**EXERCISE 3**

**MLS FOR CURVES, MESHES AND IMAGES**

To run the codes of this exercise, *../code/* must be the working directory. Then, running corresponding parts performs all the necessary calculations required and generates the plots.

1. **CURVE AND SURFACE RECONSTRUCTION USING MLS**
   1. **Curves Derivation**

The solution of moving least squares problem is acquired by solving the general form,

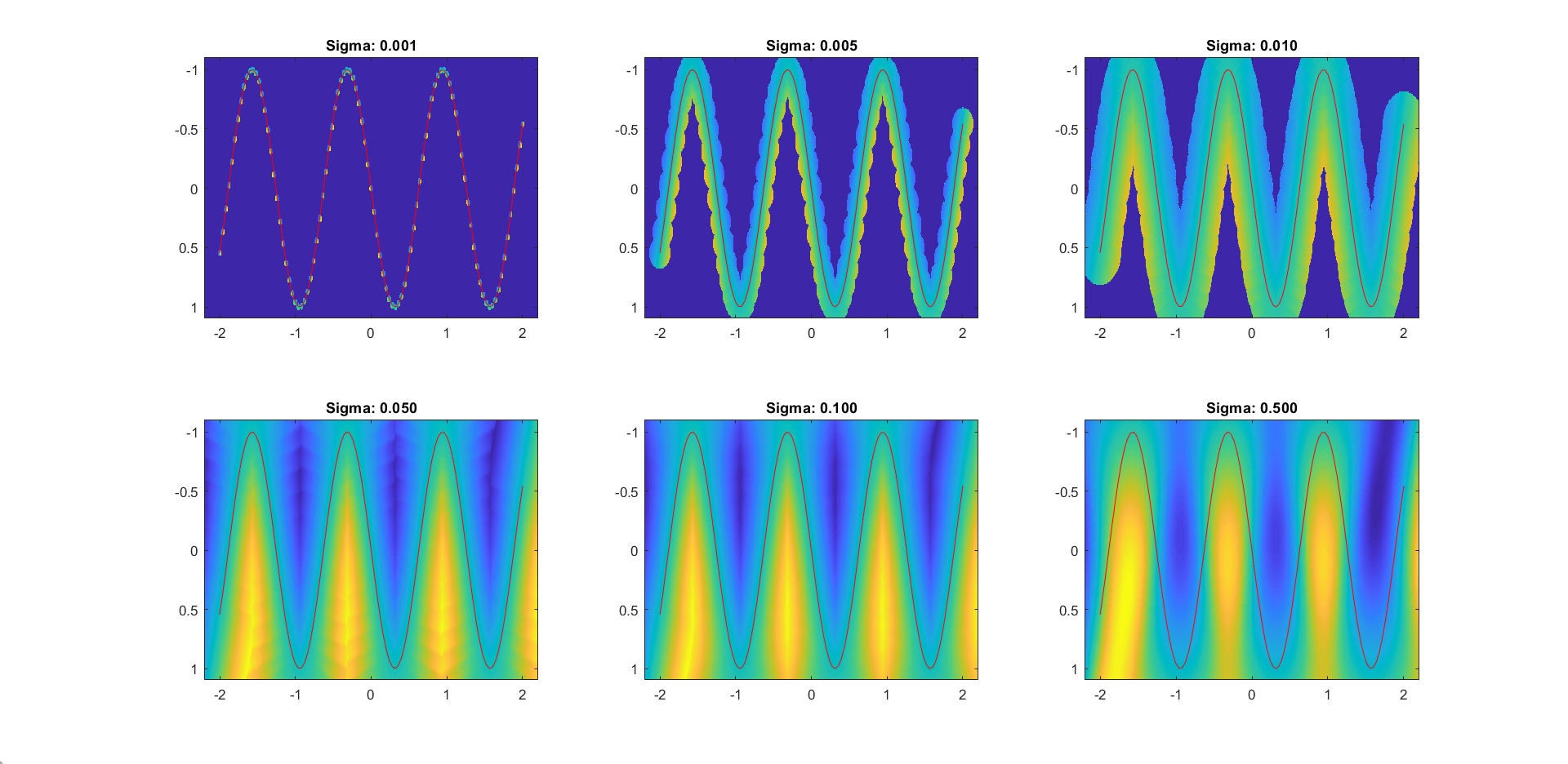
When locally fit polynomial, , consists of a single constant term, then and which results in . Therefore, the problem definition can be rewritten as,

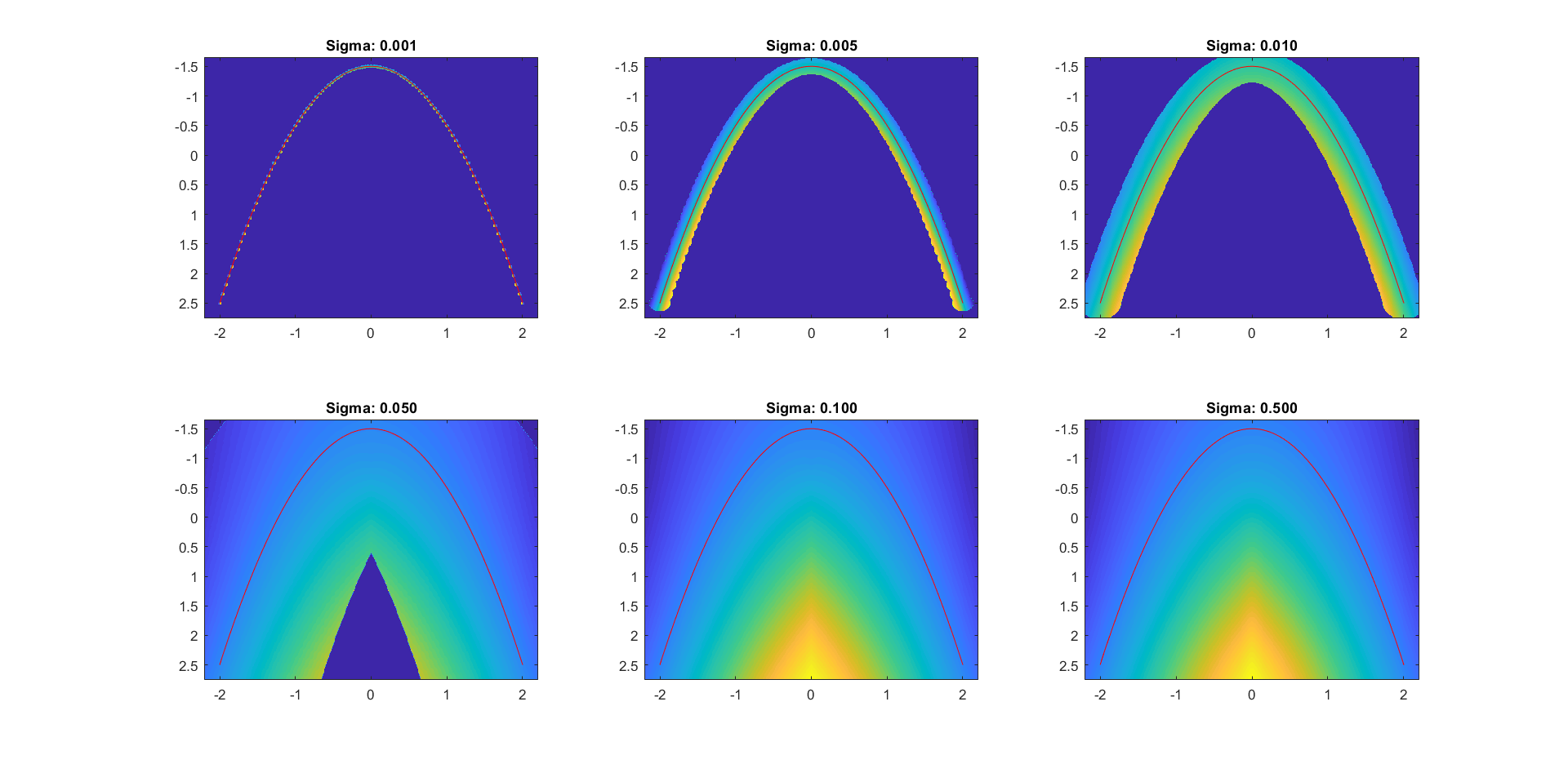
To find the value of which minimizes the error function, we can take the derivative of the error with respect to and equate to 0. Note that the result is a simple weighted average in our case.

* 1. **Curves plotting**

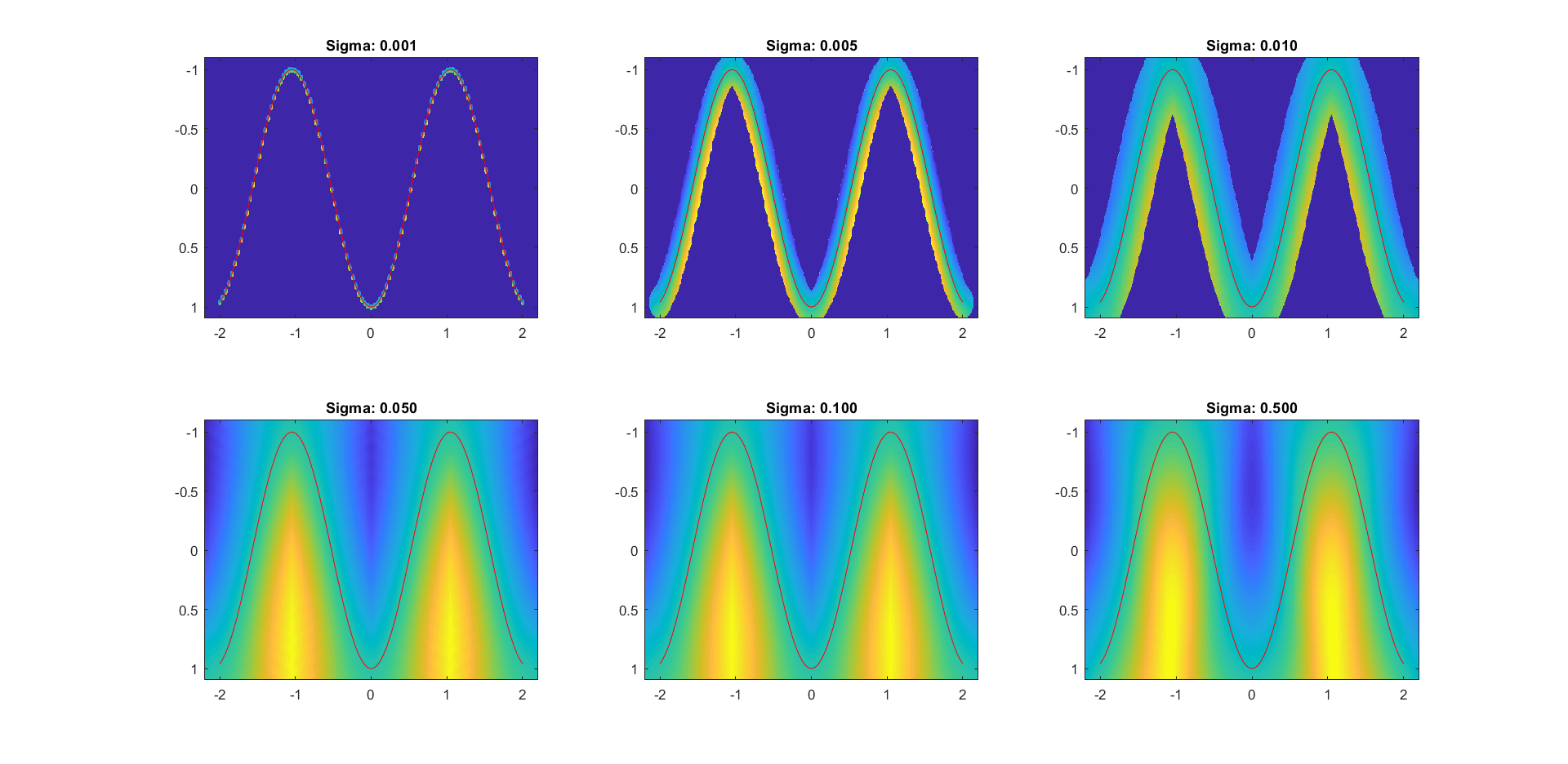
MLS for curve fitting algorithm, utilizing the found in 1.1 is tested with different values of sigma for Gaussian weighting function. The grid resolution is always kept as 0.01 in both axes. According to the results, a balanced performance between smoothness and continuity is achieved with sigma = 0.005 in all three cases. Running *part1\_2.m* is enough to get the results.

Curve Data 1:



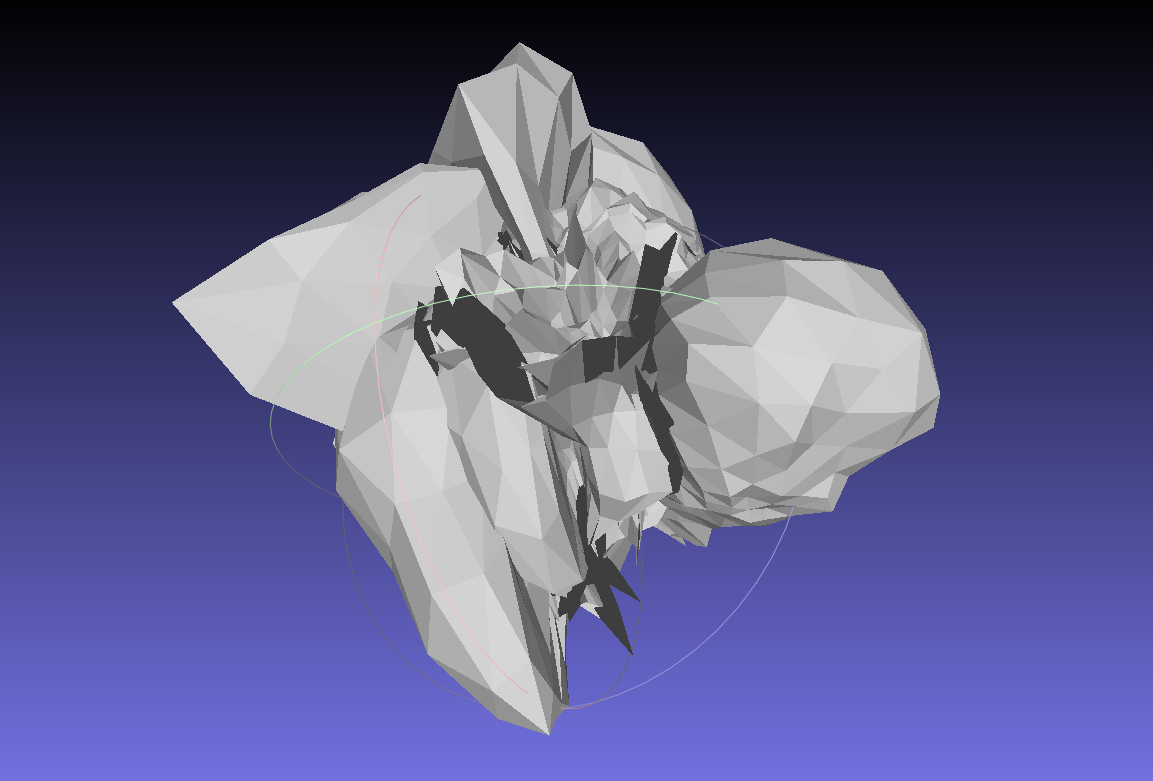
Curve Data 2:

Curve Data 3:



* 1. **Smoothing Meshes**

The derivation of the gradient function is as follows:



However, I did not manage to implement it correctly. In the code folder, part1\_3.m contains all the necessary computations but the result *trial.off* which is acquired after only one iteration of smoothing *bunny.off* is not correct.

1. **Image Deformation Using Moving Least Squares**

The algorithm is implemented according to the provided paper. Running *part2.m* performs necessary calculations and outputs the images.

For warping, reverse mapping is used to avoid artifacts (black spots in the warped image).

