CS181 Homework 4

Ruitao (Toby) Du

04/04/2015

1 Composing Kernel Functions

In d dimensions, the volume of a hypersphere of unit radius is

$$V_{ball}(d) = \frac{\pi^{d/2}}{\Gamma(\frac{d}{2}+1)},$$

In d dimensions, the volume of a hypercube with edges of length two.

$$V_{cube}(d) = 2^d,$$

The fraction is

$$\begin{aligned} f(d) &= \frac{\frac{\pi^{d/2}}{\Gamma(\frac{d}{2}+1)}}{2^d} \\ &= \frac{\frac{\pi}{4}^{d/2}}{\Gamma(\frac{d}{2}+1)}, \end{aligned}$$

We know that $\frac{\pi}{4} < 1$ and Gamma function is monotonically increasing when $\frac{d}{2}+1 > 2$. We find that when d goes larger, the fraction is closer to zero. This means less volume is contained within the hypersphere.

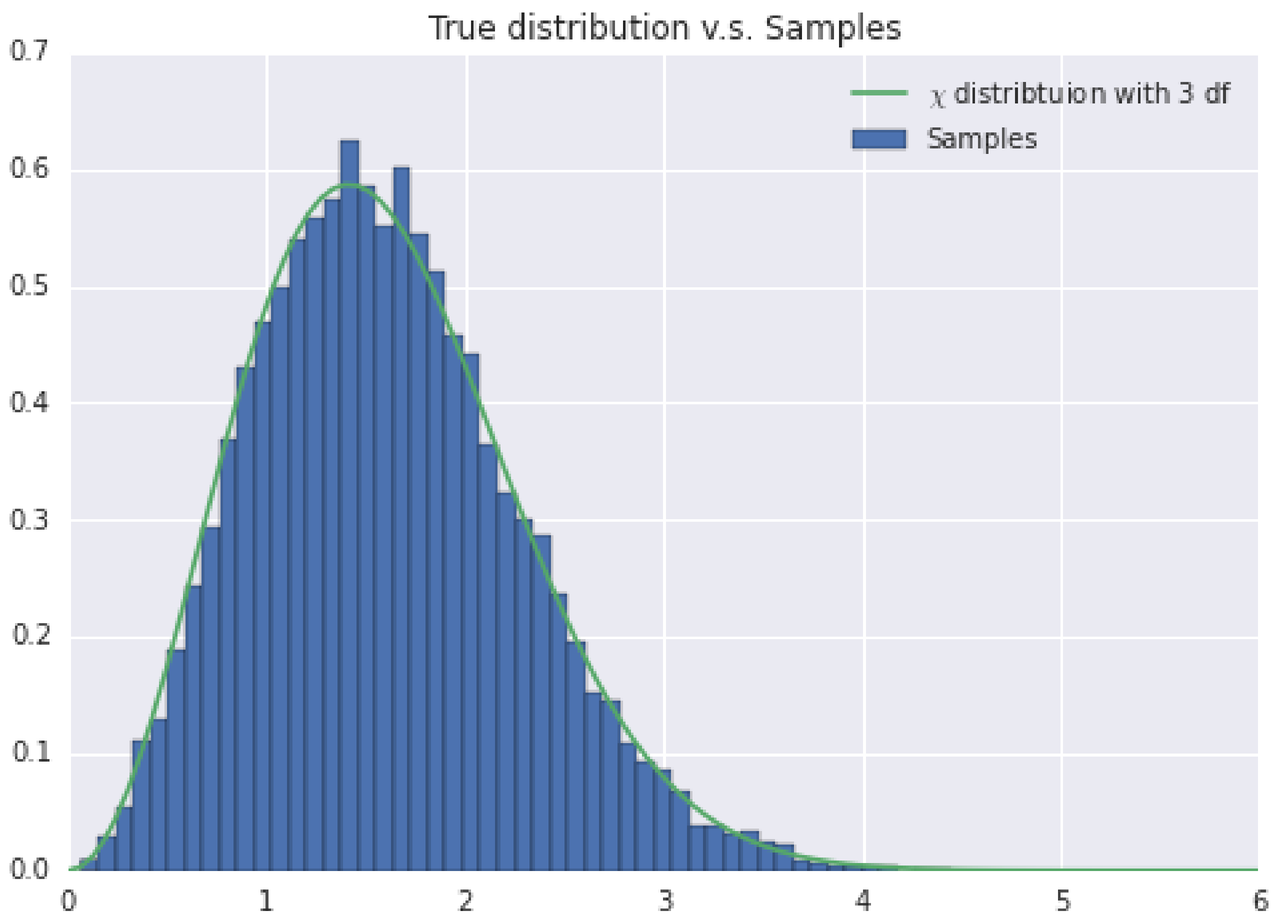
2 Slack Variables and Importances

Assume we are drawing a D dimension Gaussian distributed data $\{X_d\}_{d=1}^{D}$. The distance from the origin is

$$Dist(\{X_d\}_{d=1}^{D}) = \sqrt{\sum_{d=1}^{D} X_d^2}$$

Because our Gaussian distribution has an identity covariance matrix, data in each dimension is independent. Also, it has zero mean. We know that this satisfies the definition of χ distribution with D degree of freedom. Now I draw 10000 data from 3-dimension Gaussian distribution. And then I calculate the distance from origin and plot χ distribution with 3 degree of freedom.

$$Dist(\{X_d\}_{d=1}^{D}) \sim \chi_D$$



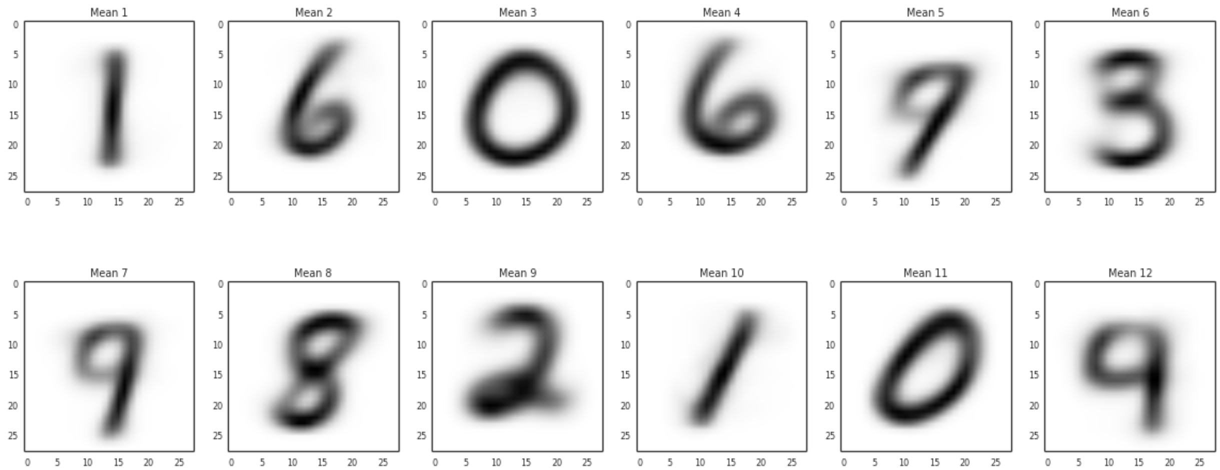
As we can see from the plot, the samples are similar to the χ distribution with 3 degree of freedom.

3 Implement K-Means

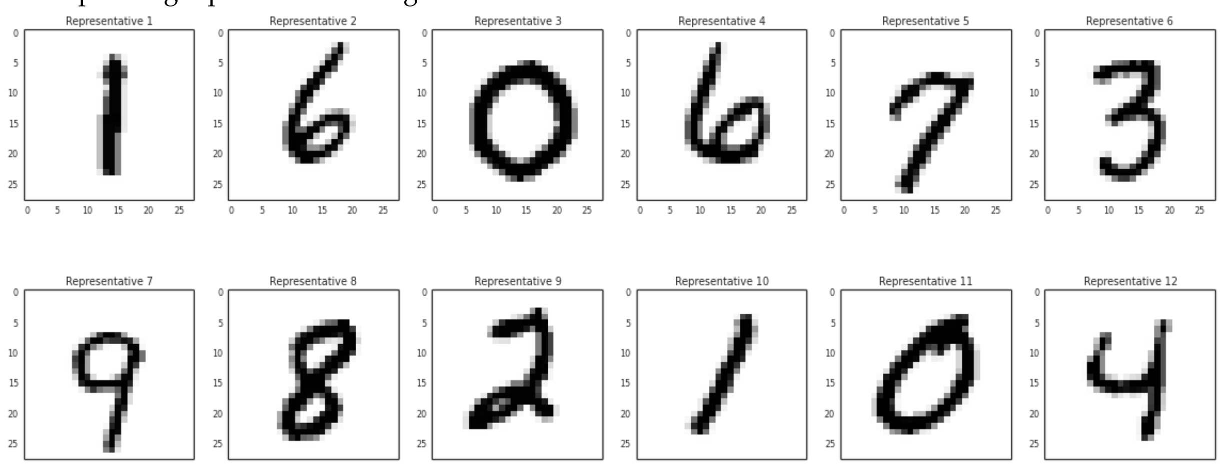
I choose the MNIST Handwritten Digits database. I use np.ravel() to turn a matrix of pixels to a vector. And then I run my K-means algorithm on $k = 6, 8, 10, 12$

$k = 12$

Mean images:

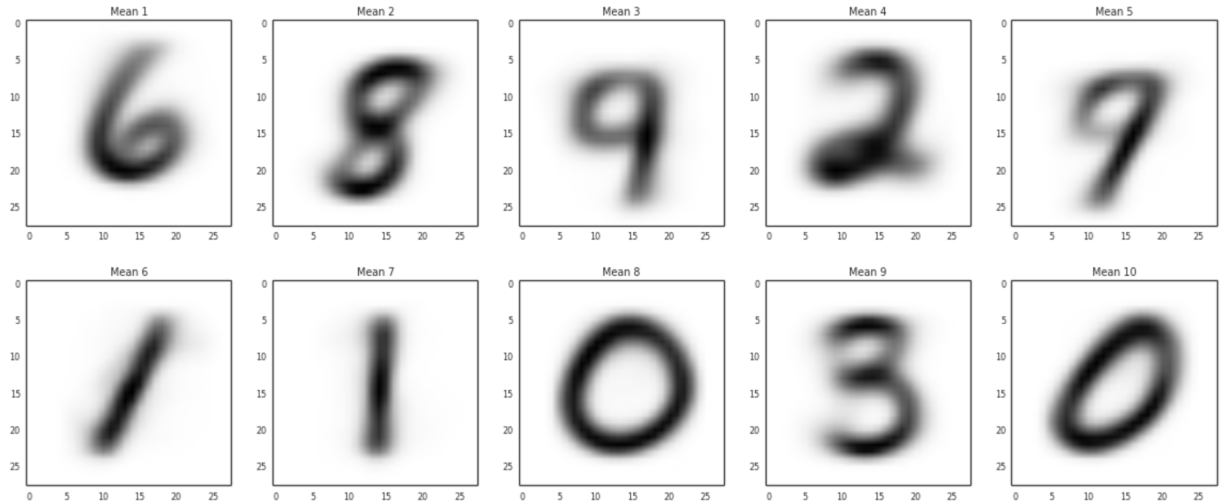


Corresponding representative images:

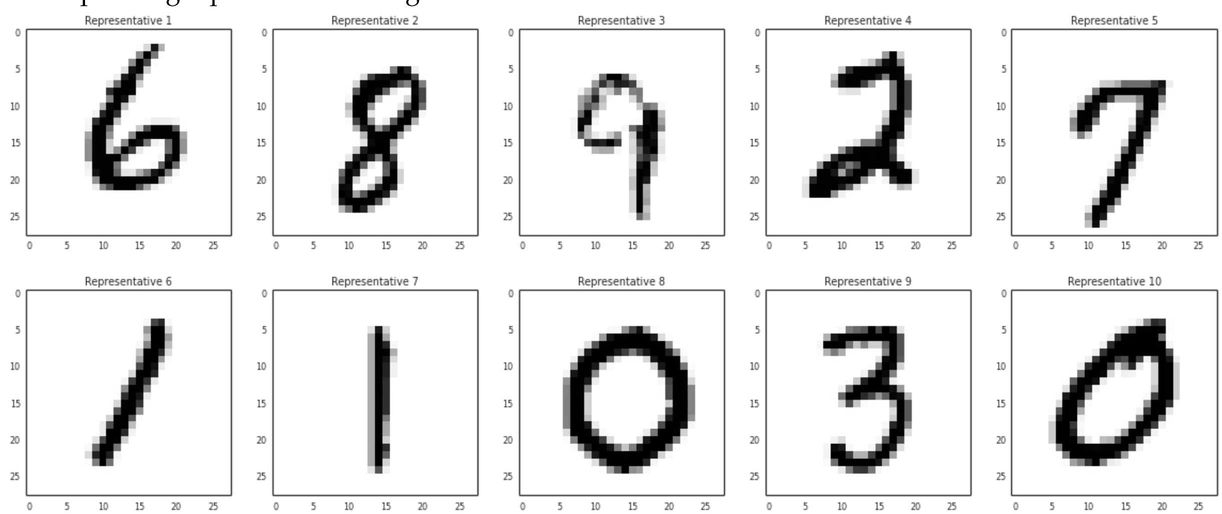


$k = 10$

Mean images:

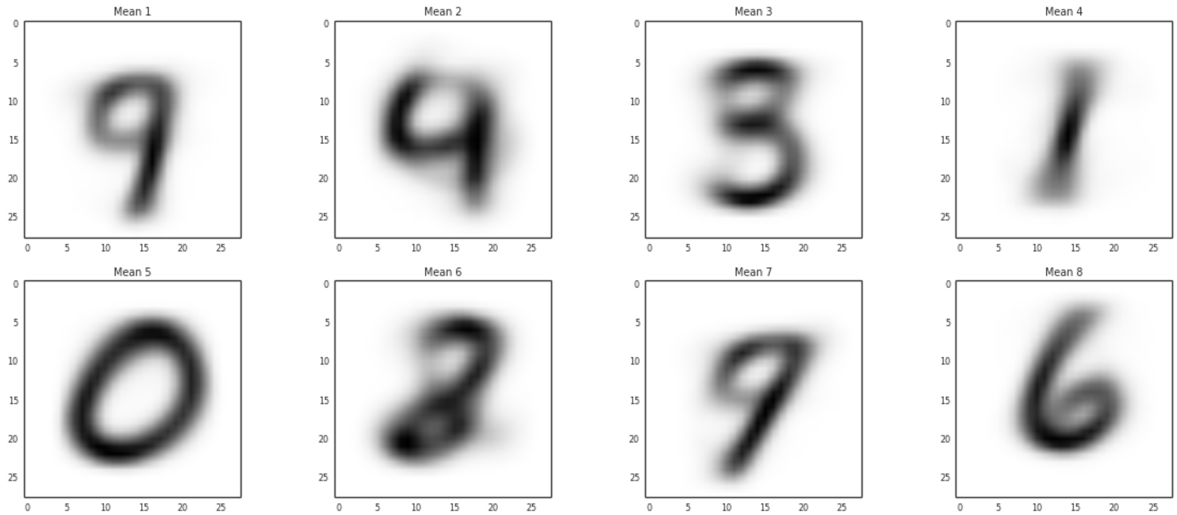


Corresponding representative images:

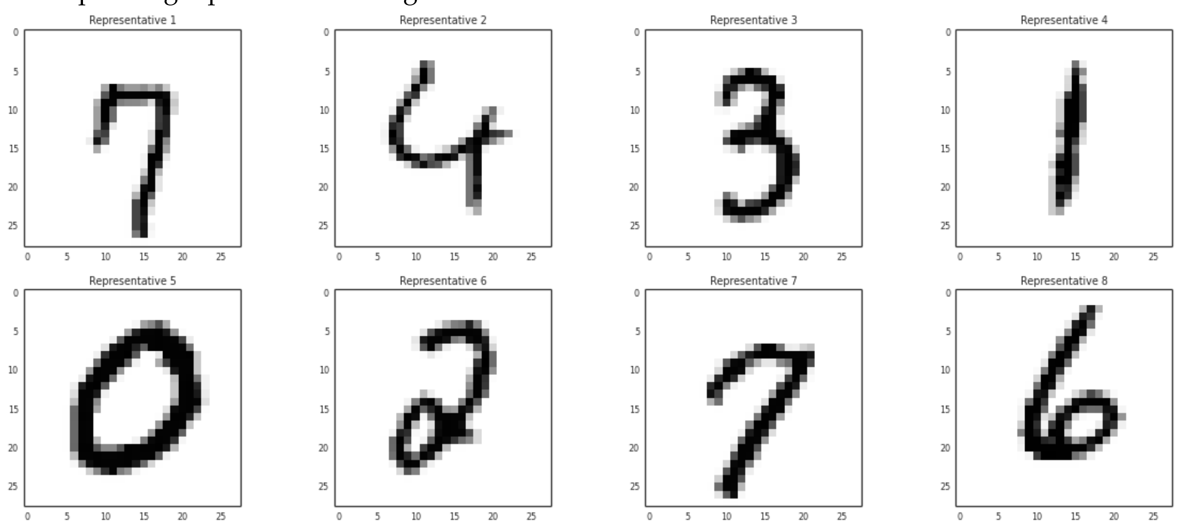


$k = 8$

Mean images:

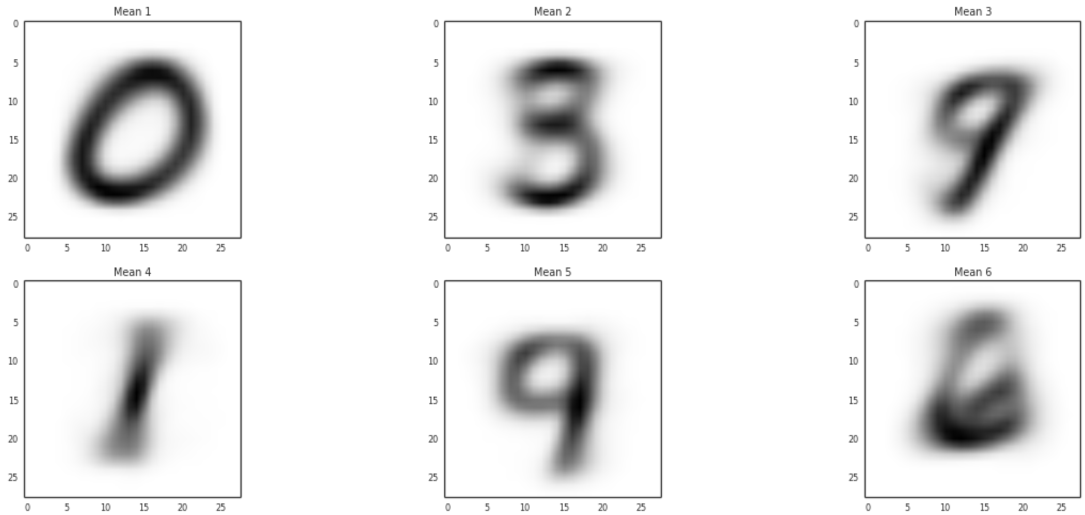


Corresponding representative images:

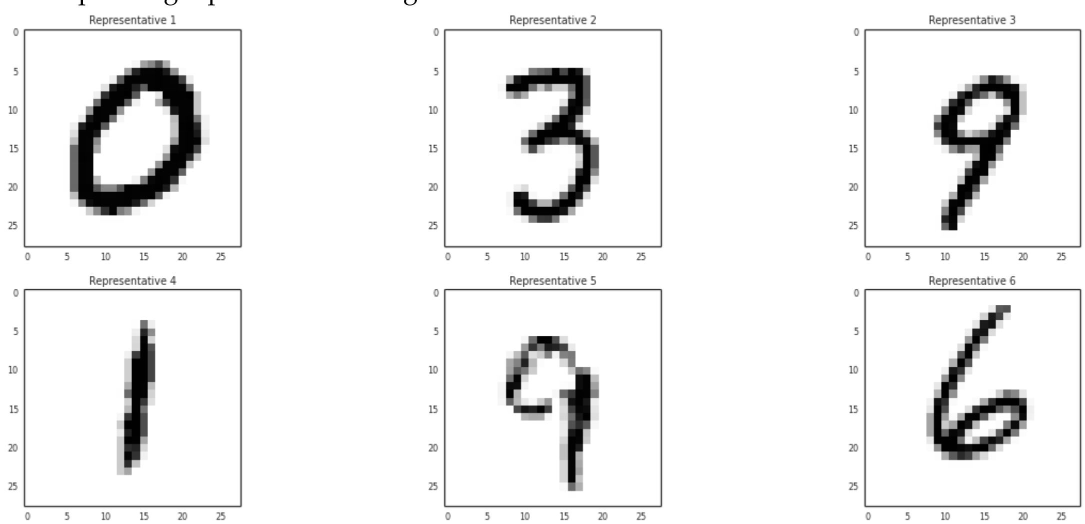


$k = 6$

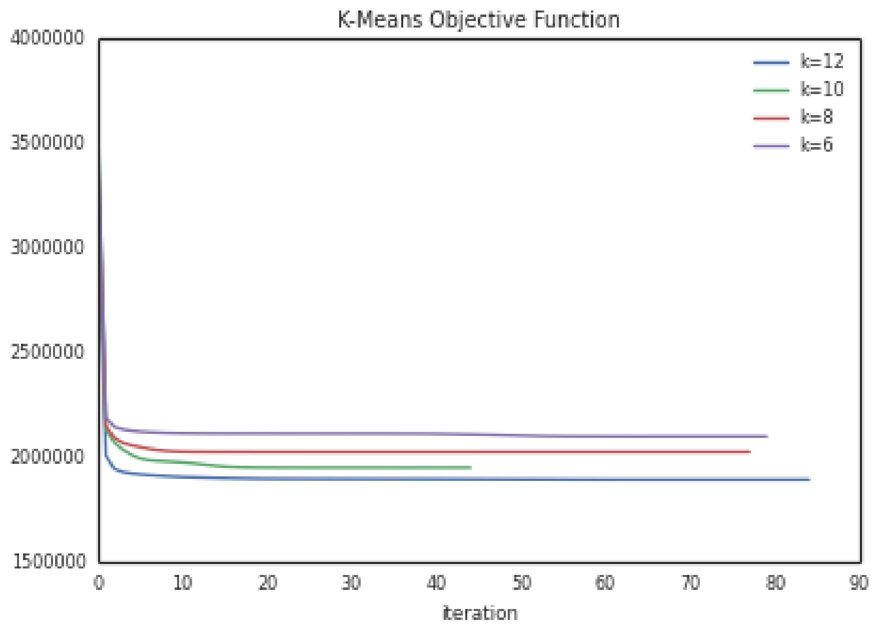
Mean images:



Corresponding representative images:



Objective function of each k is

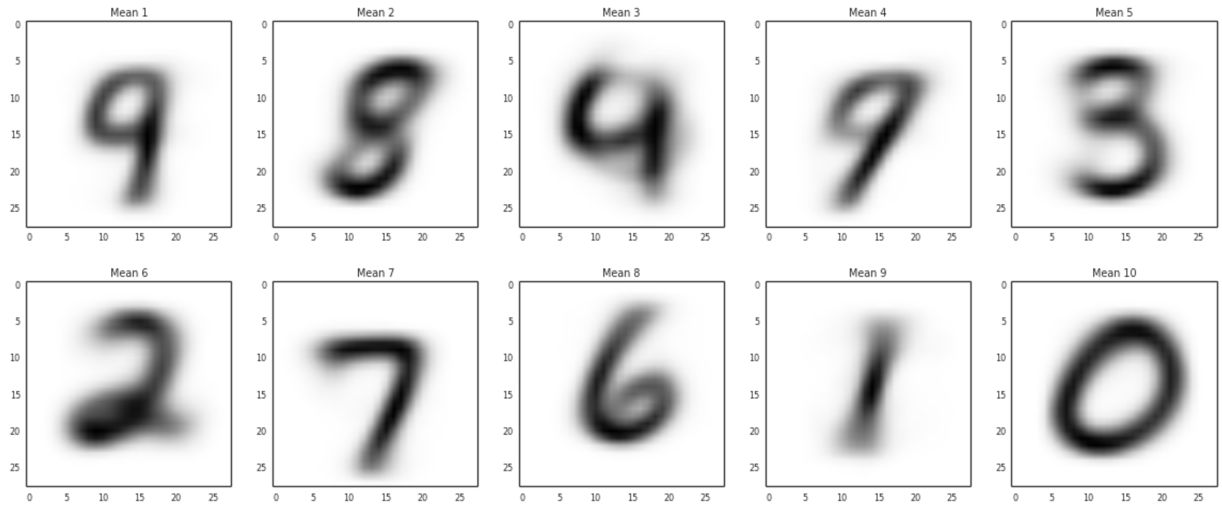


As we can see, 7 and 9 are difficult to distinguish. Ideally, we have 10 digits and we can cluster 10 different clusters in 10 digits. However, some number is difficult to distinguish, so we may have two or three similar mean images.

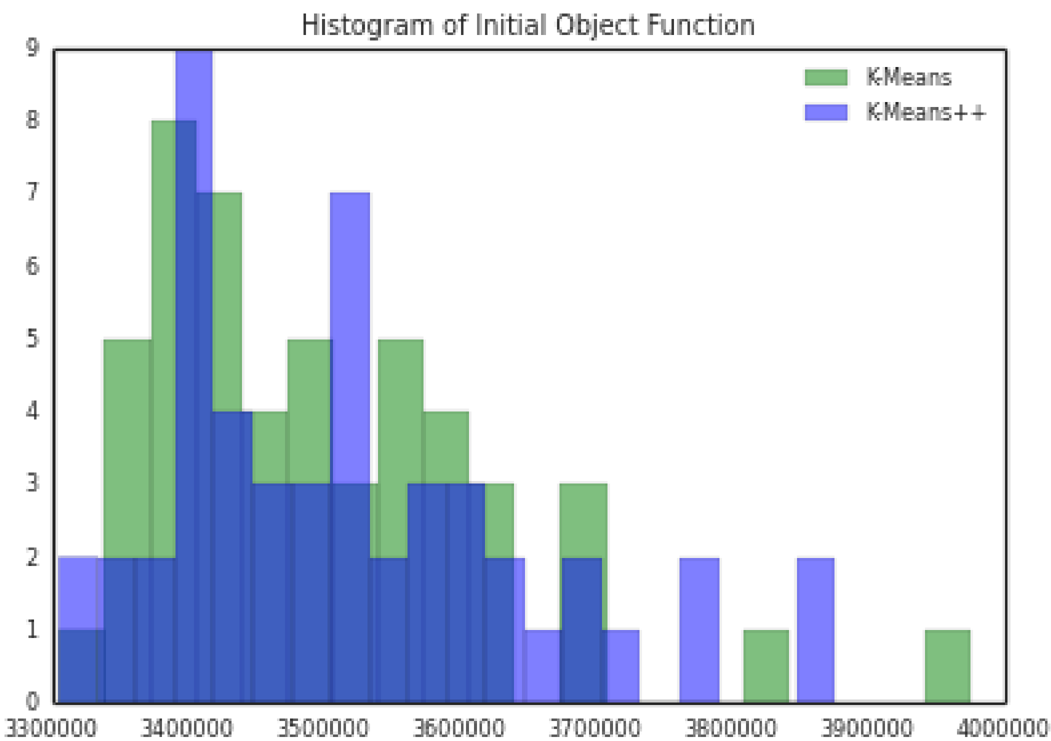
From the plot of objective function, it is monotonically decreasing after every iteration. We also learn that if we choose larger k, usually it takes more interations to converge.

4 Implement K-Means++

I implemented K-Means++ algorithm and the results are a little improved because I can train 9 distint mean images.



I try 50 times of K-means and K-means++ algorithms to initialize and calculate the objective funcion.



As we can see that K-means++ are more likely to yield a lower objective value. It means that when we use K-means++ algorithm, we may have more reasonable initialization and may converge more quickly.

5 Calibration

8 hours