Jupyter Notebook Execution Report

Name: Your Name

Date: November 18, 2025

Cell 1: 🟦 Code

```
# --- Cell 1: Setup ---
```

```
using DifferentialEquations

using Flux

using Plots

using Optimization

using OptimizationOptimisers

using Zygote

using DataFrames
```

```
# Hodgkin-Huxley Model Parameters (Global Constants)

const Cm = 1.0 # µF/cm^2

const g_Na = 120.0 # mS/cm^2

const g_K = 36.0 # mS/cm^2

const g_L = 0.3 # mS/cm^2

const E_Na = 50.0 # mV

const E_K = -77.0 # mV

const E_L = -54.387 # mV
```

Error:

```
Traceback (most recent call last):
  File "c:\Users\Admin\.vscode\extensions\ganeshkumbhar.nb2pdf-1.1.9\scripts\nb2pdf.py", line 401
    exec('\n'.join(lines[:-1]), glb)
  File "&lt;string&gt;", line 3
    using DifferentialEquations
          ^^^^^^^^^^^^^^^^^^^^^
SyntaxError: invalid syntax
During handling of the above exception, another exception occurred:
Traceback (most recent call last):
  File "c:\Users\Admin\.vscode\extensions\ganeshkumbhar.nb2pdf-1.1.9\scripts\nb2pdf.py", line 408
    exec(source, glb)
  File "&lt;string&gt;", line 3
```

```
    using DifferentialEquations
          ^^^^^^^^^^^^^^^^^^^^^
SyntaxError: invalid syntax
```

Cell 2: ■ Code

```
# --- Cell 2: Known Physics & Stimulus ---
```

```
# Voltage-gated ion channel kinetics
α_n(V) = 0.01 * (V + 55) / (1 - exp(-(V + 55) / 10))
β_n(V) = 0.125 * exp(-(V + 65) / 80)
α_m(V) = 0.1 * (V + 40) / (1 - exp(-(V + 40) / 10))
β_m(V) = 4.0 * exp(-(V + 65) / 18)
α_h(V) = 0.07 * exp(-(V + 65) / 20)
β_h(V) = 1 / (1 + exp(-(V + 35) / 10))
```

```
# Steady-state & time-constant functions for the 2D model
m_inf(V) = α_m(V) / (α_m(V) + β_m(V))
h_inf(V) = α_h(V) / (α_h(V) + β_h(V))
n_inf(V) = α_n(V) / (α_n(V) + β_n(V))
tau_n(V) = 1 / (α_n(V) + β_n(V))
```

```
# Stimulus protocol: a short, sharp pulse
function stimulus(t; start=10.0, duration=1.0, amplitude=20.0)
(start &lt;= t &lt; start + duration) ? amplitude : 0.0
end
```

Error:

```
Traceback (most recent call last):
  File "c:\Users\Admin\.vscode\extensions\ganeshkumbhar.nb2pdf-1.1.9\scripts\nb2pdf.py", line 401
    exec('\n'.join(lines[:-1]), glb)
  File "&lt;string&gt;", line 4
    α_n(V) = 0.01 * (V + 55) / (1 - exp(-(V + 55) / 10))
    ^^^^^^
SyntaxError: cannot assign to function call here. Maybe you meant '==' instead of '='?
During handling of the above exception, another exception occurred:
Traceback (most recent call last):
  File "c:\Users\Admin\.vscode\extensions\ganeshkumbhar.nb2pdf-1.1.9\scripts\nb2pdf.py", line 408
    exec(source, glb)
  File "&lt;string&gt;", line 4
    α_n(V) = 0.01 * (V + 55) / (1 - exp(-(V + 55) / 10))
    ^^^^^^
```

```
SyntaxError: cannot assign to function call here. Maybe you meant '==' instead of '='?
```

Cell 3: ■ Code

```
# ---- Hyperparameters / safety ----
nn_scale = 0.01 # scale factor for NN output (tune: 1e-3 .. 1e-1)
abstol = 1e-8
reltol = 1e-6
maxsteps = 1e7 # optional large step cap
grad_clip_norm = 5.0 # clip gradient L2 norm to this value
lr = 0.01 # learning rate for Adam
num_iters = 1000 # training iterations (or use your callback loop)
```

Cell 4: ■ Code

```
# --- Cell 3: Data Generation ---
```

```
# 2D Hodgkin-Huxley reduced model engine
function hodgkin_huxley_reduced!(du, u, p, t)
V, n = u
I_ext = stimulus(t)
I_Na = g_Na * m_inf(V)^3 * h_inf(V) * (V - E_Na)
I_K = g_K * n^4 * (V - E_K)
I_L = g_L * (V - E_L)
du[1] = (I_ext - I_Na - I_K - I_L) / Cm
du[2] = (n_inf(V) - n) / tau_n(V)
end
```

```
# Define and solve the 2D problem to generate training data
u■_reduced = Float64[-65.0, 0.3177]
tspan_reduced = (Float64(0.0), Float64(50.0))
prob_reduced = ODEProblem(hodgkin_huxley_reduced!, u■_reduced, tspan_reduced,
nothing)
sol_reduced = solve(prob_reduced, Tsit5(), saveat=0.1)
```

```
# Extract and structure the training data
training_data_2D = Array(sol_reduced)
timestamps_2D = Float64.(sol_reduced.t)
```

```
# (Optional) Verify data shape and content
df = DataFrame(t=timestamps_2D, V=training_data_2D[1, :], n=training_data_2D[2, :])
println("Generated Training Data:")
display(first(df, 5))
```

Error:

```
Traceback (most recent call last):
  File "c:\Users\Admin\.vscode\extensions\ganeshkumbhar.nb2pdf-1.1.9\scripts\nb2pdf.py", line 401
    exec('\n'.join(lines[:-1]), glb)
  File "&lt;string&gt;", line 15
    u■_reduced = Float64[-65.0, 0.3177]
     ^
SyntaxError: invalid character '■' (U+2080)
During handling of the above exception, another exception occurred:
Traceback (most recent call last):
  File "c:\Users\Admin\.vscode\extensions\ganeshkumbhar.nb2pdf-1.1.9\scripts\nb2pdf.py", line 408
    exec(source, glb)
  File "&lt;string&gt;", line 15
    u■_reduced = Float64[-65.0, 0.3177]
     ^
SyntaxError: invalid character '■' (U+2080)
```

Cell 5: ■ Code

```
# explicitly defining weights and biases to Float64
W1= randn(Float64, 10 ,1)
b1=zeros(Float64,10)
W2=randn(Float64,1,10)
b2=zeros(Float64,1)
```

```
# Creating the layer using provided Float64 arrays
layer1 = Dense(W1,b1, tanh)
layer2 = Dense(W2,b2)
# Define the Neural Network structure
U = Chain(layer1,layer2)
```

```
# Extract the trainable parameters (p_nn) and the re-structuring function (re)
p_nn, re = Flux.destructure(U)
```

```
# Define the UDE function with the embedded neural network
function ude_model!(du, u, p_nn, t)
V, n = u
```

```
# Neural network component to learn the unknown current
unknown_current = re(p_nn)([Float64(V)])[1]
```

```
# Known physics components
I_ext = stimulus(t)
I_K = g_K * n^4 * (V - E_K)
I_L = g_L * (V - E_L)
```

```
# The hybrid dynamics equation
du[1] = (I_ext + unknown_current - I_K - I_L) / Cm
du[2] = (n_inf(V) - n) / tau_n(V)
end
```

Error:

```
Traceback (most recent call last):
  File "c:\Users\Admin\.vscode\extensions\ganeshkumbhar.nb2pdf-1.1.9\scripts\nb2pdf.py", line 401
    exec('\n'.join(lines[:-1]), glb)
  File "&lt;string&gt;", line 11
    U = Chain(layer1,layer2)
IndentationError: unexpected indent
During handling of the above exception, another exception occurred:
Traceback (most recent call last):
  File "c:\Users\Admin\.vscode\extensions\ganeshkumbhar.nb2pdf-1.1.9\scripts\nb2pdf.py", line 408
    exec(source, glb)
  File "&lt;string&gt;", line 11
    U = Chain(layer1,layer2)
IndentationError: unexpected indent
```

Cell 6: ■ Code

```
# check parameter element types
println("Parameter element types:")
for param in Flux.params(U)
println(eltype(param), " size=", size(param))
end
```

```
# quick forward check
test_in = Float64[-65.0]
println("NN forward:", re(p_nn)(test_in))
```

Error:

```
Traceback (most recent call last):
  File "c:\Users\Admin\.vscode\extensions\ganeshkumbhar.nb2pdf-1.1.9\scripts\nb2pdf.py", line 401
    exec('\n'.join(lines[:-1]), glb)
  File "&lt;string&gt;", line 3
    for param in Flux.params(U)
                               ^
SyntaxError: expected ':'
During handling of the above exception, another exception occurred:
Traceback (most recent call last):
  File "c:\Users\Admin\.vscode\extensions\ganeshkumbhar.nb2pdf-1.1.9\scripts\nb2pdf.py", line 408
    exec(source, glb)
  File "&lt;string&gt;", line 3
    for param in Flux.params(U)
                               ^
SyntaxError: expected ':'
```

Cell 7: ■ Code

```
using SciMLSensitivity

using DifferentialEquations

using SciMLSensitivity # or DiffEqSensitivity if you prefer

using Zygote

using Optimisers # for optimizer & update

using LinearAlgebra
```

Error:

```
Traceback (most recent call last):
  File "c:\Users\Admin\.vscode\extensions\ganeshkumbhar.nb2pdf-1.1.9\scripts\nb2pdf.py", line 401
    exec('\n'.join(lines[:-1]), glb)
  File "&lt;string&gt;", line 1
    using SciMLSensitivity
          ^^^^^^^^^^^^^^^^
SyntaxError: invalid syntax
During handling of the above exception, another exception occurred:
Traceback (most recent call last):
  File "c:\Users\Admin\.vscode\extensions\ganeshkumbhar.nb2pdf-1.1.9\scripts\nb2pdf.py", line 408
    exec(source, glb)
  File "&lt;string&gt;", line 1
    using SciMLSensitivity
          ^^^^^^^^^^^^^^^^
SyntaxError: invalid syntax
```

Cell 8: ■ Code

```
# --- Cell 5: Loss Function ---
# ---- Stable predict function using BacksolveAdjoint and Float64 inputs ----
function predict_ude(p_nn)
# build problem with the current flattened NN params
prob_ude = ODEProblem(ude_model!, u■_reduced, tspan_reduced, p_nn)
# Use BacksolveAdjoint for stable gradients
sol = solve(prob_ude, Tsit5(),
saveat = timestamps_2D,
abstol = abstol, reltol = reltol,
maxiters = maxsteps,
sensealg = BacksolveAdjoint(autojacvec = true))
return sol
end
```

```
# ---- Loss function (keep as Float64) ----
function loss_function(p_nn)
sol = predict_ude(p_nn)
if sol.retcode != :Success
return Inf
end
# squared error (Float64)
return sum(abs2, Array(sol) .- training_data_2D)
end
```

```
# Sum of squared errors
loss = sum(abs2, Array(prediction) .- training_data_2D)
return loss
end
```

Error:

```
Traceback (most recent call last):
  File "c:\Users\Admin\.vscode\extensions\ganeshkumbhar.nb2pdf-1.1.9\scripts\nb2pdf.py", line 401
    exec('\n'.join(lines[:-1]), glb)
  File "&lt;string&gt;", line 6
    prob_ude = ODEProblem(ude_model!, u■_reduced, tspan_reduced, p_nn)
                                       ^
SyntaxError: invalid character '■' (U+2080)
During handling of the above exception, another exception occurred:
Traceback (most recent call last):
  File "c:\Users\Admin\.vscode\extensions\ganeshkumbhar.nb2pdf-1.1.9\scripts\nb2pdf.py", line 408
```

```
    exec(source, glb)
  File "&lt;string&gt;", line 6
    prob_ude = ODEProblem(ude_model!, u■_reduced, tspan_reduced, p_nn)
                                       ^
SyntaxError: invalid character '■' (U+2080)
```

Cell 9: ■ Code

```
# --- Cell 6: Training ---
```

```
# Callback function to display progress during training
losses = Float64[]
iter = 0
callback = function (p, l)
global iter += 1
push!(losses, l)
if iter % 10 == 0
println("Iteration $iter | Current Loss: $l")
end
return false # Must return false to continue training
end
```

```
# Define the optimization problem
optf = Optimization.OptimizationFunction((x, p) -&gt; loss_function(x),
Optimization.AutoZygote())
optprob = Optimization.OptimizationProblem(optf, p_nn)
```

```
# Execute the training mission
println("Commencing Training...")
# We use a lower learning rate for stability and more iterations.
# This is a full-scale training run. It may take a few minutes.
result_ude = Optimization.solve(optprob, Adam(0.01), callback = callback, maxiters
= 1000)
```

```
println("--- TRAINING COMPLETE ---")
println("Final Loss: $(result_ude.objective)")
```

Error:

```
Traceback (most recent call last):
  File "c:\Users\Admin\.vscode\extensions\ganeshkumbhar.nb2pdf-1.1.9\scripts\nb2pdf.py", line 401
    exec('\n'.join(lines[:-1]), glb)
  File "&lt;string&gt;", line 4
    losses = Float64[]
                     ^
SyntaxError: invalid syntax
During handling of the above exception, another exception occurred:
Traceback (most recent call last):
  File "c:\Users\Admin\.vscode\extensions\ganeshkumbhar.nb2pdf-1.1.9\scripts\nb2pdf.py", line 408
    exec(source, glb)
  File "&lt;string&gt;", line 4
    losses = Float64[]
                     ^
SyntaxError: invalid syntax
```

Cell 10: ■ Code

```
plot(losses,

xlabel="Iteration",

ylabel="Loss",

title="Training Loss Over Iterations",

label="Loss",

lw=2,

yscale=:log10)
```

Error:

```
Traceback (most recent call last):
  File "c:\Users\Admin\.vscode\extensions\ganeshkumbhar.nb2pdf-1.1.9\scripts\nb2pdf.py", line 401
    exec('\n'.join(lines[:-1]), glb)
  File "&lt;string&gt;", line 1
    plot(losses,
        ^
SyntaxError: '(' was never closed
During handling of the above exception, another exception occurred:
Traceback (most recent call last):
  File "c:\Users\Admin\.vscode\extensions\ganeshkumbhar.nb2pdf-1.1.9\scripts\nb2pdf.py", line 408
    exec(source, glb)
  File "&lt;string&gt;", line 9
    yscale=:log10)
           ^
SyntaxError: invalid syntax
```