Vector Quantization

Alex Sverdlov
alex@theparticle.com

1 Introduction

There are many quantization techniques. These notes describe something called "vector quantization". A vector in this sense is a tuple of numbers: a point in n-dimensional space.

These points could be sourced from n sensors, with the tuple representing the readings of all n sensors at any given time.

Suppose we have a large set of such points. Suppose also that we suspect there's a pattern to them. Perhaps the sensors are recording positions of arms, legs as someone's running. Or a frequency spectrum of speech. The input is many n-dimensional points. We would like to quantize that n-dimensional motion into one dimensional state jumping.

Assume we want to split the motion into M distinct states. The first M points get their own cluster.

For any subsequent point: find the closest cluster to the point. Also, find two nearest clusters (inter-cluster distance; closest to each other).

If new point is closer to its cluster than the two nearest clusters are to each other, then add point to the closest cluster. Otherwise merge two nearest clusters, and new point becomes its own cluster. No matter what, we end up with M clusters.

The cluster location is the average location of points in the cluster.

This can be implemented efficiently (in one pass over the data) by noting that clusters must only 'remember' the sum and counts of points in the clusters. That is enough to find the cluster location, to add points to the cluster, to merge two existing clusters, etc.

The output are M cluster centers.

For any future points, we find the closest cluster, and replace its n-dimensional coordinate with the cluster index, representing the state label.

Then we can build a state-to-state transition table, estimate probability of moving from state to state, etc.