CS136

CP4 Pengcheng Xu

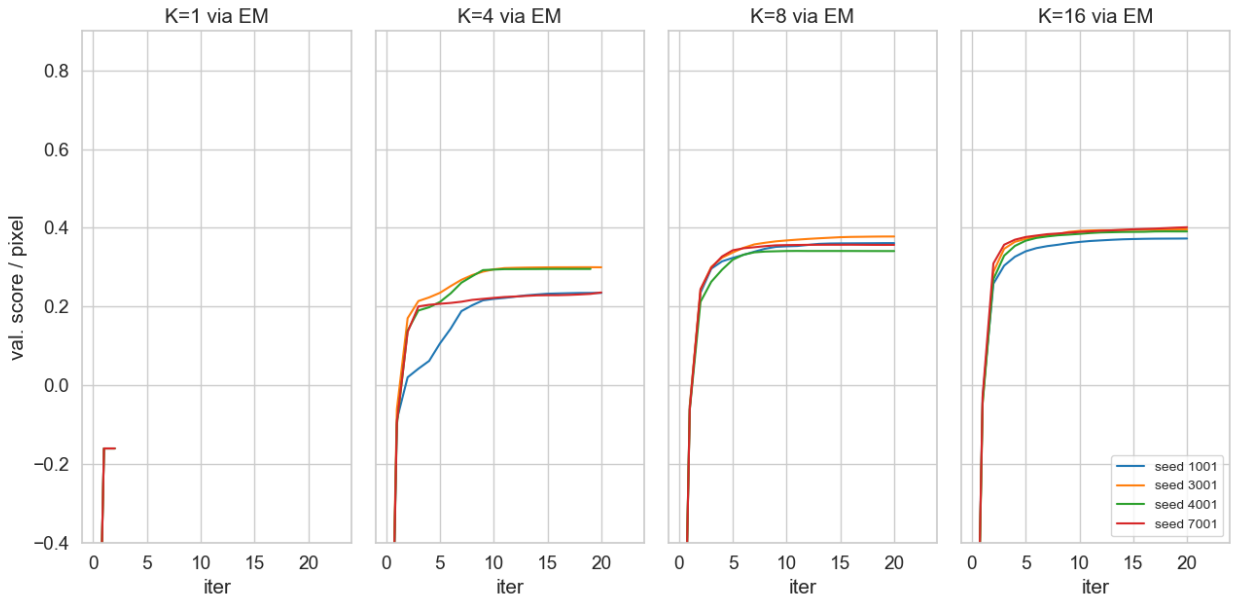
Collaboration Statement:

- Total hours spent: 6 hrs
- Consult Resources:
    - Course's website
    - Numpy website
    - Online resource
- Internet

Problem 1.

1a

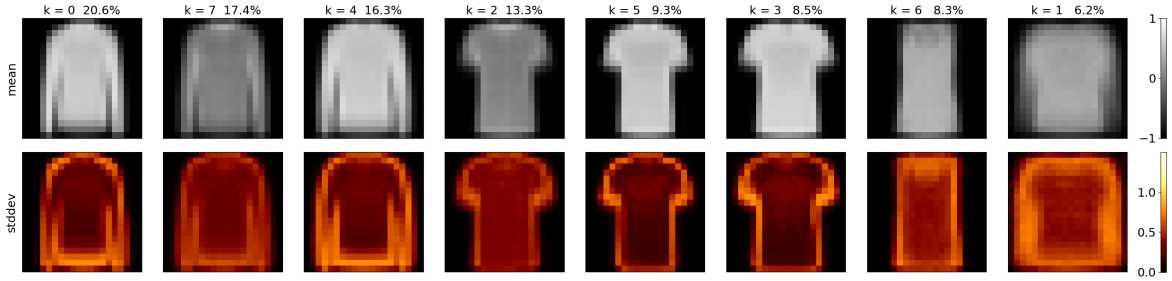
After using pretrained EM model from 1(v) in the CP4 description, and running `plot_history_for_many_runs,py` script, we could make a figure as below (validation score versus iteration, with a separate line for each random seed):



From the figure above, we could see that K = 16 seems to do best on this data, because all the random seeds converge to a higher validation score (i.e. around 0.4) compared to other K-value models, and also converging to the final stable validation score relatively quickly.

1b

After modifying and running `train_EM_gmm_with_man_runs.py` script, we could see the best run with K = 8 is when we set random seed = 3001.
The corresponding visualizaton is as follows:



From the figure above, we could get out estimation:

- **(a) long-sleeve (clusters k = 0, k= 7, and k = 4)**
  probability = 54.3% (i.e. 20.6% + 17.4% +16.3%, the summatioin of each cluster's probablity)

- **(b) short-sleeve (clusters k= 2, k = 5, k = 3, and k = 1)**

probability = 37.4% (i.e. 13.3% + 9.3% + 8.5% + 6.2%)

- **(c) non-sleeve (cluster k = 6)**
  probability = 8.3%

1c

After finishing `#TODO` parts and running the completed version of `trian_EM_gmm_with_many_runs.py` scripts, and extracting the score on both validation set and testing set, we could draw a table as below (keep 5 digits after decimal point):

K value	Validation score (log likelidhood per pixel)	Test score (log likelidhood per pixel)	See with best model
1	-0.15983	-0.17355	1001
4	0.29984	0.29848	3001
8	0.37801	0.37951	3001
16	0.40152	0.40639	7001