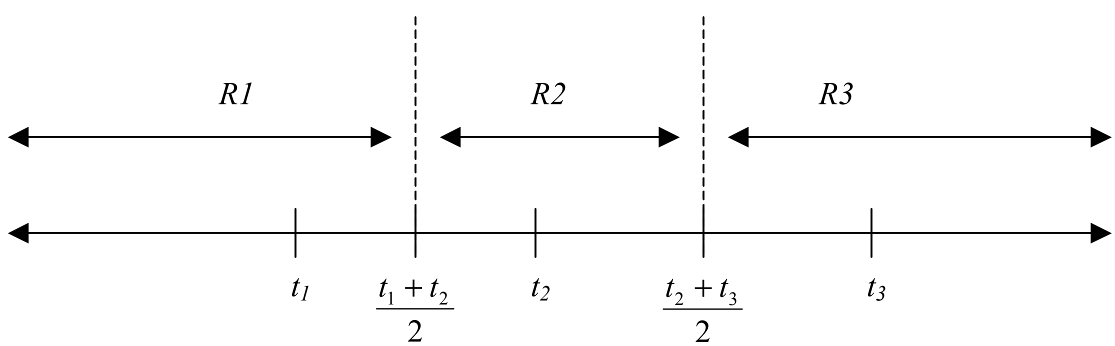
CSG142 – Digital Image Processing
Lloyd-Max Quantization
October 25, 2006
Corrected Nov 21, 2006

A quantization function is a function that maps a continuous range of input values to a single output value. For example, one useful quantization function is to simply round the input value to its closest integer:

$$Q(x) = \text{round}(x)$$

The Lloyd-Max quantizer (LMQ) is a quantization function that performs minimum mean squared error quantization. Suppose we want to find the best way to quantize a set of (continuous-valued) data to n discrete levels. If the n levels are known (let's say they are values t_1 through t_n), then the decision boundaries are at the midpoint between the values.

This concept is illustrated in the following diagram, where the real numbers are quantized into three values, t_1, t_2, and t_3. The values in region *R1* are quantized to t_1, the values in region *R2* are quantized to t_2, and the values in *R3* are quantized to t_3. The decision boundary between *R1* and *R2* is half way between t_1 and t_2, and the decision boundary between *R2* and *R3* is half way between t_2 and t_3.



It is easy to show that if levels t_1 through t_n are known, the best decision boundaries are at the midpoint between levels. However, how can we find these values t_1 through t_n in the first place? There are many ways to do this, but here I will describe an iterative method.

Iterative Lloyd-Max Quantizer

Let us define some terms:

t_i	Quantization value i. The quantization values should be ordered, such that $t_1 < t_2 < \cdots < t_n$.
s_i	The decision boundary between t_i and t_{i+1}.

Then, the algorithm is as follows

1) Estimate t_1 through t_n using any method. For example, if $v_{\min}, v_{\max}$ are the minimum and maximum input values, they could be placed uniformly throughout the range, at locations:

$$t_i = v_{\min} + \frac{i - 0.5}{n}\left(v_{\max} - v_{\min}\right)$$

2) Loop:
   a. Compute new values for s_1 through s_{n-1} that minimize mean squared error for current values of t_1 through t_n:

   $$s_i = \frac{t_i + t_{i+1}}{2}$$

   b. Compute new values for t_1 through t_n that minimize mean squared error for current values of s_1 through s_{n-1}. Thus, t_i is the mean of all input values that fall between s_{i-1} and s_i.

   It can happen that no values fall between s_{i-1} and s_i. In this case, you should choose a new location for t_i, while maintaining $t_1 < t_2 < \cdots < t_n$.

   c. Check for convergence. For example you may continue until the positions of the quantization levels have stopped changing, or if the average variance within each level has stopped changing.