Homework 3
CS 6375.502: Machine Learning
Spring 2018

Collaborative Filtering, Boosting and Bagging, and K-means clustering

Azadeh Samadian

1. Collaborative Filtering on Netflix Ratings

For this problem, I used sklearn and numpy library in the python to find similarity and mean of vectors.
Then I just implemented the memory based collaborative filtering from its paper. I used cosine as similarity measurements.

Collaborative filtering:

Memory-based RMSE (cosine): 0.8845
Memory-based MAE (cosine): 0.6949

2. Boosting and Bagging

Iteration #300

Dataset 1: AutoUniv

Base Learner	Vanilla	Bagging	Boosting
Decision stump	25.90%	25.90%	26.80%
J48	21.10%	19.00%	26.80%
Random Forest	21.70%	27.30%	21.90%

Dataset 2: Breast Cancer

Base Learner	Vanilla	Bagging	Boosting
Decision stump	31.46%	30.41%	27.27%
J48	24.47%	23.42%	31.46%
Random Forest	30.41%	28.32%	32.86%

Dataset 3: hepatitis

Base Learner	Vanilla	Bagging	Boosting
Decision stump	22.58%	18.06%	18.70%
J48	16.13%	15.48%	14.83%

Random Forest	14.83%	14.83%	15.48%

Iteration #100

Dataset 1: AutoUniv

Base Learner	Vanilla	Bagging	Boosting
Decision stump	25.90%	25.90%	26.70%
J48	21.10%	19.00%	24.50%
Random Forest	21.70%	21.50%	22.10%

Dataset 2: Breast Cancer

Base Learner	Vanilla	Bagging	Boosting
Decision stump	31.46%	24.82%	28.32%
J48	24.47%	24.12%	31.81%
Random Forest	30.41%	29.01%	33.21%

Dataset 3: hepatitis

Base Learner	Vanilla	Bagging	Boosting
Decision stump	22.58%	18.06%	18.70%
J48	16.13%	16.13%	14.19%
Random Forest	14.83%	14.83%	16.12%

Iteration #150

Dataset 1: AutoUniv

Base Learner	Vanilla	Bagging	Boosting
Decision stump	25.90%	25.90%	26.70%
J48	21.10%	19.03%	23.70%
Random Forest	21.70%	21.50%	22.00%

Dataset 2: Breast Cancer

Base Learner	Vanilla	Bagging	Boosting
Decision stump	31.46%	24.82%	28.32%
J48	24.47%	24.47%	31.81%
Random Forest	30.41%	28.67%	33.56%

Dataset 3: hepatitis

Base Learner	Vanilla	Bagging	Boosting
Decision stump	22.58%	18.06%	20.00%
J48	16.13%	16.13%	13.54%
Random Forest	14.83%	14.83%	16.12%

1. Which algorithms+data set combination is improved by Bagging?

Decision stump + Breast Cancer
Decision stump + hepatitis
J48 + hepatitis

2. Which algorithms+data set combination is improved by Boosting?

J48 + AutoUniv
Decision stump + Breast Cancer
J48 + Breast Cancer
Random Forest + Breast Cancer
Decision stump + hepatitis
Random Forest +AutoUniv (#iteration= 150)

3. Can you explain these results in terms of the bias and variance of the learning algorithms applied to these domains? Are some of the learning algorithms unbiased for some of the domains? Which ones?

Bias means measure the accuracy of the method while variance means measure the precision or specificity of the match. Bagging reduces the variance by averaging and has a little effect on bias. but, we can average and reduce bias using Boosting. We can categorize classification algorithms in unstable and stable methods. A learner is unstable if a small change to the training set D, causes a large change in the output hypothesis. Bagging doesn't work that well in stable algorithms and helps unstable procedures, while Boosting may work well.
J48 decision tree and neural networks are unstable. K-NN, Random Forest and naïve bayse are stable classifiers. Experiments on three datasets prove this. For example, looking at the J48, as it is an unstable method, bagging might help. As you can see, the value before bagging for Breast Cancer dataset in J48 is 31.46% and after bagging is 23.42%. This is almost occurred for other datasets using J48. But Boosting doesn't help for J48, except for hepatitis dataset. This is also true for Random Forest. However, bagging helped now as much as it helped in J48. For decision stump, the results are vary based on dataset like Breast Cancer and also number of iterations.

3. K-means clustering on images

- Display the images after data compression using K-means clustering for different values of K (2, 5, 10, 15, 20).

		
K= 2		

K=5		
K=10		

K=15	
K=20	

		
K= 2		

K=5		
K=10		

K=15		
K=20		

- What are the compression ratios for different values of K? Note that you have to repeat the experiment multiple times with different initializations and report the average as well as variance in the compression ratio.

The compression ratio is the size of the compressed file compared to that of the uncompressed file. The following table shows the compression ratio in percentage.

Koala:

# Cluster	File size (kb)	Compression ratio
2	45.7	0.0585
5	111.3	0.1425
10	185.6	0.2377
15	232.2	0.2973
20	260.8	0.334
original	780.8	1

Penguins:

# Cluster	File size (kb)	Compression ratio
2	26.6	0.0341
5	52.7	0.0677
10	111.2	0.1429
15	144.2	0.1853
20	170.4	0.219
original	777.8	1

`

Multiple experiments:

Koala:

cluster = 2

File size (kb)	Compression ratio
45.7	0.0585
45.7	0.0585
45.7	0.0585

Mean = 0.0585
Variance = 0

cluster = 5

File size (kb)	Compression ratio
111.3	0.1425
111.2	0.1424
111.5	0.1428

Mean = 0.142566667
Variance = 2.8888888892187E-8

cluster = 10

File size (kb)	Compression ratio
185.6	0.2377
185.2	0.2371
186	0.2382

Mean = 0.2377
Variance = 2.022222222268E-7

cluster = 15

File size (kb)	Compression ratio
232.2	0.2973
230.1	0.2946
235.6	0.3017

Mean = 0.2969
Variance = 8.666666667656E-8

cluster = 20

File size (kb)	Compression ratio
260.8	0.334
266.5	0.3413
263.3	0.3372

Mean = 0.3319
Variance = 2.8595555555548E-5

Penguins:
cluster = 2

File size (kb)	Compression ratio
26.6	0.0341
26.6	0.0341
26.6	0.0341

Mean = 0.0341
Variance = 0

cluster = 5

File size (kb)	Compression ratio
52.7	0.0677

52.9	0.068
52.7	0.0677

Mean = 0.0678
Variance = 1.9999999999881E-8

cluster = 10

File size (kb)	Compression ratio
111.2	0.1429
110.8	0.1424
112	0.1439

Mean = 0.1430
Variance= 3.8888888888542E-7

cluster = 15

File size (kb)	Compression ratio
144.2	0.1853
144.2	0.1853
144.2	0.1853

Mean = 0.1853
Variance= 0

cluster = 20

File size (kb)	Compression ratio
170.4	0.219
172.1	0.2212
170.9	0.2197

Mean = 0.2199
Variance= 8.4222222223225E-7

- Is there a tradeoff between image quality and degree of compression? What would be a good value of K for each of the two images?

For compression and segmentation, based on the result and also the following reference, K-means gives a better result for smaller values of k. For large number of clusters, many clusters appear in the image at various places. Since Euclidean distance is not a very good metric for segmentation. One of the application of k_means is image segmentation and compression. The smaller value of k gives better segmented images.

For quality, it is clear that the more compression gives the lower quality. Looking at the size of reduced images, I believe as the number of clusters get larger the quality plays a more important factor in comparison with compression. So, the quality of k-mean with k = 20 is better the other ones for both images since the quality is much closer to the original image.