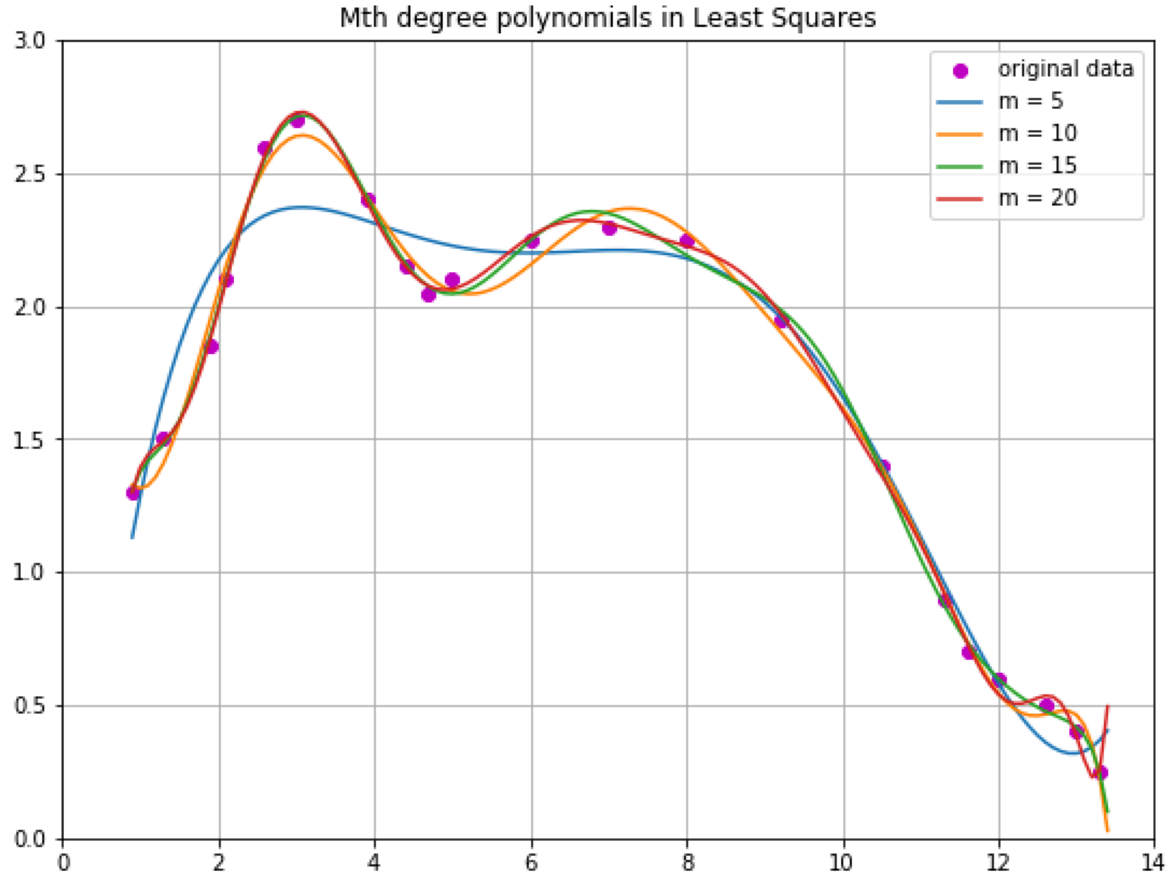
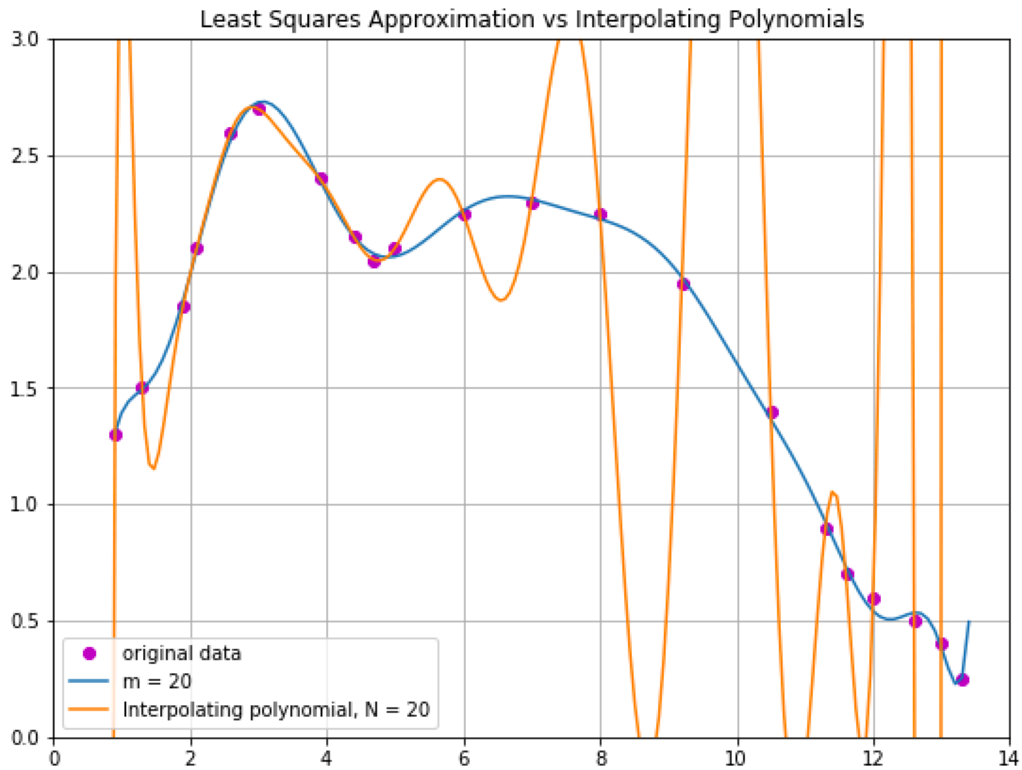
The graph for the least squares polynomial is shown below:

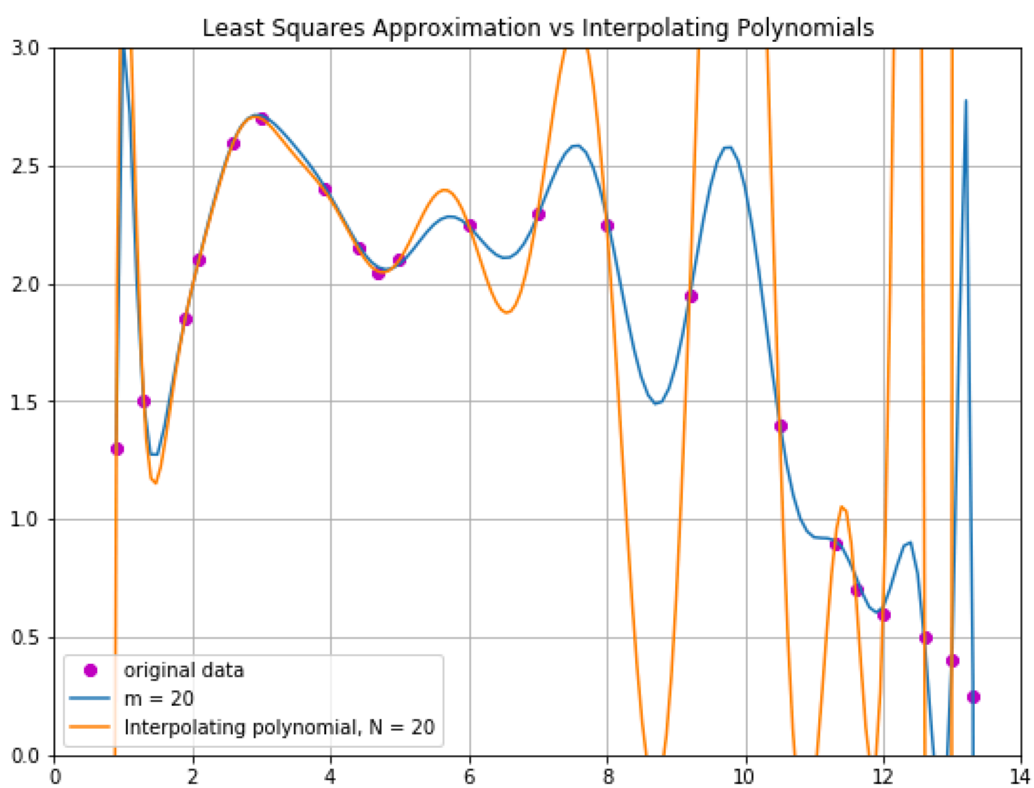


The graphs for each of the mth degree polynomials look very similar. With the exception being m=5. Where m = 10, 15, 20 the graphs look very similar in terms of shape and following the shape of the original data. The original data is shown in the magenta points. m = 5 is shown in blue and you can see that it is the only data that stand out by itself a bit more. As you increase the number of degrees in the polynomials, the data gets more and more accurate, but you don't need to increase it too much, as just increasing it by 5 points gets it very close to where it should be.

The data that is most similar to the line of original points is the 20^{th} order polynomial. I will discuss this more below.

The graphs where m = 20 are shown below. There are two graphs and I will talk about why there are two more in depth on the page after the graphs.


Least Squares Approximation vs Interpolating Polynomials
original data
m = 20
Interpolating polynomial, N = 20

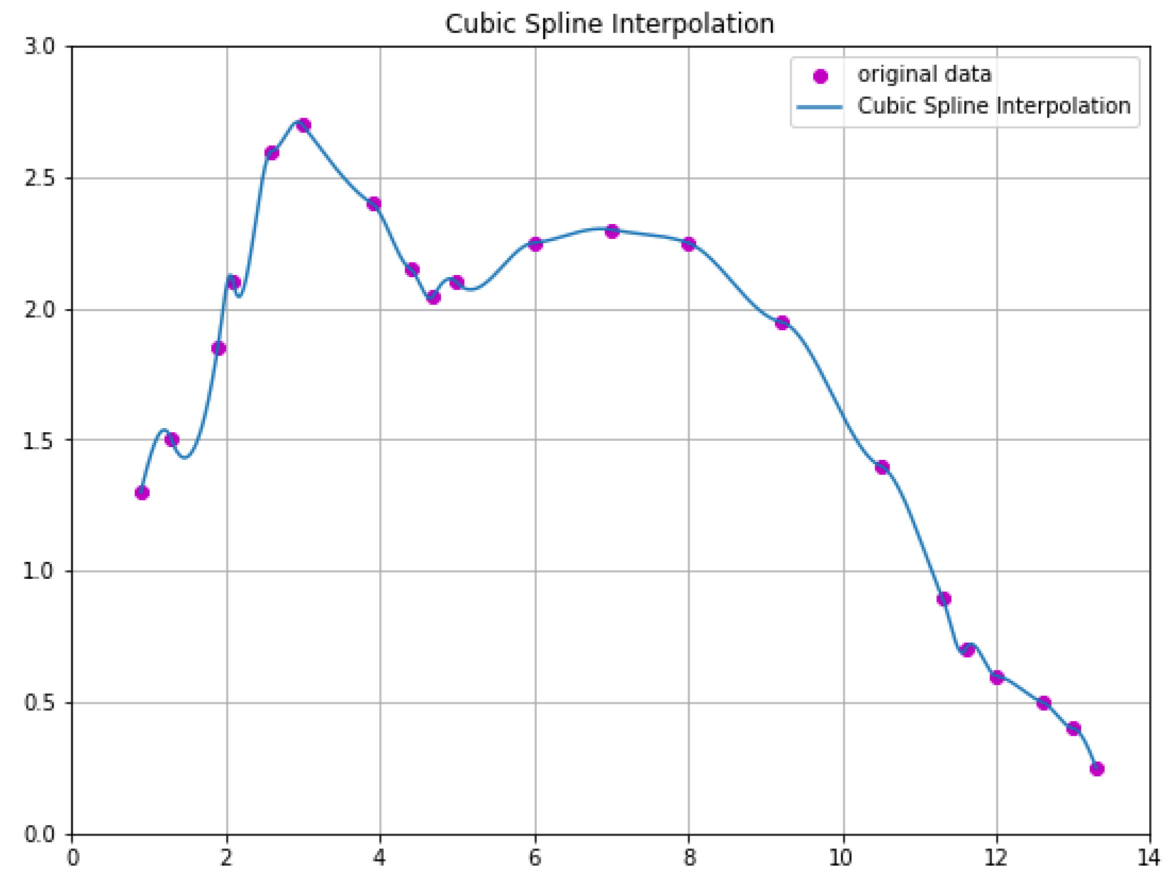

Least Squares Approximation vs Interpolating Polynomials
original data
m = 20
Interpolating polynomial, N = 20

Both the images above show the 20^{th} order polynomials in least squares along with using Lagrange interpolation. The first image show m = 20 when using the $a = XX^{-1} \bullet X^{-1}f$ and the second one uses the function a = Xf. These for some reason were coming up are two different numbers which was strange, but the functions looked different. Neither of the 20^{th} degree polynomials went exactly through all the points, which may just have been a processing error or error due to the program that I used, but nevertheless it's there.

The one where we just used Xf = a was the one that looked most similar to the Lagrange polynomials so that one is probably more representative of what it is that we were looking for. The shape is closer to that of Lagrange.

I don't think this gives a good approximation just because it oscillates a lot. The function goes up and down a considerable about of times and that is not necessarily a good thing although it does go through all the points. A function that oscillates a lot can take longer than one that does not which is not good for optimization and such.

The graph for cubic splines is shown below:



This graph is a pretty good representation of the original data. It does not exactly look like a ducks back but it is pretty close and it is a lot closer to the original data set than least squares and interpolating polynomials were.

I came up with the necessary derivations on my own and they are show on the image attached below. They are pretty much the same as the equations 7b that were given to us but the delta values are different for n=1 and n=N.

For this one, we had to take into account the end cases and when n=1 and n=N and solve for the data differently when n=2 to n=N-1.

I think this one is a lot better of a way to represent the data that we were trying to approximate and it looks a lot more similar in terms of shape than the two other methods we used.