DEPARTMENT OF INDUSTRIAL & SYSTEMS ENGINEERING, UNIVERSITY OF SOUTHERN CALIFORNIA

Machine Learning HW4

Mengchun (Marissa) Wu (*2314-9438-39*)

November 19, 2014

Collaborate with Jizhe Zhang

1.Boosting

(a)

$$g_i = \frac{\partial L(y_i, \hat{y}_i)}{\partial \hat{y}_i} = \frac{\partial (y_i - \hat{y}_i)^2}{\partial \hat{y}_i} = -2(y_i - \hat{y}_i)$$

(b) Let $A = \sum_{i=1}^{n} (-g_i - \gamma h(\mathbf{x}_i))^2$

Then $\min_{\gamma} A$ can be calculated as:

$$\frac{\partial A}{\partial \gamma} = \sum_{i=1}^{n} (2(y_i - \hat{y}_i) - \gamma h(\mathbf{x}_i))(-2h(\mathbf{x}_i)) = 0$$

$$\gamma = \frac{\sum_{i=1}^{n} 2(y_i - \hat{y}_i) h(\mathbf{x}_i)}{\sum_{i=1}^{n} h(\mathbf{x}_i)^2}$$

the optimal value of the step size γ can be computed in the closed form in this step.

$$A_{min} = \sum_{i=1}^{n} (2(y_i - \hat{y}_i) - \frac{\sum_{i=1}^{n} 2(y_i - \hat{y}_i) h(\mathbf{x}_i)}{\sum_{i=1}^{n} h(\mathbf{x}_i)^2} h(\mathbf{x}_i))^2$$

$$h^* = arg\min A_{min} \Rightarrow \frac{\partial A_{min}}{\partial h} = 0$$

h^* can be derived independent of the value of γ.

(c)

$$L_i = L(y_i, \hat{y}_i + \alpha h^*(\mathbf{x}_i)) = (y_i - \hat{y}_i - \alpha h^*(\mathbf{x}_i))^2$$

$$\alpha^* = arg\min \sum_{i=1}^{n} L_i$$

$$\frac{\partial \sum_{i=1}^{n} L_i}{\partial \alpha} = \sum_{i=1}^{n} (y_i - \hat{y}_i - \alpha h^*(\mathbf{x}_i))(-2h^*(\mathbf{x}_i)) = 0$$

$$\alpha^* = \frac{\sum\limits_{i=1}^{n} (y_i - \hat{y}_i) h^*(\mathbf{x}_i)}{\sum\limits_{i=1}^{n} h^*(\mathbf{x}_i)^2}$$

update:

$$\hat{y}_i \leftarrow \hat{y}_i + \alpha^* h^*(\mathbf{x}_i)$$

$$\hat{y}_i \leftarrow \hat{y}_i + \frac{\sum\limits_{i=1}^{n} (y_i - \hat{y}_i) h^*(\mathbf{x}_i)}{\sum\limits_{i=1}^{n} h^*(\mathbf{x}_i)^2} h^*(\mathbf{x}_i)$$

2.Neural Network

(a) Assuming there are more than one hidden layer with linear activation functions:

The output in layer i is defined as $z_k^{(i)}$, for layer 1:

$$z_j^{(1)} = \sum_i w_{ji}^{(1)} x_i$$

For layer 2:

$$z_k^{(2)} = \sum_k w_{kj}^{(2)} (\sum_i w_{ji}^{(1)} x_i) = \sum_i c_i x_i$$

is also linear.
Hence, we can conclude that with linear activation functions in the hidden layers, the output in the last hidden layer is linear.
For output layer with a single logistic output:

$$y = \sigma(\sum_i c_i x_i)$$

the output is equivalent to the logistic regression.

(b)

$$\frac{\partial L}{\partial w_{ki}} = \frac{\partial L}{\partial a_k}\frac{\partial a_k}{\partial w_{ki}} = \frac{\partial L}{\partial a_k} x_i$$

where

$$a_k = \sum_{i=1}^{3} w_{ki} x_i$$

$$\frac{\partial L}{\partial a_k} = \frac{\partial L}{\partial z_k}\frac{\partial z_k}{\partial a_k} = \frac{\partial L}{\partial z_k} h'(a_k)$$

where

$$h(x) = tanh(x)$$

$$h'(x) = 1 - tanh^2(x)$$

hence

$$h'(a_k) = 1 - tanh^2(\sum_{i=1}^{3} w_{ki} x_i)$$

$$\frac{\partial L}{\partial z_k} = \frac{\partial L}{\partial \hat{y}_j}\frac{\partial \hat{y}_j}{\partial z_k} = -(y_j - \hat{y}_j) v_{jk}$$

Thus

$$\frac{\partial L}{\partial w_{ki}} = -(y_j - \hat{y}_j) v_{jk} (1 - tanh^2(\sum_{i=1}^{3} w_{ki} x_i)) x_i$$

$$w_{ki}^{new} \leftarrow w_{ki} - \eta(y_j - \hat{y}_j)v_{jk}(1 - tanh^2(\sum_{i=1}^{3} w_{ki}x_i))x_i$$

Also, for v_{jk}:

$$\frac{\partial L}{\partial v_{jk}} = \frac{\partial L}{\partial \hat{y}_j}\frac{\partial \hat{y}_j}{\partial v_{jk}} = -(y_j - \hat{y}_j)z_k = -(y_j - \hat{y}_j)tanh(\sum_{i=1}^{3} w_{ki}x_i)$$

$$v_{jk}^{new} \leftarrow v_{jk} - \eta(y_j - \hat{y}_j)tanh(\sum_{i=1}^{3} w_{ki}x_i)$$

where η is step size.

3.Clustering

(a)

$$\frac{\partial D_k}{\partial \mu_k} = -2 \sum_{n:r_{nk}=1}^{N} \left(x_n - \mu_k\right) = 0$$

$$\Rightarrow \mu_k = \frac{\sum_{n:r_{nk}=1} x_n}{N_k}$$

where N_k is the number of $r_{nk} = 1$.
For each μ_k its second order derivative is $2I$, which is semi-positive definite. Hence μ_k is the optimal solution. So when μ_k is the mean of all data points assigned to the cluster k, for any k, then the objective D is minimized.

(b) If we use L_1 norm as the new cost function, each element of the vector is independent from each other when calculating the cost. Let's just consider the d-th element for μ_k.

$$D_{kd} = \sum_{n:r_{nk}=1} |x_{nd} - \mu_{kd}|$$

Suppose the d-th element's median is unique and there exists another optimal solution which is different from the median via δ, and $\delta > 0$. Suppose this optimal solution is bigger than median, thus for the numbers smaller than median and median itself, their cost is increased by $\left(\frac{N+1}{2}\right)\delta$ and for the numbers bigger than median, their cost is reduced by $\left(\frac{N-1}{2}\right)\delta$ at most. Hence, the new total cost is increased by δ at least and there's no such optimal solution. Median is the only optimal solution.

4.Mixture Models

(a)

$$Y=(Y_1,Y_2,...,Y_n)^T, X=(X_1,X_2,...,X_n)^T, C=(c_1,c_2,...,c_n)^T$$

log-likelihood in terms of unobserved variables:

$$L = log\prod_{i=1}^{n} P(Y_i, X_i|\lambda, c_i)$$
$$=\sum_{i=1}^{n} logP(Y_i, X_i|\lambda, c_i)$$
$$=\sum_{i=1}^{r} logP(Y_i, X_i|\lambda, c_i) + \sum_{i=r+1}^{n} logP(Y_i, X_i|\lambda, c_i)$$
$$=\sum_{i=1}^{r} log(\lambda e^{-\lambda X_i}) + \sum_{i=r+1}^{n} log(P(X_i|\lambda, c_i)P(Y_i|X_i, \lambda, c_i))$$
$$=\sum_{i=1}^{r} log(\lambda e^{-\lambda X_i}) + \sum_{i=r+1}^{n} log(\lambda e^{-\lambda(X_i-c_i)}P(X_i \geq c_i))$$
$$=\sum_{i=1}^{r} log(\lambda e^{-\lambda X_i}) + \sum_{i=r+1}^{n} log(\lambda e^{-\lambda(X_i-c_i)}e^{-\lambda c_i})$$
$$=\sum_{i=1}^{r} log(\lambda e^{-\lambda X_i}) + \sum_{i=r+1}^{n} log(\lambda e^{-\lambda X_i})$$
$$=\sum_{i=1}^{n} log(\lambda e^{-\lambda X_i})$$

Hence

$$L=\sum_{i=1}^{n}(log\lambda - \lambda X_i) = nlog\lambda - \lambda\sum_{i=1}^{n} X_i$$

(b) E-Step: let λ^{old} be the estimation for λafter i^{th} iteration:

$$Q(\lambda,\lambda^{old}) = E(logP(Y_i, X_i|\lambda, c_i)|\lambda^{old}, c_i)$$
$$=\sum_{X} logP(Y_i, X_i|\lambda, c_i)P(X_i|Y_i, \lambda^{old}, c_i)$$
$$=\sum_{X_1}^{X_r}(log\lambda - \lambda X_i) + \sum_{X_{r+1}}^{X_n}\int_{c_i}^{\infty}(log\lambda - \lambda X_i)\lambda e^{-\lambda(X_i-c_i)}dX_i$$
$$= nlog\lambda - \lambda\sum_{i=1}^{r} X_i - \lambda\sum_{i=r+1}^{n}(c_i + \frac{1}{\lambda^{old}})$$

(c) M-step:

$$\frac{\partial Q}{\partial\lambda} = 0 \Rightarrow \lambda = \frac{n}{\sum_{i=1}^{r} X_i + \sum_{i=1+r}^{n}(c_i + \frac{1}{\lambda^{old}})}$$

Hence,

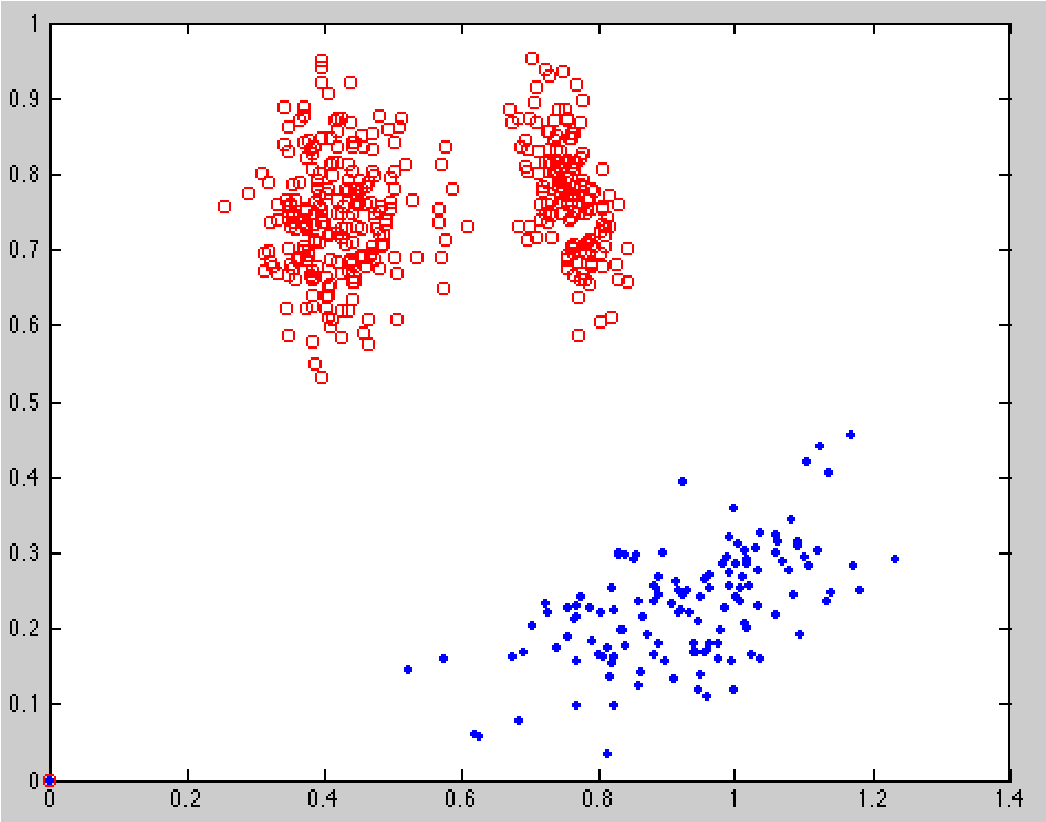
$$\lambda^{new} = \frac{n}{\sum_{i=1}^{r} X_i + \sum_{i=1+r}^{n}(c_i + \frac{1}{\lambda^{old}})}$$

5.Programming

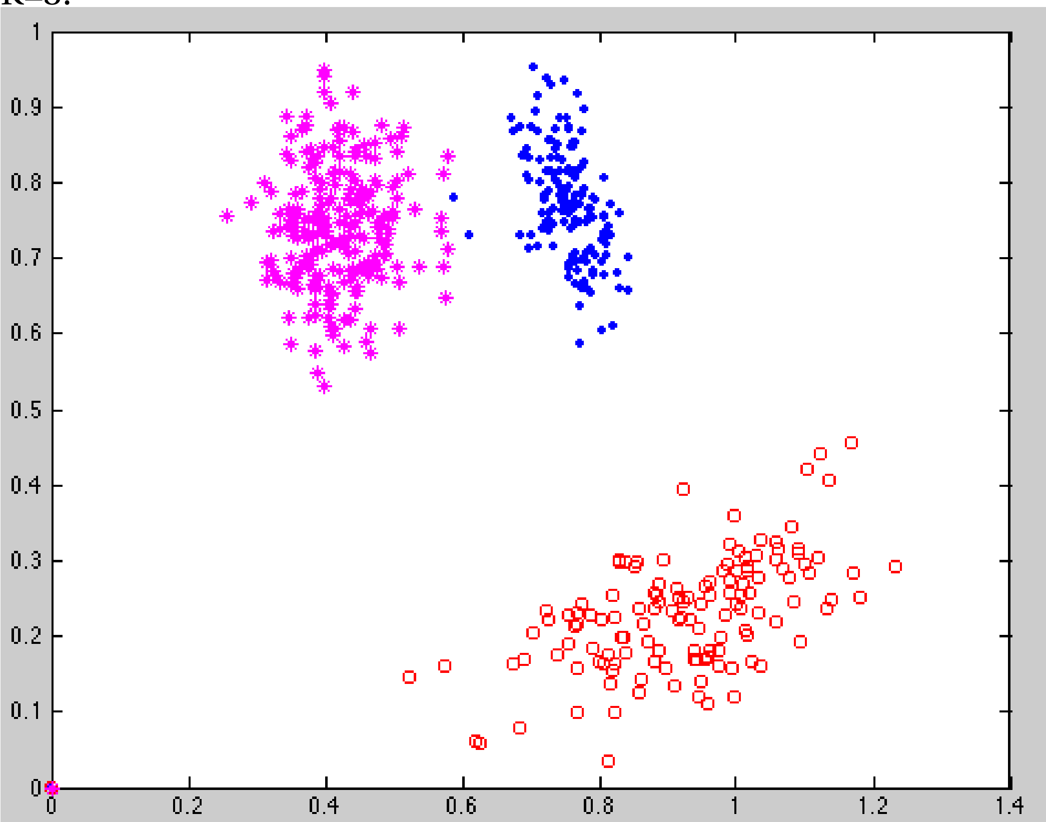
5.2 Solution:

(a)

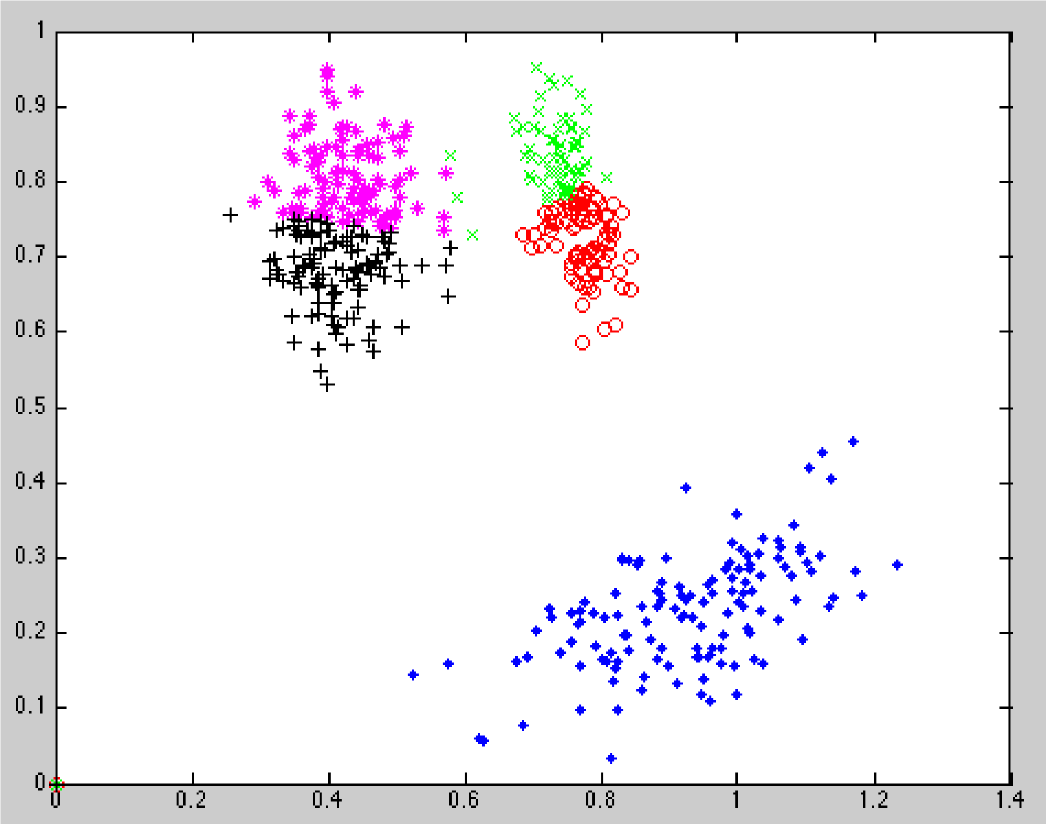
K=2:



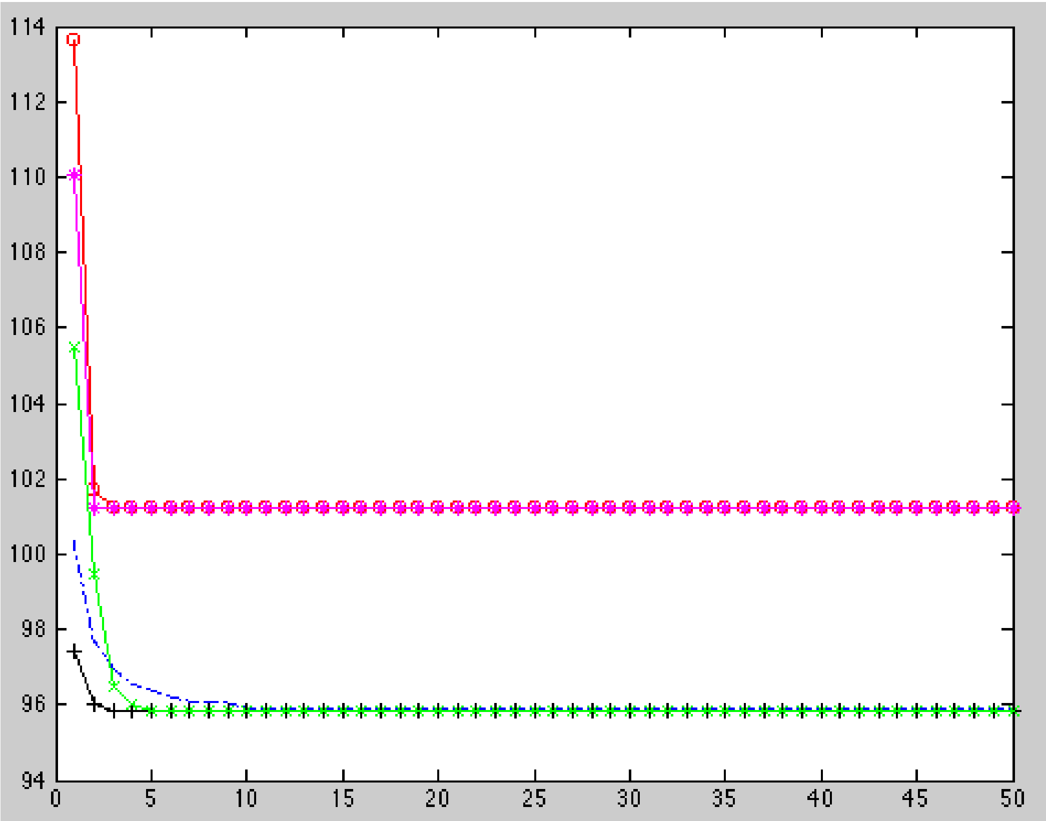
K=3:



K=5:



(b) graph with normalized data:



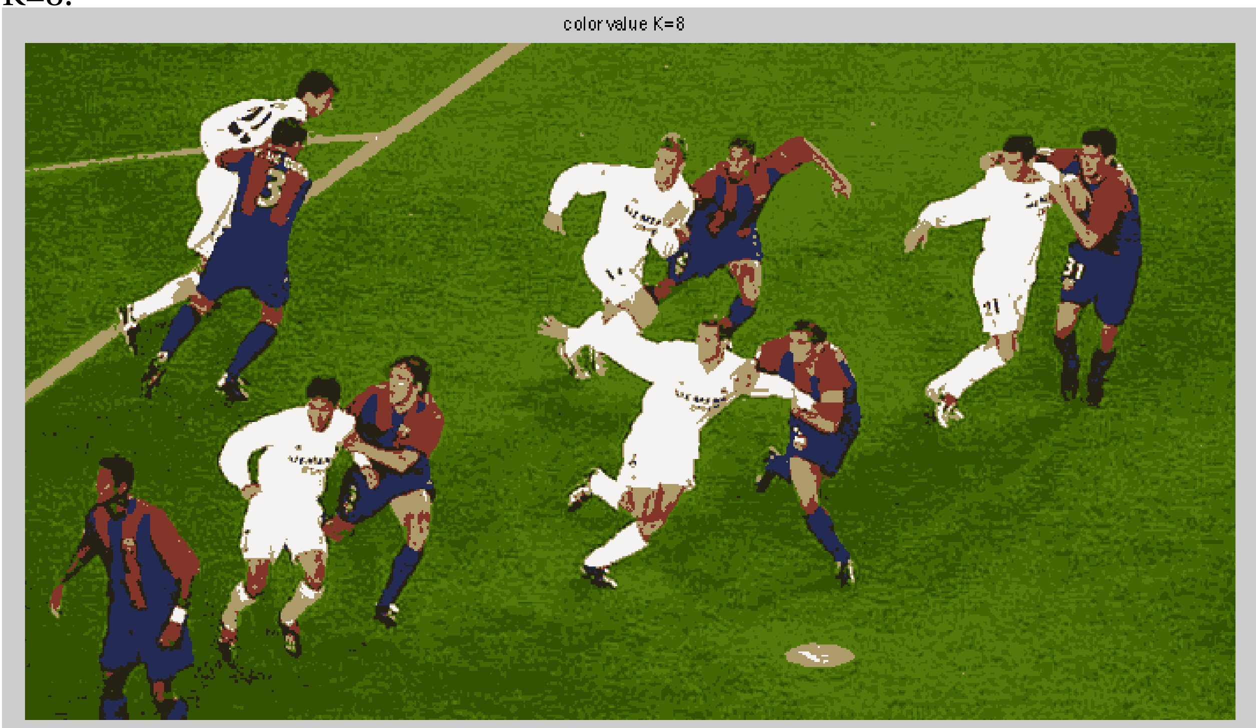
(c) k-means always converge after finite number of iterations.
For reassignment of points and re-center step in k-means is to decrease J, look at the form of J we can see that J always decrease. When J gets the local smallest value (sometimes not optimal solution), k-means converges.

5.3 Solution:

(d)

K=3:

K=8:



K=15:


color value K=15