**Theory**

**SEGMENTATION**

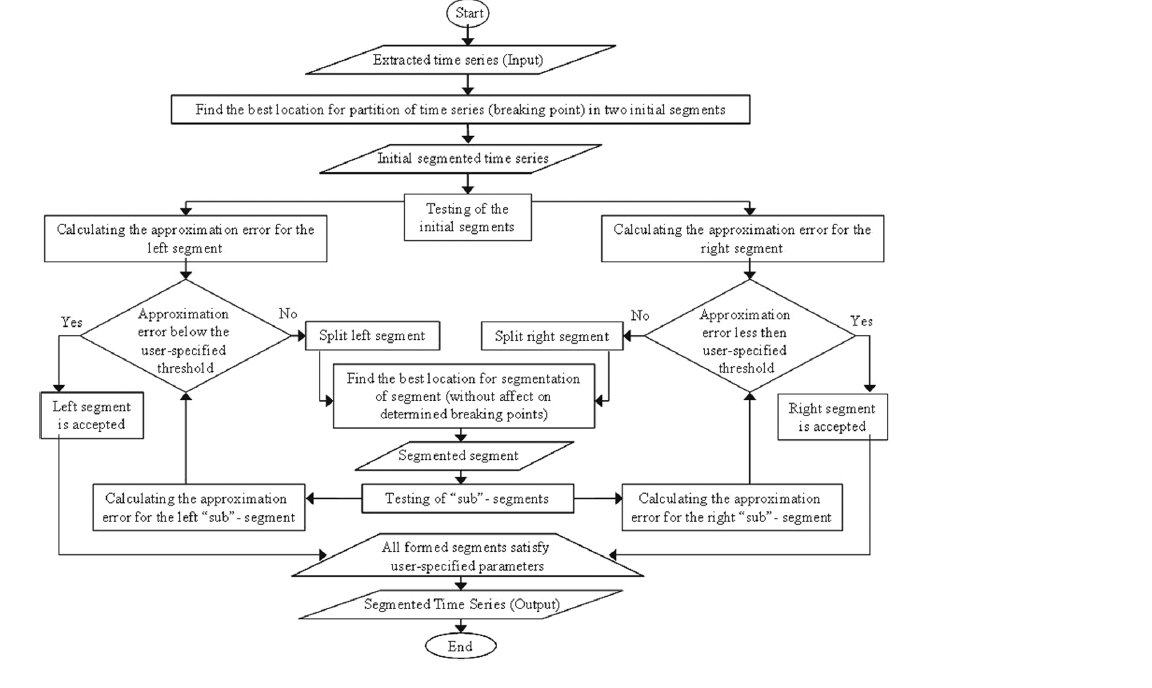
The process of decomposing a large dataset into relatively smaller homogenous chunks. Segmentation is used in Industries for Market segmentation or Customer segmentation, Image Segmentation etc. Segmentation can be either done on the basis of similarity of one or more features or on the basis of the similarity in the geometry of the data. The former one is known as Feature-based segmentation and the latter one is Geometric segmentation. We are using the Feature-based segmentation.

**SEGMENTATION ALGORITHMS**

**Top Down Algorithm**

The segments are computed in the Divide and Conquer manner. In this approach, Initially, the whole dataset is considered as a single segment then in subsequent iterations anchor is placed when the difference between the features of subsequent segments exceeds the user-defined threshold. That anchor acts as the start boundary of the new segment and the end boundary of the previous segment. And then for the newly created segments, the process is repeated until the user-defined uniformity is achieved.

Figure 4 : Flowchart from Top Down Algorithm

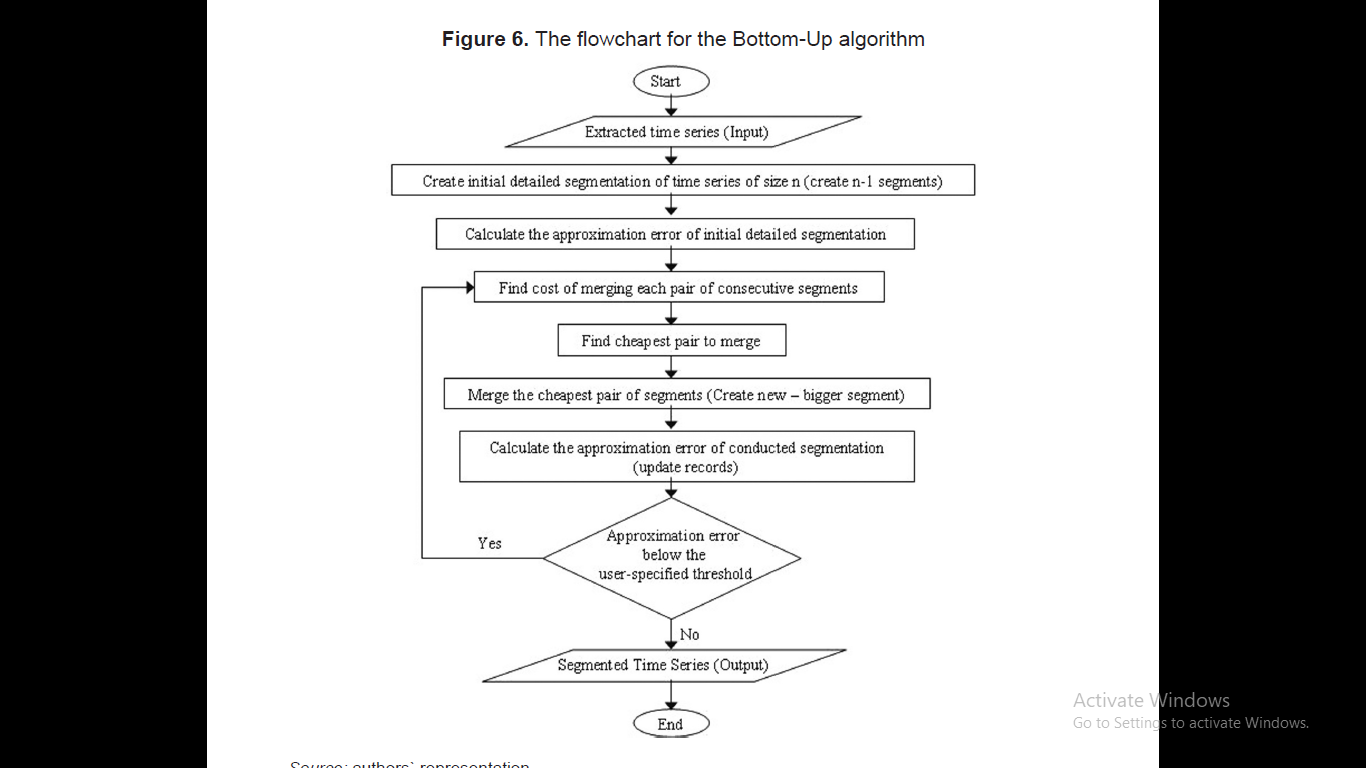


Source : [ 1 ]

**Bottom Up Algorithm**

The segments are computed in the Iterative merge manner. In this approach, Initially, the whole dataset is divided into atomic chunks. Then in further iterations, the pair of adjacent chunks with the lowest difference are merged. This process is repeated until the maximum possible chunks sizes are available which maintains the user-defined homogeneity.

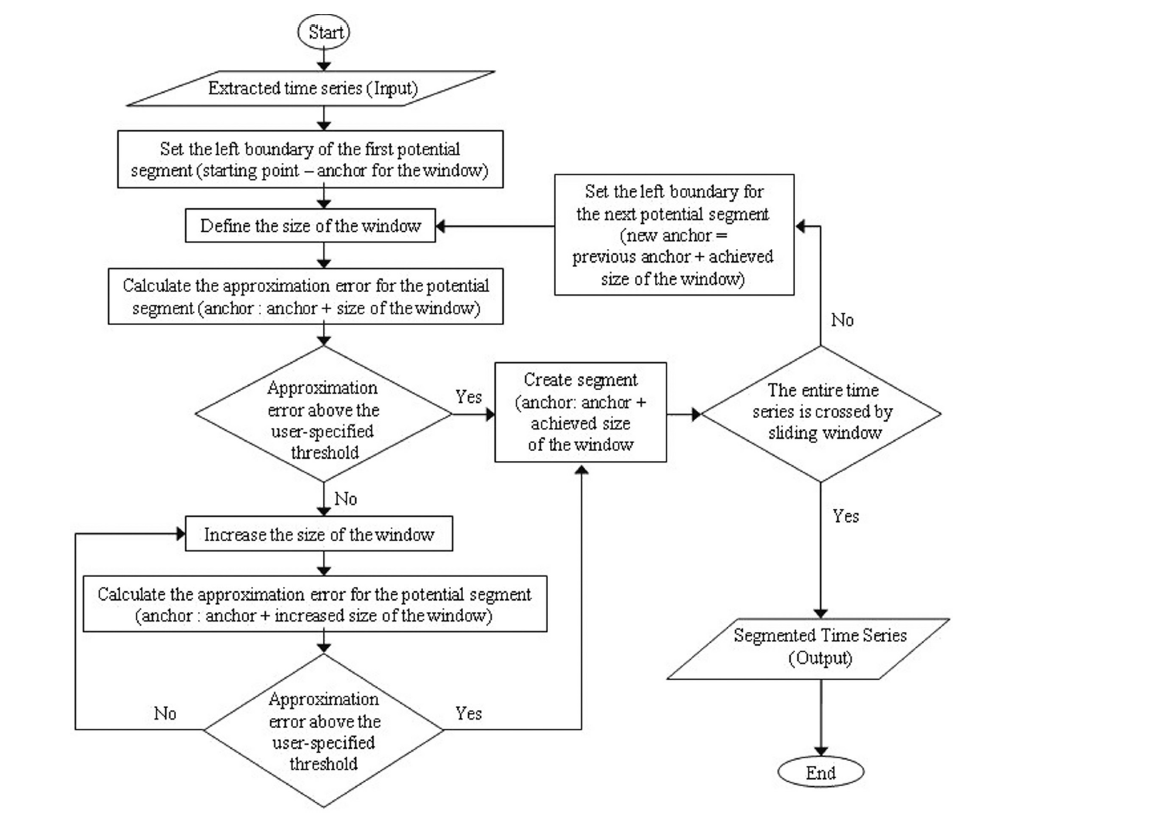
Figure 5 : Flow Chart for Bottom Up Algorithm



Source : [ 1 ]

**Sliding Window Algorithm**

This algorithm uses a Window of the user-specified length and the window is shifted over the data in iterations. The anchor is placed when the difference created by including one more data point is more than the predefined threshold.

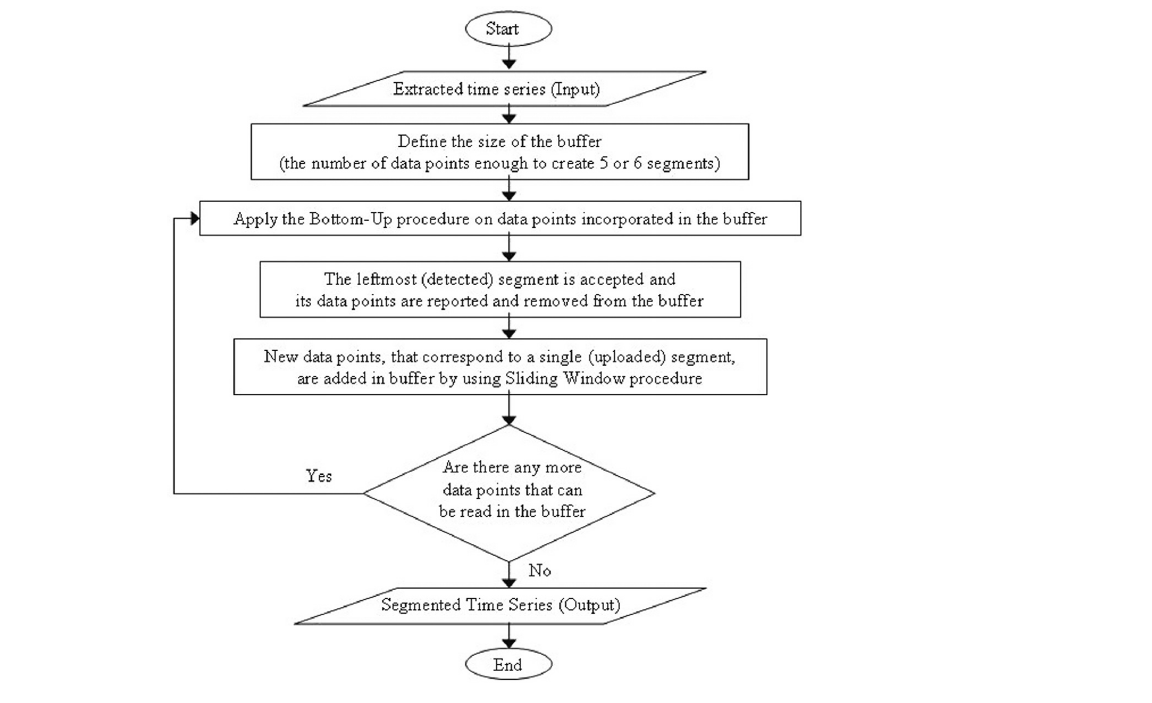
Figure 6: Flowchart for Sliding Window Algorithm

Source : [ 1 ]

**Sliding Window and Bottom Up ( SWAB ) Algorithm :**

This algorithm is the hybrid of the above two mentioned algorithms. Two fixed sized windows are created adjacent to each other and in iterations, those windows are shifted over the data. The distribution of data of both windows are then compared if the difference exceeds the limit specified by the user, then the anchor is placed between the windows.

Figure 6: Flow Chart for SWAB Algorithm



Source : [ 1 ]

**FUZZY LOGIC**

Fuzzy logic is an approach to perform computation based on "degrees of truth" rather than the usual "true or false" (1 or 0) Boolean logic on which the modern computer is based.

Fuzzy logic includes 0 and 1 as extreme cases of certainty (or "the state of matters" or "fact") but also includes the various states of certainty in between, for example, the result of a comparison between two things could be not "tall" or "short" but " 0.38 degree of tallness."

Fuzzy logic seems closer to the way our brains work. We aggregate data and form a number of partial truths which we aggregate further into higher truths which in turn, when certain thresholds are exceeded, cause certain further results such as motor reaction. A similar kind of process is used in neural networks, expert systems and other artificial intelligence applications. Fuzzy logic is essential in the development of human-like capabilities for AI, sometimes referred to as artificial general intelligence: the representation of generalized human cognitive abilities in software so that, faced with an unfamiliar task, the AI system could find a solution.

**{\displaystyle (S(x)+S(-x))\cdot (S(y)+S(-y))\cdot (S(z)+S(-z))=1}**

**Fuzzy logic operators**

Fuzzy logic works with membership values in a way that mimics Boolean logic. To this end, replacements for basic operators AND, OR, NOT are available. There are several ways to do this. A common replacement is called the Zadeh operators:

|  |  |
| --- | --- |
| Boolean | Fuzzy |
| AND(x,y) | MIN(x,y) |
| OR(x,y) | MAX(x,y) |
| NOT(x) | 1 – x |

For TRUE/1 and FALSE/0, the fuzzy expressions produce the same result as the Boolean expressions.

**FUZZY INFERENCE SYSTEM**

Fuzzy Inference System is the key unit of a fuzzy logic system having decision making as its primary work. It uses the “IF…THEN” rules along with connectors “OR” or “AND” for drawing essential decision rules.

**Characteristics of Fuzzy Inference System**

* The output from FIS is always a fuzzy set irrespective of its input which can be fuzzy or crisp.
* It is necessary to have fuzzy output when it is used as a controller.
* A defuzzification unit would be there with FIS to convert fuzzy variables into crisp variables.

**Functional Blocks of FIS**

The following five functional blocks will help you understand the construction of FIS −

**Rule** **Base** − It contains fuzzy IF-THEN rules.

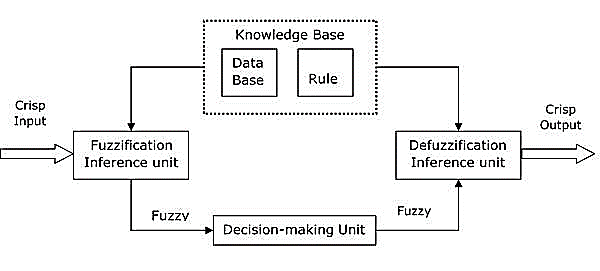
**Database** − It defines the membership functions of fuzzy sets used in fuzzy rules.

**Decision-making Unit** − It performs operation on rules.

**Fuzzification Interface Unit** − It converts the crisp quantities into fuzzy quantities.

**Defuzzification Interface Unit** − It converts the fuzzy quantities into crisp quantities. Following is a block diagram of fuzzy interference system.

Figure 7 : Block Diagram of Fuzzy Inference System



**MAHALANOBIS DISTANCE**

The Mahalanobis distance is a measure of the distance between a point P and a distribution D, introduced by P. C. Mahalanobis in 1936. It is a multi-dimensional generalization of the idea of measuring how many standard deviations away P is from the mean of D. This distance is zero if P is at the mean of D, and grows as P moves away from the mean along each principal component axis. The Mahalanobis distance measures the number of standard deviations from P to the mean of D. If each of these axes is re-scaled to have unit variance, then the Mahalanobis distance corresponds to standard Euclidean distance in the transformed space. The Mahalanobis distance is thus unitless and scale-invariant, and takes into account the correlations of the data set.



where,

de(t) = Mahalanobis Distance between two distribution

e1 = Mean of first distribution

e2 = Mean of second distribution

σ12 = Variance of first distribution

σ22 = Variance deviation of second distribution