```
%FFR135 HM3.2 Chaotic time-series prediction 2024
%Author: Siyu Hu, gushusii@student.gu.se, 19950910-3702

function RunReservoirComputer()
    clear all;
    close all;
    clc;

    trainingData = csvread('training-set.csv')'; % 19900 x 3
    testData = csvread('test-set-5.csv');       % 100 x 3

    ridgeParam = 0.01;
    numReservoir = 500;
    outputFile = 'prediction.csv';

    [numSamples, N] = size(trainingData);
    T = size(testData, 2);

    w_in = sqrt(0.002) * randn(numReservoir, N);
    w = sqrt(2 / numReservoir) * randn(numReservoir, numReservoir);
    R = zeros(numReservoir, numSamples - 1);

    % Training
    % use (1:numSamples-1) timesteps as input
    for t = 1:numSamples-1
        if t == 1
            r = zeros(numReservoir, 1);
        else
            r = R(:, t-1);
        end
        x = trainingData(t, :)';
        R(:, t) = tanh(w * r + w_in * x);
    end

    Y = trainingData(2:end, :)';
    I = eye(numReservoir);
    w_out = (Y * R') / (R * R' + ridgeParam * I);

    % Testing
    R_test = zeros(numReservoir, T + 500);
    Y_pred = zeros(3, 500);
    O = zeros(N, 1);

    for t = 1:T
    if t == 1
        r = zeros(numReservoir, 1);
    else
        r = R_test(:, t-1);
    end
    x = testData(:, t);
    R_test(:, t) = tanh(w * r + w_in * x);
    O = w_out * R_test(:, t);  %  O(t)
```

```
    end

    % 500 timesteps (1 timestep = 0.02s)
    for t = T+1:T+500
        r = R_test(:, t-1);
        x = O;  %  set O(t) as input x
        R_test(:, t) = tanh(w * r + w_in * x);
        O = w_out * R_test(:, t);  % update O(t+1)
        Y_pred(:, t - T) = O;      % save x1, x2, x3
    end

    % save x2 to csv file
    writematrix(Y_pred(2, :)', outputFile);
    disp('Prediction completed and saved to csv file');

    % plot Lorenz system
    figure;
    plot3(Y_pred(1, :), Y_pred(2, :), Y_pred(3, :), 'b', 'LineWidth', 1.5);
    hold on;
    plot3(testData(1, :), testData(2, :), testData(3, :), 'r', 'LineWidth',
1.5);
    xlabel('x_1(t)');
    ylabel('x_2(t)');
    zlabel('x_3(t)');
    title('3D Prediction vs. TestData of Lorenz dynamics');
    legend('Prediction', 'Actual');
    grid on;
    hold off;

    % plot 2D Y_pred(501-601 timesteps) and testData(x2)(1-101 timesteps)
    figure;
    timeSteps = (1:100) * 0.02;
    plot(timeSteps, Y_pred(2, 1:100), 'b', 'LineWidth', 1.5);
    hold on;
    plot(timeSteps, testData(2, :), 'r', 'LineWidth', 1.5);
    xlabel('Time (seconds)');
    ylabel('x_2(t)');
    title('Prediction VS TestData for x_2 component (first 100 timesteps)');
    legend('Prediction (Y_{pred})', 'Actual (testData)');
    grid on;
    hold off;

end
```

Published with MATLAB® R2024a