ECEN 521  
Oct 06 2015  
Matthew James  
Connor Smith  
Ricky Wyman’s Brother

Design Project 1

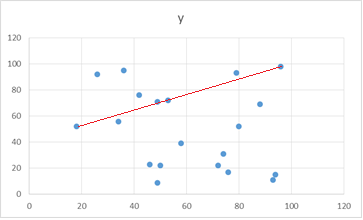
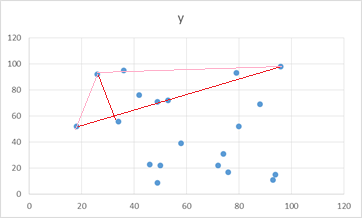
 To implement our Convex Hull algorithm we implemented the Quick Hull algorithm. This design is based on the Quicksort algorithm. Its normal complexity is but in the worst case can be as bad as . Our implementation of this algorithm takes the point with the lowest x value and the highest x value (and adds them to our list of convex hull points also see Figure 1) and connects a line between them. It then iterates through all the points above it and finds the one that is furthest away from the line (and adds it to the list of points in our convex hull also see Figure 2). Using that point it makes a triangle between the two extreme points and this new point. With that triangle we go through each point and using the cross product of the different vectors we find out if the different points are within the triangle or outside of it. Using that data, we can make a new line and find the furthest point and add it to the list. We continue this recursively, throwing away any points with in the triangles until we have all of them on that side of the line. We then do the same thing for the bottom half.

Figure 2

Figure 1

The problem is not too difficult as the algorithm was well documented. The only difficulties were finding out implementation details such as computing cross products and comparing the sign of two different ones. Our implementation does not return the data in sorted form so we did need to sort it afterwards which we did by computing the angle of the different points in relation to the first point and then using that to make a counter clockwise sorted list.

Running several test cases we found the performance to be quite good. We were able to run a 10,000 point set in around 5 seconds. This seemed to be much quicker than what would have been possible using different algorithms.