CS 6375 – Machine Learning
Homework#4

Name: Revanth Segu

1. Kmeans:

The program expects three arguments (< input_filepath> < cluster_no> < output_filepath>).
input_filepath : absolute path of input image (or just the file name if the image is in same path as the class file)
cluster_no : number of clusters the image has to be clustered into
output_filepath : absolute path of output image (or just the file name if the image is in same path as the class file)

The program can be executed by following the two steps given below:

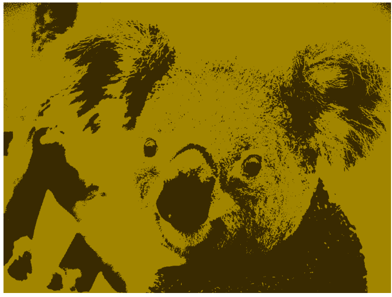
```
1.javac KMeans.java
2.java KMeans Koala.png 15 Koala_Out_15.png
```

For selecting the initial cluster centers a new method has been adopted. For deciding 'k' cluster centers each of the entire range RGB values will be divided p(say 10) segments which entire range of RGBs into (10*10*10 segments) and the occurrences of RGB color values in each segment is recorded and the most frequent k segment center is taken as cluster centers.

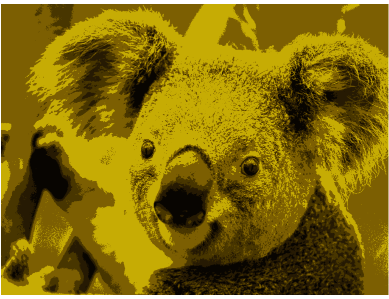
The algorithm is run till convergence of RGB values or maximum to a max number of iterations specified by MAX_ITR variable in the code.

Compression Ratios:

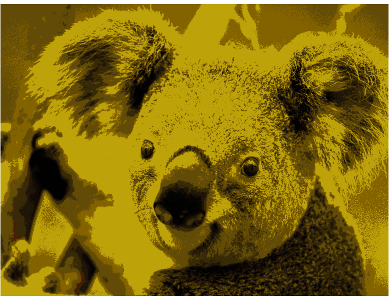
Kola.png



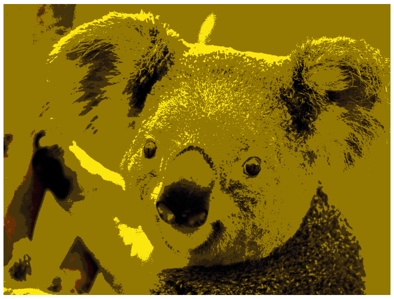
K=2



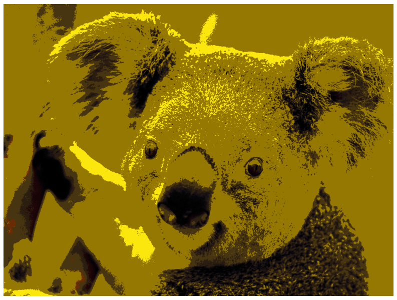
k=5



k=10



K=15

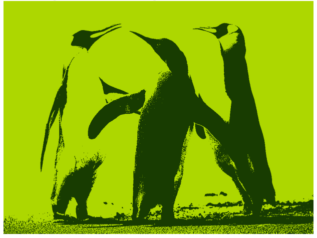


k=20

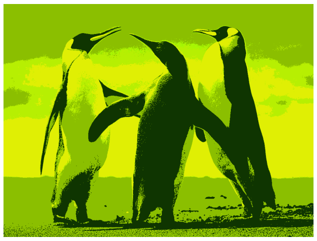
Table showing Number of clusters compression ratios

Number of clusters(k)	Compression Ratio
2	98.12%
5	95.35%
10	93.45%
15	91.64%
20	89.56%

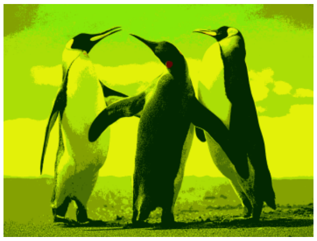
Penguins.png



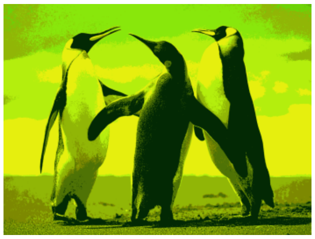
K=2



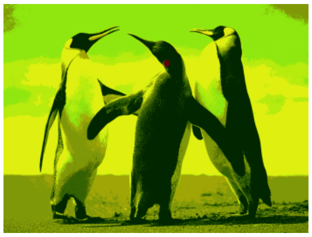
k=5



k=10



K=15



k=20

Table showing Number of clusters compression ratios

Number of clusters(k)	Compression Ratio
2	98.4%
5	97.1%
10	95.44%
15	94.94%
20	94.7%

It has been observed that the image quality decreases as the compression ratio increases. The more number of clusters clearer the image looks best k for penguins.png is 50 and for kola.png 25.

2. EM Algorithm:

EMM algorithm is implemented in EMGMM.java it expects one argument the complete path of file “em_data.txt” . ‘.’ Can be used if the file em_data.txt is in the same directory as the java class file

The program can be executed by following the two steps given below:

1.javac EMGMM.java
2.java EMGMM.java em_data.txt

It has been observed that randomly selecting the 3 initial data points and variances the three clusters converge to means
5.7648231744639435, 4.858437514325959, 4.858437514325959 with variances 61.82956942315852,55.789625609382426,55.789625609382426 respectively.

And by initializing random probabilities to cluster points clusters converge to means are 8.945564806912449,3.268066698101703,3.268066698101703 with variances 70.28735678083237,40.30811372083744,40.30811372083744

EMM algorithm with known variances EMGMMVar.java it expects one argument the complete path of file “em_data.txt” . ‘.’ Can be used if the file em_data.txt is in the same directory as the java class file

The program can be executed by following the two steps given below:

```
1.javac EMGMMVar.java
2.java EMGMMVar.java em_data.txt
```

It has been observed that randomly selecting the 3 initial data points and variances the three clusters converge to means
13.741564313399916,0.870066944857979,0.870066944857979

And by initializing random probabilities to cluster points clusters converge to means are 13.741564313399916,0.870066944857979,0.870066944857979

It has been observed that if we initialize to random probabilities than clusters the cluster means converge quickly in less number of iterations.

3. Boosting and Bagging:

Contact-lenses Dataset

Base Learner	**Vanilla**	**Bagging**	**Boosting**
Decision stump	70.8333 %	66.6667 %	70.833 %
Naïve Bayes	70.8333 %	75 %	79.1667 %
J48	83.3333%	83.3333 %	75 %

Supermarket Dataset

Base Learner	**Vanilla**	**Bagging**	**Boosting**
Decision stump	64.4046%	70.7586%	77.761 %
Naïve Bayes	63.713%	63.713%	63.713%
J48	63.713 %	63.713 %	63.713%

segment-challenge Dataset

Base Learner	**Vanilla**	**Bagging**	**Boosting**
Decision stump	30.4%	45.1333%	30.4%
Naïve Bayes	81.0667%	81.2667%	81.0667%
J48	95.7333%	96.2%	97.8%

1. Algorithms-Data set combination improved by bagging:
   Naïve Bayes - Contact-lenses
   Decision stump – Supermarket
   Decision stump - segment-challenge
   J48 - segment-challenge

2. Algorithms-Data set combination improved by boosting:
   Naïve Bayes - Contact-lenses
   Decision stump – Supermarket
   J48 - segment-challenge

3. Bagging reduces variance on data sets that are mentioned in 1.

   On unstable data sets boosting reduced the accuracy one such data set is J48 classifier for Contact-lenses Dataset

   It has been observed that few learning algorithms are biased for few domains.J48, Naïve Bayes are biased for segment-challenge Dataset. J48 is biased for Contact-lenses Dataset