Name: Quang Khai Le

Class: EE 104

Professor: Christopher Pham

Lab 3: Curve fit demonstration

Github link: <https://github.com/blade199916/Lab3_Le_QuangKhai.git>

Youtube link:

1&2 part: <https://youtu.be/OOTPpasJ3xY>

Continued part: <https://youtu.be/ufxN9HIUyCQ>

First step: make sure the other file in installed and ready

import numpy as np

import matplotlib.pyplot as plt

import scipy.optimize as opt

import scipy.stats as st

import math

import matplotlib.pyplot as plt

from sklearn.linear\_model import Ridge,LinearRegression

from sklearn.preprocessing import PolynomialFeatures

from sklearn.pipeline import make\_pipeline

1. Polynomial Regression Curve Fitting (Lecture 1):

Utilize Polynomial Regression to fit a curve (from Lecture 1).

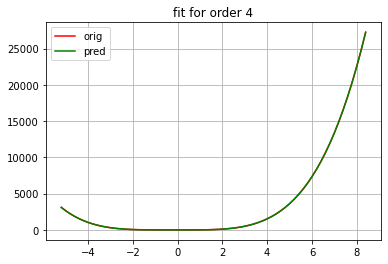
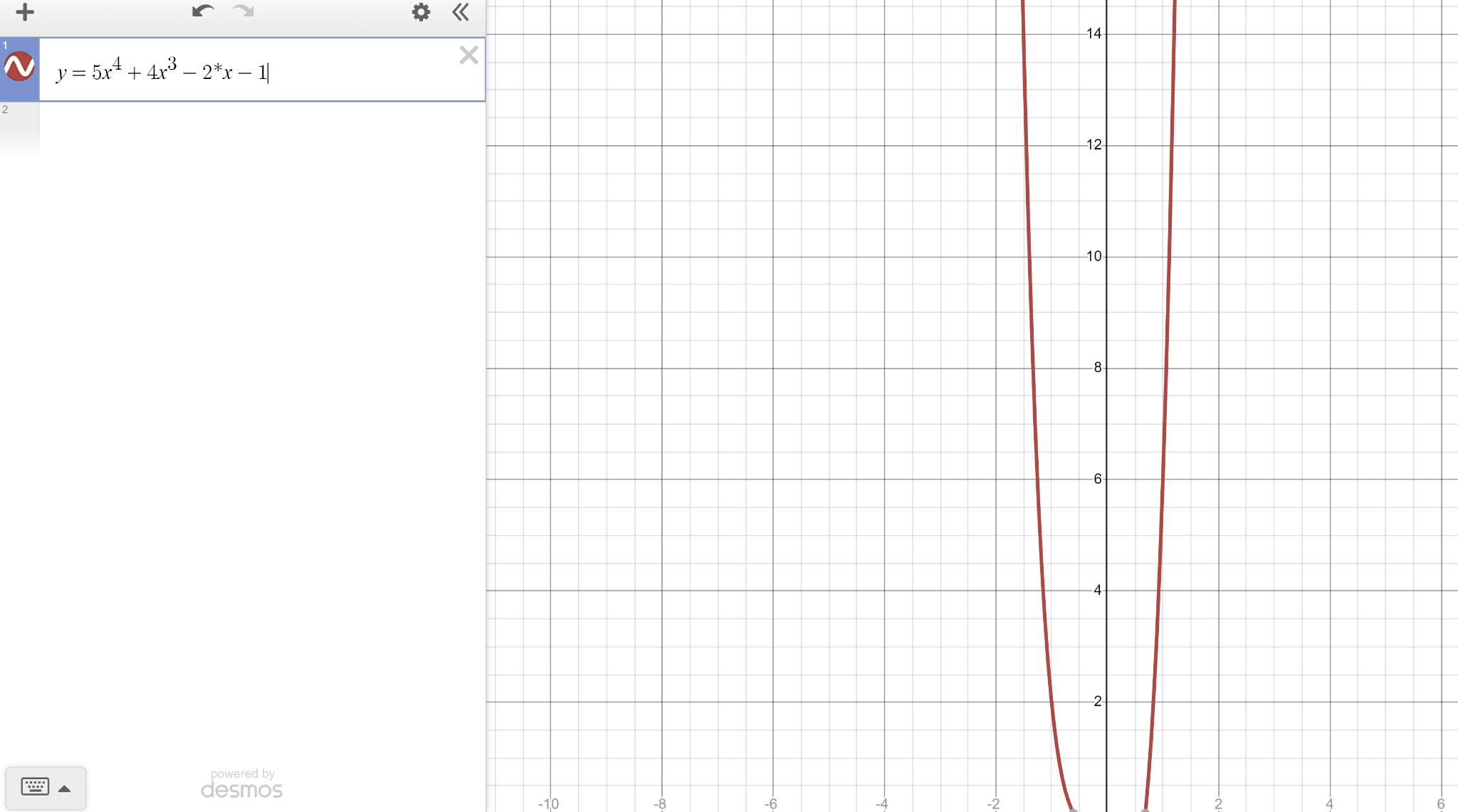
Specify the range of x-values with a step size (e.g., [-2, 2, 0.1]).

Provide the Y-values, representing the dependent variable in the dataset (3rd order or higher Polynomial).

Enter the order of the Polynomial (pres(order)).

Enhance the plot with a title, legend, and grid lines.

Execute the program to visualize the plot.



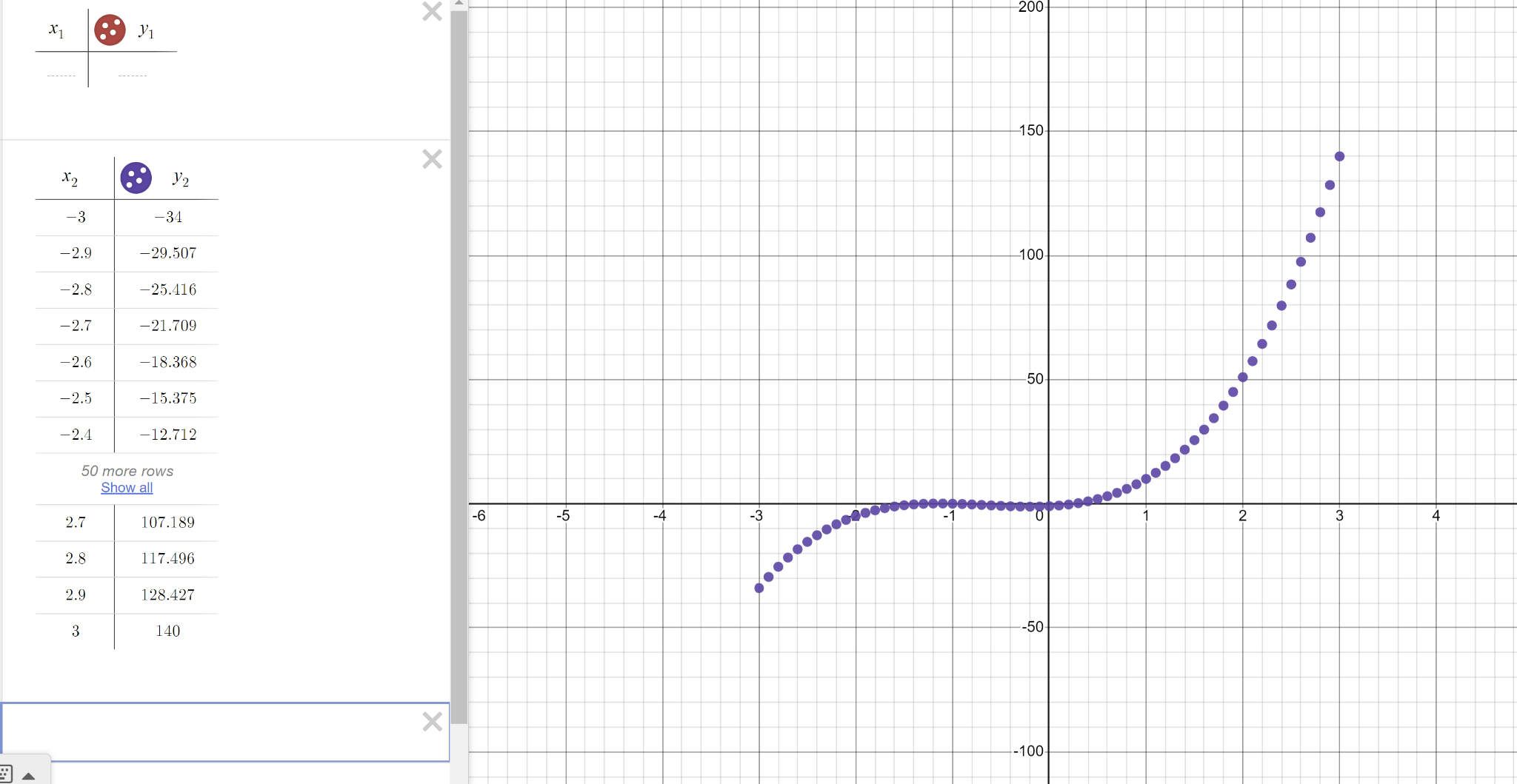
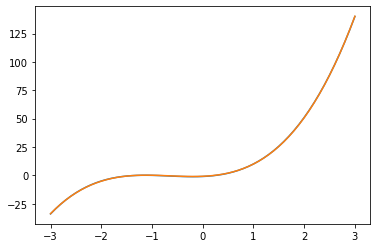
1. Pipeline Ridge or Pipeline Linear Regression Curve Fitting (Lecture 4):

Create an Excel data sheet and save it as a .csv file (e.g., pp.csv).

Use "float(things[number])" to select the desired column in the file.

Enter a number to display the desired result.

Run the program, review the output, and examine the plot.



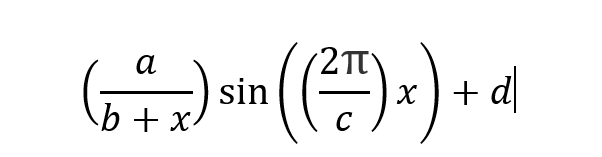
1. Damped Sine Wave Curve Fitting (Lecture 2):

Define an array "xin" to set the parameters for generating a sine wave.

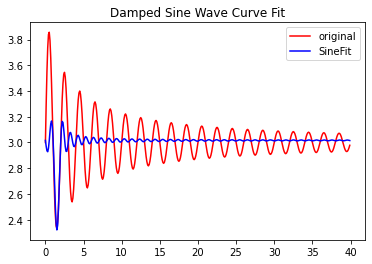
Incorporate the damped sine wave function.

Generate random numbers and add them to the data.

Execute the program to visualize the resulting plot.



Let a,b,c,d = 3



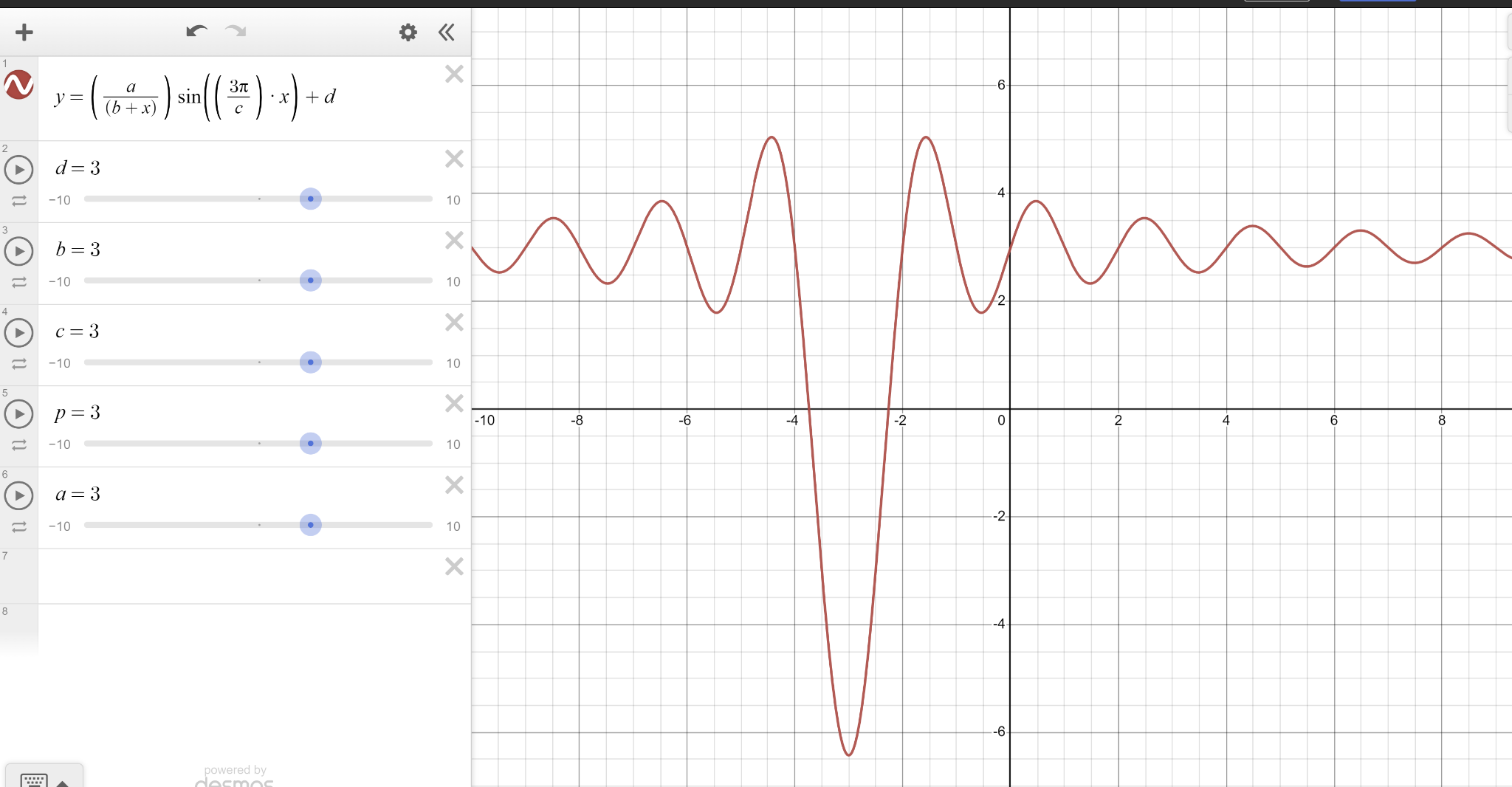
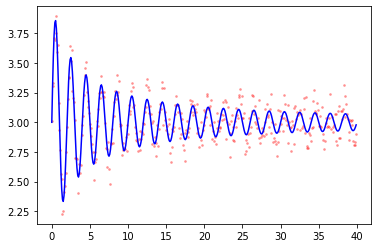
1. Curve Fitting with Noise (Lecture 2):

Specify parameters in the array "xin" to generate a sine wave.

Add the damped sine wave function.

Create x-values and choose perfect y-values.

Apply noise to the y-values (e.g., yy = [y + random.gauss(0, 0.1) for y in y]).

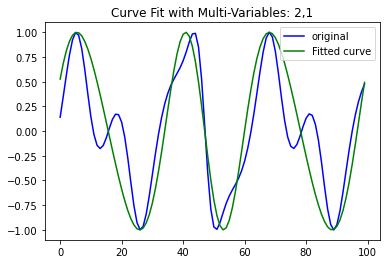


1. Curve Fitting with Multiple Variables (Lecture 3):

Generate data with two dependent variables and two independent variables.

Set up five different scenarios for testing curve fitting.

Run the program and examine the resulting plots.



(2,1) is the best fit in 5 pair of variables:

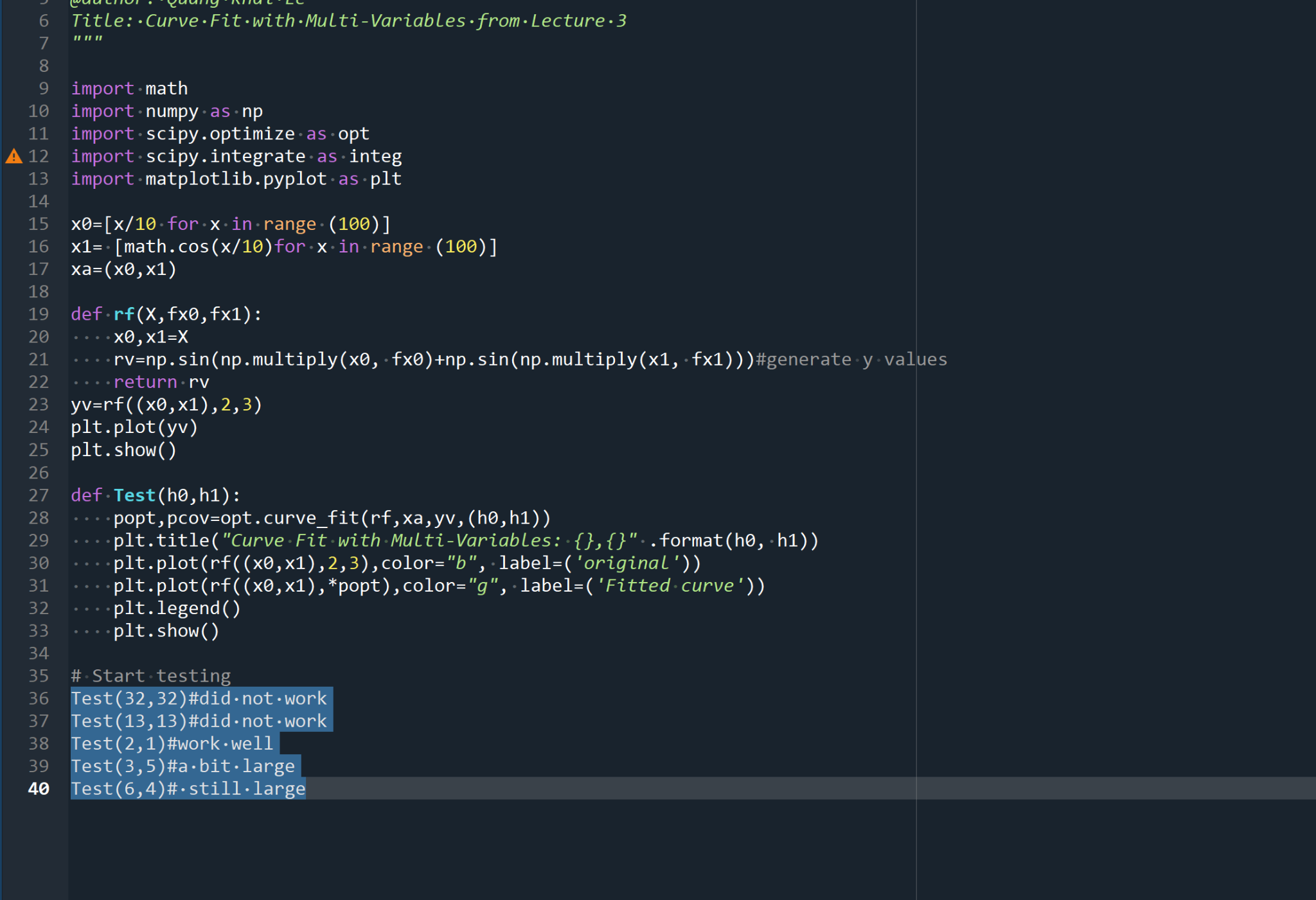
Test(32,32)#did not work

Test(13,13)#did not work

Test(2,1)#work well

Test(3,5)#a bit large

Test(6,4)# still large



1. Game Development: Coin Collector:

Modify the actor (character) according to your preference.

Adjust the playing area to your desired specifications.

Include additional time or increase the game speed if desired.

Execute the program and enjoy playing the game.

