

# Data Collection

Libg++ currently provides two classes for *data collection* and analysis of the collected data.

## SampleStatistic

Class **SampleStatistic** provides a means of accumulating samples of **double** values and providing common sample statistics.

Assume declaration of **double x**.

<b>SampleStatistic a;</b>	declares and initializes a.
<b>a.reset();</b>	re-initializes a.
<b>a += x;</b>	adds sample x.
<b>int n = a.samples();</b>	returns the number of samples.
<b>x = a.mean;</b>	returns the means of the samples.
<b>x = a.var()</b>	returns the sample variance of the samples.
<b>x = a.stdDev()</b>	returns the sample standard deviation of the samples.
<b>x = a.min()</b>	returns the minimum encountered sample.

<b>x = a.max()</b>	returns the maximum encountered sample.
<b>x = a.confidence(int p)</b>	returns the p-percent ( $0 \leq p < 100$ ) confidence interval.
<b>x = a.confidence(double p)</b>	returns the p-probability ( $0 \leq p < 1$ ) confidence interval.

## SampleHistogram

Class **SampleHistogram** is a derived class of **SampleStatistic** that supports collection and display of samples in bucketed intervals. It supports the following in addition to **SampleStatistic** operations.

**SampleHistogram h(double lo, double hi, double width);** declares and initializes h to have buckets of size width from lo to hi. If the optional argument width is not specified, 10 buckets are created. The first bucket and also holds samples less than lo, and the last one holds samples greater than hi.

**int n = h.similarSamples(x)** returns the number of samples in the same bucket as x.

**int n = h.inBucket(int i)** returns the number of samples in bucket i.

**int b = h.buckets()** returns the number of buckets.

**h.printBuckets(ostream s)** prints bucket counts on ostream s.

**double bound = h.bucketThreshold(int i)** returns the upper bound of bucket i.

