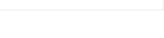


MITx 6.86x

## **Machine Learning with Python-From Linear Models to Deep Learning**

Discussion Course **Progress** Resources Dates

A Course / Unit 4. Unsupervised Learning... / Project 4: Collaborative Filtering vi



2. K-means

< Previous

☐ Bookmark this page

Project due Apr 26, 2023 08:59 -03 Completed

## K-means

1.0/1.0 point (graded)

For this part of the project you will compare clustering obtained via K-means to the (so EM. In order to do so, our K-means algorithm will differ a bit from the one you learned. It estimated exactly as before but the algorithm returns additional information. More specific resulting clusters of points to estimate a Gaussian model for each cluster. Thus, our K-returns a mixture model where the means of the component Gaussians are the  $\boldsymbol{K}$  cent K-means algorithm. This is to make it such that we can now directly plot and compare stop two algorithms as if they were both estimating mixtures.

Read a 2D toy dataset using  $[X = np.loadtxt('toy_data.txt')]$ . Your task is to run the this data using the implementation we have provided in kmeans.py Initialize K-means common.init(X, K, seed), where K is the number of clusters and seed is the rand randomly initialize the parameters.

Note that init(X,K) returns a K-component mixture model with means, variances an The K-means algorithm will only care about the means, however, and returns a mixture on the K-means solution.

Try K=[1,2,3,4] on this data, plotting each solution using our <code>common.plot</code> function initialization is random, please use seeds 0,1,2,3,4 to and select the one that minimize the associated plots (best solution for each K). The code for this task can be written in

$$ec_{X}^{\circ}$$
rt the lowest cost for each  $K$ :

## Legal

Terms of Service & Have is copies 50 attempts

<u>Privacy Policy</u>

Accessibility Policy

**Trademark Policy** 









Next >





© 2023 edX LLC. All rights reserved.

深圳市恒宇博科技有限公司 <u>粤ICP备17044299号-2</u>