Title: Numerical Methods Lecture

Speakers: Erik Schnetter

Collection: Numerical Methods 2023/24

Date: January 29, 2024 - 2:00 PM

URL: https://pirsa.org/24010018

Safari File Edit View History Bookmarks Develop Window Help Mon Jan 29 14:01

symmetry.pi.local

Ordinary Dif... - JupyterLab | Getting Started with Differential Equations in Julia · DifferentialEquations.jl

File Edit View Run Kernel Tabs Settings Help

Ordinary Differential Equation

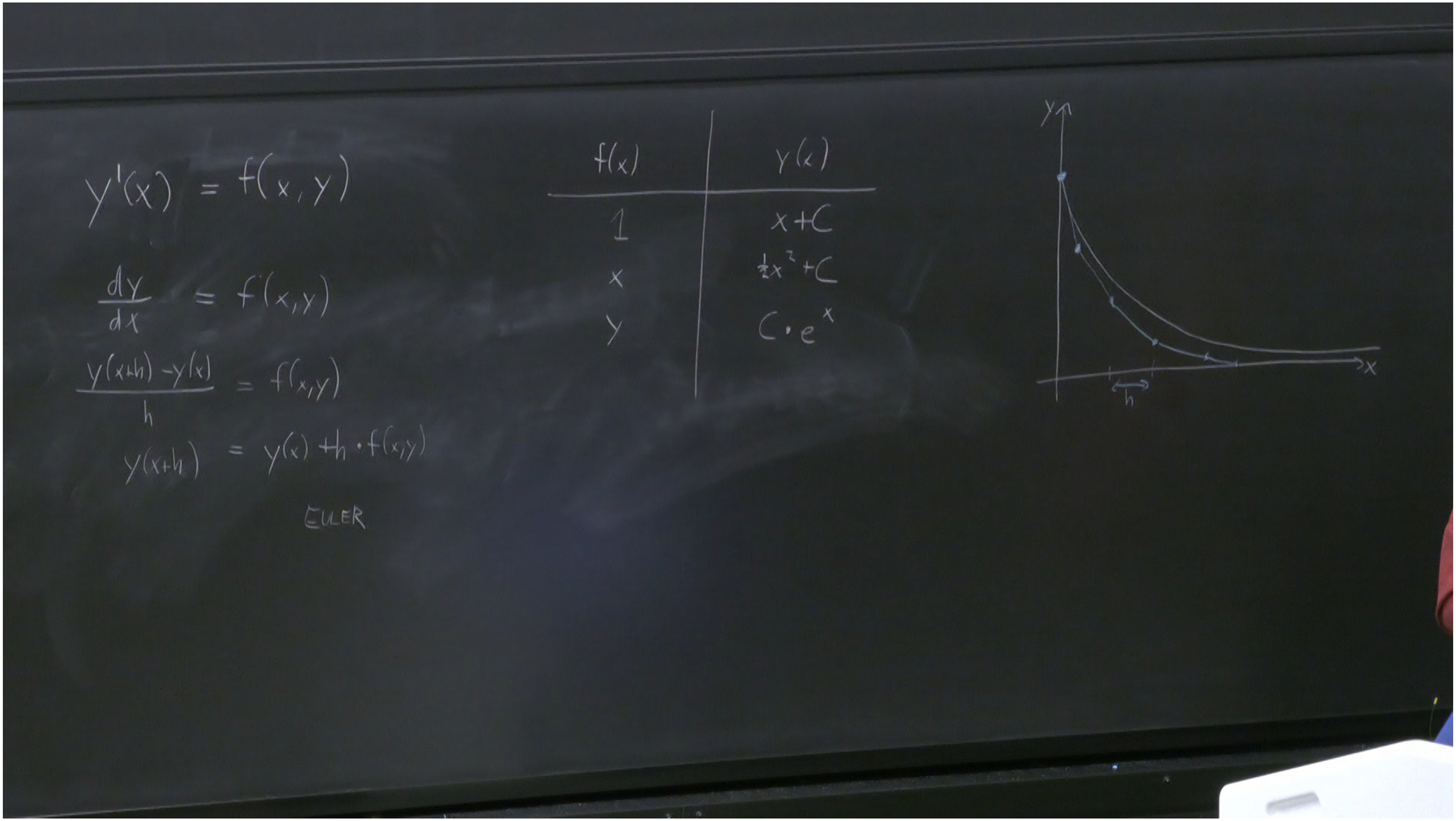
Code Julia 1.10.0

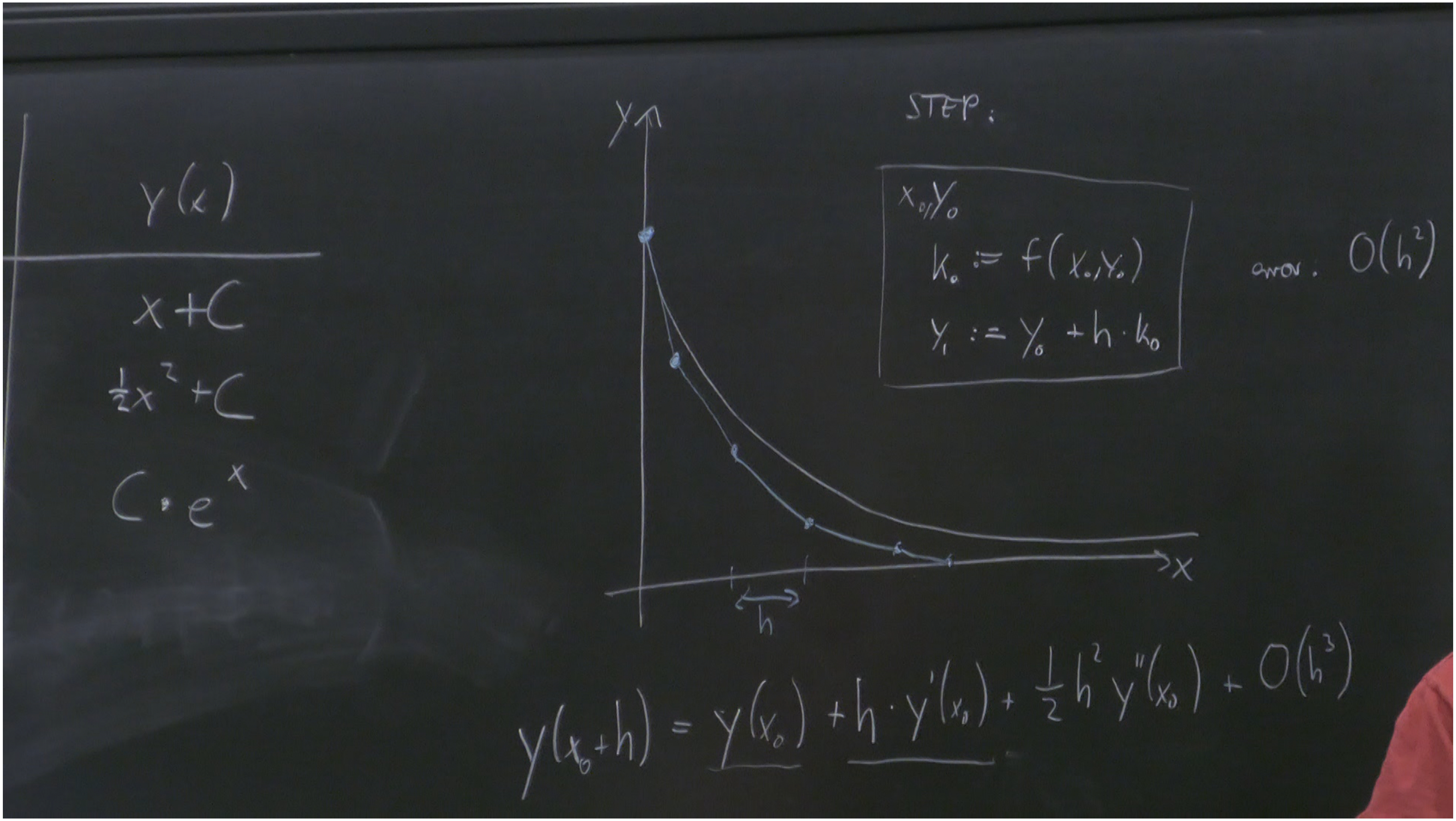
Ordinary Differential Equations (ODEs)

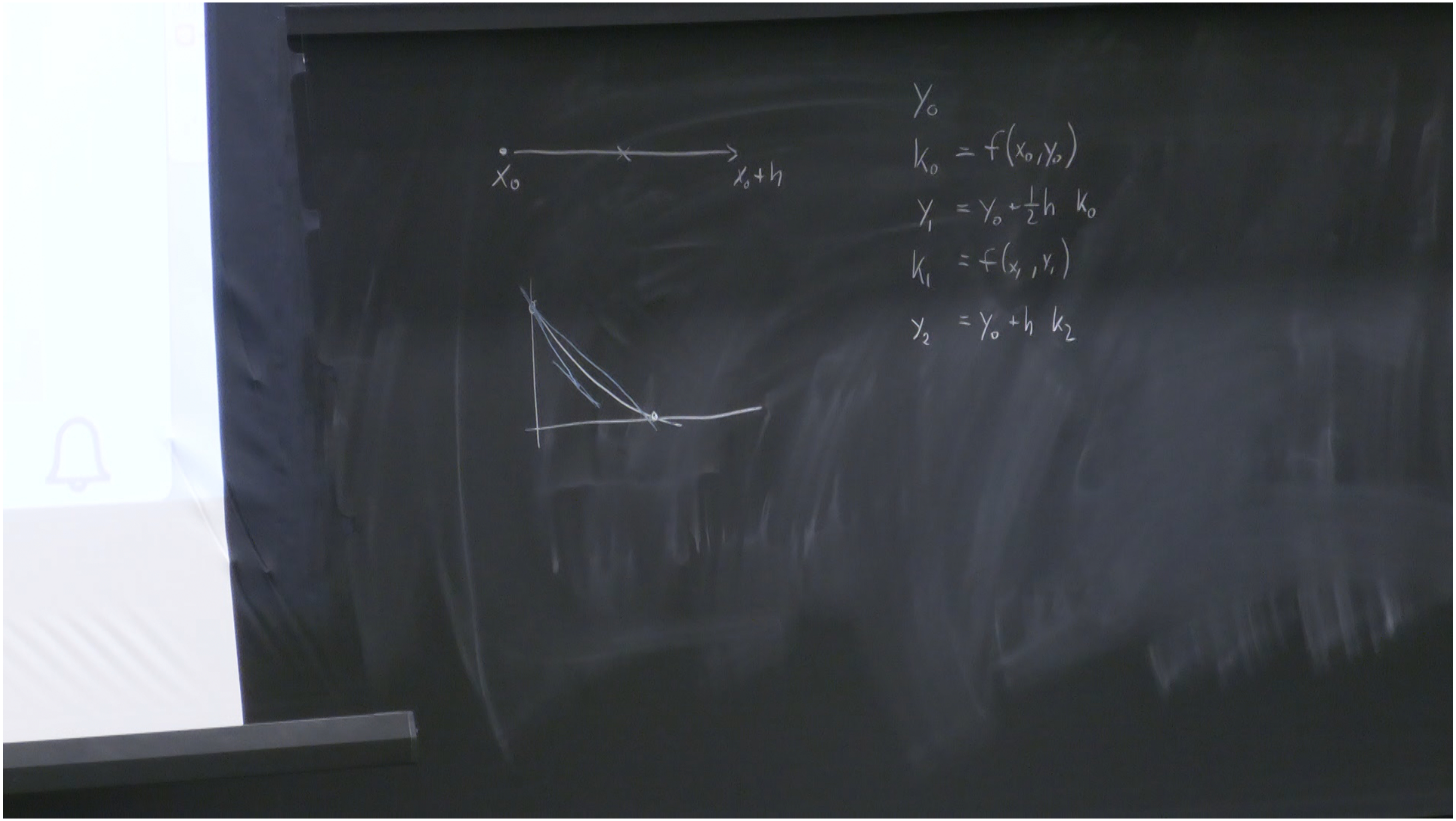
[3]:
```julia
using DifferentialEquations
```

[ ]:

Simple 0 5 Julia 1.10.0 | ... Mode: Comm... Ln 1, Col... Ordinary Differential Equations... 0


y'(x) = f(x, y)
dy/dx = f(x,y)
(y(x+h) - y(x))/h = f(x,y)
y(x+h) = y(x) + h · f(x,y)
EULER
f(x) | y(x)
1 | x+C
x | ½x²+C
y | C · e^x
y
x
h


y(x)
x+C
½x²+C
C·e^x
y
x
h
STEP:
x0,y0
k0 := f(x0,y0)
y1 := y0 + h·k0
error: O(h²)
y(x0+h) = y(x0) + h·y'(x0) + ½h²y''(x0) + O(h³)


x_0
$x_0 + h$
y_0
$k_0 = f(x_0, y_0)$
$y_1 = y_0 + \frac{1}{2} h \ k_0$
$k_1 = f(x_1, y_1)$
$y_2 = y_0 + h \ k_2$

$$= f(x_0, y_0)$$

$$= y_0 + \tfrac{1}{2} h \, k_0$$

$$= f(x_1, y_1)$$

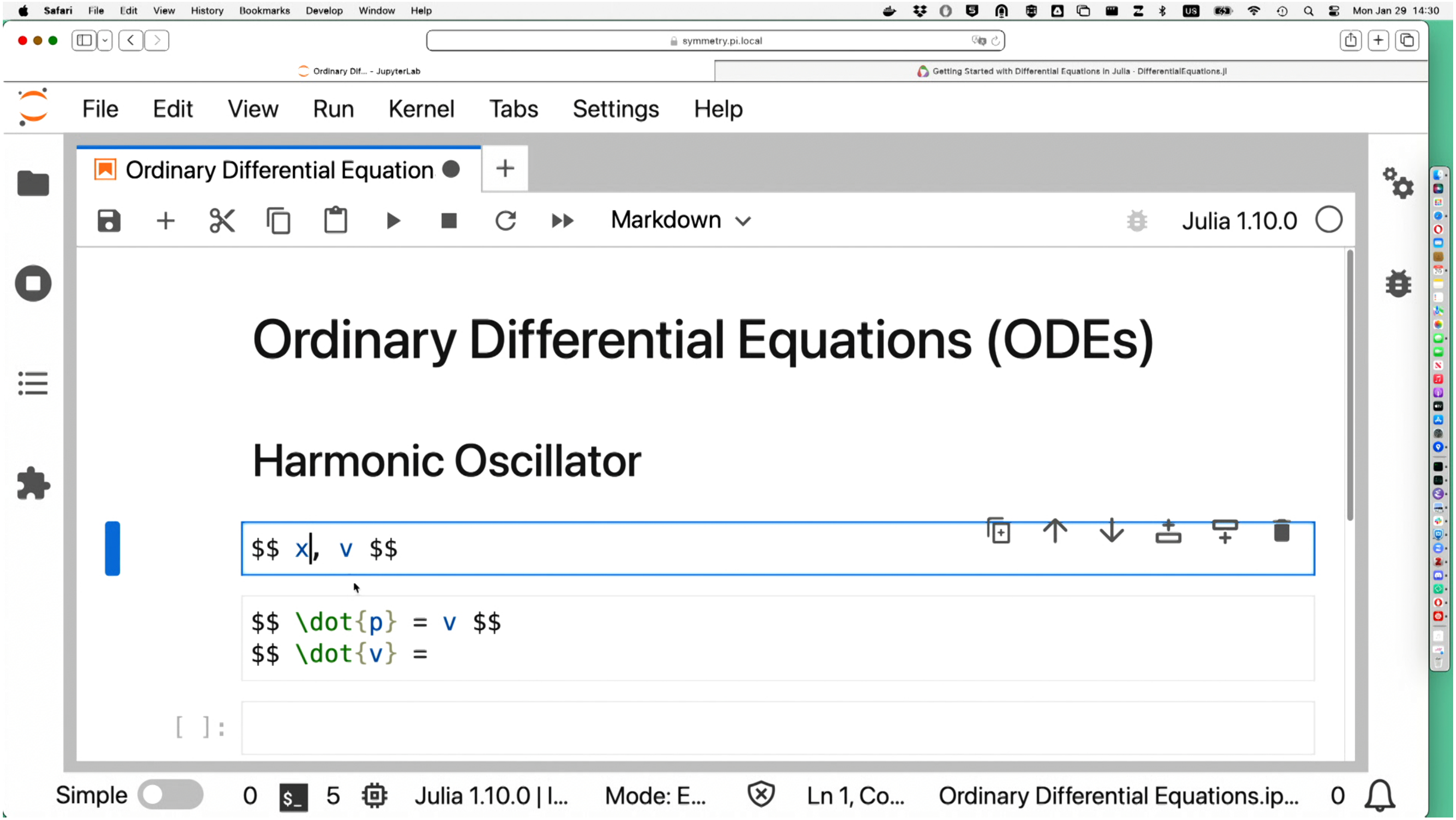
$$= y_0 + h \, k_1$$

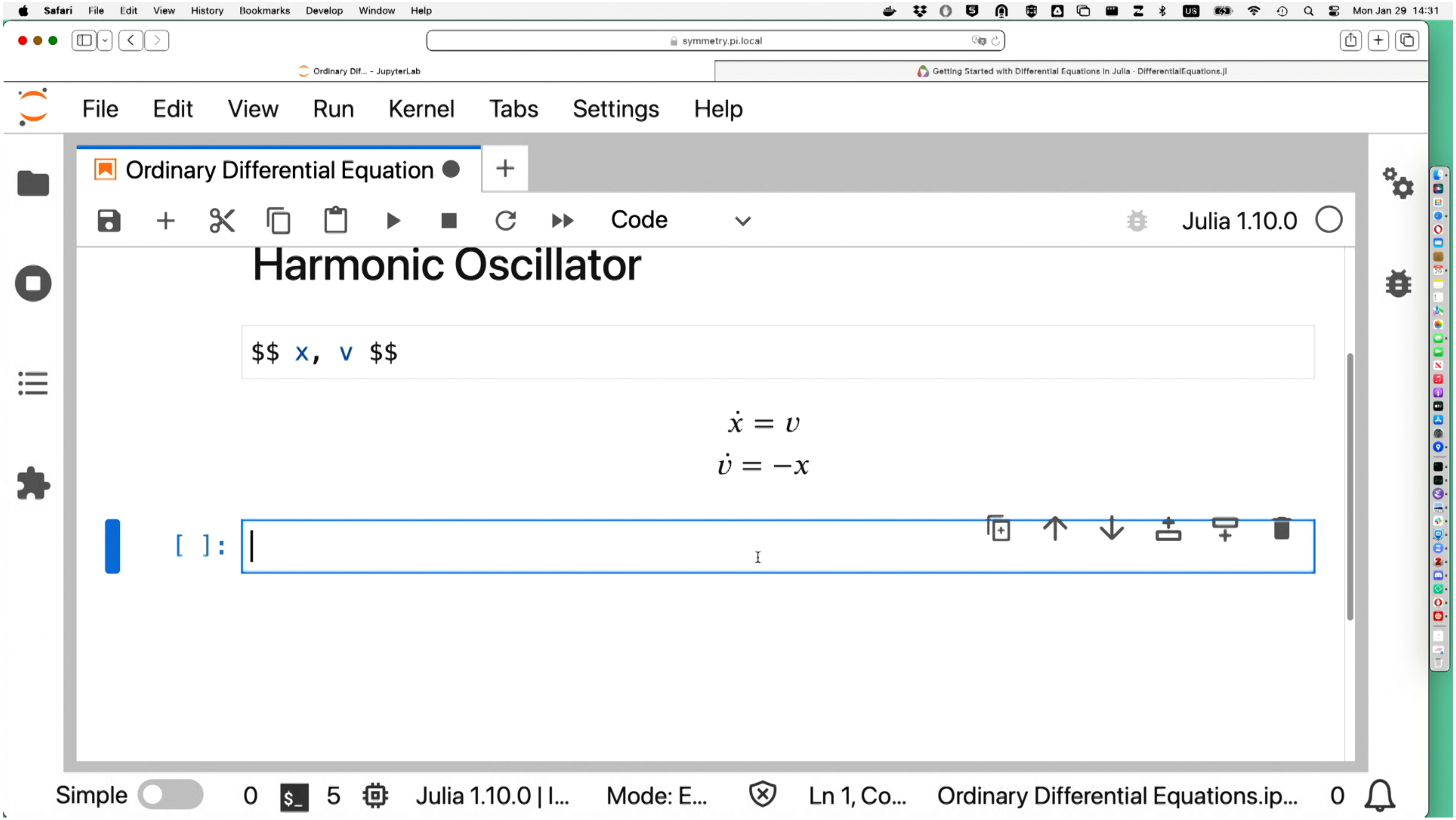
$$y(x_0 + \tfrac{1}{2}h) = y_0 + \tfrac{1}{2} h \, y_0' + \tfrac{1}{8} h^2 \, y_0'' + O(h^3)$$

$$y_1 = y(x_0 + \tfrac{1}{2}h) + O(h^2) = y_0 + \tfrac{1}{2} h \, y_0' + O(h^2)$$

$$k_1 = y'(x_0 + \tfrac{1}{2}h) + O(h^2) = y_0' + \tfrac{1}{2} h \, y_0'' + O(h^2)$$

$$y_2 = y_0 + h \, k_1 = \underbrace{y_0 + h \, y_0' + \tfrac{1}{2} h^2 \, y_0''} + O(h^3)$$


symmetry.pi.local
Ordinary Dif... - JupyterLab
Getting Started with Differential Equations in Julia · DifferentialEquations.jl
File
Edit
View
Run
Kernel
Tabs
Settings
Help
Ordinary Differential Equation
Markdown
Julia 1.10.0
Ordinary Differential Equations (ODEs)
Harmonic Oscillator
$$ x, v $$
$$ \dot{p} = v $$
$$ \dot{v} =
[ ]:
Simple
0
5
Julia 1.10.0 | I...
Mode: E...
Ln 1, Co...
Ordinary Differential Equations.ip...
0


Safari File Edit View History Bookmarks Develop Window Help
Mon Jan 29 14:31
symmetry.pi.local
Ordinary Dif... - JupyterLab
Getting Started with Differential Equations in Julia · DifferentialEquations.jl
File Edit View Run Kernel Tabs Settings Help
Ordinary Differential Equation
Code
Julia 1.10.0
Harmonic Oscillator
$$ x, v $$
$$\dot{x} = v$$
$$\dot{v} = -x$$
[ ]:
Simple
0
5
Julia 1.10.0 | I...
Mode: E...
Ln 1, Co...
Ordinary Differential Equations.ip...
0

Safari File Edit View History Bookmarks Develop Window Help Mon Jan 29 14:33

symmetry.pi.local

Ordinary Dif... - JupyterLab | Getting Started with Differential Equations in Julia · DifferentialEquations.jl

File Edit View Run Kernel Tabs Settings Help

Ordinary Differential Equatio ●

Code | Julia 1.10.0

Ordinary Differential Equations (ODEs)

Harmonic Oscillator

$$x, v$$

$$\dot{x} = v$$
$$\dot{v} = -x$$

• [6]:

```
function fho(U::Vector)

end
```

```
ParseError:
# Error @ ]8;;file:///home/eschnetter/In[6]#1:16\In[6]:1:16]8;;\
function fho(U)
#              └ ── premature end of input

Stacktrace:
 [1] top-level scope
   @ In[6]:1
```

[ ]:

Simple 0 5 Julia 1.10.0 | Idle Mode: Edit Ln 3, Col 4 Ordinary Differential Equations.ipynb 0

Safari File Edit View History Bookmarks Develop Window Help Mon Jan 29 14:36

symmetry.pi.local

Ordinary Dif... - JupyterLab | Getting Started with Differential Equations in Julia · DifferentialEquations.jl

File Edit View Run Kernel Tabs Settings Help

Ordinary Differential Equatio ● +

Code | Julia 1.10.0

```
    return udot
end
```

```
[7]: fho (generic function with 1 method)
```

```
[8]: fho([1, 0])
```

```
[8]: 2-element Vector{Int64}:
  0
 -1
```

```
[9]: function euler(f, y₀, h)
         k₀ = f(y₀)
         y₁ = y₀ + h*k₀
         return y₁
     end
```

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

```
[10]: euler(fho, [1,0], 0.1)
```

```
[10]: 2-element Vector{Float64}:
  1.0
 -0.1
```

```
[ ]:
```

Simple 0 5 Julia 1.10.0 | Idle Mode: Edit Ln 1, Col 1 Ordinary Differential Equations.ipynb 0

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

[10]: euler(fho, [1,0], 0.1)

[10]: 2-element Vector{Float64}:
  1.0
 -0.1

[11]: U = [1,0]
h = 0.1
Uall = [U]
for n in 1:10
    U = euler(fho, U, h)
    push!(Uall, U)
end
Uall

InexactError: Int64(-0.1)

Stacktrace:
  [1] Int64
    @ ./float.jl:909 [inlined]
  [2] convert
    @ ./number.jl:7 [inlined]
  [3] setindex!(A::Array{T}, x::Any, i1::Int64) where T
    @ Base ./array.jl:1021 [inlined]
  [4] _unsafe_copyto!(dest::Vector{Int64}, doffs::Int64, src::Vector{Float64}, soffs::Int64, n::Int64)
    @ Base ./array.jl:299
  [5] unsafe_copyto!
    @ ./array.jl:353 [inlined]
```

Safari File Edit View History Bookmarks Develop Window Help Mon Jan 29 14:38

symmetry.pi.local

Ordinary Dif... - JupyterLab | Getting Started with Differential Equations in Julia · DifferentialEquations.jl

File Edit View Run Kernel Tabs Settings Help

Ordinary Differential Equatio

Code Julia 1.10.0

```
 1.0
 -0.1
```

[12]:

```julia
U = [1.0, 0.0]
h = 0.1
Uall = [U]
for n in 1:10
    U = euler(fho, U, h)
    push!(Uall, U)
end
Uall
```

[12]:

```
11-element Vector{Vector{Float64}}:
 [1.0, 0.0]
 [1.0, -0.1]
 [0.99, -0.2]
 [0.97, -0.29900000000000004]
 [0.9400999999999999, -0.396]
 [0.9005, -0.49001]
 [0.851499, -0.58006]
 [0.793493, -0.6652099]
 [0.72697201, -0.7445592000000001]
 [0.65251609, -0.8172564010000001]
 [0.5707904498999999, -0.8825080100000001]
```

[ ]:

Simple 0 5 Julia 1.10.0 | Idle Mode: Command Ln 1, Col 1 Ordinary Differential Equations.ipynb 0

```
end
Uall
```

```
[13]: 21-element Vector{Vector{Float64}}:
 [1.0, 0.0]
 [1.0, -0.1]
 [0.99, -0.2]
 [0.97, -0.29900000000000004]
 [0.9400999999999999, -0.396]
 [0.9005, -0.49001]
 [0.851499, -0.58006]
 [0.793493, -0.6652099]
 [0.72697201, -0.7445592000000001]
 [0.65251609, -0.8172564010000001]
 [0.5707904498999999, -0.8825080100000001]
 [0.4825396488999999, -0.9395870549900001]
 [0.3885809434009999, -0.9878410198800001]
 [0.2897968414129999, -1.0266991142201]
 [0.18712692999098987, -1.0556787983614]
 [0.08155905015484986, -1.074391491360499]
 [-0.025880098981200042, -1.082547396375984]
 [-0.13413483861879844, -1.079959386477864]
 [-0.24213077726658483, -1.0665459026159843]
 [-0.34878536752818323, -1.0423328248893258]
 [-0.45301865001711583, -1.0074542881365074]
```

Safari File Edit View History Bookmarks Develop Window Help Mon Jan 29 14:40

symmetry.pi.local

Ordinary Dif... - JupyterLab | Getting Started with Differential Equations in Julia · DifferentialEquations.jl

File Edit View Run Kernel Tabs Settings Help

Ordinary Differential Equatio ● | +

Code | Julia 1.10.0

```
-0.1
```

[16]:

```julia
U = [1.0, 0.0]
h = 0.1/2
Uall = [U]
for n in 1:30*2
    U = euler(fho, U, h)
    push!(Uall, U)
end
```

```
[[1.0, 0.0], [1.0, -0.05], [0.9975, -0.1], [0.9925, -0.149875], [0.9850062500000001, -0.1995], [0.9750312500000001, -0.2487503125000000
1], [0.9625937343750002, -0.297501875], [0.9477186406250002, -0.34563156171875004], [0.9304370625390627, -0.39301749375000006], [0.9107
861878515626, -0.4395393468769532], [0.8888092205077149, -0.4850786562695313], [0.8645552876942384, -0.5295191172949171], [0.8380793318
294926, -0.572746881679629], [0.8094419877455111, -0.6146508482711036], [0.778709445331956, -0.6551229476583792], [0.745953297949037, -
0.694058419924977], [0.7112503769527881, -0.7313560848224289], [0.6746825727116667, -0.7669186036700683], [0.6363366425281632, -0.80065
27323056516], [0.5963040059128807, -0.8324695644320598], [0.5546805276912777, -0.8622847647277038], [0.5115662894548926, -0.89001879111
22677], [0.46706534989927917, -0.9155971055850123], [0.42128549462002857, -0.9389503730799762], [0.37433797596602975, -0.96001464781097
77], [0.3263372435754809, -0.9787315466092792], [0.2774006662450169, -0.9950484087880532], [0.22764824580561424, -1.0089184421003041],
[0.17720232370059902, -1.020300854390585], [0.12618728098106977, -1.0291609705756148], [0.07472923245228902, -1.0354703346246683], [0.0
229557157210556, -1.0392067962472828], [-0.029004624091308544, -1.0403545820333355], [-0.08102235319297532, -1.03890435082877], [-0.132
96757073441384, -1.0348532331691214], [-0.1847102323928699, -1.0282048546324007], [-0.23612047512448994, -1.0189693430127573], [-0.2870
689422751278, -1.0071633192565328], [-0.33742710823795447, -0.9928098721427764], [-0.38706760184509326, -0.9759385167308786], [-0.43586
45276816372, -0.956585136638624], [-0.48369378451356837, -0.9347919102545421], [-0.5304333800262955, -0.9106072210288637], [-0.57596374
10777386, -0.8840855520275489], [-0.6201680186791161, -0.855287364973662], [-0.6629323869277992, -0.8242789640397061], [-0.704146335129
7845, -0.7911323446933162], [-0.7437029523644504, -0.7559250279368269], [-0.7814992037612918, -0.7187398803186044], [-0.81743619777722
2, -0.6796649201305398], [-0.851419443783749, -0.6387931102416787], [-0.8833590992958329, -0.5962221380524912], [-0.9131702061984575, -
0.5520541830876996], [-0.9407729153528425, -0.5063956727777768], [-0.9660926989917313, -0.45935702701013464], [-0.989060550342238, -0.4
1105239206054806], [-1.0096131699452655, -0.36159936454343616], [-1.0276931381724372, -0.31111870604617287], [-1.043249073474746, -0.25
9734049137551], [-1.0562357759316234, -0.2075715954638137], [-1.0666143557048142, -0.15475980666723255]]
```

[ ]:

Display a menu | 0 | 5 | Julia 1.10.0 | Idle | Mode: Edit | Ln 8, Col 1 | Ordinary Differential Equations.ipynb | 0

Safari File Edit View History Bookmarks Develop Window Help Mon Jan 29 14:41

symmetry.pi.local

Ordinary Dif... - JupyterLab | Getting Started with Differential Equations in Julia · DifferentialEquations.jl

File Edit View Run Kernel Tabs Settings Help

Ordinary Differential Equatio

Code | Julia 1.10.0

```
Uall = [U]
for n in 1:30*2
    U = euler(fho, U, h)
    push!(Uall, U)
end
```

[19]:
```
for U in Uall
    println(round(U; sigdig=3))
end
```

```
MethodError: no method matching round(::Vector{Float64}; sigdig::Int64)

Closest candidates are:
  round(::Type{Bool}, ::AbstractFloat) got unsupported keyword argument "sigdig"
   @ Base float.jl:391
  round(::Type{BigInt}, ::BigFloat, ::RoundingMode) got unsupported keyword argument "sigdig"
   @ Base mpfr.jl:353
  round(::Type{BigInt}, ::BigFloat, ::Union{Base.MPFR.MPFRRoundingMode, RoundingMode}) got unsupported keyword argument "sigdig"
   @ Base mpfr.jl:344
  ...


Stacktrace:
 [1] top-level scope
   @ ./In[19]:2
```

[ ]:
```
?
```

Simple 0 5 Julia 1.10.0 | Idle Mode: Edit Ln 1, Col 2 Ordinary Differential Equations.ipynb 0

`RoundingModeReal` and `RoundingModeImaginary` default to `RoundNearest`, which rounds to the nearest integer, with ties (fractional values of 0.5) being rounded to the nearest even integer.

Example

```
julia> round(3.14 + 4.5im)
3.0 + 4.0im

julia> round(3.14 + 4.5im, RoundUp, RoundNearestTiesUp)
4.0 + 5.0im

julia> round(3.14159 + 4.512im; digits = 1)
3.1 + 4.5im

julia> round(3.14159 + 4.512im; sigdigits = 3)
3.14 + 4.51im
```

```
round([T,] x, [r::RoundingMode])
round(x, [r::RoundingMode]; digits::Integer=0, base = 10)
round(x, [r::RoundingMode]; sigdigits::Integer, base = 10)
```

Rounds the number `x`.

Without keyword arguments, `x` is rounded to an integer value, returning a value of type `T`, or of the same type of `x` if no `T` is provided. An `InexactError` will be thrown if the value is not representable by `T`, similar to `convert`.

If the `digits` keyword argument is provided, it rounds to the specified number of digits after the decimal place (or before if negative), in base `base`.

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

[10]: euler(fho, [1,0], 0.1)

[10]: 2-element Vector{Float64}:
  1.0
 -0.1

[17]: U = [1.0, 0.0]
h = 0.1/2
Uall = [U]
for n in 1:30*2
    U = euler(fho, U, h)
    push!(Uall, U)
end

[22]: function 
for U in Uall
    println(round.(U; sigdigits=3))
end

[1.0, 0.0]
[1.0, -0.05]
[0.998, -0.1]
[0.992, -0.15]
[0.985, -0.2]
[0.975, -0.249]
[0.963, -0.298]
[0.948, -0.346]
[0.93, -0.393]
[0.911, -0.44]
```

Safari File Edit View History Bookmarks Develop Window Help Mon Jan 29 14:45

symmetry.pi.local

Ordinary Dif... - JupyterLab | Getting Started with Differential Equations in Julia · DifferentialEquations.jl

File Edit View Run Kernel Tabs Settings Help

Ordinary Differential Equatio

Code Julia 1.10.0

```
[0.825, -0.566]
[0.764, -0.645]
[0.696, -0.718]
[0.621, -0.784]
[0.539, -0.842]
[0.452, -0.892]
[0.361, -0.933]
[0.265, -0.964]
[0.168, -0.986]
[0.0683, -0.998]
[-0.0319, -1.0]
[-0.132, -0.992]
[-0.23, -0.973]
[-0.326, -0.945]
[-0.419, -0.908]
[-0.508, -0.862]
[-0.592, -0.807]
[-0.669, -0.743]
[-0.74, -0.673]
[-0.804, -0.595]
[-0.859, -0.512]
[-0.906, -0.423]
[-0.944, -0.331]
[-0.972, -0.235]
[-0.991, -0.136]
```

[ ]:

Simple 0 5 Julia 1.10.0 | Idle Mode: Edit Ln 1, Col 1 Ordinary Differential Equations.ipynb 0

Safari File Edit View History Bookmarks Develop Window Help Mon Jan 29 14:46

symmetry.pi.local

Ordinary Dif... - JupyterLab | Getting Started with Differential Equations in Julia · DifferentialEquations.jl

File Edit View Run Kernel Tabs Settings Help

Ordinary Differential Equatior

Code Julia 1.10.0

```
[0.0683, -0.998]
[-0.0319, -1.0]
[-0.132, -0.992]
[-0.23, -0.973]
[-0.326, -0.945]
[-0.419, -0.908]
[-0.508, -0.862]
[-0.592, -0.807]
[-0.669, -0.743]
[-0.74, -0.673]
[-0.804, -0.595]
[-0.859, -0.512]
[-0.906, -0.423]
[-0.944, -0.331]
[-0.972, -0.235]
[-0.991, -0.136]
[-1.0, -0.0364]
```

[ ]:

Simple 0 5 Julia 1.10.0 | Idle Mode: Command Ln 1, Col 1 Ordinary Differential Equations.ipynb 0

Safari File Edit View History Bookmarks Develop Window Help Mon Jan 29 14:46

symmetry.pi.local

Ordinary Dif... - JupyterLab | Getting Started with Differential Equations in Julia · DifferentialEquations.jl

File Edit View Run Kernel Tabs Settings Help

Ordinary Differential Equation

Code | Julia 1.10.0

```
[-0.417, -0.909]
[-0.462, -0.887]
[-0.506, -0.863]
[-0.548, -0.836]
[-0.589, -0.808]
[-0.629, -0.778]
[-0.667, -0.745]
[-0.703, -0.711]
[-0.738, -0.675]
[-0.771, -0.637]
[-0.802, -0.598]
[-0.831, -0.557]
[-0.857, -0.515]
[-0.882, -0.471]
[-0.905, -0.426]
[-0.925, -0.381]
[-0.943, -0.334]
[-0.958, -0.286]
[-0.971, -0.238]
[-0.982, -0.189]
[-0.99, -0.14]
[-0.996, -0.0902]
```

[ ]:

Simple 0 5 Julia 1.10.0 | Idle Mode: Command Ln 1, Col 1 Ordinary Differential Equations.ipynb 0

Safari File Edit View History Bookmarks Develop Window Help Mon Jan 29 14:48

symmetry.pi.local

Ordinary Dif... - JupyterLab | Getting Started with Differential Equations in Julia · DifferentialEquations.jl

File Edit View Run Kernel Tabs Settings Help

Ordinary Differential Equation

Code | Julia 1.10.0

[7]:
```
function fho(U::Vector)
    x, v = U
    xdot = v
    vdot = -x
    Udot = [xdot, vdot]
    return Udot
end
```

[7]: fho (generic function with 1 method)

[8]:
```
fho([1, 0])
```

[8]:
```
2-element Vector{Int64}:
  0
 -1
```

[ ]:

[9]:
```
function euler(f, y₀, h)
    k₀ = f(y₀)
    y₁ = y₀ + h*k₀
    return y₁
end
```

[9]: euler (generic function with 1 method)

[10]:
```
euler(fho, [1,0], 0.1)
```

[10]:
```
2-element Vector{Float64}:
  1.0
```

Simple 0 5 Julia 1.10.0 | Idle Mode: Edit Ln 1, Col 1 Ordinary Differential Equations.ipynb 0

Safari File Edit View History Bookmarks Develop Window Help Mon Jan 29 14:49

docs.sciml.ai

Ordinary Dif... - JupyterLab | Getting Started with Differential Equations in Julia · DifferentialEquations.jl

HOME MODELING SOLVERS ANALYSIS MACHINE LEARNING DEVELOPER TOOLS Search...

DifferentialEquations.jl

Search docs (Ctrl + /)

Version v7.8.0

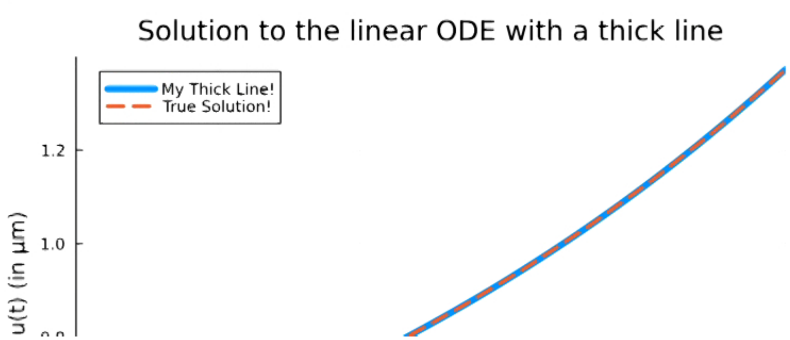
on the time interval $t \in [0,1]$ where $f(u,p,t) = \alpha u$. Here, u is the current state variable, p is our parameter variable (containing things like a reaction rate or the constant of gravity), and t is the current time.

(In our example, we know by calculus that the solution to this equation is $u(t) = u_0 \exp(\alpha t)$, but we will use DifferentialEquations.jl to solve this problem *numerically*, which is essential for problems where a symbolic solution is not known.)

The general workflow is to define a problem, solve the problem, and then analyze the solution. The full code for solving this problem is:

```
using DifferentialEquations
f(u, p, t) = 1.01 * u
u0 = 1 / 2
tspan = (0.0, 1.0)
prob = ODEProblem(f, u0, tspan)
sol = solve(prob, Tsit5(), reltol = 1e-8, abstol = 1e-8)

using Plots
plot(sol, linewidth = 5, title = "Solution to the linear ODE with a thick line",
    xaxis = "Time (t)", yaxis = "u(t) (in μm)", label = "My Thick Line!") # legend=false
plot!(sol.t, t -> 0.5 * exp(1.01t), lw = 3, ls = :dash, label = "True Solution!")
```



Safari File Edit View History Bookmarks Develop Window Help Mon Jan 29 14:50

docs.sciml.ai

Ordinary Dif... - JupyterLab | DifferentialEquations.jl: Efficient Differential Equation Solving in Julia · DifferentialEquations.jl

HOME MODELING SOLVERS ANALYSIS MACHINE LEARNING DEVELOPER TOOLS

DifferentialEquations.jl

Search docs (Ctrl + /)

Differential Equation Solving in Julia

- Contributing
- Supporting and Citing
- Getting Started: Installation And First Steps
- Acknowledgements
- Reproducibility

Version v7.8.0

DifferentialEquations.jl: Efficient Differential Equation Solving in Julia

Search... GitHub

DifferentialEquations.jl: Efficient Differential Equation Solving in Julia

This is a suite for numerically solving differential equations written in Julia and available for use in Julia, Python, and R. The purpose of this package is to supply efficient Julia implementations of solvers for various differential equations. Equations within the realm of this package include:

- Discrete equations (function maps, discrete stochastic (Gillespie/Markov) simulations)
- Ordinary differential equations (ODEs)
- Split and Partitioned ODEs (Symplectic integrators, IMEX Methods)
- Stochastic ordinary differential equations (SODEs or SDEs)
- Stochastic differential-algebraic equations (SDAEs)
- Random differential equations (RODEs or RDEs)
- Differential algebraic equations (DAEs)
- Delay differential equations (DDEs)
- Neutral, retarded, and algebraic delay differential equations (NDDEs, RDDEs, and DDAEs)

Safari File Edit View History Bookmarks Develop Window Help Mon Jan 29 14:53

docs.sciml.ai

Ordinary Dif... - JupyterLab | ODE Problems · DifferentialEquations.jl

HOME MODELING SOLVERS ANALYSIS MACHINE LEARNING DEVELOPER TOOLS

DifferentialEquations.jl

Search docs (Ctrl + /)

- ODE Problems
  - Solution Type
  - Example Problems
- Non-autonomous Linear ODE / Lie Group Problems
- Dynamical, Hamiltonian and 2nd Order ODE Problems

Version v7.8.0

Search...

```
# Test that it worked
using OrdinaryDiffEq
sol = solve(prob,Tsit5())
using Plots; plot(sol,vars=(1,2,3))
```

More Example Problems

Example problems can be found in DiffEqProblemLibrary.jl.

To use a sample problem, such as `prob_ode_linear`, you can do something like:

```
#] add ODEProblemLibrary
using ODEProblemLibrary
prob = ODEProblemLibrary.prob_ode_linear
sol = solve(prob)
```

source

SciMLBase.ODEFunction — Type

```
ODEFunction{iip,F,TMM,Ta,Tt,TJ,JVP,VJP,JP,SP,TW,TWt,TPJ,S,S2,S3,O,TCV} <: AbstractC
```

A representation of an ODE function `f`, defined by:

Safari File Edit View History Bookmarks Develop Window Help Mon Jan 29 14:54

symmetry.pi.local

Ordinary Dif... - JupyterLab | Getting Started with Differential Equations in Julia - DifferentialEquations.jl

File Edit View Run Kernel Tabs Settings Help

Ordinary Differential Equatio ● +

Code | Julia 1.10.0

```
[-0.771, -0.637]
[-0.802, -0.598]
[-0.831, -0.557]
[-0.857, -0.515]
[-0.882, -0.471]
[-0.905, -0.426]
[-0.925, -0.381]
[-0.943, -0.334]
[-0.958, -0.286]
[-0.971, -0.238]
[-0.982, -0.189]
[-0.99, -0.14]
[-0.996, -0.0902]
```

Using a package

[36]:
```julia
using DifferentialEquations
```

[ ]:
```julia
prob = ODEProblem(f, u0, tspan)
```

Simple 0 5 Julia 1.10.0 | Idle Mode: Edit Ln 1, Col 20 Ordinary Differential Equations.ipynb 0

Safari File Edit View History Bookmarks Develop Window Help Mon Jan 29 14:56

symmetry.pi.local

Ordinary Dif... - JupyterLab | Getting Started with Differential Equations in Julia · DifferentialEquations.jl

File Edit View Run Kernel Tabs Settings Help

Ordinary Differential Equatio ● +

Code Julia 1.10.0

```
[-0.771, -0.637]
[-0.802, -0.598]
[-0.831, -0.557]
[-0.857, -0.515]
[-0.882, -0.471]
[-0.905, -0.426]
[-0.925, -0.381]
[-0.943, -0.334]
[-0.958, -0.286]
[-0.971, -0.238]
[-0.982, -0.189]
[-0.99, -0.14]
[-0.996, -0.0902]
```

Using a package

[36]:
```julia
using DifferentialEquations
```

[ ]:
```julia
U0 = [1.0, 0.0]
prob = ODEProblem((U,p,t) -> fho(U), U0, (0.0, 3.14))
sol = solve(prob)
```

Simple 0 5 Julia 1.10.0 | Idle Mode: Edit Ln 2, Col 19 Ordinary Differential Equations.ipynb 0

Ordinary Differential Equations.ipynb — Julia 1.10.0

```
 0.0
 0.0009990009990009992
 0.010989010989010992
 0.07985922249873038
 0.2403882280626971
 0.48125583199780264
 0.7872724750204353
 1.1770558023021032
 1.630266513673432
 2.1040018658103854
 2.6513321708517577
 3.14
u: 12-element Vector{Vector{Float64}}:
 [1.0, 0.0]
 [0.9999995009985435, -0.0009990000832833342]
 [0.9999396214263468, -0.010988789821184269]
 [0.9968129466114029, -0.07977436592568168]
 [0.9712456180254766, -0.23807970964822978]
 [0.8864142948207253, -0.4628927349710369]
 [0.7057801319307302, -0.7084309070362929]
 [0.3836450082688139, -0.9234805642321077]
 [-0.05943639522180999, -0.9982319800633562]
 [-0.5082985573738581, -0.8611807771069901]
 [-0.8822130995277943, -0.47085082387073185]
 [-0.9999989945731433, -0.0015855238215814471]
```