**LBG Algorithm**

LBG algorithm is like a K-means clustering algorithm which takes a set of input vectors \( S = \{ x_i \in \mathbb{R}^d \mid i = 1, 2, \ldots, n \} \) as input and generates a representative subset of vectors \( C = \{ c_j \in \mathbb{R}^d \mid j = 1, 2, \ldots, K \} \) with a user specified \( K << n \) as output according to the similarity measure. For the application of Vector Quantization (VQ), \( d = 16 \), \( K = 256 \) or 512 are commonly used.

**LBG Algorithm**

1. Input training vectors \( S = \{ x_i \in \mathbb{R}^d \mid i = 1, 2, \ldots, n \} \).
2. Initiate a codebook \( C = \{ c_j \in \mathbb{R}^d \mid j = 1, 2, \ldots, K \} \).
3. Set \( D_0 = 0 \) and let \( k = 0 \).
4. Classify the \( n \) training vectors into \( K \) clusters according to \( x_i \in S_q \) if \( \| x_i - c_q \|_p \leq \| x_i - c_j \|_p \) for \( j \neq q \).
5. Update cluster centers \( c_j \), \( j = 1, 2, \ldots, K \) by \( c_j = \frac{1}{|S_j|} \sum_{x_i \in S_j} x_i \).
6. Set \( k \leftarrow k + 1 \) and compute the distortion \( D_k = \sum_{j=1}^{K} \sum_{x_i \in S_j} \| x_i - c_j \|_p \).
7. If \( (D_{k-1} - D_k)/D_k > \epsilon \) (a small number), repeat steps 4 \~ 6.
8. Output the codebook \( C = \{ c_j \in \mathbb{R}^d \mid j = 1, 2, \ldots, K \} \).

The convergence of LBG algorithm depends on the initial codebook \( C \), the distortion \( D_k \), and the threshold \( \epsilon \), in implementation, we need to provide a maximum number of iterations to guarantee the convergence.