2. From re_der I get -2.4783497330, which agrees with manually calculated derivative for all but last shown decimal. Relative errors of derivatives with various n are shown below. They differ somewhat from re_der and WolframAlpha.

n	central diff	re_der
4	-4.5247938828e-09	4.1015831904e-11
6	-6.3249950314e-11	-5.2888637544e-09
8	-3.0310423649e-09	1.3720468244e-07
10	-2.1805600221e-07	2.9124720360e-05
12	6.7649196950e-05	2.5709034249e-03
14	-3.2697173674e-03	2.8927047776e-02

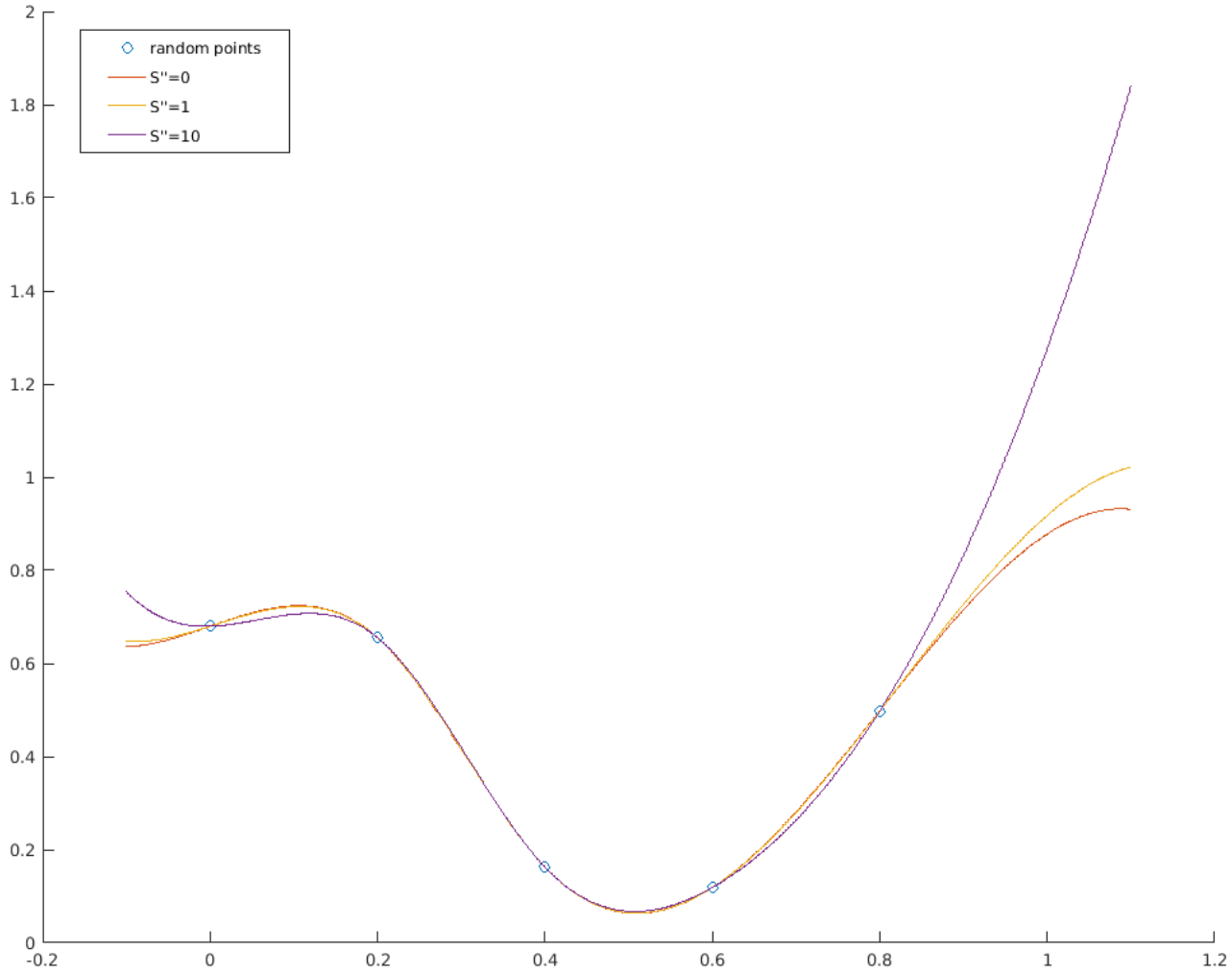
Error of re_der seems to grow when h gets smaller from 10^-4 whereas central difference method gives best results with h=10^-6.

3. I generated 5 or 20 random y-values in range [0,1] wit equally many evenly spaced values in range [0, 1[(spacing with which range [0,1] would have contained 11 points ie. 0.2 or 0.05). For getting the spline I used matlab function csape giving ‘second’ and wanted second derivatives like for S’’=0

```
p1 = csape(x, y, 'second', [0, 0])
```

Csape returns a polynomial, values of which I evaluated and then plotted together with the original points.

Plot with 5 points:



plot with 20 points: