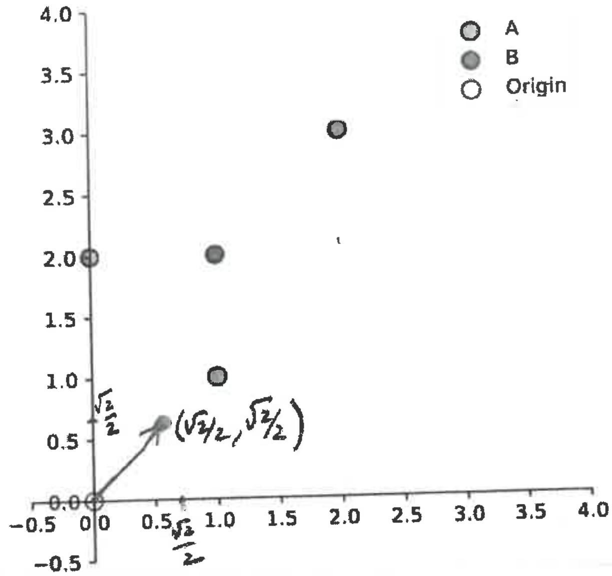
Brian Bertness
1478201

1. a)

A: $\begin{pmatrix} 0 \\ 2 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \end{pmatrix}$

B: $\begin{pmatrix} 1 \\ 2 \end{pmatrix}, \begin{pmatrix} 2 \\ 3 \end{pmatrix}$



1) Subtract off the mean.

$$m = \frac{1}{4}\left(\begin{bmatrix} 0 \\ 2 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \end{bmatrix} + \begin{bmatrix} 1 \\ 2 \end{bmatrix} + \begin{bmatrix} 2 \\ 3 \end{bmatrix} \right)$$

$$= \frac{1}{4}\begin{bmatrix} 4 \\ 8 \end{bmatrix}$$

$$= \begin{bmatrix} 1 \\ 2 \end{bmatrix}$$

So $X = \begin{bmatrix} 0-1 & 1-1 & 1-1 & 2-1 \\ 2-2 & 1-2 & 2-2 & 3-2 \end{bmatrix}$

$$= \begin{bmatrix} -1 & 0 & 0 & 1 \\ 0 & -1 & 0 & 1 \end{bmatrix}$$

2) Form the Co-Variance Matrix

$$\Sigma = Cov(X) = \frac{1}{n-1} X X^T$$

$$= \frac{1}{3}\begin{bmatrix} -1 & 0 & 0 & 1 \\ 0 & -1 & 0 & 1 \end{bmatrix}\begin{bmatrix} -1 & 0 \\ 0 & -1 \\ 0 & 0 \\ 1 & 1 \end{bmatrix}$$

$$= \frac{1}{3}\begin{bmatrix} 1+0+0+1 & 0+0+0+1 \\ 0+0+0+1 & 0+1+0+1 \end{bmatrix}$$

$$= \frac{1}{3}\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} = \begin{bmatrix} 2/3 & 1/3 \\ 1/3 & 2/3 \end{bmatrix}$$

3) Find Eigenvalues/ Eigenvectors

$$\det(\Sigma - \lambda I) = \vec{0}$$

$$\Rightarrow \det\begin{bmatrix} 2/3 - \lambda & 1/3 \\ 1/3 & 2/3 - \lambda \end{bmatrix} = \vec{0}$$

$$\Rightarrow (2/3 - \lambda)^2 - (1/3)(1/3) = \vec{0}$$

$$\Rightarrow 4/9 - 2(2/3)(\lambda) + \lambda^2 - 1/9 = \vec{0}$$

$$\Rightarrow 4/9 - 4/3\,\lambda + \lambda^2 - 1/9 = \vec{0}$$

$$\Rightarrow 1/3 - 4/3\,\lambda + \lambda^2 = \vec{0}$$

$$\Rightarrow 1 - 4\lambda + 3\lambda^2 = \vec{0}$$

$$\Rightarrow (1 - 3\lambda)(1 - \lambda) = 0$$

so $(1-3\lambda) = 0$ or $(1-\lambda) = 0$

$\Rightarrow -3\lambda = -1$ or $\lambda = 1$

$\Rightarrow \lambda = 1/3$ so let $\lambda = 1$

Now, $(\Sigma - \lambda I)\vec{x} = \vec{0}$. Since $\lambda = 1$

$$\Rightarrow \begin{bmatrix} 2/3 - 1 & 1/3 \\ 1/3 & 2/3 - 1 \end{bmatrix} \vec{x} = \vec{0}$$

$$\Rightarrow \begin{bmatrix} -1/3 & 1/3 \\ 1/3 & -1/3 \end{bmatrix} \vec{x} = \vec{0}$$

So, Row Reduce:

$$\left[\begin{array}{cc|c} -1/3 & 1/3 & 0 \\ 1/3 & -1/3 & 0 \end{array}\right] \sim \left[\begin{array}{cc|c} 1 & -1 & 0 \\ 0 & 0 & 0 \end{array}\right]$$

Therefore, $x_1 = x_2$ and x_2 is free. Thus,

$$\vec{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} x_2 \\ x_2 \end{bmatrix} = x_2 \begin{bmatrix} 1 \\ 1 \end{bmatrix} = x_2 \vec{v}$$ where $x_2 \neq 0$, $\vec{v} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

So a general solution is

$\begin{bmatrix} 1 \\ 1 \end{bmatrix}$ but we want a unit vector so:

$$\vec{w} = \frac{1}{\|\vec{v}\|}\vec{v} = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} \sqrt{2}/2 \\ \sqrt{2}/2 \end{bmatrix}$$

1.b) Consider $z^t = W^T x^t \Rightarrow W\vec{z}^t = WW^T\vec{x}^t$
$\Rightarrow \vec{v}^t = WW^T\vec{x}^t$

where $\vec{v}^t = W\vec{z}^t$.

We note that $\vec{v}^t \neq \vec{x}^t$ for all t.

This is due to the fact that WW^T is not the identity matrix since we did not put all of the eigenvectors into W^T. I believe if we were to put all of the eigenvectors into W^T then $\vec{v}^t$ would equal $\vec{x}^t$ since then WW^T would be an identity matrix. Of course if we did that we would not be decreasing dimensionality which is the whole point of PCA.

1 c) $m_1 = \begin{bmatrix} (0+1)/2 \\ (2+1)/2 \end{bmatrix} = \begin{bmatrix} 1/2 \\ 3/2 \end{bmatrix}$

$m_2 = \begin{bmatrix} (1+2)/2 \\ (2+3)/2 \end{bmatrix} = \begin{bmatrix} 3/2 \\ 5/2 \end{bmatrix}$

For S_1

$\begin{bmatrix} 0 \\ 2 \end{bmatrix} - m_1 = \begin{bmatrix} 0 \\ 2 \end{bmatrix} - \begin{bmatrix} 1/2 \\ 3/2 \end{bmatrix} = \begin{bmatrix} -1/2 \\ 1/2 \end{bmatrix}$

$\begin{bmatrix} 1 \\ 1 \end{bmatrix} - m_1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix} - \begin{bmatrix} 1/2 \\ 3/2 \end{bmatrix} = \begin{bmatrix} 1/2 \\ -1/2 \end{bmatrix}$

$S_1 = \begin{bmatrix} -1/2 \\ 1/2 \end{bmatrix} \begin{bmatrix} -1/2 & 1/2 \end{bmatrix} + \begin{bmatrix} 1/2 \\ -1/2 \end{bmatrix} \begin{bmatrix} 1/2 & -1/2 \end{bmatrix}$

$= \begin{bmatrix} 1/4 & -1/4 \\ -1/4 & 1/4 \end{bmatrix} + \begin{bmatrix} 1/4 & -1/4 \\ -1/4 & 1/4 \end{bmatrix}$

$= \begin{bmatrix} 1/2 & -1/2 \\ -1/2 & 1/2 \end{bmatrix}$

For S_2

$$\begin{bmatrix} 1 \\ 2 \end{bmatrix} - m_2 = \begin{bmatrix} 1 \\ 2 \end{bmatrix} - \begin{bmatrix} 3/2 \\ 5/2 \end{bmatrix} = \begin{bmatrix} -1/2 \\ -1/2 \end{bmatrix}$$

$$\begin{bmatrix} 2 \\ 3 \end{bmatrix} - m_2 = \begin{bmatrix} 2 \\ 3 \end{bmatrix} - \begin{bmatrix} 3/2 \\ 5/2 \end{bmatrix} = \begin{bmatrix} 1/2 \\ 1/2 \end{bmatrix}$$

$$S_2 = \begin{bmatrix} -1/2 \\ -1/2 \end{bmatrix} \begin{bmatrix} -1/2 & -1/2 \end{bmatrix} + \begin{bmatrix} 1/2 \\ 1/2 \end{bmatrix} \begin{bmatrix} 1/2 & 1/2 \end{bmatrix}$$

$$= \begin{bmatrix} 1/4 & 1/4 \\ 1/4 & 1/4 \end{bmatrix} + \begin{bmatrix} 1/4 & 1/4 \\ 1/4 & 1/4 \end{bmatrix}$$

$$= \begin{bmatrix} 1/2 & 1/2 \\ 1/2 & 1/2 \end{bmatrix}$$

S_w

$$S_w = S_1 + S_2 = \begin{bmatrix} 1/2 & -1/2 \\ -1/2 & 1/2 \end{bmatrix} + \begin{bmatrix} 1/2 & 1/2 \\ 1/2 & 1/2 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

S_B

$$S_B = (\vec{m}_1 - \vec{m}_2)(\vec{m}_1 - \vec{m}_2)^T \qquad \vec{m}_1 - \vec{m}_2 = \begin{bmatrix} 1/2 \\ 3/2 \end{bmatrix} - \begin{bmatrix} 3/2 \\ 5/2 \end{bmatrix} = \begin{bmatrix} -1 \\ -1 \end{bmatrix}$$

$$= \begin{bmatrix} -1 \\ -1 \end{bmatrix} \begin{bmatrix} -1 & -1 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$$

d) $\vec{w}_1 = c\, S_w^{-1}(\vec{m}_1 - \vec{m}_2)$ now, $S_w^{-1} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$

$= c \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} -1 \\ -1 \end{bmatrix}$ Since inverse of identity matrix is identity matrix and

$= c \begin{bmatrix} -1 \\ -1 \end{bmatrix}$ $\vec{m}_1 - \vec{m}_2 = \begin{bmatrix} -1 \\ -1 \end{bmatrix}$

$= c_2 \begin{bmatrix} 1 \\ 1 \end{bmatrix}$ from S_B calculation in part c).

where $c_2 = -1 \cdot c$

Choose $c_2 = 1$ and we have, $\vec{w}_1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

We normalize $\vec{w}_1$ by

$$\vec{w}_{unit} = \frac{1}{\|\vec{w}_1\|}\vec{w}_1 = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} \sqrt{2}/2 \\ \sqrt{2}/2 \end{bmatrix}$$

2. a) $X_1 = 4,\ X_2 = 6,\ X_3 = 7,\ X_4 = 8,\ X_5 = 10,\ X_6 = 12$
$X_7 = 14,\ X_8 = 16.\quad k = 2$

Iter 1 $\quad c_1 = 3,\quad c_2 = 12$

Assignment

$X_1: |4-3| = 1,\quad |4-12| = 8$
$X_2: |6-3| = 3,\quad |6-12| = 6$
$X_3: |7-3| = 4,\quad |7-12| = 5$
$X_4: |8-3| = 5,\quad |8-12| = 4$
$X_5: |10-3| = 7\quad |10-12| = 2$
$X_6: |12-3| = 9\quad |12-12| = 0$
$X_7: |14-3| = 11\quad |14-12| = 2$
$X_8: |16-3| = 13\quad |16-12| = 4$

$C_1 = \{X_1, X_2, X_3\} = \{4, 6, 7\}\quad C_2 = \{X_4, X_5, X_6, X_7, X_8\} = \{8, 10, 12, 14, 16\}$

Update Means

$c_1 = \frac{4+6+7}{3} = \frac{17}{3} \approx 5.67\qquad c_2 = \frac{8+10+12+14+16}{5} = \frac{60}{5} \approx 12$

Iter 2

Assignment

$x_1: |4-5.67| = 1.67\quad |4-12| = 8$
$x_2: |6-5.67| = 0.33\quad |6-12| = 6$
$x_3: |7-5.67| = 1.33\quad |7-12| = 5$
$x_4: |8-5.67| = 2.33\quad |8-12| = 4$
$X_5: |10-5.67| = 4.33\quad |10-12| = 2$
$X_6: |12-5.67| = 6.33\quad |12-12| = 0$
$X_7: |14-5.67| = 8.33\quad |14-12| = 2$
$X_8: |16-5.67| = 10.33\quad |16-12| = 4$

Update Means

$C_1 = \{x_1, x_2, x_3, x_4\} = \{4, 6, 7, 8\}\quad C_2 = \{x_5, x_6, x_7, x_8\} = \{10, 12, 14, 16\}$

$c_1 = \frac{4+6+7+8}{4} = \frac{25}{4} = 6.25\qquad c_2 = \frac{10+12+14+16}{4} = \frac{52}{4} = 13$

Iter 3 Assignment

x_1: $|4-6.25| = 2.25$ $|4-13| = 9$ x_5: $|10-6.25| = 3.75$ $|10-13| = 3$

x_2: $|6-6.25| = 0.25$ $|6-13| = 7$ x_6: $|12-6.25| = 5.75$ $|12-13|$ 1

x_3: $|7-6.25| = 1.25$ $|7-13| = 6$ x_7: $|14-6.25| = 7.75$ $|14-13|$ 1

x_4 $|8-6.25| = 2.25$ $|8-13| = 5$ x_8: $|16-6.25| = 9.75$ $|16-13|$ 3

Update Means

$C_1 = \{x_1, x_2, x_3, x_4\} = \{4, 6, 7, 8\}$ $C_2 = \{x_5, x_6, x_7, x_8\} = \{10, 12, 14, 16\}$

$C_1 = \frac{4+6+7+8}{4} = \frac{25}{4} = 6.25$ $C_2 = \frac{10+12+14+16}{4} = \frac{52}{4} = 13$

2\. b) It took 3 iterations for the k-means algorithm to converge.

$$J = \sum_{n=1}^{N} \sum_{k=1}^{k} r_{nk} \| X_n - M_k \|^2$$ From part a) we have

$n = 8$

$k = 2$

$M_1 = 6.25$

$M_2 = 13$

$C_1 = \{4, 6, 7, 8\}$

$C_2 = \{10, 12, 14, 16\}$

$$= \sum_{n=1}^{8} \sum_{k=1}^{2} r_{nk} \| X_n - M_k \|^2$$

(Ignoring terms where $r_{nk} = 0$ we have)

$$= (4-6.25)^2 + (6-6.25)^2 + (7-6.25)^2 + (8-6.25)^2 + (10-13)^2 + (12-13)^2 + (14-13)^2 + (16-13)^2$$

$$= (-2.25)^2 + (0.25)^2 + (1.25)^2 + (2.25)^2 + (-3)^2 + (-1)^2 + (1)^2 + (3)^2$$

$= 28.75$

2.c) $C_1 = 8$, $C_2 = 14$

Iter 1 Assignment

$x_1: |4-8| = 4$, $|4-14| = 10$ $x_5: |10-8| = 2$, $|10-14| = 4$

$x_2: |6-8| = 2$, $|6-14| = 8$ $x_6: |12-8| = 4$, $|12-14| = 2$

$x_3: |7-8| = 1$, $|7-14| = 7$ $x_7: |14-8| = 6$, $|14-14| = 0$

$x_4: |8-8| = 0$, $|8-14| = 6$ $x_8: |16-8| = 8$, $|16-14| = 2$

$C_1 = \{x_1, x_2, x_3, x_4, x_5\} = \{4, 6, 7, 8, 10\}$ $C_2 = \{x_6, x_7, x_8\} = \{12, 14, 16\}$

Update Means

$C_1 = \frac{4+6+7+8+10}{5} = 7$ $C_2 = \frac{12+14+16}{3} = 14$

Iter 2. Assignment

$x_1: |4-7| = 3$, $x_5: |4-14| = 10$ $x_5: |10-7| = 3$ $|10-14| = 4$

$x_2: |6-7| = 1$, $x_6: |6-14| = 8$ $x_6: |12-7| = 5$ $|12-14| = 2$

$x_3: |7-7| = 0$, $x_7: |7-14| = 7$ $x_7: |14-7| = 7$ $|14-14| = 0$

$x_4: |8-7| = 1$, $x_8: |8-14| = 6$ $x_6: |16-7| = 9$ $|16-14| = 2$

$C_1 = \{x_1, x_2, x_3, x_4, x_5\} = \{4, 6, 7, 8, 10\}$ $C_2 = \{x_6, x_7, x_8\} = \{12, 14, 16\}$

$C_1 = \frac{4+6+7+8+10}{5} = 7$ $C_2 = \frac{12+14+16}{3} = 14$

d) It took 2 iterations for the k-means algorithm to converge.

2. d) cont'd

$$J = \sum_{n=1}^{N} \sum_{k=1}^{2} r_{nk} \| x_n - \mu_k \|^2$$

From part c) we have $n=8$

$k=2$

$\mu_1 = 7$

$\mu_2 = 14$

$C_1 = \{x_1, x_2, x_3, x_4, x_5\}$

$C_2 = \{x_6, x_7, x_8\}$

$$= \sum_{n=1}^{8} \sum_{k=1}^{2} r_{nk} \| x_n - \mu_k \|^2$$

(ignoring terms where $r_{nk} = 0$ we have)

$$= (4-7)^2 + (6-7)^2 + (7-7)^2 + (8-7)^2 + (10-7)^2 + (12-14)^2 + (14-14)^2 + (16-14)^2$$

$$= (-3)^2 + (-1)^2 + 0^2 + 1^2 + 3^2 + (-2)^2 + 0^2 + (-2)^2$$

$$= 28.0$$

2. e) If we compare a) with c) we find that solution c) is better since c) had a smaller reconstruction error and converged in less iterations.

3.a) Yes, the plot shapes are following what I would expect as the reconstruction error decreases rapidly at first and then levels off (see the graphs at the end).

We see that when k=8 the end accuracy based on reconconstruction error was better, classification accuracy 0.94 compared to 0.83. However, k=8 took longer to converge compared to k=4, 27 iterations compared to 20 iterations. The simple reason is the number of centers. With more centers the algorithm can be more accurate classifying but the trade-off is that there are more numbers to crunch so the algorithm is slower.

3.b) In this case 127 dimensions were needed. PCA did not help the accuracy as the classification accuracy was the same as in part 3.a) without PCA. However, the algorithm convursed faster in both the k=8 and k=4 cases with PCA restricting to 127 dimensions.

3.c) No, the results are worse, both in terms of accuracy and number of iterations. This is due to the fact that as the number of dimensions decrease so does the variance of the sample. This implies less accuracy and more work by kNN for convergence.

k=8

Console Output:

Using raw data converged in 27 iteration (5.17 seconds)

Classification accuracy: 0.94

#################

Project data into 127 dimensions with PCA converged in 26 iteration (4.74 seconds)

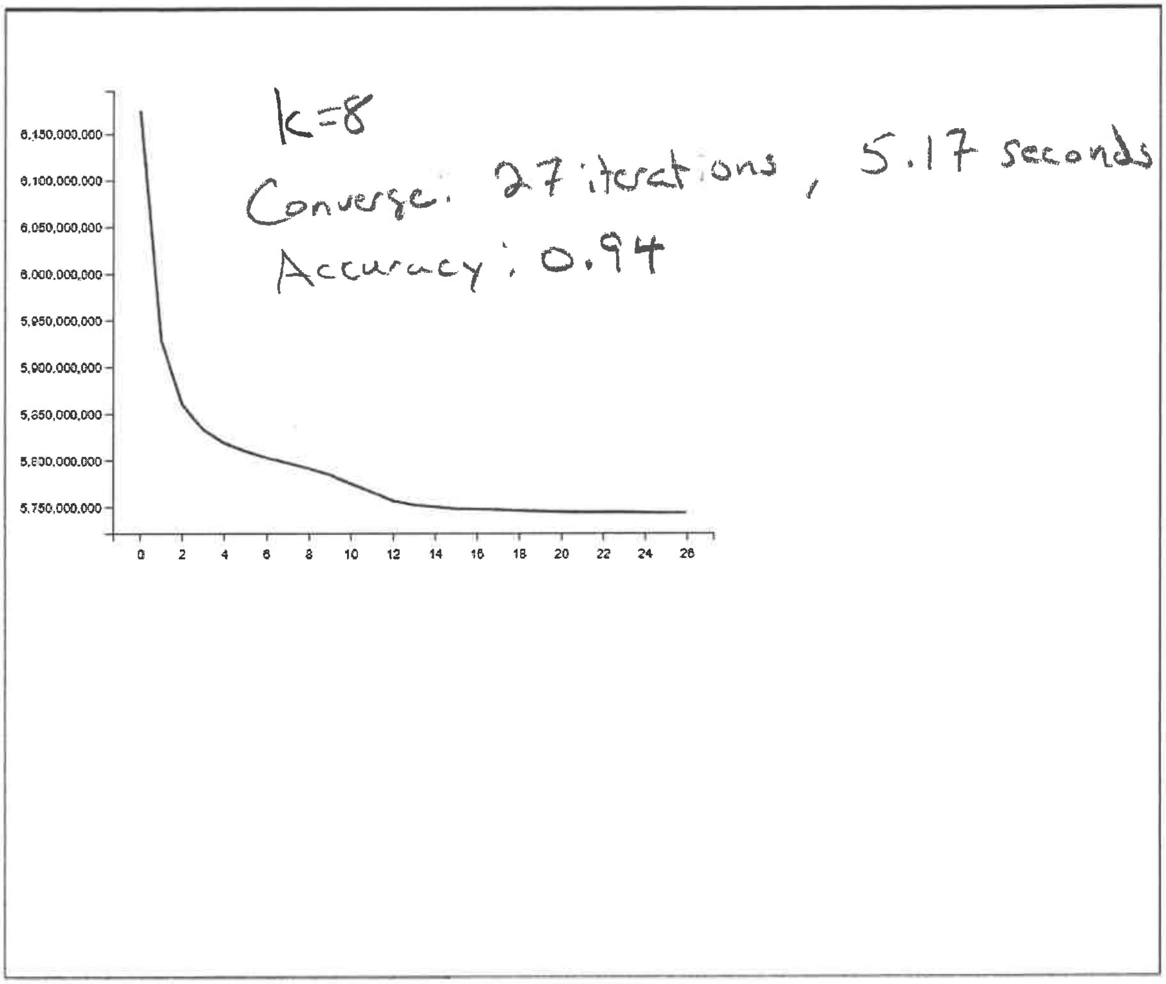
Classification accuracy: 0.94

#################

Project data into 1 dimension with PCA converged in 33 iteration (5.79 seconds)

Classification accuracy: 0.74

Process finished with exit code 0

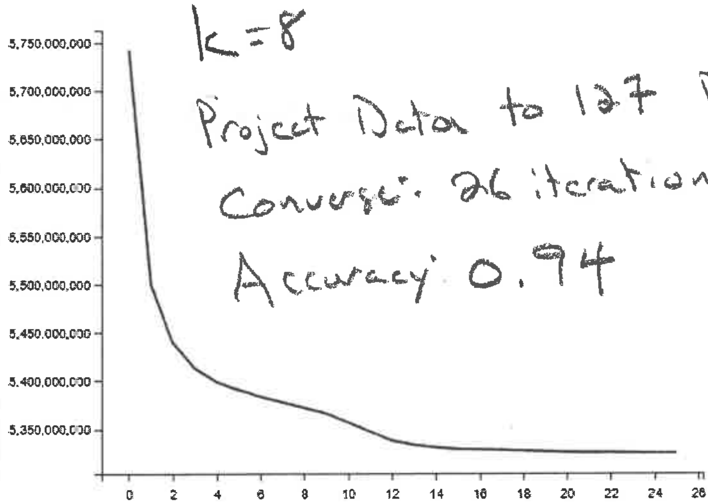

k=8
Converge: 27 iterations, 5.17 seconds
Accuracy: 0.94
6,150,000,000
6,100,000,000
6,050,000,000
6,000,000,000
5,950,000,000
5,900,000,000
5,850,000,000
5,800,000,000
5,750,000,000
0 2 4 6 8 10 12 14 16 18 20 22 24 26

k=8

Project Data to 127 Dimensions (PCA)

Converge: 26 iterations, 4.74 Seconds

Accuracy: 0.94

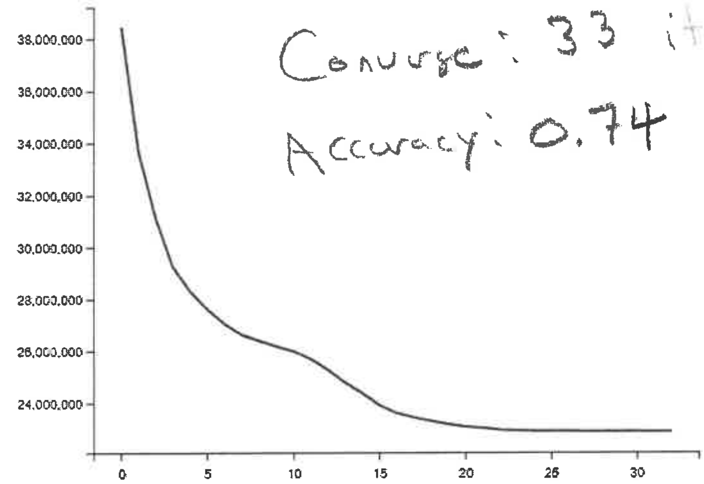


k = 8

Project Data to 1 Dimension (PCA)

Converge: 33 iterations, 5.79 seconds

Accuracy: 0.74



k=4

Using raw data converged in 20 iteration (2.67 seconds)

Classification accuracy: 0.83

#################

Project data into 127 dimensions with PCA converged in 20 iteration (2.66 seconds)

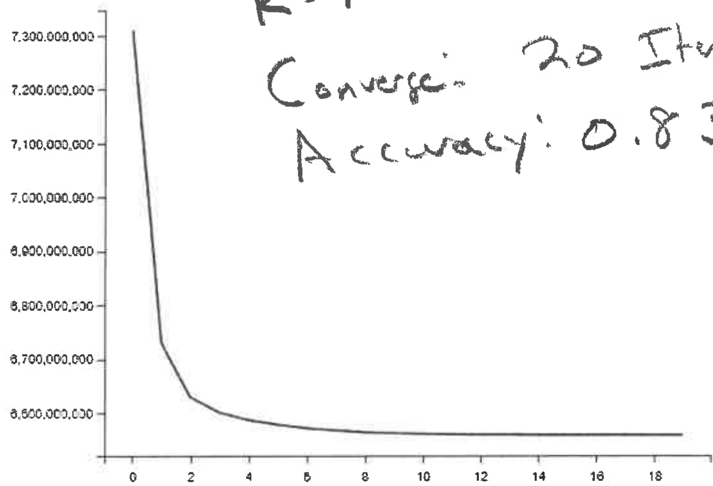
Classification accuracy: 0.83

#################

Project data into 1 dimension with PCA converged in 22 iteration (2.85 seconds)

Classification accuracy: 0.72

Process finished with exit code 0



k=4

Converge: 20 Iterations in 2.67 Seconds

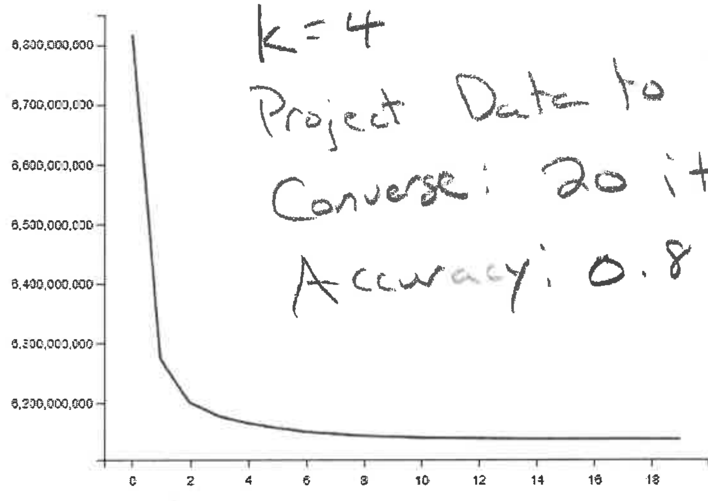
Accuracy: 0.83

k = 4

Project Data to 127 Dimensions (PCA)

Converse: 20 iterations in 2.66 seconds

Accuracy: 0.83



k=4

Project Data to 1 Dimensions (PCA)

Converge: 22 iterations, 2.85 seconds

Accuracy: 0.72