Eliminating Outliers

What are Outliers?

- Outliers have extreme values (either small or large) that can largely influence statistical analysis
- Here is a sample set of data to show the effects:

  Resting Heart Rate (bpm): 80, 67, 76, 78, 66
  Average: 73.4
  Standard Deviation: 6.47

  Rest Heart Rate with Outlier (bpm): 80, 67, 76, 78, 66, **120**
  Average: 81.2
  Standard Deviation: 19.9

  Just adding that one data point of 120 beats per minutes has totally changed the average and the standard deviation.

How to Deal with Outliers

Reasons to take out outliers:

- The data was measured wrong (the students might have misread the heart rate monitor)
- The data was recorded wrong (the students might have recorded it in the wrong units)
- The conditions for the experiment might have been wrong (the students might have measured exercising heart beat)

Reasons against taking out outliers:

- It might represent natural variation in data
- Will taking out the outlier affect the analysis for the better or for the worse

These reasons however cannot actually justify eliminating an outlier from the data but an actual quantitative test must be conducted.

There are multiple ways to find out whether an outlier is statistically significant or not.

- Tiejen-Moore Test: http://www.itl.nist.gov/div898/handbook/eda/section3/eda35h2.htm
  - Can detect multiple outliers without accumulating error but you need to specify the number of outliers
- Generalized Extreme Studentized Deviate Test: www.itl.nist.gov/div898/handbook/eda/section3/eda35h3.htm
  - Only need to specify an upper limit on the number of outliers and can detect multiple outliers.

A Grubbs test can also test for multiple outliers but at a cost of accumulating type I error. The reason why it is used in this tutorial is because it can be implemented very easily by hand and using MATLAB

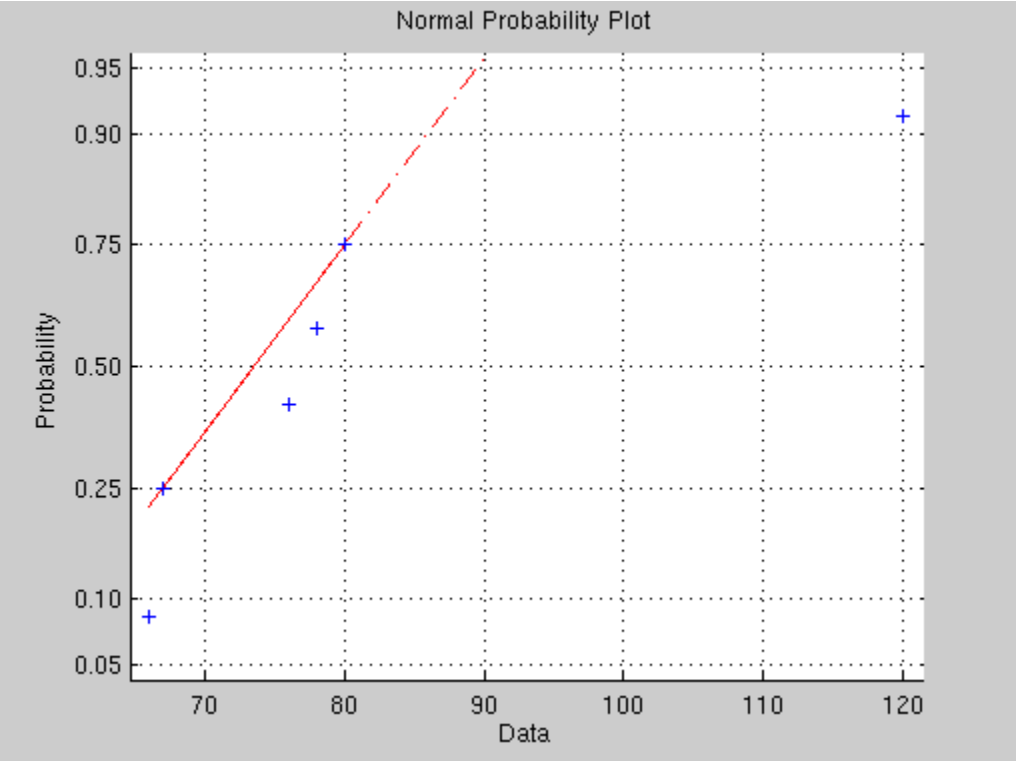
Grubbs Test for Outliers

- Quantitative test to eliminate outliers on approximately normal distribution
- Test one point at a time with the null hypothesis being the point is not an outlier.
- This test can be iterated multiple times in order to test for more than one outlier.
- The test basically calculates how far the outlier is from the mean and compares this to a critical value. If the outlier is too far out then it can be justifiably eliminated.
- For more information and a step by step tutorial, check out: http://www.itl.nist.gov/div898/handbook/eda/section3/eda35h1.htm

Example:

We can conduct this test on the data set we had above of heart rates:
80, 67, 76, 78, 66, **120**

- First make a normal probability plot (if not normal, then this test cannot be used)



- This plot looks almost linear except for the point at 120. Therefore this is probably an outlier
- Use the Grubbs one sided test because you know it is a large outlier

$$G = \frac{Y_{max} - \bar{Y}}{s} = \frac{120 - 81.2}{19.9} = 1.95$$

$$G_{crit} = \frac{N-1}{\sqrt{N}} * \sqrt{\frac{t^2_{\frac{alpha}{N},N-2}}{N-2+t^2_{\frac{alpha}{N},N-2}}} = \frac{6-1}{\sqrt{6}} * \sqrt{\frac{3.964^2}{6-2+3.964^2}} = 1.82$$

$$Y_{max} - the\ outlier\ that\ you're\ checking$$

$$\bar{Y} - mean\ of\ data$$

$$s - standard\ deviation\ of\ data$$

$$N - number\ of\ data\ points$$

$$t_{\frac{alpha}{N},N-2} - the\ t\ value\ for\ probability\ of\ \frac{alpha}{N}\ and\ dF\ of\ N-2$$

- Because $G>G_{crit}$, the null can be rejected and the point is an outlier.
- This point can be removed if you choose to remove it.

MATLAB Implementation:

- The MATLAB code is here: http://www.mathworks.com/matlabcentral/fileexchange/3961-deleteoutliers
- Download the file from that website and save the file to the folder that you will be using in MATLAB.
- The file is a function so you will have to call it from either another MATLAB file or the command line.
- Before calling the function, make sure to put your data into a row or column vector

```
data= [80, 67, 76, 78, 66, 120];
alpha=0.05;
[newdata, idx, outliers] = deleteoutliers(data, alpha)
```

```
>> data= [80, 67, 76, 78, 66, 120];
>> alpha=0.05;
>> [newdata, idx, outliers] = deleteoutliers(data, alpha)

newdata =

    80    67    76    78    66

idx =

     6

outliers =

   120
```

Significance Level 0f 0.05

New Data without Outliers

The index of where the outlier was

The value of the outlier

- This program will automatically conduct the Grubb's test multiple times until all the outliers are deleted

Example with Multiple Outliers

- The same exact same command can be run on data sets that have more than one outlier.

  data= [80, 67, 76, 78, 66, 120 200];
  alpha=0.05;
  [newdata, idx, outliers] = deleteoutliers(data, alpha)

```
>> data= [80, 67, 76, 78, 66, 120 200];
>> alpha=0.05;
>> [newdata, idx, outliers] = deleteoutliers(data, alpha)

newdata =

    80    67    76    78    66

idx =

     6     7

outliers =

   120   200
```

- As you can see, the MATLAB version of the Grubbs test will continuously iterate and eliminate all the significant outliers.

Conclusion

There are many reasons to remove or to keep outliers but there needs to be a statistical test done in order to remove any point in a data set. Once a Grubbs tests is done and the point is shown to be an outlier then you can make a choice of whether to take the point out or not but if the point does not pass the Grubbs test, then you should not take the point out.