# Overview

This document presents the UML diagram for the Statistics class, a C++ template class designed for statistical calculations on a dataset. The dataset can be treated as a sample (true) or population (false). The QuartileValues struct is a nested type within Statistics, used to store quartile-related data.

# UML

Template class for statistical calculations on a dataset.   
Dataset type: true = sample, false = population.   
Assumes the array is sorted for some operations (e.g., median, mode).

|  |
| --- |
| Statistics Class |
| * MUTATORS * bool datasetType |
| * ACCESSORS * DynamicArray<T> dArr * Statistics() * ~Statistics() * bool setDatasetType(const bool& populationOrSample) * bool getDatasetType() const * T minimum() const * T maximum() const * double range() const * size\_t size() const * T sum() const * double sumPowSub(const double& exponent, const double& subtractValue) const * + double sumPowSubDiv(const double& exp, const double& sub, const double& div) const * double mean() const * double median() const * double mode() const * double standardDeviation() const * double variance() const * double midrange() const * struct QuartileValues"  {  + double Q1  + double Q2  + double Q3  + bool q1Known  + bool q3Known  } * QuartileValues quartilesCalculation() const * void quartiles() const * void interquartile() const * void outliers() const * void sumOfSquares() const * void meanAbsoluteDeviation() const * void rootMeanSquare() const * void standardErrorMean() const * double skewness() const * double kurtosis() const * double kurtosisExcess() const * double coefficientOfVariation() const * double relativeStandardDeviation() const * map displayFrequencyTable() const * void displayOut() |
|  |