945984
0.0034973310276704694
0.005559713316600767
0.005847629321357406
0.03157195259820256
[29]: # Calculate convergence factor
# f(x) = log(exp(1+x) + exp(1-x))
df(x) = (exp(1+x) - exp(1-x)) / (exp(1+x) + exp(1-x))
cf = (df0 - df.(x0)) ./ (restrict(df1) - df.(x0))
[29]: 5-element Vector{Float64}:
1.984654369197082
3.933221349610925
3.9496748394912804
3.9703360393249305
2.1290507632838613
Simple
0
4
Julia 1.8.5 | Idle
Mem: 2.52 GB
Saving completed
Mode: Command
Ln 1, Col 1
Convergence 2023 b.ipynb

Safari File Edit View History Bookmarks Develop Window Help

symmetry.pi.local

JupyterLab

File Edit View Run Kernel Git Tabs Settings Help

Mem:2.71 GB

Untitled10.ipynb | Convergence 2023 b.ipynb

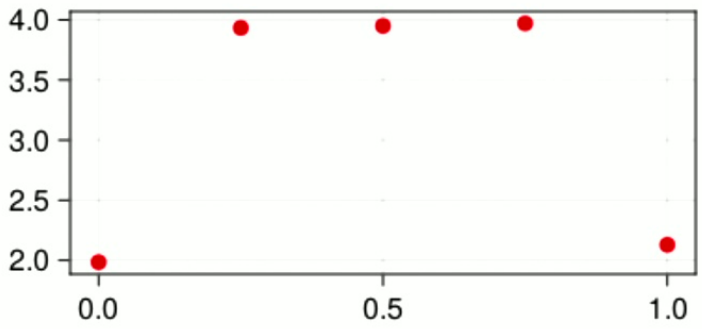
Code git Julia 1.8.5

```
0.005847629321357406
0.03157195259820256
```

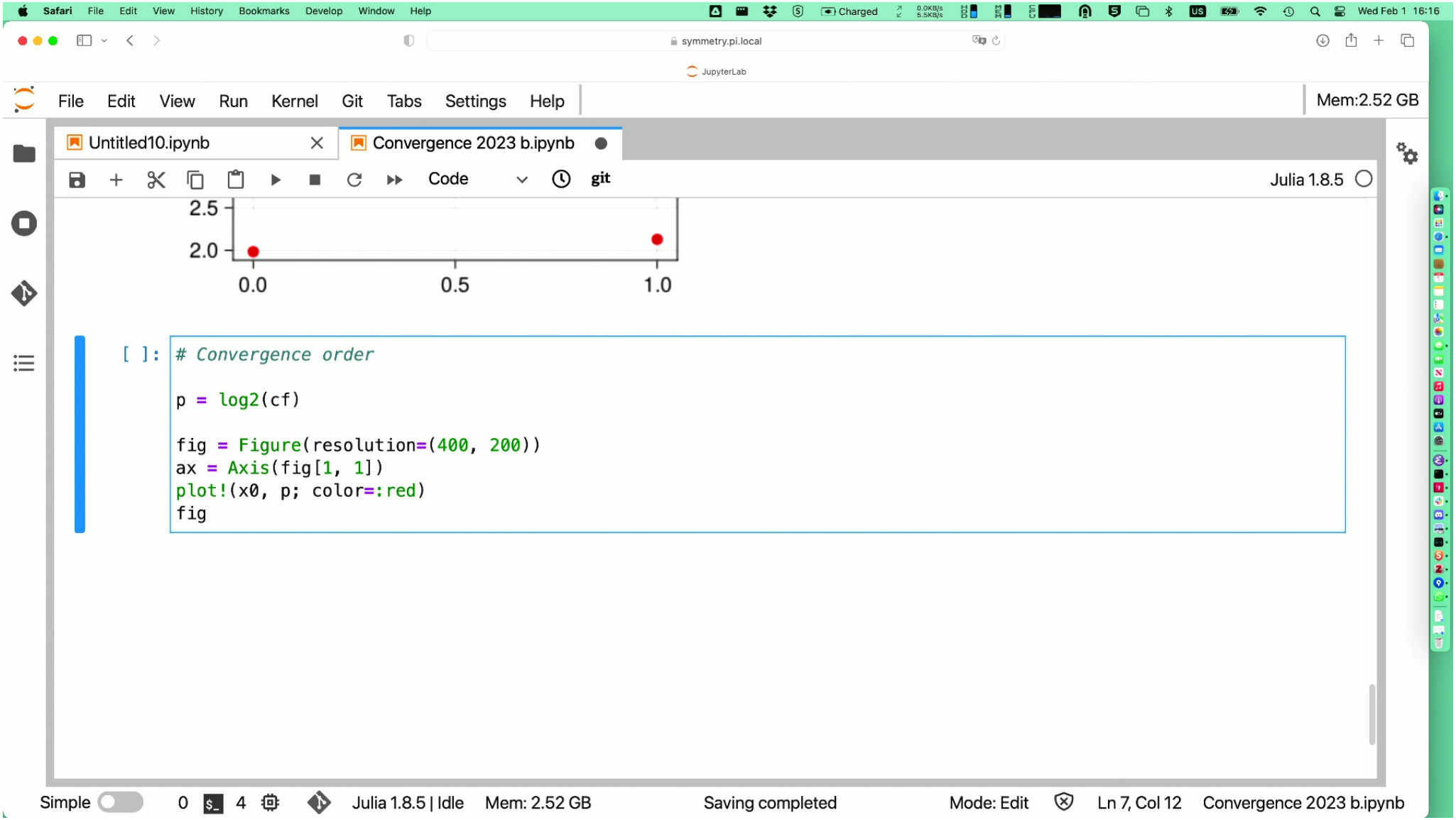
[30]:

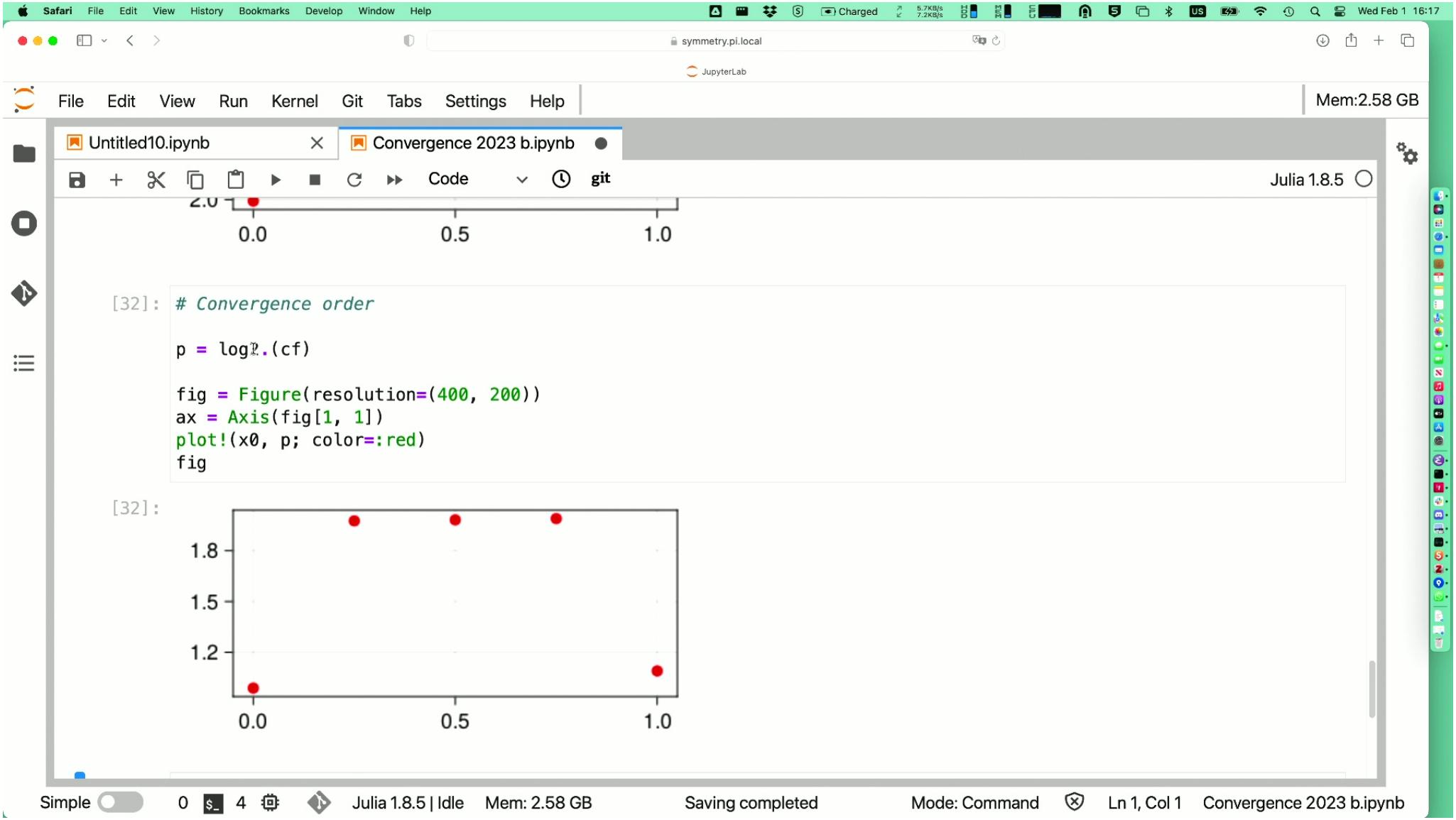
```
# Calculate convergence factor
# f(x) = log(exp(1+x) + exp(1-x))
df(x) = (exp(1+x) - exp(1-x)) / (exp(1+x) + exp(1-x))

cf = (df0 - df.(x0)) ./ (restrict(df1) - df.(x0))

fig = Figure(resolution=(400, 200))
ax = Axis(fig[1, 1])
plot!(x0, cf; color=:red)
fig
```

[30]:



Simple 0 4 Julia 1.8.5 | Idle Mem: 2.71 GB Saving completed Mode: Command Ln 1, Col 1 Convergence 2023 b.ipynb


Safari File Edit View History Bookmarks Develop Window Help
Wed Feb 1 16:16
symmetry.pi.local
JupyterLab
File Edit View Run Kernel Git Tabs Settings Help
Mem:2.52 GB
Untitled10.ipynb
Convergence 2023 b.ipynb
Code
git
Julia 1.8.5
2.5
2.0
0.0
0.5
1.0
[ ]:
# Convergence order
p = log2(cf)
fig = Figure(resolution=(400, 200))
ax = Axis(fig[1, 1])
plot!(x0, p; color=:red)
fig
Simple
0
4
Julia 1.8.5 | Idle
Mem: 2.52 GB
Saving completed
Mode: Edit
Ln 7, Col 12
Convergence 2023 b.ipynb


Safari File Edit View History Bookmarks Develop Window Help
Wed Feb 1 16:17
symmetry.pi.local
JupyterLab
File Edit View Run Kernel Git Tabs Settings Help
Mem:2.58 GB
Untitled10.ipynb
Convergence 2023 b.ipynb
Code
git
Julia 1.8.5
2.0
0.0
0.5
1.0
[32]:
# Convergence order
p = log2.(cf)
fig = Figure(resolution=(400, 200))
ax = Axis(fig[1, 1])
plot!(x0, p; color=:red)
fig
[32]:
1.8
1.5
1.2
0.0
0.5
1.0
Simple
0
4
Julia 1.8.5 | Idle
Mem: 2.58 GB
Saving completed
Mode: Command
Ln 1, Col 1
Convergence 2023 b.ipynb

Safari File Edit View History Bookmarks Develop Window Help

symmetry.pi.local

JupyterLab

File Edit View Run Kernel Git Tabs Settings Help

Mem:2.58 GB

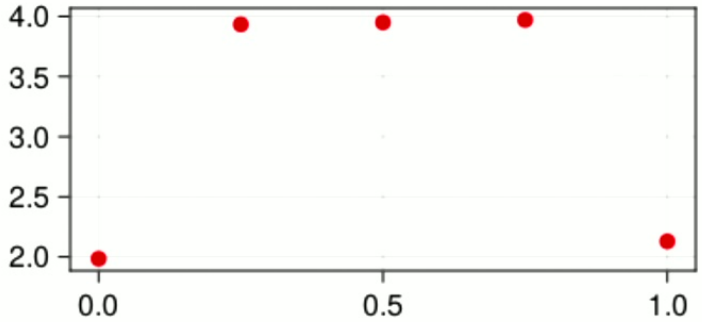
Untitled10.ipynb

Convergence 2023 b.ipynb

Code git

Julia 1.8.5

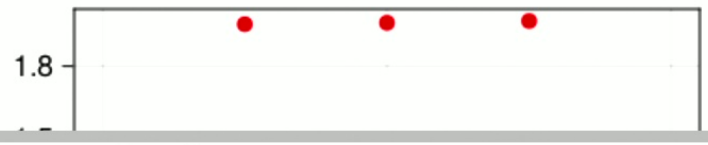
[30]:



[32]:

```
# Convergence order

p = log2.(cf)

fig = Figure(resolution=(400, 200))
ax = Axis(fig[1, 1])
plot!(x0, p; color=:red)
fig
```

[32]:



Simple 0 4 Julia 1.8.5 | Idle Mem: 2.58 GB Saving completed Mode: Command Ln 10, Col 4 Convergence 2023 b.ipynb

Safari File Edit View History Bookmarks Develop Window Help — Wed Feb 1 16:19

symmetry.pi.local

JupyterLab

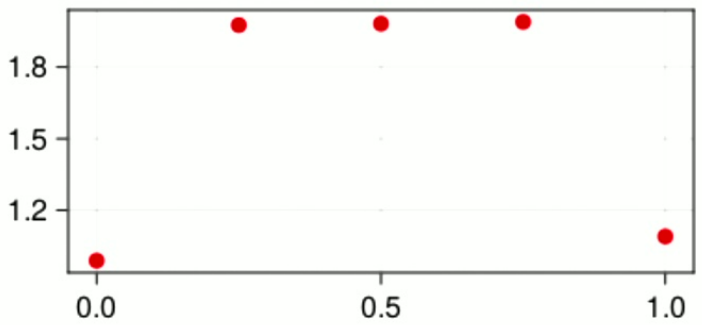
File Edit View Run Kernel Git Tabs Settings Help — Mem:2.73 GB

Untitled10.ipynb | Convergence 2023 b.ipynb

Code — git — Julia 1.8.5

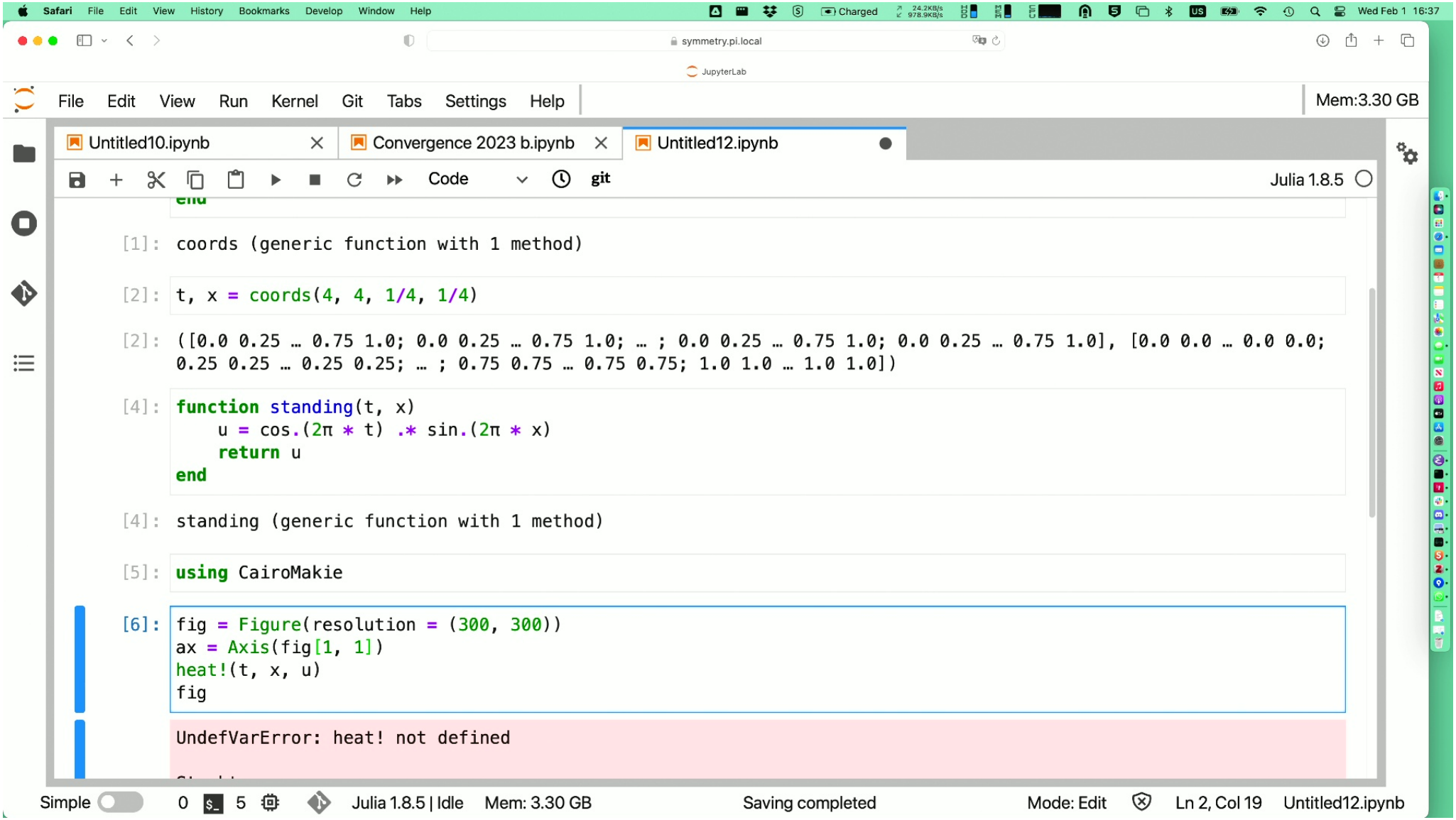
[32]:
```
# Convergence order

p = log2.(cf)

fig = Figure(resolution=(400, 200))
ax = Axis(fig[1, 1])
plot!(x0, p; color=:red)
fig
```

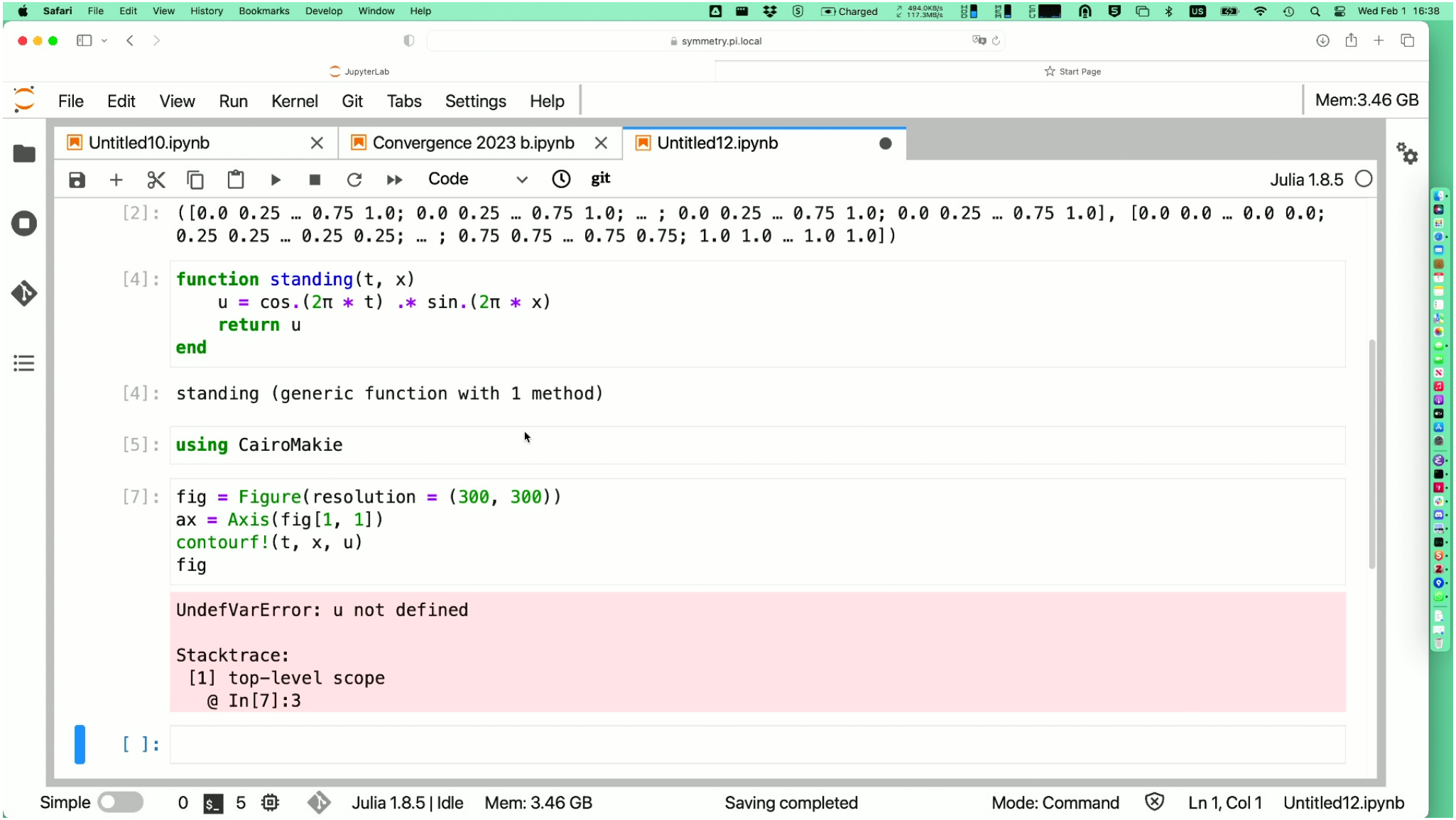
[32]:

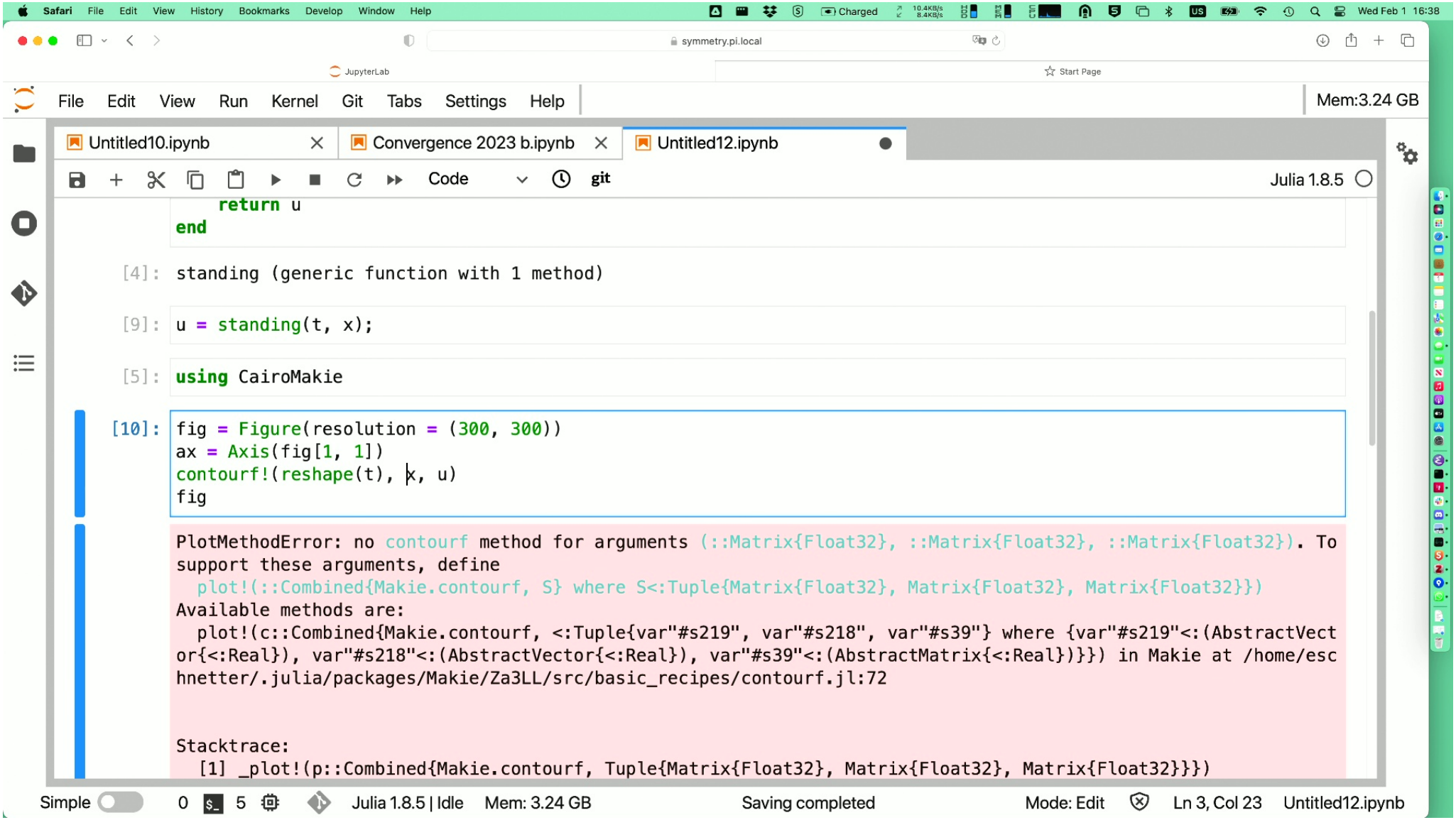


[ ]:

Simple — 0 — 4 — Julia 1.8.5 | Idle — Mem: 2.73 GB — Saving completed — Mode: Command — Ln 10, Col 4 — Convergence 2023 b.ipynb


Safari File Edit View History Bookmarks Develop Window Help
Wed Feb 1 16:37
symmetry.pi.local
JupyterLab
File Edit View Run Kernel Git Tabs Settings Help
Mem:3.30 GB
Untitled10.ipynb
Convergence 2023 b.ipynb
Untitled12.ipynb
Code
git
Julia 1.8.5
end
[1]: coords (generic function with 1 method)
[2]: t, x = coords(4, 4, 1/4, 1/4)
[2]: ([0.0 0.25 … 0.75 1.0; 0.0 0.25 … 0.75 1.0; … ; 0.0 0.25 … 0.75 1.0; 0.0 0.25 … 0.75 1.0], [0.0 0.0 … 0.0 0.0; 0.25 0.25 … 0.25 0.25; … ; 0.75 0.75 … 0.75 0.75; 1.0 1.0 … 1.0 1.0])
[4]: function standing(t, x)
u = cos.(2π * t) .* sin.(2π * x)
return u
end
[4]: standing (generic function with 1 method)
[5]: using CairoMakie
[6]: fig = Figure(resolution = (300, 300))
ax = Axis(fig[1, 1])
heat!(t, x, u)
fig
UndefVarError: heat! not defined
Simple
0 5
Julia 1.8.5 | Idle
Mem: 3.30 GB
Saving completed
Mode: Edit
Ln 2, Col 19
Untitled12.ipynb


Safari File Edit View History Bookmarks Develop Window Help
Wed Feb 1 16:38
symmetry.pi.local
JupyterLab
Start Page
File Edit View Run Kernel Git Tabs Settings Help
Mem:3.46 GB
Untitled10.ipynb
Convergence 2023 b.ipynb
Untitled12.ipynb
Code
git
Julia 1.8.5
[2]: ([0.0 0.25 … 0.75 1.0; 0.0 0.25 … 0.75 1.0; … ; 0.0 0.25 … 0.75 1.0; 0.0 0.25 … 0.75 1.0], [0.0 0.0 … 0.0 0.0; 0.25 0.25 … 0.25 0.25; … ; 0.75 0.75 … 0.75 0.75; 1.0 1.0 … 1.0 1.0])
[4]: function standing(t, x)
u = cos.(2π * t) .* sin.(2π * x)
return u
end
[4]: standing (generic function with 1 method)
[5]: using CairoMakie
[7]: fig = Figure(resolution = (300, 300))
ax = Axis(fig[1, 1])
contourf!(t, x, u)
fig
UndefVarError: u not defined
Stacktrace:
[1] top-level scope
@ In[7]:3
[ ]:
Simple
0 5
Julia 1.8.5 | Idle
Mem: 3.46 GB
Saving completed
Mode: Command
Ln 1, Col 1
Untitled12.ipynb


Safari File Edit View History Bookmarks Develop Window Help
Wed Feb 1 16:38
symmetry.pi.local
JupyterLab
Start Page
File Edit View Run Kernel Git Tabs Settings Help
Mem:3.24 GB
Untitled10.ipynb
Convergence 2023 b.ipynb
Untitled12.ipynb
Code
git
Julia 1.8.5
return u
end
[4]: standing (generic function with 1 method)
[9]: u = standing(t, x);
[5]: using CairoMakie
[10]: fig = Figure(resolution = (300, 300))
ax = Axis(fig[1, 1])
contourf!(reshape(t), x, u)
fig
PlotMethodError: no contourf method for arguments (::Matrix{Float32}, ::Matrix{Float32}, ::Matrix{Float32}). To support these arguments, define
plot!(::Combined{Makie.contourf, S} where S<:Tuple{Matrix{Float32}, Matrix{Float32}, Matrix{Float32}})
Available methods are:
plot!(c::Combined{Makie.contourf, <:Tuple{var"#s219", var"#s218", var"#s39"} where {var"#s219"<:(AbstractVector{<:Real}), var"#s218"<:(AbstractVector{<:Real}), var"#s39"<:(AbstractMatrix{<:Real})}}) in Makie at /home/eschnetter/.julia/packages/Makie/Za3LL/src/basic_recipes/contourf.jl:72
Stacktrace:
[1] _plot!(p::Combined{Makie.contourf, Tuple{Matrix{Float32}, Matrix{Float32}, Matrix{Float32}}})
Simple
0 5
Julia 1.8.5 | Idle
Mem: 3.24 GB
Saving completed
Mode: Edit
Ln 3, Col 23
Untitled12.ipynb

Safari File Edit View History Bookmarks Develop Window Help

symmetry.pi.local

JupyterLab

Start Page

File Edit View Run Kernel Git Tabs Settings Help

Mem:3.28 GB

Untitled10.ipynb | Convergence 2023 b.ipynb | Untitled12.ipynb

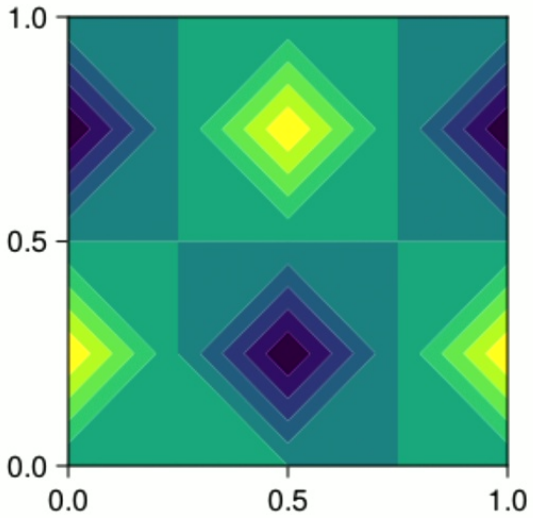
Code git Julia 1.8.5

```
using CairoMakie
```

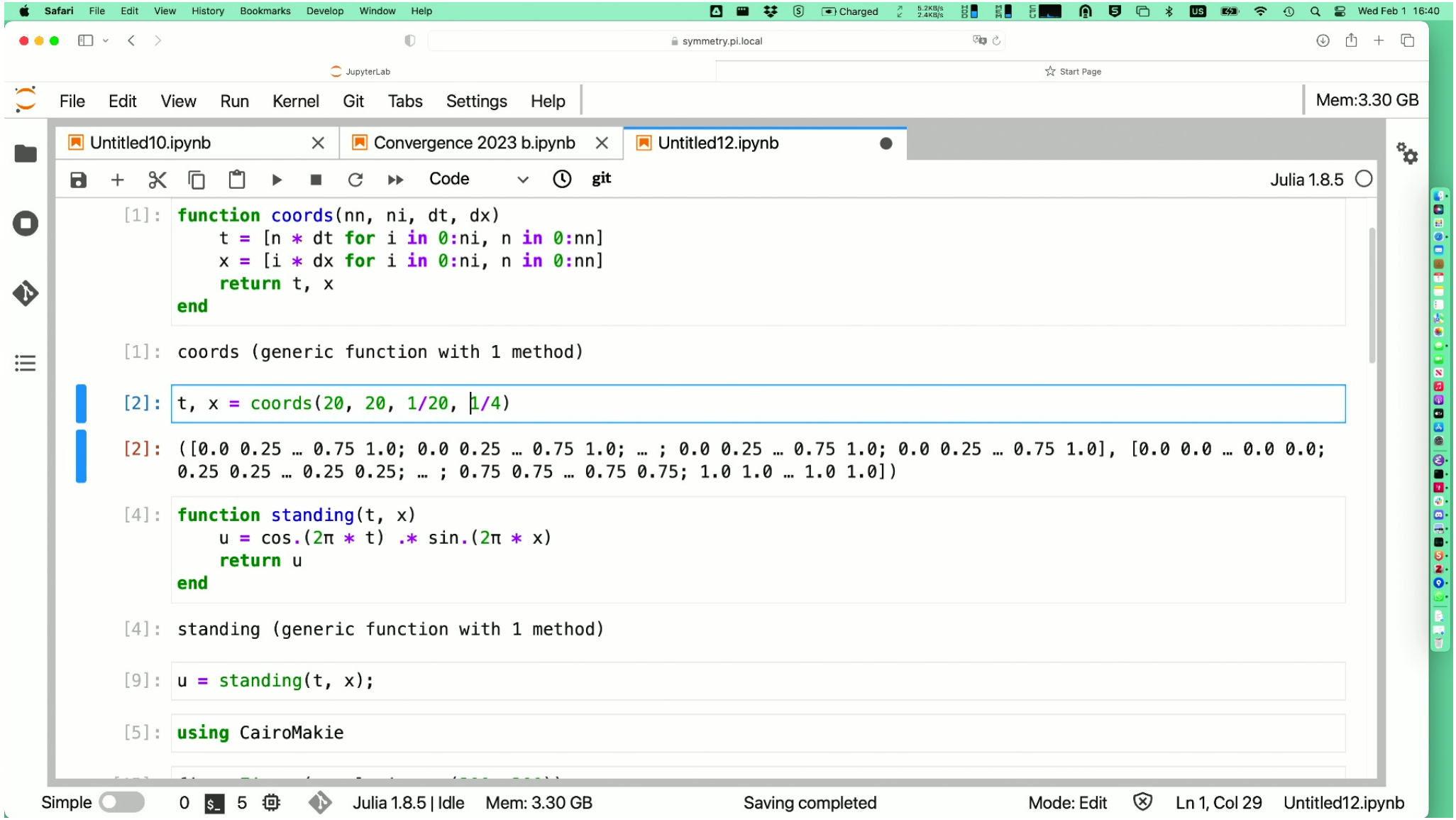
[13]:

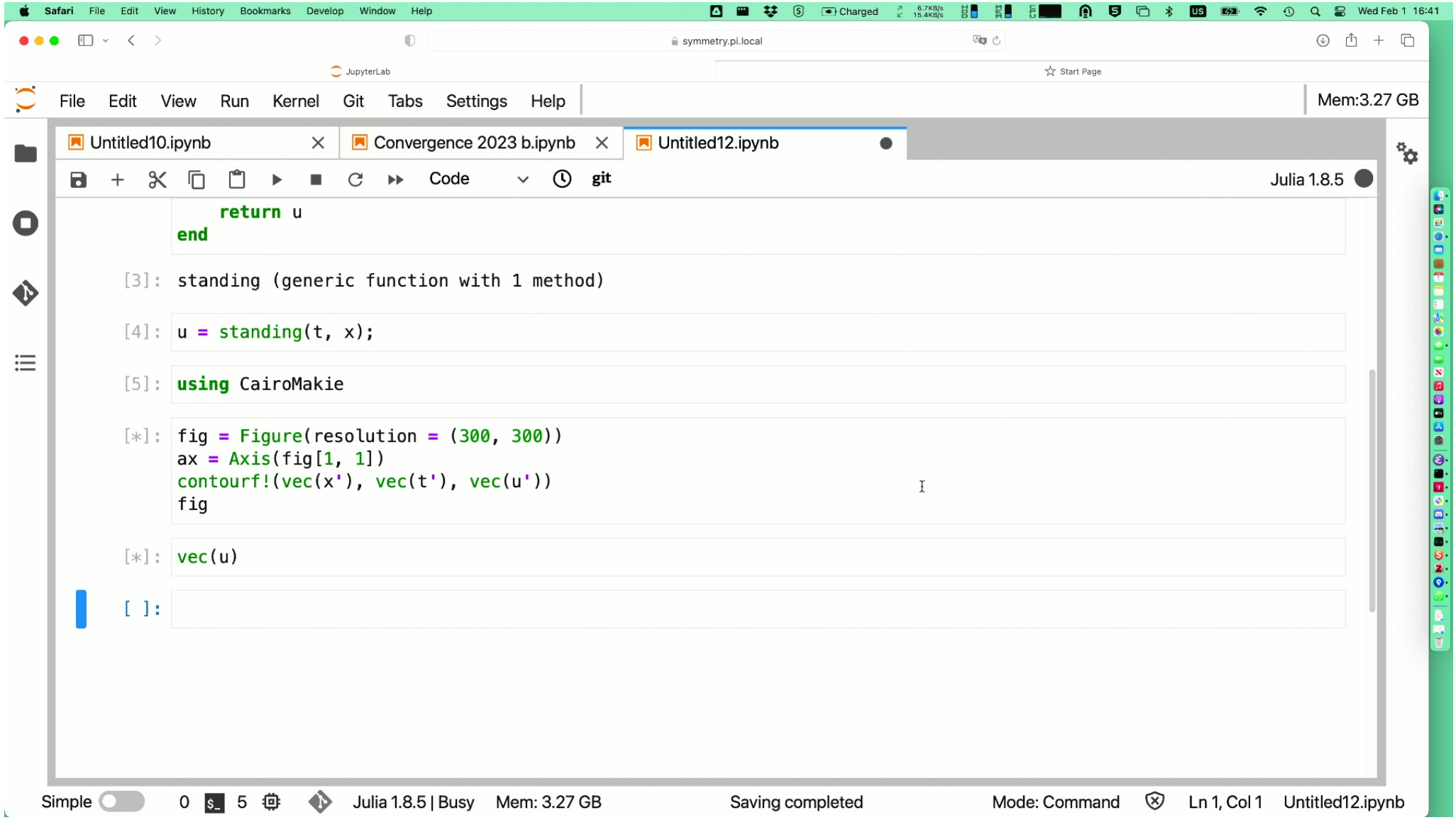
```
fig = Figure(resolution = (300, 300))
ax = Axis(fig[1, 1])
contourf!(reshape(t, :), reshape(x, :), reshape(u, :))
fig
```

[13]:



Simple 0 5 Julia 1.8.5 | Idle Mem: 3.28 GB Saving completed Mode: Command Ln 1, Col 1 Untitled12.ipynb


Safari File Edit View History Bookmarks Develop Window Help
Wed Feb 1 16:40
symmetry.pi.local
JupyterLab
Start Page
File Edit View Run Kernel Git Tabs Settings Help
Mem:3.30 GB
Untitled10.ipynb
Convergence 2023 b.ipynb
Untitled12.ipynb
Code
git
Julia 1.8.5
[1]: function coords(nn, ni, dt, dx)
t = [n * dt for i in 0:ni, n in 0:nn]
x = [i * dx for i in 0:ni, n in 0:nn]
return t, x
end
[1]: coords (generic function with 1 method)
[2]: t, x = coords(20, 20, 1/20, 1/4)
[2]: ([0.0 0.25 … 0.75 1.0; 0.0 0.25 … 0.75 1.0; … ; 0.0 0.25 … 0.75 1.0; 0.0 0.25 … 0.75 1.0], [0.0 0.0 … 0.0 0.0;
0.25 0.25 … 0.25 0.25; … ; 0.75 0.75 … 0.75 0.75; 1.0 1.0 … 1.0 1.0])
[4]: function standing(t, x)
u = cos.(2π * t) .* sin.(2π * x)
return u
end
[4]: standing (generic function with 1 method)
[9]: u = standing(t, x);
[5]: using CairoMakie
Simple
0 5
Julia 1.8.5 | Idle
Mem: 3.30 GB
Saving completed
Mode: Edit
Ln 1, Col 29
Untitled12.ipynb


Safari File Edit View History Bookmarks Develop Window Help
Wed Feb 1 16:41
symmetry.pi.local
JupyterLab
Start Page
File Edit View Run Kernel Git Tabs Settings Help
Mem:3.27 GB
Untitled10.ipynb
Convergence 2023 b.ipynb
Untitled12.ipynb
Code
git
Julia 1.8.5
return u
end
[3]: standing (generic function with 1 method)
[4]: u = standing(t, x);
[5]: using CairoMakie
[*]: fig = Figure(resolution = (300, 300))
ax = Axis(fig[1, 1])
contourf!(vec(x'), vec(t'), vec(u'))
fig
[*]: vec(u)
[ ]:
Simple
0 5
Julia 1.8.5 | Busy
Mem: 3.27 GB
Saving completed
Mode: Command
Ln 1, Col 1
Untitled12.ipynb

Safari File Edit View History Bookmarks Develop Window Help — Wed Feb 1 16:42

symmetry.pi.local

JupyterLab — Start Page

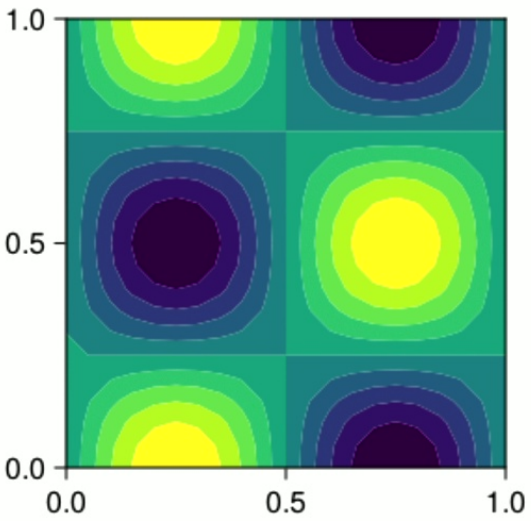
File Edit View Run Kernel Git Tabs Settings Help — Mem:3.24 GB

Untitled10.ipynb | Convergence 2023 b.ipynb | Untitled12.ipynb

Code — git — Julia 1.8.5

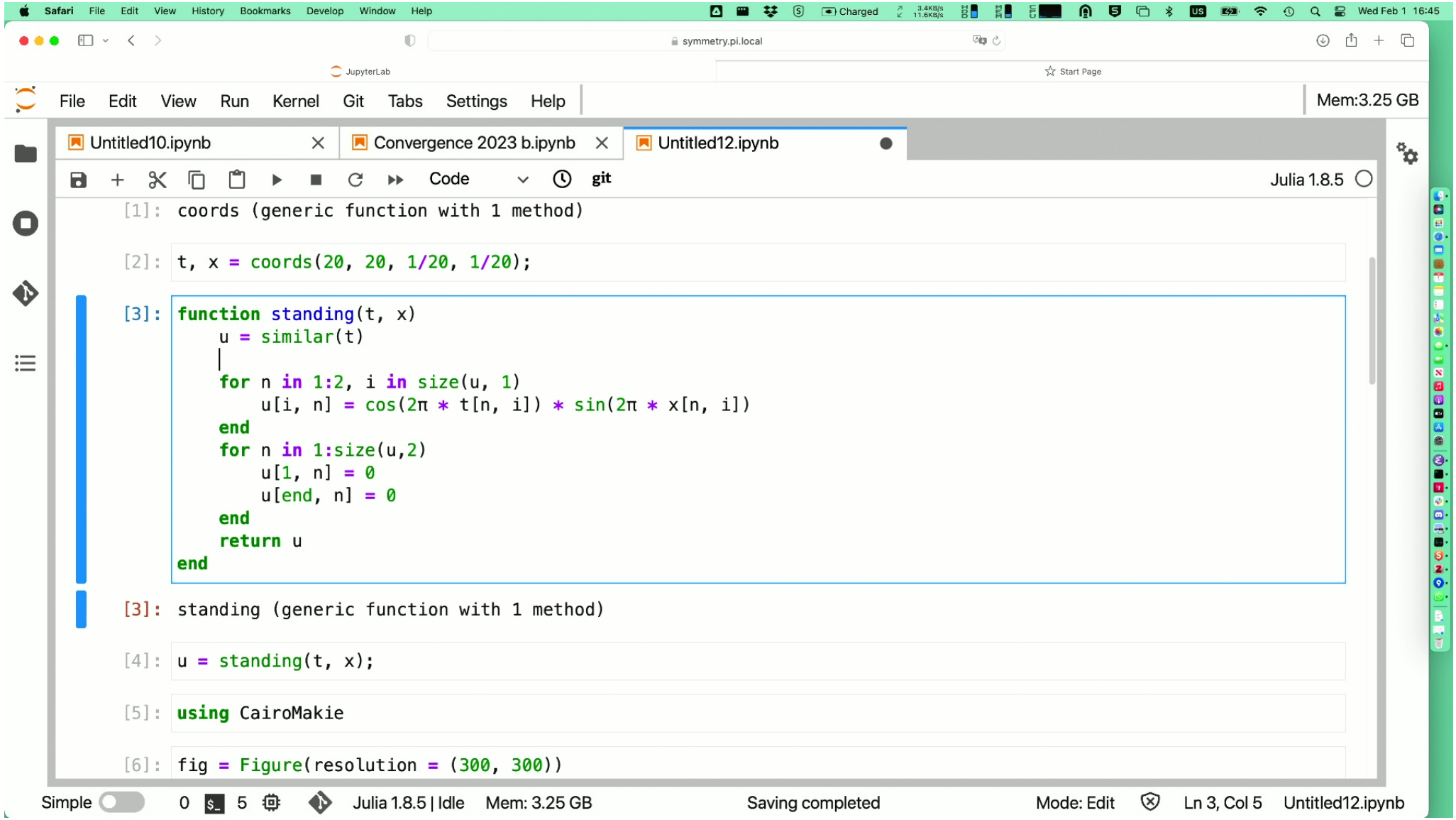
[6]:
```
fig = Figure(resolution = (300, 300))
ax = Axis(fig[1, 1])
contourf!(vec(x'), vec(t'), vec(u'))
fig
```

[6]:



[7]:
```
vec(u)
```

Simple — 0 — 5 — Julia 1.8.5 | Idle — Mem: 3.24 GB — Saving completed — Mode: Edit — Ln 2, Col 19 — Untitled12.ipynb


Safari File Edit View History Bookmarks Develop Window Help
Wed Feb 1 16:45
symmetry.pi.local
JupyterLab
Start Page
File Edit View Run Kernel Git Tabs Settings Help
Mem:3.25 GB
Untitled10.ipynb
Convergence 2023 b.ipynb
Untitled12.ipynb
Code
git
Julia 1.8.5
[1]: coords (generic function with 1 method)
[2]: t, x = coords(20, 20, 1/20, 1/20);
[3]:
function standing(t, x)
u = similar(t)
for n in 1:2, i in size(u, 1)
u[i, n] = cos(2π * t[n, i]) * sin(2π * x[n, i])
end
for n in 1:size(u,2)
u[1, n] = 0
u[end, n] = 0
end
return u
end
[3]: standing (generic function with 1 method)
[4]: u = standing(t, x);
[5]: using CairoMakie
[6]: fig = Figure(resolution = (300, 300))
Simple
0 5
Julia 1.8.5 | Idle
Mem: 3.25 GB
Saving completed
Mode: Edit
Ln 3, Col 5
Untitled12.ipynb

Safari File Edit View History Bookmarks Develop Window Help

symmetry.pi.local

JupyterLab

Start Page

File Edit View Run Kernel Git Tabs Settings Help

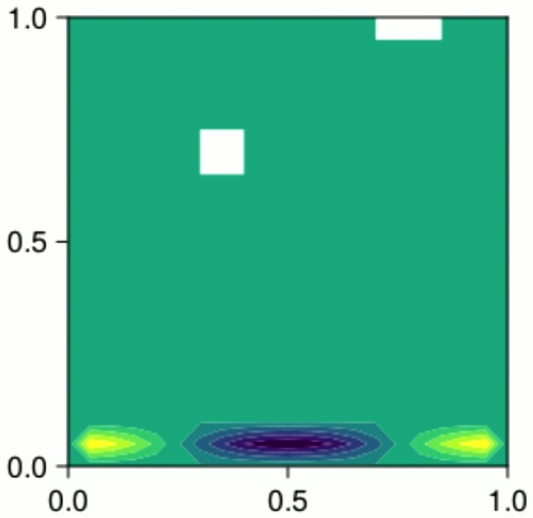
Mem:3.23 GB

Untitled10.ipynb | Convergence 2023 b.ipynb | Untitled12.ipynb

Code git Julia 1.8.5

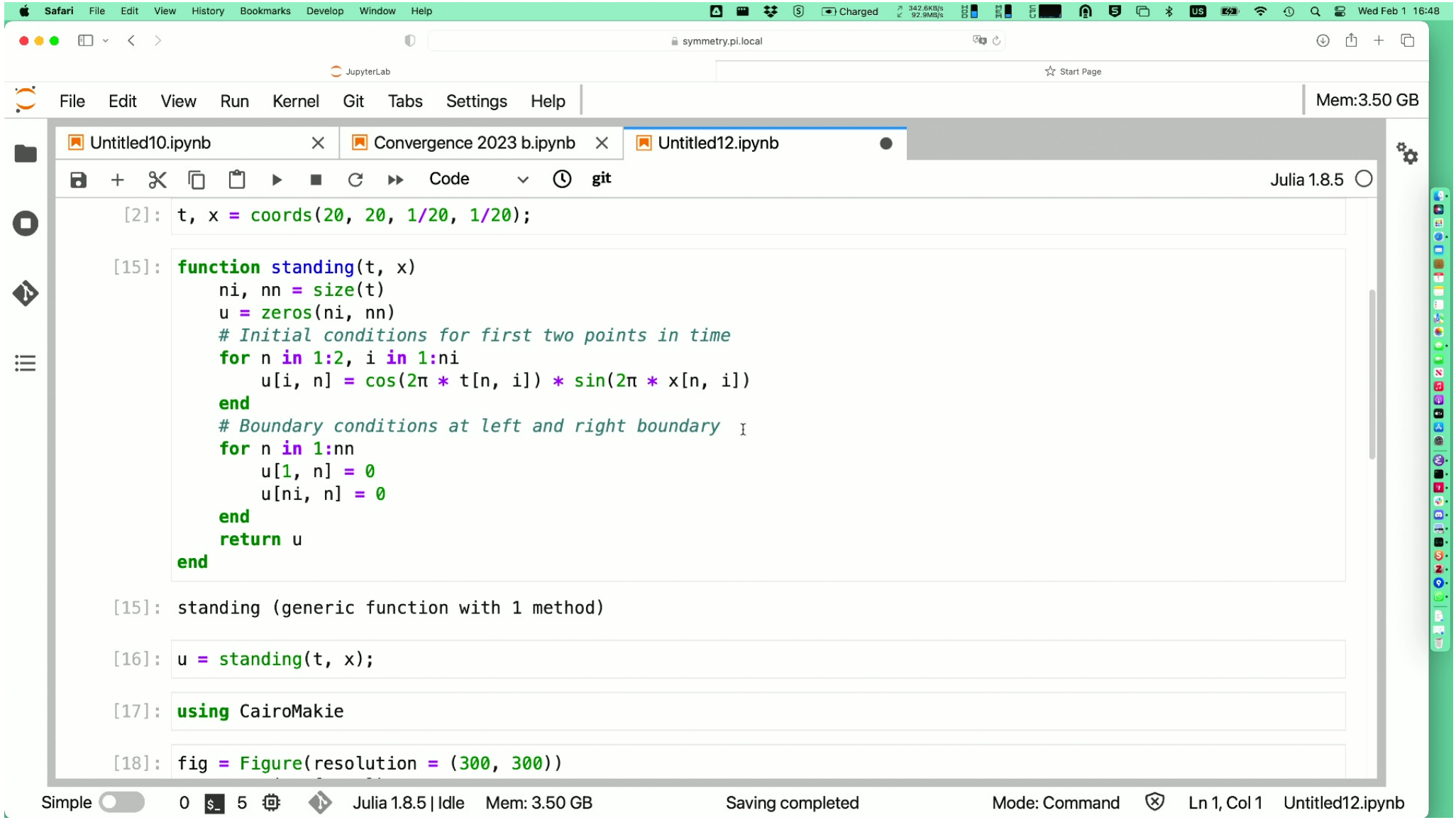
[10]:
```
fig = Figure(resolution = (300, 300))
ax = Axis(fig[1, 1])
contourf!(vec(x'), vec(t'), vec(u'))
fig
```

[10]:



[7]:
```
vec(u)
```

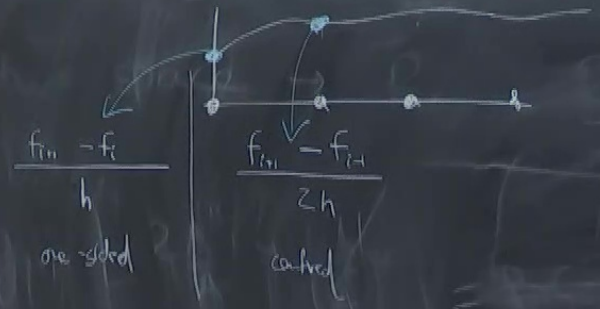
Simple 0 5 Julia 1.8.5 | Idle Mem: 3.23 GB Saving completed Mode: Command Ln 1, Col 7 Untitled12.ipynb


Safari File Edit View History Bookmarks Develop Window Help
Wed Feb 1 16:48
symmetry.pi.local
JupyterLab
Start Page
File Edit View Run Kernel Git Tabs Settings Help
Mem:3.50 GB
Untitled10.ipynb
Convergence 2023 b.ipynb
Untitled12.ipynb
Code
git
Julia 1.8.5
[2]: t, x = coords(20, 20, 1/20, 1/20);
[15]: function standing(t, x)
ni, nn = size(t)
u = zeros(ni, nn)
# Initial conditions for first two points in time
for n in 1:2, i in 1:ni
u[i, n] = cos(2π * t[n, i]) * sin(2π * x[n, i])
end
Boundary conditions at left and right boundary
for n in 1:nn
u[1, n] = 0
u[ni, n] = 0
end
return u
end
[15]: standing (generic function with 1 method)
[16]: u = standing(t, x);
[17]: using CairoMakie
[18]: fig = Figure(resolution = (300, 300))
Simple
0 5
Julia 1.8.5 | Idle
Mem: 3.50 GB
Saving completed
Mode: Command
Ln 1, Col 1
Untitled12.ipynb

boundaries

1. ghost zones

2. one-sided FD



$\frac{A''}{A} =$

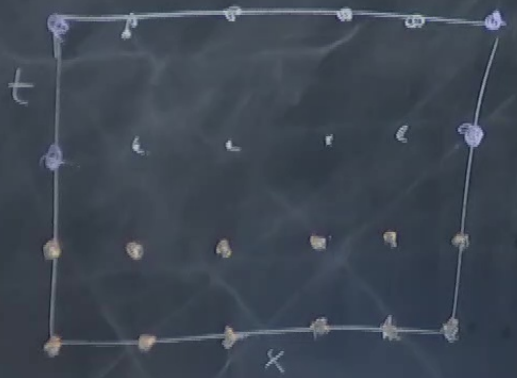
$A =$

$=$

→ gives

$$\partial_t^2 u(t,x) = \partial_x^2 u(t,x)$$

$$f''(x) = \frac{f(x-h) - 2f(x) + f(x+h)}{h^2} + \dots$$



$$\frac{f_{i-1} - 2f_i + f_{i+1}}{h^2}$$