

## Contents

- [Import data in Tab-separated columns](#)
- [Display all data in histograms by cores and MIS](#)
- [Statistical Power test](#)
- [Plots the results of the Statistical Power test](#)
- [Alternative formulation of Statistical Power test](#)
- [Import valve area measurements from two samples](#)
- [Test data for normality vs. Chi-square distribution \(Chi-square test\)](#)
- [Optional: Test them for normality vs. standard normal distribution](#)
- [Kolmogorov-Smirnov test](#)
- [Apply Kolmogorov-Smirnov test to each of the three cores \(compare MIS 1 to MIS 2 populations\)](#)
- [Apply the K-S test to all measurements from MIS 1 and 2 \(irrespective of core\)](#)
- [Apply Mann-Whitney U-test \(ranksum in MATLAB = Wilcoxon ranksum test\)](#)
- [Apply the Mann-Whitney U-test to all measurements from MIS 1 and 2](#)

### Import data in Tab-separated columns

As each column has a different number of datapoints, in order to avoid the matrix being filled by zeroes/NaN till the maximum row number, we specify, when reading in data and assigning variable names, the number of datapoints per column

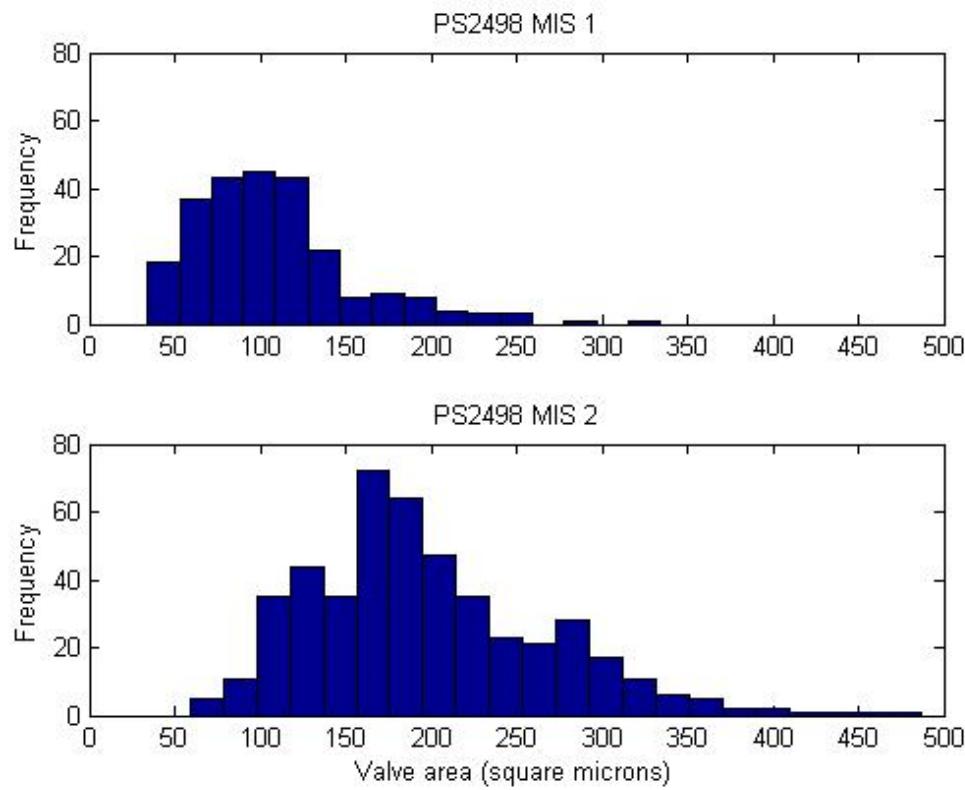
```
MIS1and2Area = dlmread('AlldataMIS1and2.txt','\t',0,0);
PS2498MIS1 = MIS1and2Area(1:245,1);
PS2498MIS2 = MIS1and2Area(1:467,2);
PS2499MIS1 = MIS1and2Area(1:88,3);
PS2499MIS2 = MIS1and2Area(1:1269,4);
PS1654MIS1 = MIS1and2Area(1:222,5);
PS1654MIS2 = MIS1and2Area(1:178,6);
AllMIS1 = [PS2498MIS1',PS2499MIS1',PS1654MIS1'];
AllMIS2 = [PS2498MIS2',PS2499MIS2',PS1654MIS2'];
```

### Display all data in histograms by cores and MIS

In order to get an idea of the structure of the data, and directly compare glacial and interglacial "population structure" Median valve area for all interglacial (MIS 1) measurements is: 112.31 Median valve area for all glacial (MIS 2) measurements is: 188.85

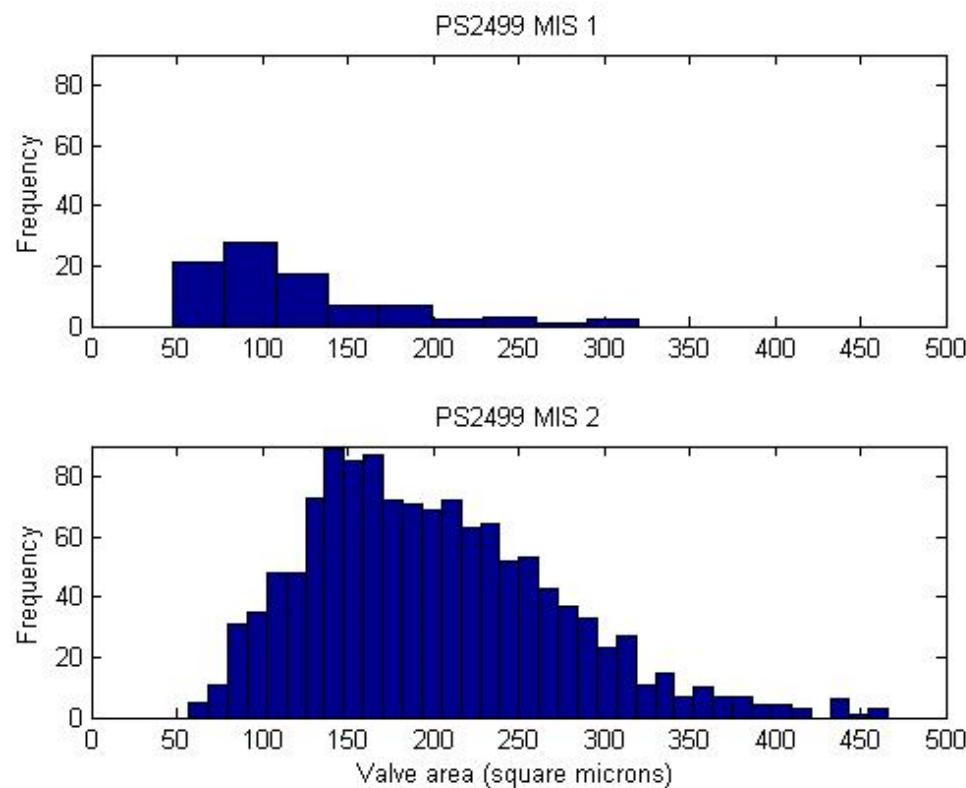
```
subplot(2,1,1); hist(PS2498MIS1,16)
title('PS2498 MIS 1')
ylabel('Frequency');
axis([0 500 0 80])

subplot(2,1,2); hist(PS2498MIS2,22)
title('PS2498 MIS 2')
xlabel('Valve area (square microns)');
ylabel('Frequency');
axis([0 500 0 80])
```



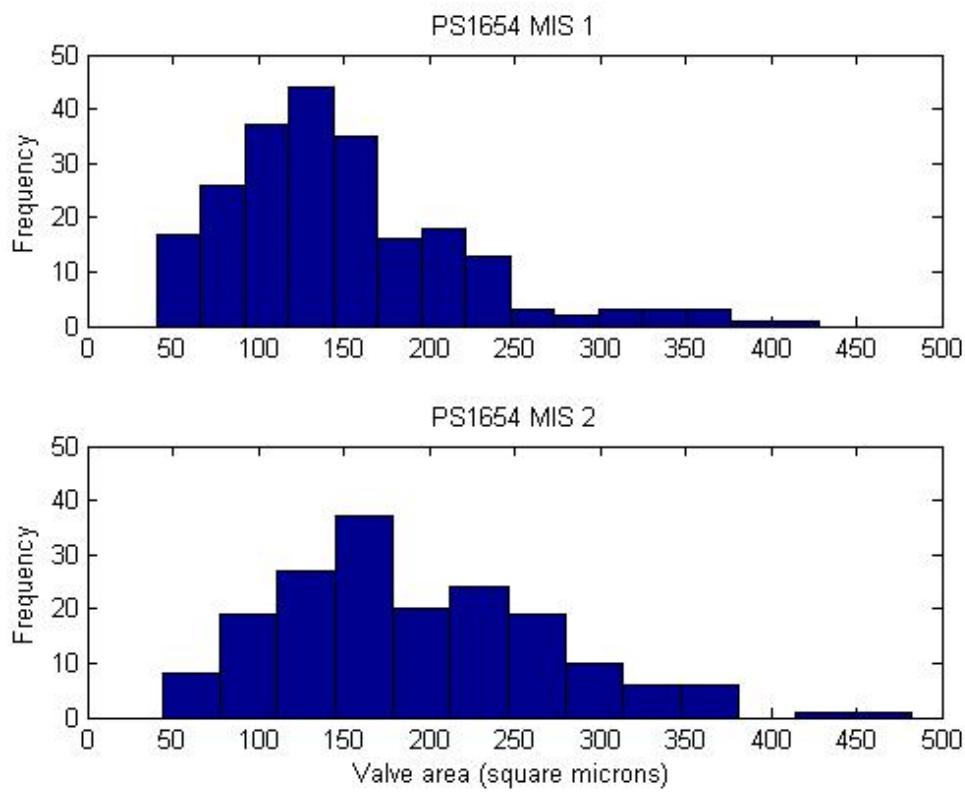
```
clf
subplot(2,1,1); hist(PS2499MIS1,9)
    title('PS2499 MIS 1')
    ylabel('Frequency');
    axis([0 500 0 90])

subplot(2,1,2); hist(PS2499MIS2,36)
    title('PS2499 MIS 2')
    xlabel('Valve area (square microns)');
    ylabel('Frequency');
    axis([0 500 0 90])
```



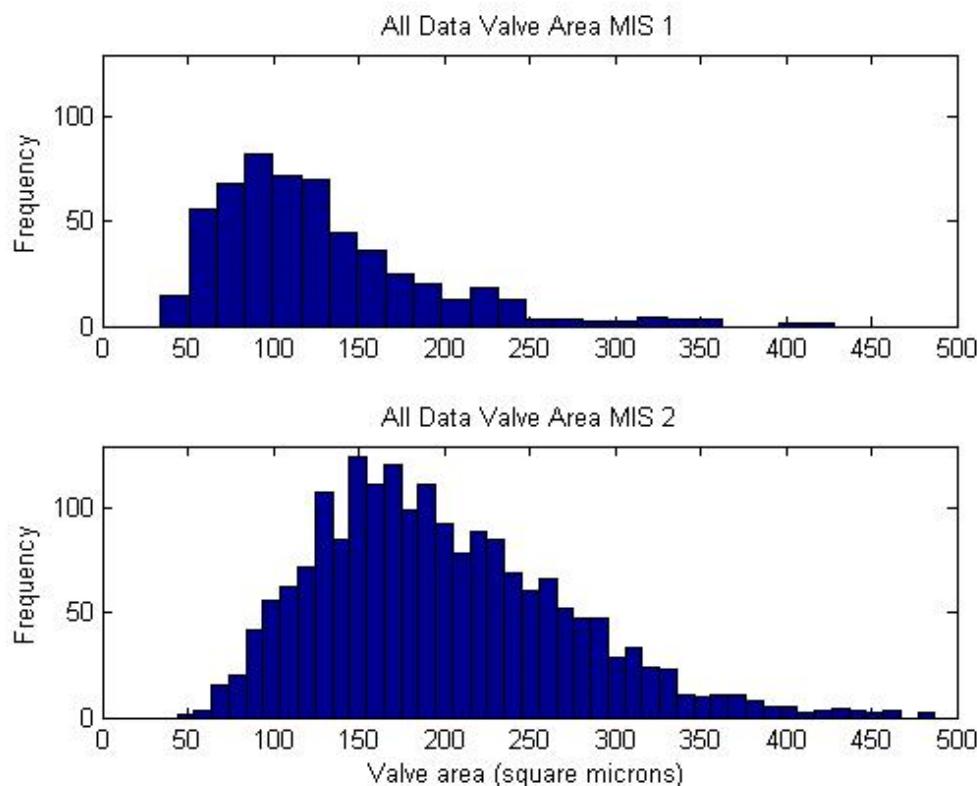
```
clf
subplot(2,1,1); hist(PS1654MIS1,15)
    title('PS1654 MIS 1')
    ylabel('Frequency');
    axis([0 500 0 50])

subplot(2,1,2); hist(PS1654MIS2,13)
    title('PS1654 MIS 2')
    xlabel('Valve area (square microns)');
    ylabel('Frequency');
    axis([0 500 0 50])
```



```
clf
subplot(2,1,1); hist(AllMIS1,24)
    title('All Data Valve Area MIS 1')
    ylabel('Frequency');
    axis([0 500 0 130])

subplot(2,1,2); hist(AllMIS2,44)
    title('All Data Valve Area MIS 2')
    xlabel('Valve area (square microns)');
    ylabel('Frequency');
    axis([0 500 0 130])
```



```
Median_valve_area_MIS1_Interglacial = median(AllMIS1)
Median_valve_area_MIS2_Glacial = median(AllMIS2)
```

```
Median_valve_area_MIS1_Interglacial =
```

```
112.3145
```

```
Median_valve_area_MIS2_Glacial =
```

```
188.8507
```

## Statistical Power test

This test provides the number of measured specimens (n) which needs to be measured in order to be able to separate, via a t-test and at 0.9 power, the estimated average value for the population from another measurement within the same population

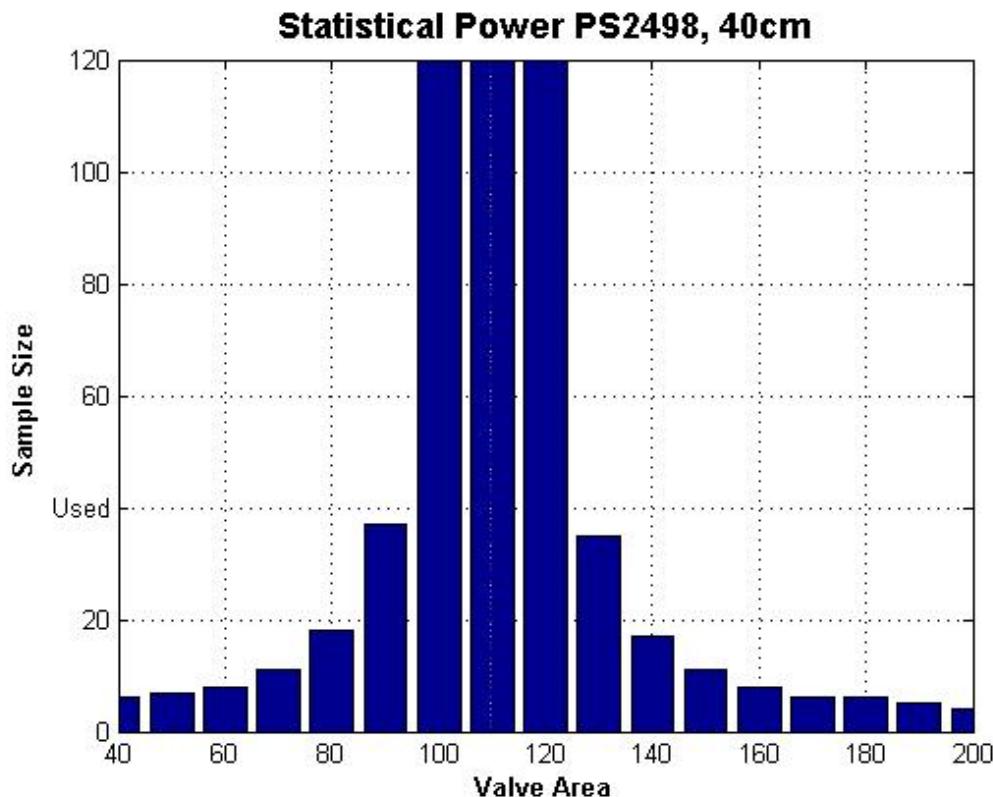
```
Mat_out=zeros(46,3);
ind_row=0;
for avgsize = 40:10:490
    ind_row=ind_row+1;
    n_PS2498_40cm=sampsizepwr('t',[109.66 35.88],avgsize);
    n_PS1654_1202cm=sampsizepwr('t',[189.36 90.68],avgsize);
    Mat_out(ind_row,1)=avgsize;
    Mat_out(ind_row,2)=n_PS2498_40cm;
    Mat_out(ind_row,3)=n_PS1654_1202cm;
end
```

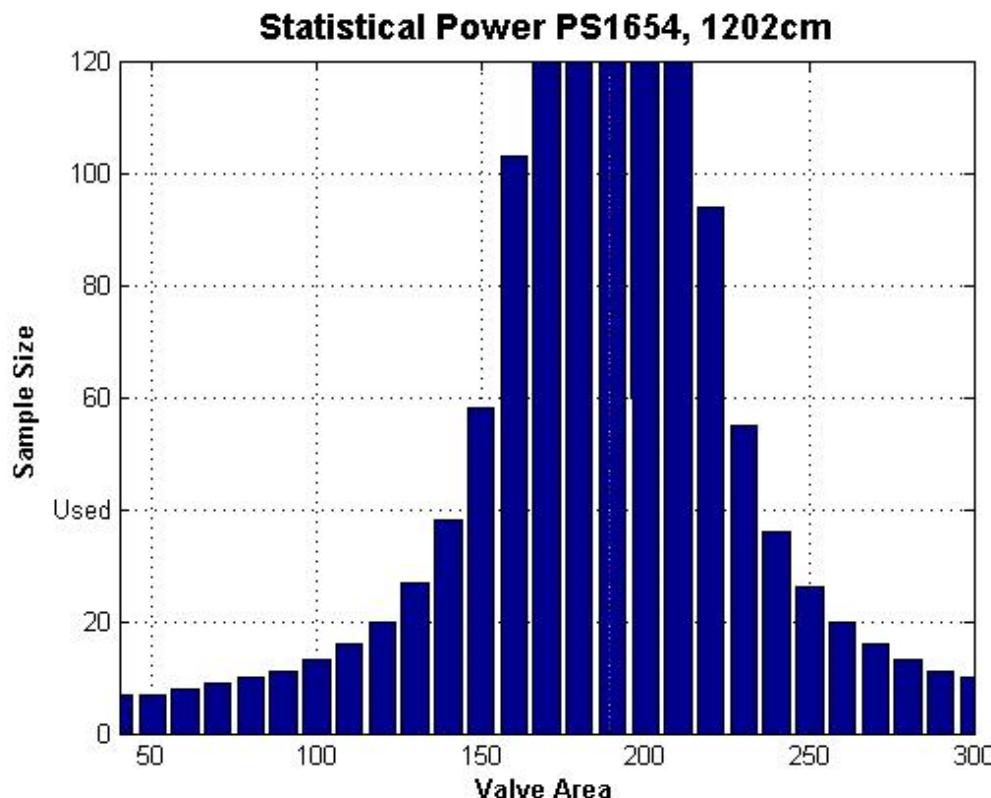
## Plots the results of the Statistical Power test

We apply it to the sample with maximum (PS2498\_40cm) and minimum (PS1654\_1202cm) standard deviation values for the Termination I dataset

```
figure
bar(Mat_out(1:46,1),Mat_out(1:46,2))
    title('Statistical Power PS2498, 40cm','fontweight','b','fontsize',14)
    xlabel('Valve Area','fontweight','b')
    ylabel('Sample Size','fontweight','b')
    axis([40,200,0,120])
    set(gca,'YTickLabel','0|20|Used|60|80|100|120')
    grid on
    hold on
    vertical_at_average = 0:1:120;
    plot(109.66,vertical_at_average)

figure
bar(Mat_out(1:46,1),Mat_out(1:46,3))
    title('Statistical Power PS1654, 1202cm','fontweight','b','fontsize',14)
    xlabel('Valve Area','fontweight','b')
    ylabel('Sample Size','fontweight','b')
    axis([40,300,0,120])
    set(gca,'YTickLabel','0|20|Used|60|80|100|120')
    grid on
    hold on
    vertical_at_average = 0:1:120;
    plot(189.36,vertical_at_average)
```





### Alternative formulation of Statistical Power test

What is the detectable/separable area value with a specified sample size N (40 in our case), and power 0.90 We test this for the whole MIS1 measurements, and the whole MIS2 measurements

```

average_MIS1 = mean(AllMIS1);
average_MIS2 = mean(AllMIS2);
stdev_MIS1 = std(AllMIS1);
stdev_MIS2 = std(AllMIS2);
average_MIS1, separable_area_MIS1=sampsizepwr('t',[average_MIS1 stdev_MIS1],[],0.90,40);
separable_area_MIS2step1=sampsizepwr('t',[average_MIS2 stdev_MIS2],[],0.90,40);
separable_area_MIS2=average_MIS2-(separable_area_MIS2step1-average_MIS2);
average_MIS2, separable_area_MIS2

average_MIS1 =
126.1805

separable_area_MIS1 =
158.5018

average_MIS2 =
198.3935

separable_area_MIS2 =
160.1666

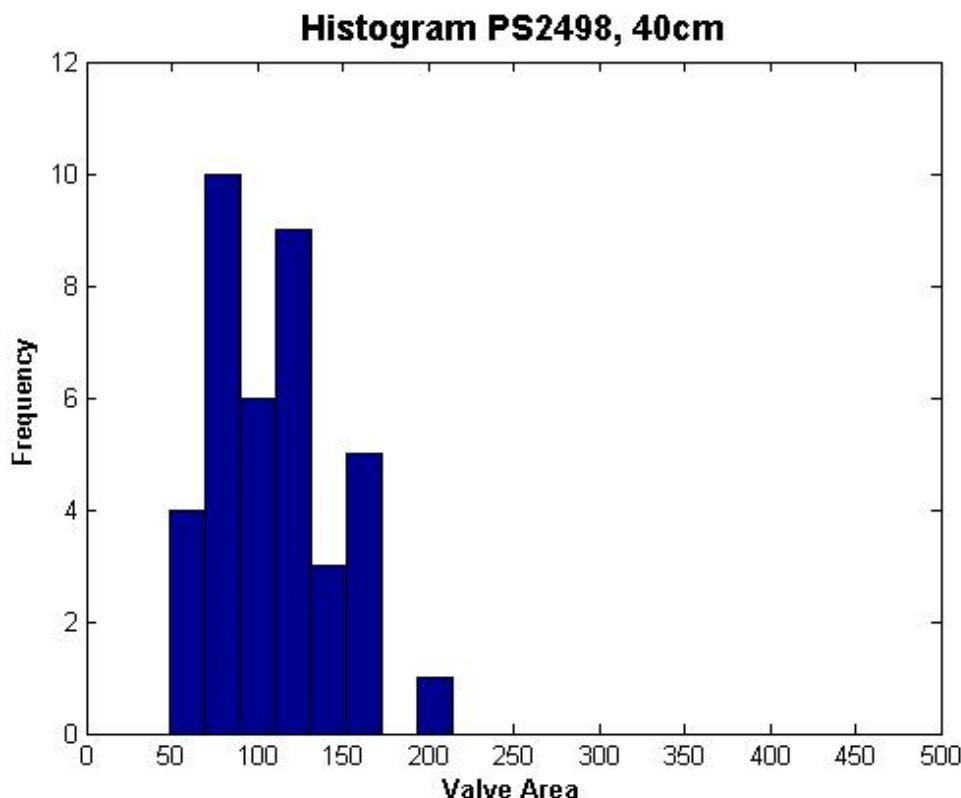
```

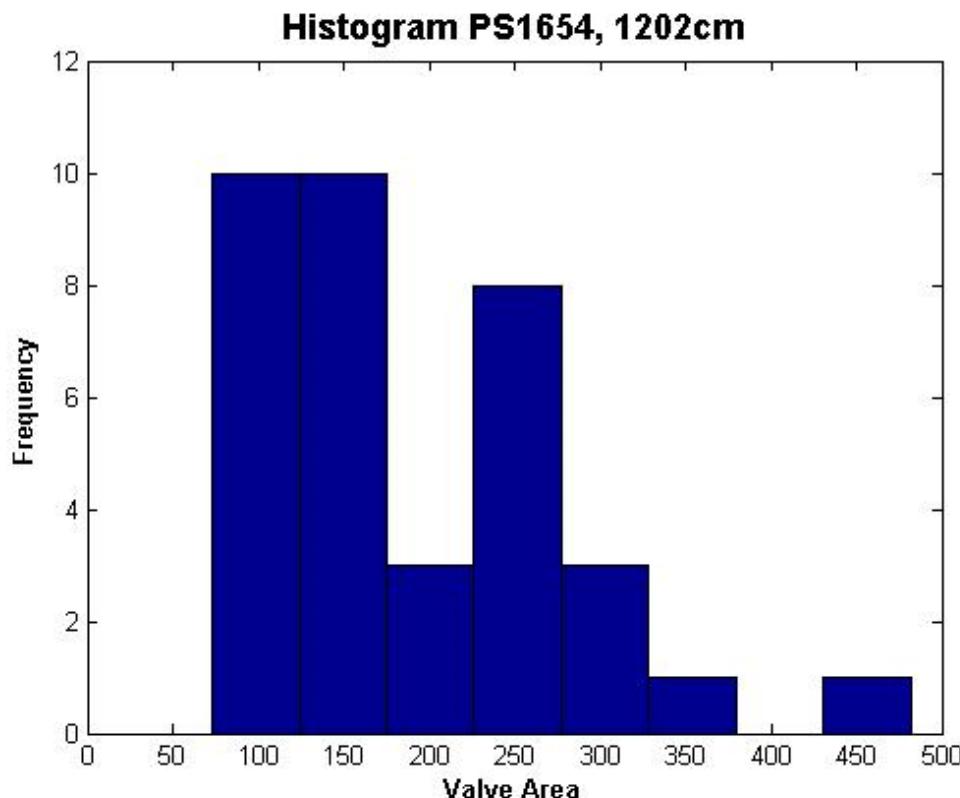
### Import valve area measurements from two samples

Two samples are analyzed, the one with minimum (PS2498\_40cm) and maximum

```
% (PS1654_1202cm) standard deviation values for the Termination I dataset
PS2498Area40cm=load('PS2498_40cm.txt');
PS1654Area1202cm=load('PS1654_1202cm.txt');
figure
hist(PS2498Area40cm,8)
    title('Histogram PS2498, 40cm','fontweight','b','fontsize',14)
    xlabel('Valve Area','fontweight','b')
    ylabel('Frequency','fontweight','b')
    axis([0,500,0,12])

figure
hist(PS1654Area1202cm,8)
    title('Histogram PS1654, 1202cm','fontweight','b','fontsize',14)
    xlabel('Valve Area','fontweight','b')
    ylabel('Frequency','fontweight','b')
    axis([0,500,0,12])
```





### Test data for normality vs. Chi-square distribution (Chi-square test)

We test for two samples only, but possible to do the same for all of them

```
[h,p,stats] = chi2gof(PS2498Area40cm,'nbins',10)
[h,p,stats] = chi2gof(PS1654Area1202cm,'nbins',10)
[h,p,stats] = chi2gof(AllMIS1,'nbins',10)
[h,p,stats] = chi2gof(AllMIS2,'nbins',10)

h =
0

p =
0.4040

stats =
chi2stat: 1.8124
    df: 2
edges: [48.4500 81.5440 98.0910 114.6380 131.1850 213.9200]
    O: [11 4 7 7 9]
    E: [8.2343 5.9630 6.9004 6.4798 10.4223]

h =
1

p =
0.0098
```

```

stats =
chi2stat: 9.2577
    df: 2
edges: [73.0200 113.9210 154.8220 195.7230 236.6240 482.0300]
    O: [8 11 2 5 10]
    E: [7.2979 5.3611 6.3476 6.1532 10.8402]

h =
1

p =
6.6912e-018

stats =
chi2stat: 86.6751
    df: 4
edges: [1x8 double]
    O: [95 186 132 68 39 17 18]
    E: [1x7 double]

h =
1

p =
1.9218e-034

stats =
chi2stat: 171.7409
    df: 6
edges: [1x10 double]
    O: [58 300 483 429 307 191 84 38 24]
    E: [1x9 double]

```

### Optional: Test them for normality vs. standard normal distribution

```

[h,p] = chi2gof(PS2498Area40cm,'cdf',@normcdf)
[h,p] = chi2gof(PS1654Area1202cm,'cdf',@normcdf)
[h,p] = chi2gof(AllMIS1,'cdf',@normcdf)
[h,p] = chi2gof(AllMIS2,'cdf',@normcdf)

```

Warning: After pooling, some bins still have low expected counts.  
The chi-square approximation may not be accurate

```

h =
1

p =
0

```

Warning: After pooling, some bins still have low expected counts.

```
The chi-square approximation may not be accurate
```

```
h =
```

```
1
```

```
p =
```

```
0
```

```
Warning: After pooling, some bins still have low expected counts.  
The chi-square approximation may not be accurate
```

```
h =
```

```
1
```

```
p =
```

```
0
```

```
Warning: After pooling, some bins still have low expected counts.  
The chi-square approximation may not be accurate
```

```
h =
```

```
1
```

```
p =
```

```
0
```

## Kolmogorov-Smirnov test

As the previous tests strongly suggest non-normal distributions, a non-parametric test like Kolmogorov-Smirnov(rather than a t-test, which requires a normal distribution) is necessary to evaluate whether two sampled populations are statistically different. We apply this test to all the valve area measurements for MIS 1 (interglacial) and MIS 2 (glacial), both running tests for each core (PS2498, PS2499, PS1654), and for the whole MIS 1 and MIS 2 measurements (i.e., grouping the three cores together).

### Apply Kolmogorov-Smirnov test to each of the three cores (compare MIS 1 to MIS 2 populations)

```
[h,p,ks2stat]=kstest2(PS2498MIS1,PS2498MIS2)  
[h,p,ks2stat]=kstest2(PS2499MIS1,PS2499MIS2)  
[h,p,ks2stat]=kstest2(PS1654MIS1,PS1654MIS2)
```

```
h =
```

```
1
```

```
p =
```

```
8.6461e-056
```

```
ks2stat =
```

```
0.6235
```

```
h =
```

```
1
```

```
p =
```

```
1.1019e-023
```

```
ks2stat =
```

```
0.5622
```

```
h =
```

```
1
```

```
p =
```

```
3.2180e-007
```

```
ks2stat =
```

```
0.2777
```

### Apply the K-S test to all measurements from MIS 1 and 2 (irrespective of core)

```
[h,p,ks2stat]=kstest2(AllMIS1,AllMIS2)
```

```
h =
```

```
1
```

```
p =
```

```
7.7999e-087
```

```
ks2stat =
```

```
0.4780
```

### Apply Mann-Whitney U-test (ranksum in MATLAB = Wilcoxon ranksum test)

Another non-parametric test to check whether two samples come from the same population.  
Apply the Mann-Whitney U-test to each of the three cores (comparing MIS 1 to MIS 2 populations)

```
[p,h,stats]=ranksum(PS2498MIS1,PS2498MIS2)
[p,h,stats]=ranksum(PS2499MIS1,PS2499MIS2)
[p,h,stats]=ranksum(PS1654MIS1,PS1654MIS2)
```

```
p =
```

```
5.1987e-061
```

```
h =
1

stats =
    zval: -16.4789
    ranksum: 44376

p =
3.1650e-025

h =
1

stats =
    zval: -10.3767
    ranksum: 22863

p =
1.7228e-008

h =
1

stats =
    zval: 5.6378
    ranksum: 42168
```

## Apply the Mann-Whitney U-test to all measurements from MIS 1 and 2

(irrespective of core)

```
[p,h,stats]=ranksum(AllMIS1,AllMIS2)

p =
1.1297e-101

h =
1

stats =
    zval: -21.4078
    ranksum: 3.6887e+005
```

*Published with MATLAB® 7.11*