K-means Implementation

on Iris Dataset

Data Mining and Machine Learning

Professor

蔡崇煒

Members

4103056002 資工三 杜杰

4103056029 資工三 黃冕

4103056041 資工三 陳仲彥

4104056009 資工二 洪浩祐

Abstract

K-means clustering is one of the most popular way of cluster analysis (clustering). Clustering, a branch of unsupervised machine learning, is aim for finding a pattern to describe hidden structure from "unlabeled" data. The examples (input) to the learner (algorithm) are unlabeled, and the output is what the learner have learned in the process.

Introduction

K-means clustering aims to partition ***N*** observations into ***k*** clusters in which each observation belongs to the cluster with the nearest mean (high cohesion) and best with low coupling.

Main idea

Given a set of observations $(x_1, x_2, \cdots, x_n)$, where each observation is a d-dimensional real vector, k-means clustering aims to partition the n observations into $k(\leq n)$ sets $S = \{S_1, S_2, \cdots, S_k\}$ so as to minimize the within-cluster sum of squares (WCSS) (i.e. variance). Formally, the objective is to find:

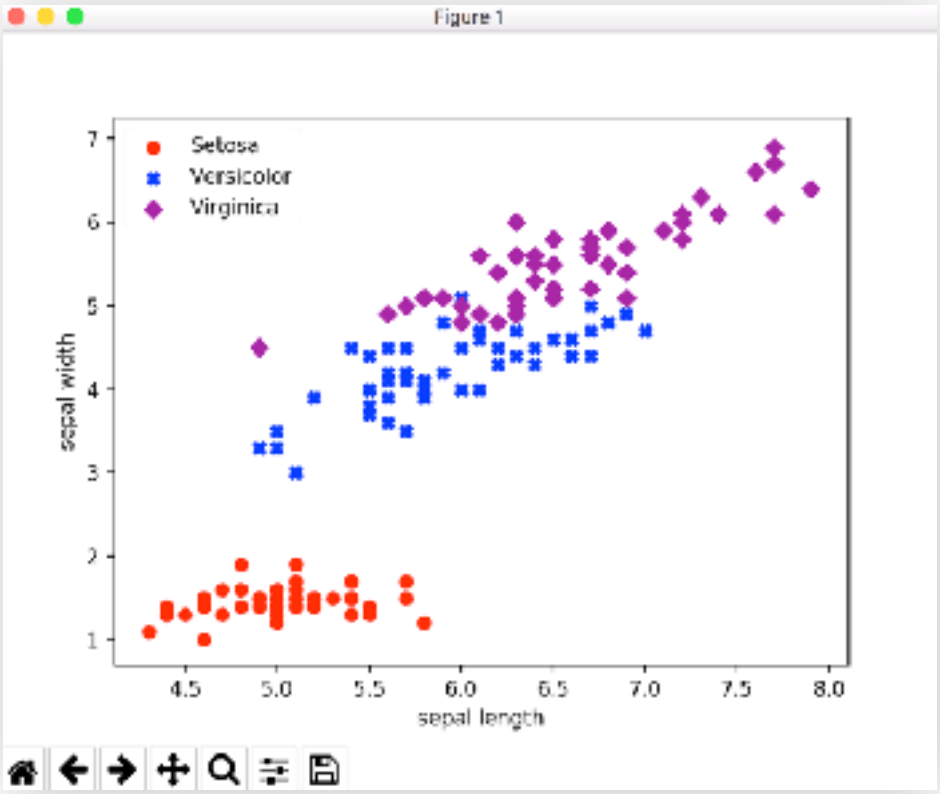
$$\sum_{j=1}^{k} \sum_{x_i \in Sj} \| x_i - \mu_j \|^2$$

where μ_j is the mean (**centroid**) of points in Sj.

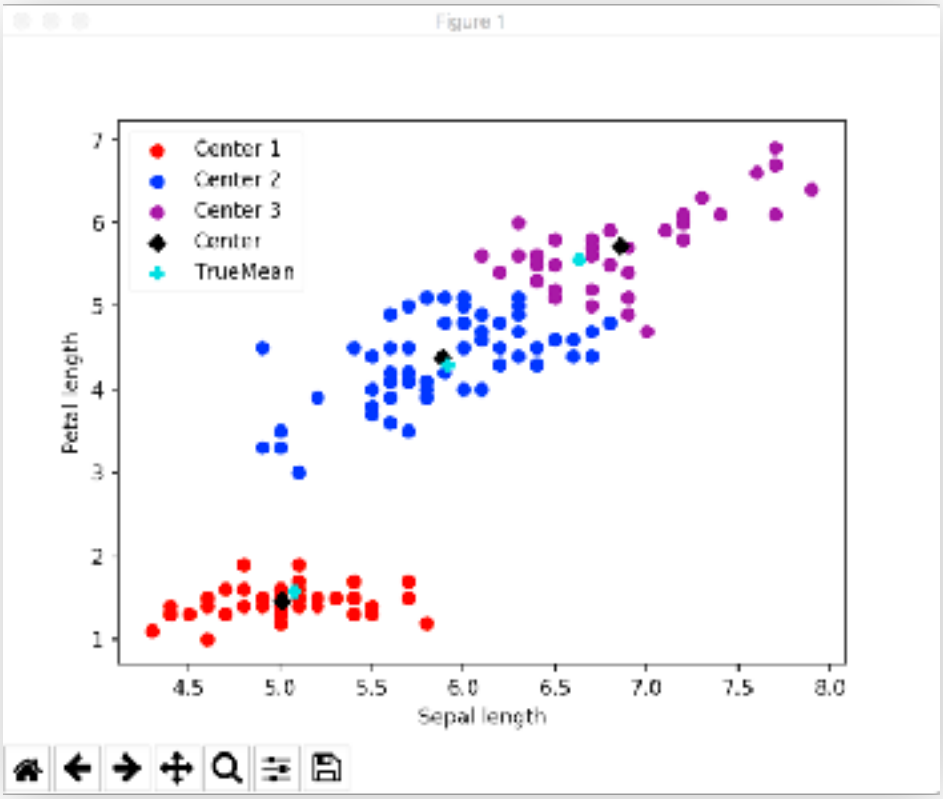
Procedure

1. Initialization: randomly chose k centroids
2. Assignment: assign each observation to the nearest centroid (cluster)
3. Update: calculate new means to be the centroids in the new clusters

Implementation



Original Data

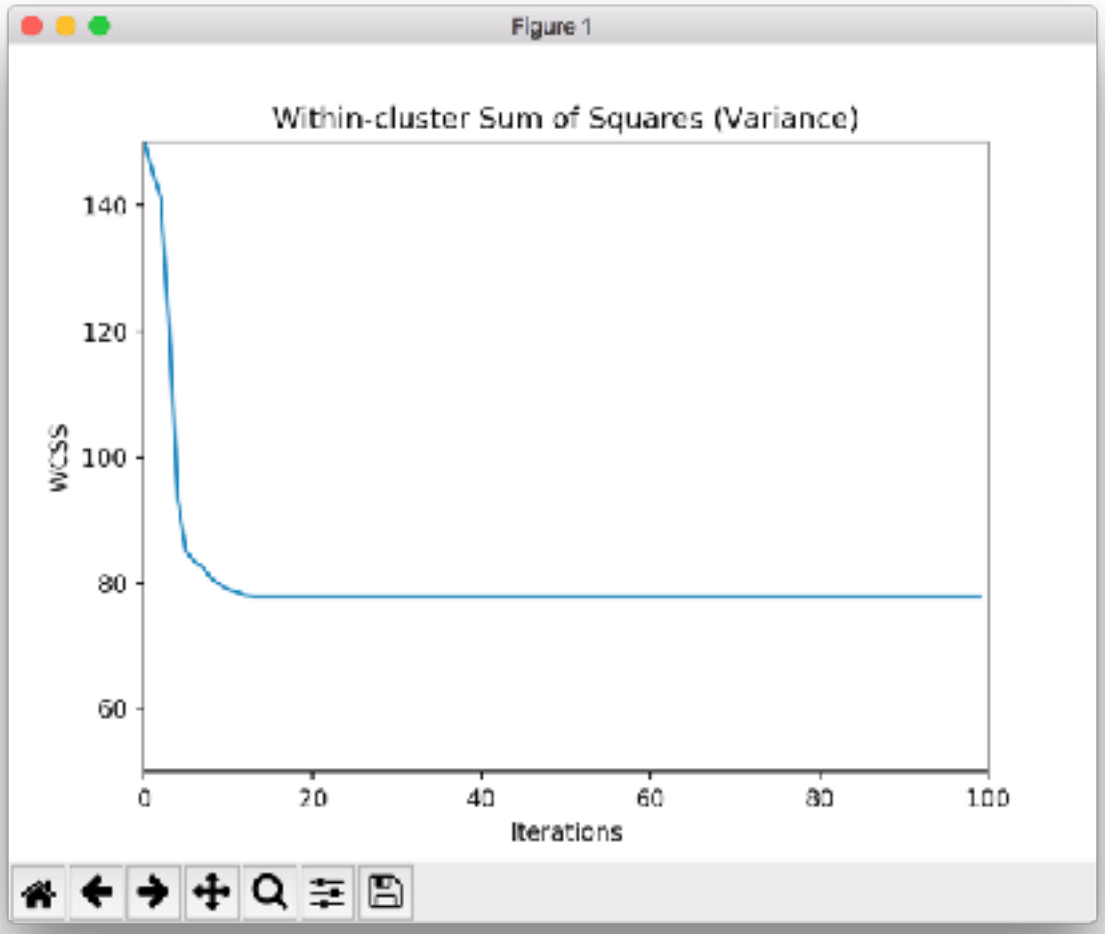


Our result

To begin with, our algorithm is implemented in Python. We use the library **pyplot** in **matplotlib** to visualize the result. What we can tell from the the two pictures is the great difference of the purple (Virginica) and blue (Versicolor) clusters. The original data shows that the observations of Virginica scatters in a wider range and overlaps with Versicolor.

Next, lets have a look at the accuracy of our algorithm. Notice the difference between black diamond ♦ and cyan plus ✚ sign. In the updating progress, we can see that the black diamond ♦ approaches closer and closer to the cyan plus ✚ , however, goes further than them.

Also, we use the within-cluster sum of squares (WCSS) (i.e. variance) to check the work.We can find that after the **fourteenth** iteration, the variance remains the same, that is, the assign and update step doesn't produce a new collection of centroids anymore because it tries to assign observations into a closer cluster but cannot find a better solution.



Variance