

**NLLScompare.m**

Calls GD, GN, and LM to solve the NLLS problem for PDE

**GD.m**

Solves the NLLS problem by gradient descend

**GaussNewton.m**

Solves the NLLS problem by Gauss-Newton

**LevenbergMarquardt.m**

Solves the NLLS problem by Gauss-Newton

**Res\_and\_Jac.m**

Computes the vector of residuals and the Jacobian for the NLLS problem

**ActivationFunction.m**

Defines the activation function (sigmoid or tanh) and its derivatives

**res.m**

Computes individual residuals and their gradients

**param.m**

Extracts  $v, W, u$  from the parameter vector

**setup.m**

Defines the boundary conditions, the right-hand side, the function  $h$  that is zero on the boundary, and the exact solution

**NN.m**

Computes the derivatives of  $N, N_x, N_y, N_{xx},$  and  $N_{yy}$  w. r. t. the parameters