color extraction

1. Kmeans
2. introduction

*k*-means clustering aims to [partition](https://en.wikipedia.org/wiki/Partition_of_a_set) *n* observations into *k* clusters in which each observation belongs to the [cluster](https://en.wikipedia.org/wiki/Cluster_(statistics)) with the nearest [mean](https://en.wikipedia.org/wiki/Mean), serving as a prototype of the cluster. This results in a partitioning of the data space into [Voronoi cells](https://en.wikipedia.org/wiki/Voronoi_cell" \o "Voronoi cell). (<https://en.wikipedia.org/wiki/K-means_clustering>)

1. details

Given an initial set of k means , the algorithm proceeds by alternating between two steps:

* 1. assignment step

Assign each observation to the cluster whose mean has the least squared Euclidean distance, this is intuitively the “nearest” mean. (Mathematically, Voronoi diagram generated by the means is applied to partition the observation).

where each is assigned to exactly one , even if it could be assigned to two or more of them.

* 1. update step

We then calculate the new means to be the centroids of the observations in the new cluster.

1. application

Given that the distance used by k-means clustering algorithm is squared Euclidean distance, it is a natural fit for color quantization in both RGB color space and CIE L\*a\*b\* color space. For these two color spaces, there are three channels in each one representing different color-related information, so we can use squared Euclidean distance to find the centroids of the color clusters which are selected to the part of the color theme palette.

After obtaining the color theme palette of the image, all colors can be remapped into the nearest selected color based on the squared Euclidean distance. (<https://lmcaraig.com/color-quantization-using-k-means/#selectionwithkmeans>)