Numerical Analysis Project 2

Nathaniel Valla

Function used:

newtons_div_dff(x_i,y_i,t): Takes two arrays of the same size and uses divided differences to find largangian polnoimal

bezier_curves(x_i,y_i,y_d,t,h):Takes in two arrays of function values and the dervitative, point spacing, and test points

Code Overview

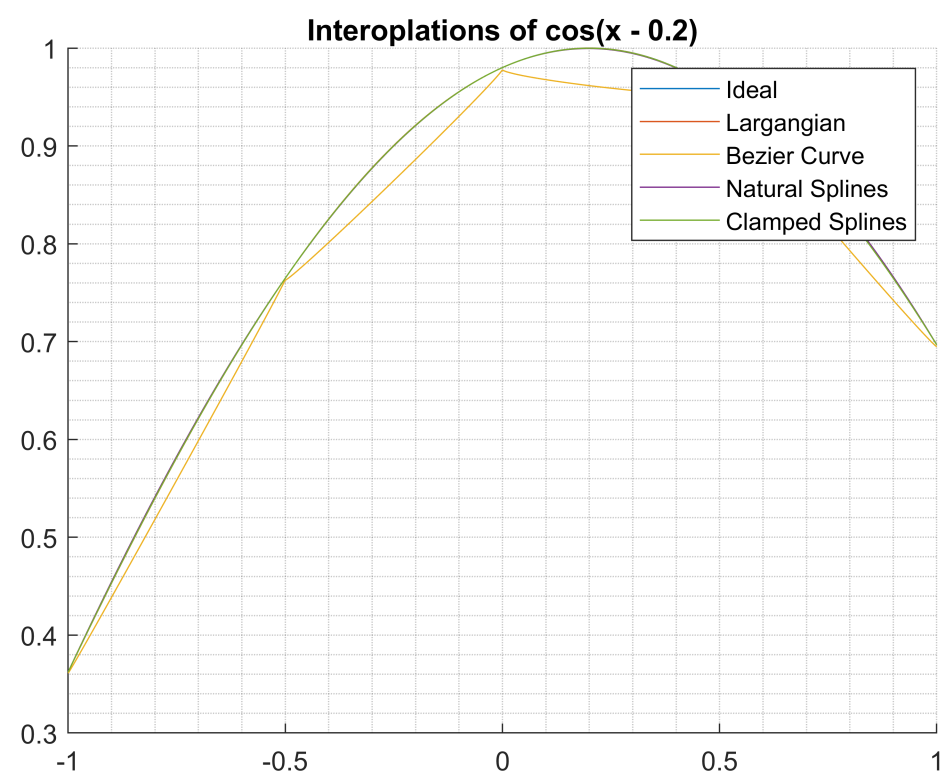
In this code three function are used to tests different methods of interpolation largangian, cubic splines and and Bezier curves. The largangian is found using newtons divided difference method. The bezier curvers are found by finding the root of the cubic of the x at x_i and then plugged into the cubic of y for y_i. The cubic splines are found using the spline function from MatLab library. One is a Natural spline and the other is a clamped splines.

After that the absolute errors are calculated for each tests function and interpolation methods. These are then grouped by method. This is so you can see how each method is able to interpolation different functions.

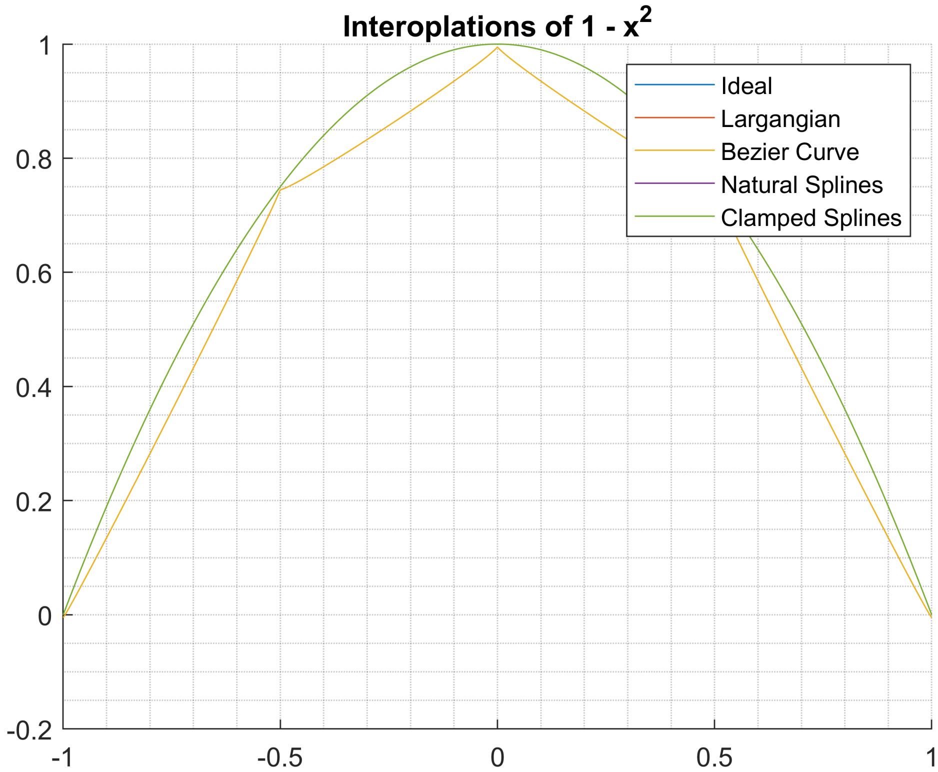
Then the data provided is used to interpolate the headlamps, and the change in data is used to see how moving one data points effect the overall curve.

Test cases:

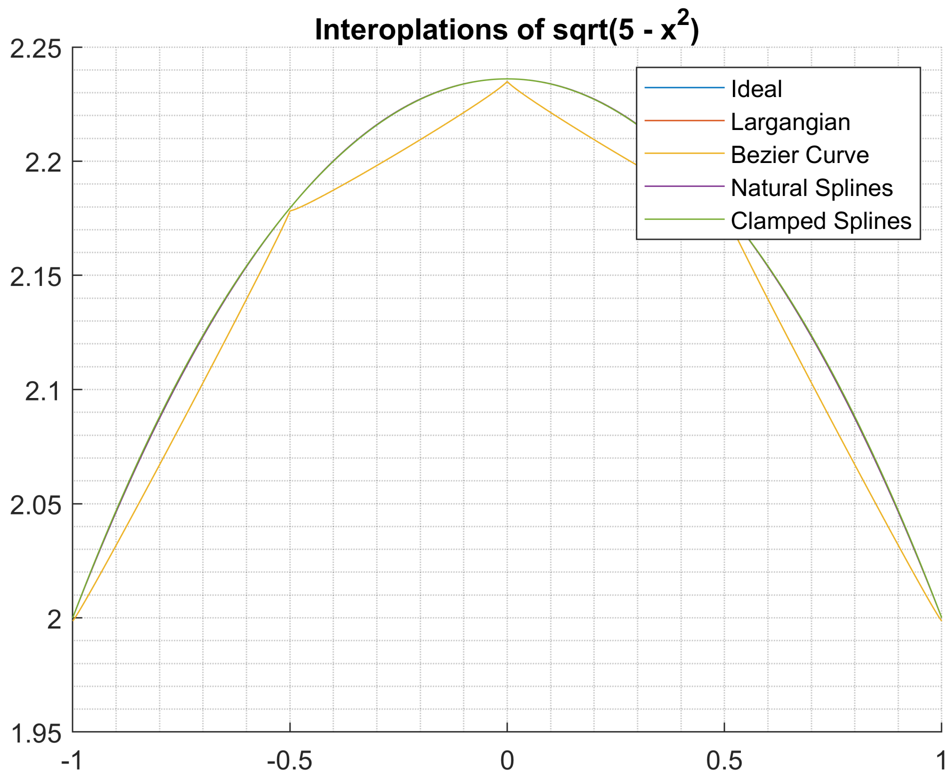
1. cos(x-0.2)



2. 1-x^2

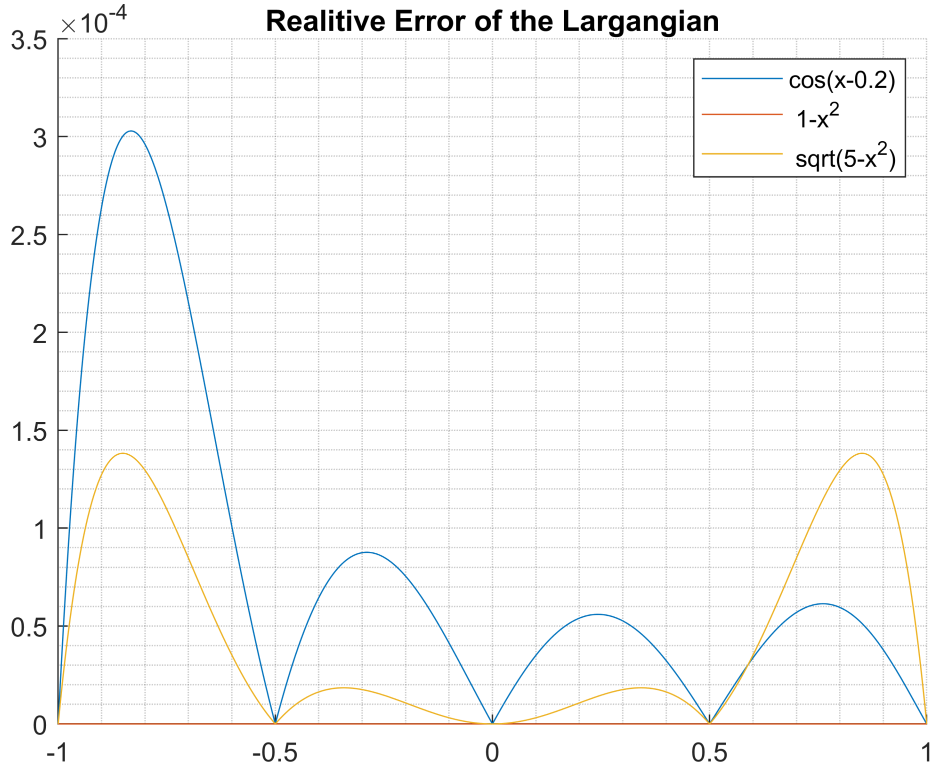


3. sqrt(5-x^2)

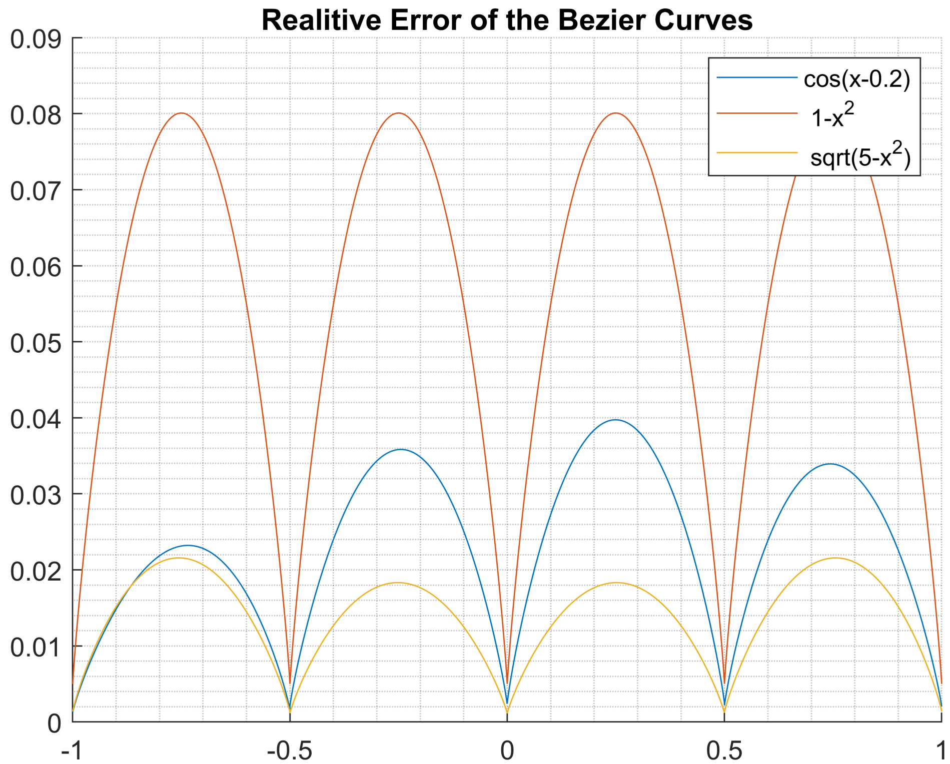


In all three test case the Bezier curves are visually the least useful in interpolating. The other three methods it is harder to tell since the over lap pretty well.

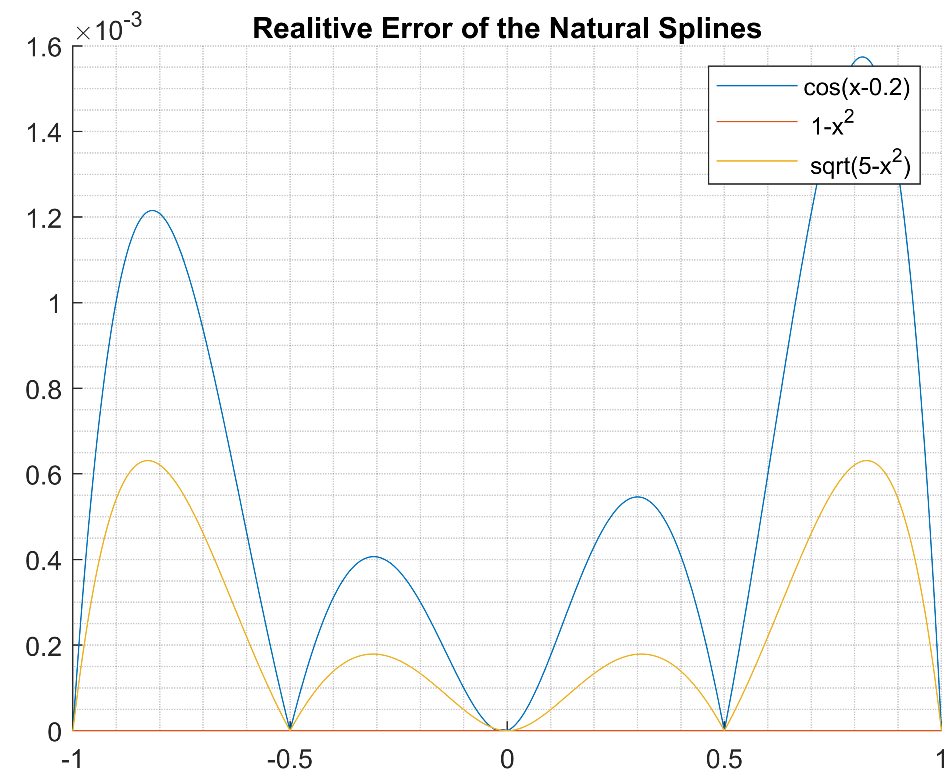
Methods Error:



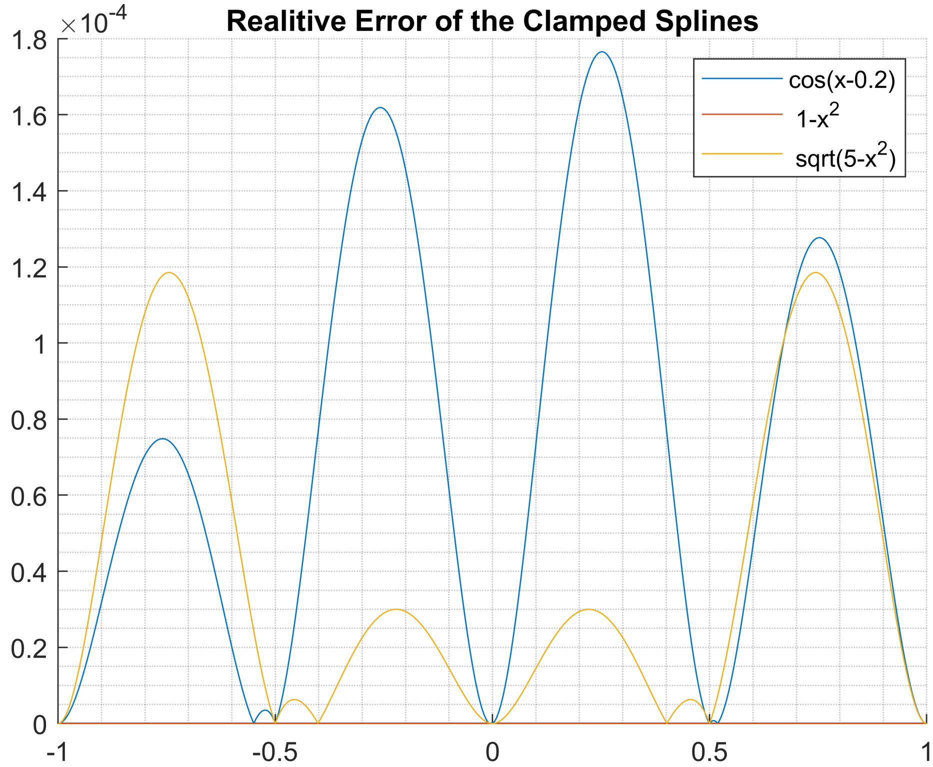
In cos(x-0.2) and sqrt(5-x^2) test senerio the largangian realtively well on about the scale of 10^-4 and the 1-x^2 sit at 10^-16. This is to be expected since the larganian is by defination the lowest order polynimial to fit through all point so actual polynomial is found. The error that is achived is down to machine error.



In all three cases the bezier curves performed bad with the error in scale 10^-1.

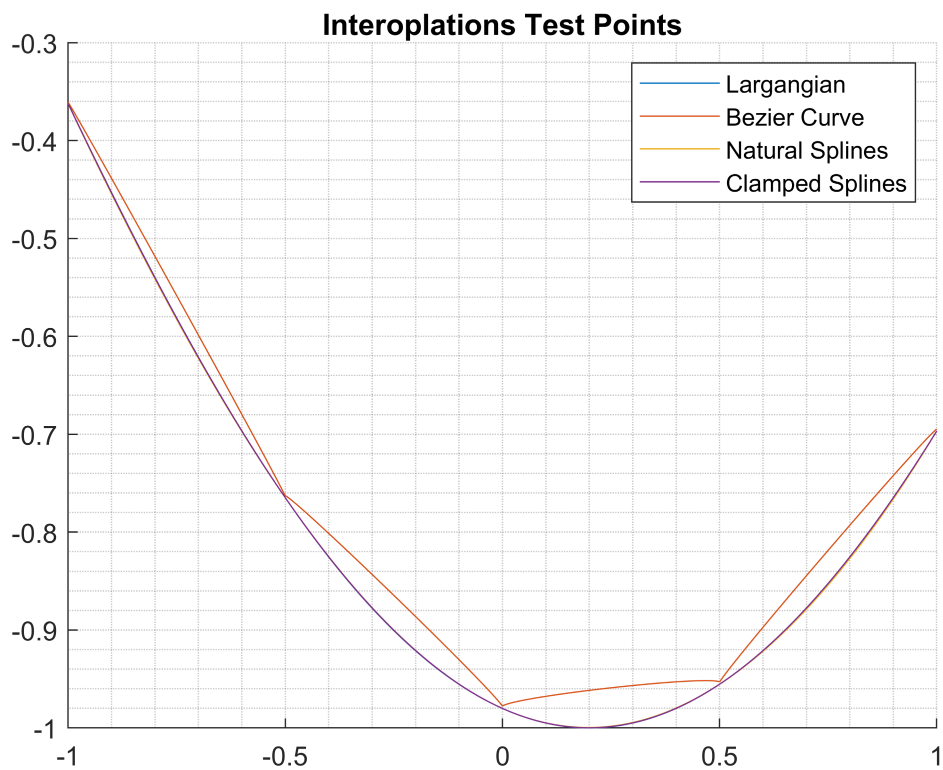


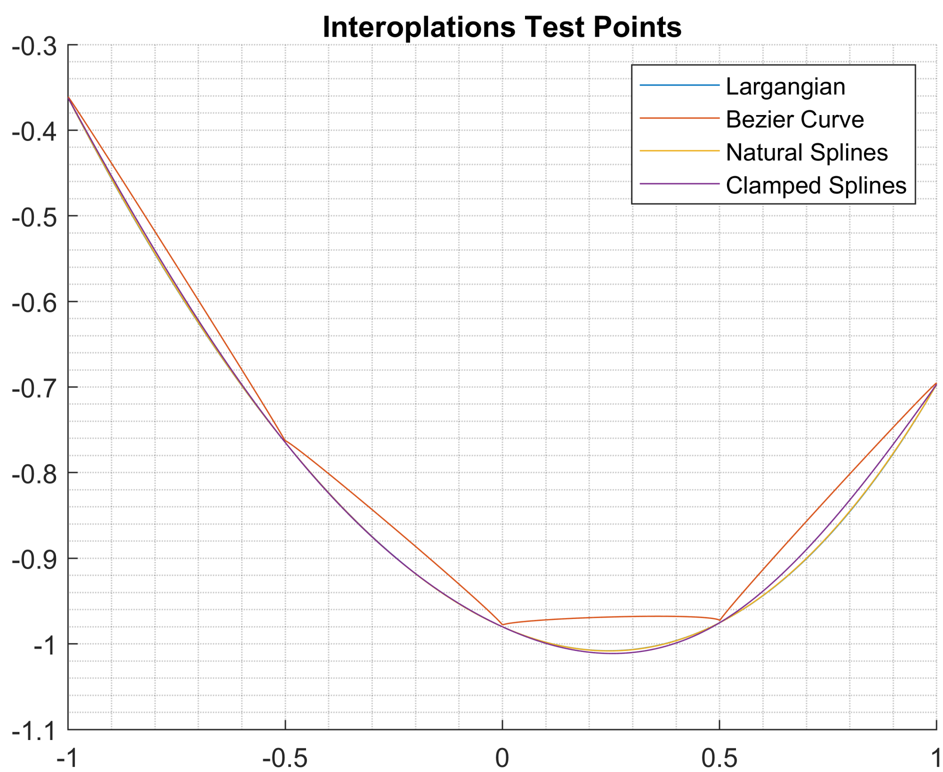
In cos(x-0.2) and sqrt(5-x^2) test senerio the Natural Splines realtively well on about the scale of 10^-3. The polynomial is approxmated really well since it using polynomials to aproximate it.



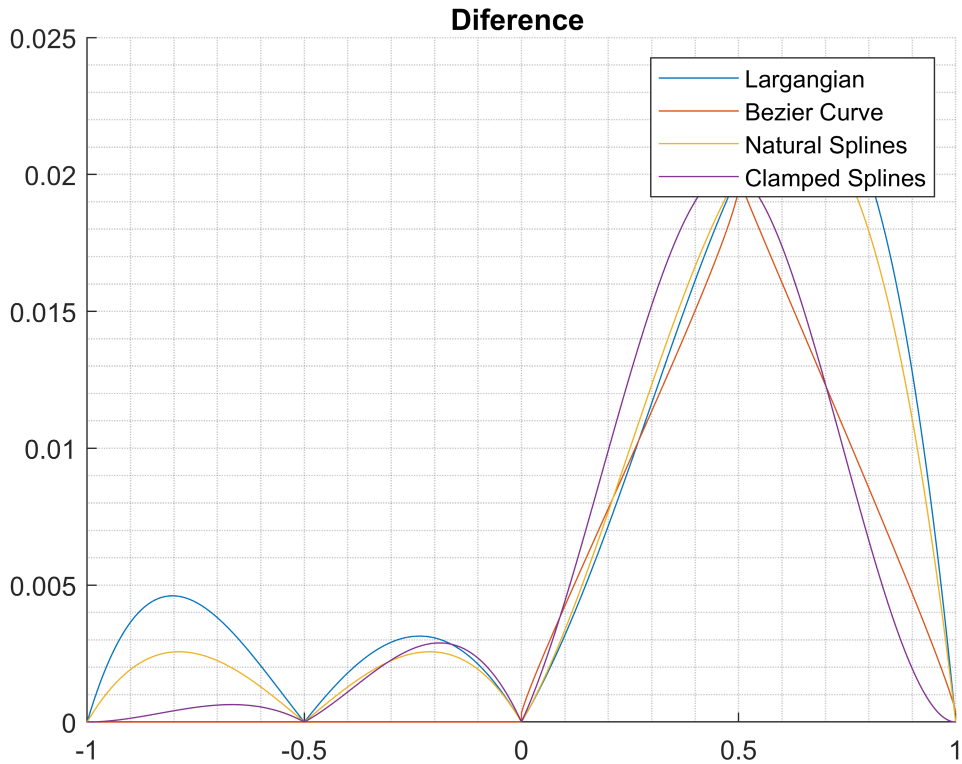
In cos(x-0.2) and sqrt(5-x^2) test senerio theClamped Splines realtively well on about the scale of 10^-4. it actually does better then the larganian because it take the derviates into account. The polynomial is again approxmated really well since it using polynomials to aproximate it.

Test Points






```
Current plot released
```




```
Current plot released
```

Between all four methods used when changing a single data point bezier changed the least and then followed by clamped splines. These makes sense since both take in the derviates into considerations and isolate the other changes quite well.

Conclusion

The most usefully method for interpolation with these test is the clamped cubic splines since it interpolate the functions quite well in all senerios and isolates any large changes in data. Beizer curve are cleary the worst and that this is without approxmating the derviates maybe if the test point are changed, using richard interpolation is probably need, but some date will have to be done using forward and backward difference since each of the beizer curzes needs to guide points. My process of the determinting the guide points is most likely needed to be reavaluated to better guide the functions since it purely guess and check.