



There are characteristics associated with each labeled peak in a spectrum, such as mass/charge, normalized intensity, area, and signal to noise ratio. These data are generated by Ciphergen's Peaks™ software and constitute the 'processed data' from a spectrum. PeakMiner uses this processed data to analyze the statistical distribution and relative intensities of individual proteins between groups. Analysis of these differences in presence or absence of some proteins and the relative expression of others across groups is used to build a 'profile' of a particular group.

### Step 1: Clustering

The first step in this process is to align or 'cluster' related proteins from different samples, that is, to assign a cluster number to every protein found in all spectra such that protein (1) in sample *A* represents the same protein (1) in sample *B*. The clustering algorithm first assembles a list of sorted protein masses from lowest to highest. For each mass, there is an associated 'error value', that is, the algorithm looks ahead to the next value and assigns a number related to the difference between value *A* and value *B* (see figure 1). For example, in the table below the algorithm would start at value *A* and calculate the error value as follows:

	Mass	Error value	
<b>A</b>	1002.12	0.000502312	<b>errorVal for A = (B – A) / B = 0.000502</b>
<b>B</b>	1002.62	0.000706998	
<b>C</b>	1003.33	0.000475103	
<b>D</b>	1003.81	0.49857172	
<b>E</b>	2001.89	0.000326835	
<b>F</b>	2002.55	0.000111027	

Figure 1

In an ideal situation, the largest local error value encountered is where the split between cluster 1 and cluster 2 is made (between **D** and **E**). However, the split is also controlled by the *mass window* selected by the user. For instance, if a mass window of 0.3% was specified, then the group of values bracketed by *A* thru *D* are averaged and if any masses are 0.3% away from the mean, another split is made. However, when there are multiple peaks from the same spectrum within the mass window, they cannot be included in the same cluster as they obviously represent different proteins. In the Zero Discard Method algorithm, when replicates are encountered, the replicate mass farthest away from the mean is assigned a cluster value of "0" and can later be re-clustered.

### Step 2: Cluster Distribution Analysis

The best way to explain distribution analysis is by example. Imagine you are analyzing three profile groups called Cancer, Benign, and Normal and that there are 100

samples from each group. After clustering the peak masses you find that the distribution of cluster 1 (mass = 2000 Da) is as follows:

	<u>Cancer</u>	<u>Benign</u>	<u>Normal</u>
Cluster 1:	78	12	3
Cluster 2:	96	92	100
...			

The protein represented by cluster 1 is found in 78% of all the cancer samples that were analyzed but only 3% of the normal samples. Therefore, Cluster 1 would be considered a good classifier of the cancer group. PeakMiner analyzes the differences in peak distribution of tens or hundreds of clusters to construct the 'Presence/Absence' element of a profile.

### Step 3: Expression Analysis

In addition to the mass values associated with each cluster, there are intensity values that can also reveal differences between groups. For each cluster, and again for each group *within* a cluster, the mean, standard deviation, and coefficient of variation of each cluster's intensity are calculated. The resulting intensity data for the two clusters illustrated above might be as follows:

	<u>Cancer</u>	<u>Benign</u>	<u>Normal</u>
Cluster 1:	9.6 ± 2.1	10.4 ± 3.6	8.7 ± 2.2
Cluster 2:	<b>32.2 ± 7</b>	7.7 ± 4.6	4.6 ± 3.1

The protein represented in cluster 2 is over-expressed 7-fold relative to the normal group, while having a nearly equivalent incidence between groups. Cluster 2 could then be used as one of the classifiers of the cancer group in the cancer profile. Combinations of expression differences for each group are compiled and used to construct an expression profile.

### Step 4: Classification

The PeakMiner Classification Algorithm is based on cumulative probability. For each cluster, the program calculates the probability that the distribution of that protein exceeds the normal expected distribution across the groups. Essentially, a separate profile is generated for the expression data and the Presence/Absence (P/A) data that takes into account each cluster's Overall Incidence (how often does that peak appear in the whole sample population), the group incidence (what percent of the samples in a particular group have that peak), and for expression data, CV% of the intensity. In addition, for the P/A data, a function was written that calculates the degree of variability of a cluster, so that the incidence for a peak of m/z = 5000 where Cancer (100%), Benign(2%), Normal(0%) would be more weighted for Cancer than a peak at m/z = 2000 where Cancer(50%), Benign(10%), Normal(5%).