

## HW6 Adaboost Algorithm

106034061 曾靖渝

### 1. Code for programming

#### (1)Linear Classifier:

just make a little modification of previous code:

```
std::uniform_real_distribution<double> weight_range(-1.0,1.0);
class Perception_Learning//Perception Learning Classifier(Linear Separability)
{
private:
    vector<example> training_set;
    vector<string> class_labels;
    vector<double> weight;//Memorize all of Weights
    int c_x=0;//c(x)
    int h_x=0;//h(x)
    int attribute_number;
    int instance_number;
    int class_label_number;
    int modifying_number;//Memorize Modifying times
    ...
}
```

data structure:

for each example:vector<double>, and store the class label at the end of vector  
for each training set:vector<double>

```

int compute_ClassLabel(example ex)//Compute c(x)
{
    cout<<"->c(x)="<<ex[attribute_number]<<endl;
    c_x=ex[attribute_number];
    return c_x;
}
int compute_H_ClassLabel(example ex)//Compute h(x)
{
    double sum=0;

    for(int i=0;i<=attribute_number;i++)
    {
        if(i==0)
            sum+=weight[i];
        else
            sum+=ex[i-1]*weight[i];

    }
    if(sum>0)h_x=1;
    else h_x=0;
    cout<<"->h(x)="<<h_x<<endl;
    return h_x;
}
```

```

void modify_weight(example ex)//Refresh Weights
{
    cout<<"->Modify Weight"<<endl;
    for(int i=0;i<=attribute_number;i++)
    {
        if(i==0)
            weight[i]+=ETA*(double)1*(double)(c_x-h_x);
        else
            weight[i]+=ETA*(double)ex[i-1]*(double)(c_x-h_x);
    }
    modifying_number++;
}
void print_Modifying_Number(void)
{
    cout<<"->Modify:<<modifying_number<<" times"<<endl;
}

```

Weight modification :according to the textbook;

(2)k-NN Classifier:Use previous code:

```

lass Induce_k_NN

private:
    vector<string> class_labels;
    vector<example> training_set;
    int attribute_number;
    int instance_number;
    int class_label_number;

```

```

6     //1-NN Classifier=====
7     double calculate_error(example ex1, example ex2)//to calculate the distance between two examples
8     {
9         cout<<"====calculating=====<<endl;
10        double sum=0;
11        for(int i=0;i<ex1.size()-1;i++)
12            sum+=pow((ex1[i]-ex2[i]),2);
13        sum=sqrt(sum);
14        return sum;
15    }
16    int classifier(example ex)//the classifier
17    {
18        int index=0;
19        double error=1000000;
20        for(auto s:training_set)
21        {
22            double new_error=calculate_error(s, ex);
23            if(error>=new_error)
24            {
25                index=s.size()-1;
26                error=new_error;
27                cout<<"->refresh:index:<<s.size()-1<<":<<class_labels[s.size()-1]<<":error=<<error<<" ,new_error=<<new_error<<endl;
28            }
29        }
30        else cout<<"->No refresh:<<s.size()-1<<":<<class_labels[s.size()-1]<<":error=<<error<<" ,new_error=<<new_error<<endl;
31    }
32    cout<<"->Choose index:<<index<<endl;
33    return index;
34 }

```

### (3)Adaboost algorithm:(including textbook version and original version)

```
4
5 class Adaboost
6 {
7     private:
8     int attribute_number;
9     int instance_number;
0     int class_label_number;
1     vector<string> class_labels;
2     vector<example> training_set;
3     Induce_k_NN Induce_Classifier_Textbook[training_subset_number], Induce_Classifier_Original[training_subset_number];
4     Perception_Learning Linear_Classifier[training_subset_number];
5     vector<double> px;
```

The data structure of adaboost algorithm, we need to include previous classifier :k-NN. And Linear Classifier,

```
    }
void add_to_training_set(example ex, string class_label)
{
    bool found=false;
    for(int i=0;i<class_labels.size();i++)
    {
        if(class_labels[i]==class_label)
        {
            found=true;
            ex.push_back(i);
            break;
        }
    }
    if(!found)
    {
        ex.push_back(class_labels.size());
        class_labels.push_back(class_label);
    }
    training_set.push_back(ex);
}
//get example according to probability
example get_example(void)
{
    double d= dis(gen);
    for(int i=0;i<px.size();i++)
    {
        if(d<px[i])return training_set[i];
        d-=px[i];
    }
    return training_set[training_set.size()-1];
}
```

Get example:get example according the probability

```

//Subset Evaluation
void subset_evaluation_textbook(int round)
{
    double ERROR=0;
    double BETA=0;
    vector<bool>check;
    for(int i=0;i<training_set.size();i++)
    {
        example ex=training_set[i];
        int label=Induce_Classifier_Textbook[round].classifier(ex);
        if(label!=ex[attribute_number])
        {
            ERROR+=px[i];
            cout<<"->Example:<<i<<" : ->Misclassify" << endl;
        }
        check.push_back(label==ex[attribute_number]);
    }
    cout<<"->Error=" <<ERROR << endl;
    BETA=(double)ERROR/(double)(1-ERROR);
    cout<<"->BETA=" <<BETA << endl;
    if(BETA<-0.001||BETA>0.001)
    {
        for(int i=0;i<training_set.size();i++)
            if(check[i])px[i]*=BETA;
        double sum=0;
        for(int i=0;i<px.size();i++)sum+=px[i];
        for(int i=0;i<px.size();i++)px[i]/=sum;
    }
}

```

```

}

void subset_evaluation_original(int round)
{
    double ERROR=0;
    double BETA=0;
    vector<bool>check;
    for(int i=0;i<training_set.size();i++)
    {
        example ex=training_set[i];
        int label=Induce_Classifier_Original[round].classifier(ex);
        if(label!=ex[attribute_number])
        {
            ERROR+=px[i];
            cout<<"->Example:<<i<<" : ->Misclassify" << endl;
        }
        check.push_back(label==ex[attribute_number]);
    }
    cout<<"->Error=" <<ERROR << endl;
    BETA=sqrt((double)ERROR/(double)(1-ERROR));
    cout<<"->BETA=" <<BETA << endl;
    if(BETA<-0.001||BETA>0.001)
    {
        for(int i=0;i<training_set.size();i++)
            if(check[i])px[i]*=BETA;
            else px[i]/=BETA;
        double sum=0;
        for(int i=0;i<px.size();i++)sum+=px[i];
        for(int i=0;i<px.size();i++)px[i]/=sum;
    }
}

```

The above are two version of adboost impliment, in order to creat sub set and refresh probability in each round.

## Classification of tow version of adaboost

```
//Subset Classification
void subset_classification_textbook(int round)
{
    cout<<"Induce Classifier:<<round<<">:Subset Classification(Textbook Version)>"<<endl;
    Induce_Classifier_Textbook[round].Initialization(attribute_number, instance_number, class_label_number);
    vector<example> subset=creating_subset();
    for(auto ex:subset)
        Induce_Classifier_Textbook[round].add_to_training_set(ex, class_labels[ex[ex.size()-1]]);
    Induce_Classifier_Textbook[round].print_training_set();
    subset_evaluation_textbook(round);
}
void subset_classification_original(int round)
{
    cout<<"Induce Classifier:<<round<<">:Subset Classification(Original Version)>"<<endl;
    Induce_Classifier_Original[round].Initialization(attribute_number, instance_number, class_label_number);
    vector<example> subset=creating_subset();
    for(auto ex:subset)
        Induce_Classifier_Original[round].add_to_training_set(ex, class_labels[ex[ex.size()-1]]);
    Induce_Classifier_Original[round].print_training_set();
    subset_evaluation_original(round);
}
```

```
:39
:40     void Textbook_Version_Classification(void)
:41     {
:42         px=vector<double>(training_set.size(), (double)1/(double)training_set.size());
:43         //          print_all_probability();
:44         cout<<"Textbook Version....."<<endl;
:45         for(int i=1;i<training_subset_number;i++)
:46         {
:47             subset_classification_textbook(i);
:48             //          print_all_probability();
:49         }
:50     }
:51     void Original_Version_Classification(void)
:52     {
:53         px=vector<double>(training_set.size(), (double)1/(double)training_set.size());
:54         //          print_all_probability();
:55         cout<<"Original Version....."<<endl;
:56         for(int i=0;i<training_subset_number;i++)
:57         {
:58             subset_classification_original(i);
:59             //          print_all_probability();
:60         }
:61     }
```

```

//Main Classification(including two version of adaboost algorithm)
void Classification(void)
{
    Textbook_Version_Classification();
    Original_Version_Classification();
}

```

## Main classification

```

//Master Classifier
void master_classifier_textbook(void)
{
    classifier_weight_textbook=vector<double>(training_subset_number,0.2);
    for(int i=0;i<training_set.size();i++)
    {
        cout<<">>>Start Master Classifier at:<<i<<" Example....."<<endl;
        double positive=0, negative=0;
        example ex=training_set[i];
        for(int j=0;j<training_subset_number;j++)
        {
            int subset_classifier_label=Induce_Classifier_Textbook[j].classifier(ex);
            if(subset_classifier_label)positive+=classifier_weight_textbook[j];
            else negative+=classifier_weight_textbook[j];
        }
        int master_classifier_label=0;
        if(positive>negative)master_classifier_label=1;
        else master_classifier_label=0;
        cout<<">>Master Classifier Label:<<master_classifier_label<<endl;
        cout<<">>Class Label:<<ex[attribute_number]<<endl;
        if(ex[attribute_number]!=master_classifier_label)cout<<">>>Misclassified....."<<endl;
        for(int j=0;j<training_subset_number;j++)
        {
            int subset_classifier_label=Induce_Classifier_Textbook[j].classifier(ex);
            classifier_weight_textbook[j]+=ETA*((double)ex[attribute_number]-(double)master_classifier_label)*(double)subset_classifier_label;
            cout<<">>For "<<j<<" subset Classifier....."<<endl;
        }
        print_subset_classifier_weight();
        cout<<">>>Finish Master Classifier at:<<i<<" Example....."<<endl;
    }
}

```

```

}

void master_classifier_original(void)
{
    classifier_weight_original=vector<double>(training_subset_number,0.2);
    for(int i=0;i<training_set.size();i++)
    {
        cout<<">>>Start Master Classifier at:<<i<<" Example....."<<endl;
        double positive=0, negative=0;
        example ex=training_set[i];
        for(int j=0;j<training_subset_number;j++)
        {
            int subset_classifier_label=Induce_Classifier_Original[j].classifier(ex);
            if(subset_classifier_label)positive+=classifier_weight_original[j];
            else negative+=classifier_weight_original[j];
        }
        int master_classifier_label=0;
        if(positive>negative)master_classifier_label=1;
        else master_classifier_label=0;
        cout<<">>Master Classifier Label:<<master_classifier_label<<endl;
        cout<<">>Class Label:<<ex[attribute_number]<<endl;
        if(ex[attribute_number]!=master_classifier_label)cout<<">>>Misclassified....."<<endl;
        for(int j=0;j<training_subset_number;j++)
        {
            int subset_classifier_label=Induce_Classifier_Original[j].classifier(ex);
            classifier_weight_original[j]+=ETA*((double)ex[attribute_number]-(double)master_classifier_label)*(double)subset_classifier_label;
            cout<<">>For "<<j<<" subset Classifier....."<<endl;
        }
        print_subset_classifier_weight();
        cout<<">>>Finish Master Classifier at:<<i<<" Example....."<<endl;
    }
}

```

Using perception learning :to make weight of subset classifier, and main classifier

2. the outcome of each classifier

(1) Linear Classifier

```
5.0,2.2,1.0,1.0,Iris-versicolor  
->Learning.....  
->h(x)=1  
->c(x)=0  
->Modify Weight  
->Weights:0 -0.8 -0.44 -0.04 0.16  
->Misclassified:1 examples  
1: ->Learning.....  
->h(x)=0  
->c(x)=0  
->Weights:0 -0.8 -0.44 -0.04 0.16  
->Misclassified:1 examples  
2: ->Learning.....  
->h(x)=0  
->c(x)=0  
->Weights:0 -0.8 -0.44 -0.04 0.16  
->Misclassified:1 examples  
3: ->Learning.....  
->h(x)=0  
->c(x)=0  
->Weights:0 -0.8 -0.44 -0.04 0.16  
->Misclassified:1 examples  
4: ->Learning.....  
->h(x)=0  
->c(x)=0  
->Weights:0 -0.8 -0.44 -0.04 0.16  
->Misclassified:1 examples  
5: ->Learning.....  
->h(x)=0  
->c(x)=0  
->Weights:0 -0.8 -0.44 -0.04 0.16  
->Misclassified:1 examples  
6: ->Learning.....  
->h(x)=0  
->c(x)=0  
->Weights:0 -0.8 -0.44 -0.04 0.16  
->Misclassified:1 examples  
7: ->Learning.....  
->h(x)=0  
->c(x)=0  
->Weights:0 -0.8 -0.44 -0.04 0.16  
->Misclassified:1 examples  
8: ->Learning.....  
->h(x)=0  
->Misclassified:2 examples  
28: ->Learning.....  
->h(x)=1  
->c(x)=1  
->Weights:0.2 0.16 0.24 0.34 0.2  
->Misclassified:2 examples  
29: 5.1,3.5,1.4,0.2,Iris-setosa  
4.9,3.0,1.4,0.2,Iris-setosa  
4.7,3.2,1.3,0.2,Iris-setosa  
4.6,3.1,1.5,0.2,Iris-setosa  
4.6,3.4,1.4,0.3,Iris-setosa  
5.0,3.4,1.5,0.2,Iris-setosa  
4.4,2.9,1.4,0.2,Iris-setosa  
4.9,3.1,1.5,0.1,Iris-setosa  
5.4,3.7,1.5,0.2,Iris-setosa  
4.8,3.0,1.4,0.1,Iris-setosa  
4.3,3.0,1.1,0.1,Iris-setosa  
5.8,4.0,1.2,0.2,Iris-setosa  
5.7,4.4,1.5,0.4,Iris-setosa  
5.4,3.9,1.3,0.4,Iris-setosa  
5.7,3.8,1.7,0.3,Iris-setosa  
5.1,3.8,1.5,0.3,Iris-setosa  
5.4,3.4,1.7,0.2,Iris-setosa  
5.1,3.7,1.5,0.4,Iris-setosa  
4.6,3.6,1.0,0.2,Iris-setosa  
5.1,3.3,1.7,0.5,Iris-setosa  
5.6,2.5,1.9,1.1,Iris-setosa  
5.0,3.0,1.6,0.2,Iris-setosa  
5.0,3.4,1.6,0.4,Iris-setosa  
5.2,3.5,1.5,0.2,Iris-setosa  
5.2,3.4,1.4,0.2,Iris-setosa  
4.7,3.2,1.6,0.2,Iris-setosa  
4.8,3.1,1.6,0.2,Iris-setosa  
5.4,3.4,1.5,0.4,Iris-setosa  
5.5,4.2,1.4,0.2,Iris-setosa  
4.9,3.1,1.5,0.1,Iris-setosa  
->Learning.....  
->h(x)=1  
->c(x)=0  
->Modify Weight  
->Weights:0 -0.86 -0.46 0.06 0.16  
->Misclassified:3 examples  
30: ->Learning.....
```

```

->c(x)=1
->Weights:0.2 0.26 0.12 0.78 0.42
->Misclassified:4 examples
32: ->Learning.....
->h(x)=1
->c(x)=1
->Weights:0.2 0.26 0.12 0.78 0.42
->Misclassified:4 examples
33: ->Learning.....
->h(x)=1
->c(x)=1
->Weights:0.2 0.26 0.12 0.78 0.42
->Misclassified:4 examples
34: ->Learning.....
->h(x)=1
->c(x)=1
->Weights:0.2 0.26 0.12 0.78 0.42
->Misclassified:4 examples
35: ->Learning.....
->h(x)=1
->c(x)=1
->Weights:0.2 0.26 0.12 0.78 0.42
->Misclassified:4 examples
36: ->Learning.....
->h(x)=1
->c(x)=1
->Weights:0.2 0.26 0.12 0.78 0.42
->Misclassified:4 examples
37: ->Learning.....
->h(x)=1
->c(x)=1
->Weights:0.2 0.26 0.12 0.78 0.42
->Misclassified:4 examples
38: ->Learning.....
->h(x)=1
->c(x)=1
->Weights:0.2 0.26 0.12 0.78 0.42
->Misclassified:4 examples
39:
->Learning.....
->h(x)=1
->c(x)=1
->Weights:0.2 0.26 0.12 0.78 0.42
->Misclassified:4 examples

```

The result show the number of example misclassified

(2) adaboost algorithm(textbook version)

```

5.4,3.4,1.5,0.4,Iris-setosa
5.5,4.2,1.4,0.2,Iris-setosa
4.9,3.1,1.5,0.1,Iris-setosa31: 32: 33: 34: 35: 36: 37: 38: 39: 40: 41: 42: 43: 44: 45: 46: 47: 48: 49: 50: 51: 52: 53: 54: 55: 56: 57: 58: 59:
60: 6.7,3.1,4.4,1.4,Iris-versicolor
5.6,3.0,4.5,1.5,Iris-versicolor
5.8,2.7,4.1,1.0,Iris-versicolor
6.2,2.2,4.5,1.5,Iris-versicolor
5.9,3.2,4.8,1.8,Iris-versicolor
6.1,2.8,4.0,1.3,Iris-versicolor
6.3,2.5,4.9,1.5,Iris-versicolor
6.1,2.8,4.7,1.2,Iris-versicolor
6.4,2.9,4.3,1.3,Iris-versicolor
6.6,3.0,4.4,1.4,Iris-versicolor
6.8,2.8,4.8,1.4,Iris-versicolor
6.0,2.9,4.5,1.5,Iris-versicolor
5.7,2.6,3.5,1.0,Iris-versicolor
5.5,2.4,3.8,1.1,Iris-versicolor
5.5,2.4,3.7,1.0,Iris-versicolor
5.8,2.7,3.9,1.2,Iris-versicolor
5.4,3.0,4.5,1.5,Iris-versicolor
6.0,3.4,4.5,1.6,Iris-versicolor
6.7,3.1,4.7,1.5,Iris-versicolor
6.3,2.3,4.4,1.3,Iris-versicolor
5.6,3.0,4.1,1.3,Iris-versicolor
5.5,2.6,4.4,1.2,Iris-versicolor
6.1,3.0,4.6,1.4,Iris-versicolor
5.8,2.6,4.0,1.2,Iris-versicolor
5.6,2.7,4.2,1.3,Iris-versicolor
5.7,3.0,4.2,1.2,Iris-versicolor
5.7,2.9,4.2,1.3,Iris-versicolor
6.2,2.9,4.3,1.3,Iris-versicolor
5.1,2.5,3.0,1.1,Iris-versicolor
5.7,2.8,4.1,1.3,Iris-versicolor61: 62: 63: 64: 65: 66: 67: 68: 69: 70: 71: 72: 73: 74: 75: 76: 77: 78: 79: 80: 81: 82: 83: 84: 85: 86: 87: 88: 89:

```

```
Textbook Version.....  
Induce Classifier:1->:Subset Classification(Textbook Version)->  
>Example:14:->Misclassify  
>Example:15:->Misclassify  
>Example:23:->Misclassify  
>Example:50:->Misclassify  
>Error=0.0444444  
>BETA=0.0465116  
Induce Classifier:2->:Subset Classification(Textbook Version)->  
>Example:15:->Misclassify  
>Example:20:->Misclassify  
>Example:25:->Misclassify  
>Example:27:->Misclassify  
>Example:29:->Misclassify  
>Example:62:->Misclassify  
>Example:72:->Misclassify  
>Example:73:->Misclassify  
>Example:74:->Misclassify  
>Example:75:->Misclassify  
>Example:76:->Misclassify  
>Example:80:->Misclassify  
>Example:81:->Misclassify  
>Example:83:->Misclassify  
>Example:84:->Misclassify  
>Example:85:->Misclassify  
>Example:86:->Misclassify  
>Example:88:->Misclassify  
>Example:89:->Misclassify  
>Error=0.229651  
>BETA=0.298113  
Induce Classifier:3->:Subset Classification(Textbook Version)->  
>Example:1:->Misclassify  
>Example:2:->Misclassify  
>Example:4:->Misclassify  
>Example:8:->Misclassify  
>Example:10:->Misclassify  
>Example:12:->Misclassify  
>Example:13:->Misclassify  
>Example:14:->Misclassify  
>Example:20:->Misclassify  
>Example:23:->Misclassify  
>Example:30:->Misclassify  
>Example:35:->Misclassify  
>Example:37:->Misclassify
```

Show the misclassified example :

```
>->>Start Master Classifier at:0 Example.....  
>Master Classifier Label:0  
>Class Label:0  
>For 0 subset Classifier.....  
>For 1 subset Classifier.....  
>For 2 subset Classifier.....  
>For 3 subset Classifier.....  
>For 4 subset Classifier.....  
>For 5 subset Classifier.....  
>For 6 subset Classifier.....  
>For 7 subset Classifier.....  
>For 8 subset Classifier.....  
>Subset Classifier Weight:  
0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2  
>->>Finish Master Classifier at:0 Example.....  
>->>Start Master Classifier at:1 Example.....  
>Master Classifier Label:0  
>Class Label:0  
>For 0 subset Classifier.....  
>For 1 subset Classifier.....  
>For 2 subset Classifier.....  
>For 3 subset Classifier.....  
>For 4 subset Classifier.....  
>For 5 subset Classifier.....  
>For 6 subset Classifier.....  
>For 7 subset Classifier.....  
>For 8 subset Classifier.....  
>Subset Classifier Weight:  
0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2  
>->>Finish Master Classifier at:1 Example.....  
>->>Start Master Classifier at:2 Example.....  
>Master Classifier Label:0  
>Class Label:0  
>For 0 subset Classifier.....  
>For 1 subset Classifier.....  
>For 2 subset Classifier.....  
>For 3 subset Classifier.....  
>For 4 subset Classifier.....  
>For 5 subset Classifier.....  
>For 6 subset Classifier.....  
>For 7 subset Classifier.....  
>For 8 subset Classifier.....  
>Subset Classifier Weight:  
0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2
```

```
->Subset Classifier Weight:  
0.2 0.4 0.4 0.4 0.4 0.4 0.4 0.4 0.4 0.6  
>->>Finish Master Classifier at:26 Example.....  
>->>Start Master Classifier at:27 Example.....  
>Master Classifier Label:1  
>Class Label:1  
>For 0 subset Classifier.....  
>For 1 subset Classifier.....  
>For 2 subset Classifier.....  
>For 3 subset Classifier.....  
>For 4 subset Classifier.....  
>For 5 subset Classifier.....  
>For 6 subset Classifier.....  
>For 7 subset Classifier.....  
>For 8 subset Classifier.....  
>Subset Classifier Weight:  
0.2 0.4 0.4 0.4 0.4 0.4 0.4 0.4 0.4 0.6  
>->>Finish Master Classifier at:27 Example.....  
>->>Start Master Classifier at:28 Example.....  
>Master Classifier Label:1  
>Class Label:1  
>For 0 subset Classifier.....  
>For 1 subset Classifier.....  
>For 2 subset Classifier.....  
>For 3 subset Classifier.....  
>For 4 subset Classifier.....  
>For 5 subset Classifier.....  
>For 6 subset Classifier.....  
>For 7 subset Classifier.....  
>For 8 subset Classifier.....  
>Subset Classifier Weight:  
0.2 0.4 0.4 0.4 0.4 0.4 0.4 0.4 0.4 0.6  
>->>Finish Master Classifier at:28 Example.....
```

Weight classifier by perception learning

(2) Original Version of adaboost

```
=====
Original Version.....  
Induce Classifier:0->:Subset Classification(Original Version)->  
->Example:20:->Misclassify  
->Example:39:->Misclassify  
->Example:44:->Misclassify  
->Example:45:->Misclassify  
->Example:53:->Misclassify  
->Error=0.0555556  
->BETA=0.242536  
Induce Classifier:1->:Subset Classification(Original Version)->  
->Example:22:->Misclassify  
->Example:38:->Misclassify  
->Example:50:->Misclassify  
->Example:55:->Misclassify  
->Example:57:->Misclassify  
->Example:59:->Misclassify  
->Example:72:->Misclassify  
->Example:73:->Misclassify  
->Example:74:->Misclassify  
->Example:75:->Misclassify  
->Example:83:->Misclassify  
->Example:88:->Misclassify  
->Error=0.0705882  
->BETA=0.275589  
Induce Classifier:2->:Subset Classification(Original Version)->  
->Example:5:->Misclassify  
->Example:7:->Misclassify  
->Example:19:->Misclassify  
->Example:21:->Misclassify  
->Example:22:->Misclassify  
->Example:25:->Misclassify  
->Example:26:->Misclassify  
->Example:29:->Misclassify  
->Example:32:->Misclassify  
->Example:38:->Misclassify  
->Example:40:->Misclassify  
->Example:55:->Misclassify  
->Example:88:->Misclassify  
->Error=0.195148  
->BETA=0.492406  
Induce Classifier:3->:Subset Classification(Original Version)->  
->Example:10:->Misclassify  
->Example:20:->Misclassify
```

```
->Class Label:1
->For 0 subset Classifier.....
->For 1 subset Classifier.....
->For 2 subset Classifier.....
->For 3 subset Classifier.....
->For 4 subset Classifier.....
->For 5 subset Classifier.....
->For 6 subset Classifier.....
->For 7 subset Classifier.....
->For 8 subset Classifier.....
->Subset Classifier Weight:
0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2
>->>Finish Master Classifier at:87 Example.....
>->>Start Master Classifier at:88 Example.....
->Master Classifier Label:1
->Class Label:1
->For 0 subset Classifier.....
->For 1 subset Classifier.....
->For 2 subset Classifier.....
->For 3 subset Classifier.....
->For 4 subset Classifier.....
->For 5 subset Classifier.....
->For 6 subset Classifier.....
->For 7 subset Classifier.....
->For 8 subset Classifier.....
->Subset Classifier Weight:
0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2
>->>Finish Master Classifier at:88 Example.....
>->>Start Master Classifier at:89 Example.....
->Master Classifier Label:1
->Class Label:1
->For 0 subset Classifier.....
->For 1 subset Classifier.....
->For 2 subset Classifier.....
->For 3 subset Classifier.....
->For 4 subset Classifier.....
->For 5 subset Classifier.....
->For 6 subset Classifier.....
->For 7 subset Classifier.....
->For 8 subset Classifier.....
->Subset Classifier Weight:
0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2
>->>Finish Master Classifier at:89 Example.....
```

### 3. Discussion:

(1) According to the data, we can know the linear classifier perform better initially, However, as the number of example growing, Linear Classifier gradually lost its accuracy

(2) In great number of examples, adaboost classifier perform better. In detail, the original also perform better then textbook version. However, we don't know if the original version will overfit.