Bayesian Network and Unsupervised Learning

(CSE 5522 Homework #2, Spring 2017)

Cheng Zhang

Dept. Computer Science and Engineering, The Ohio State University
zhang.7804@osu.edu

1 Problem 1

1.1 Step 1

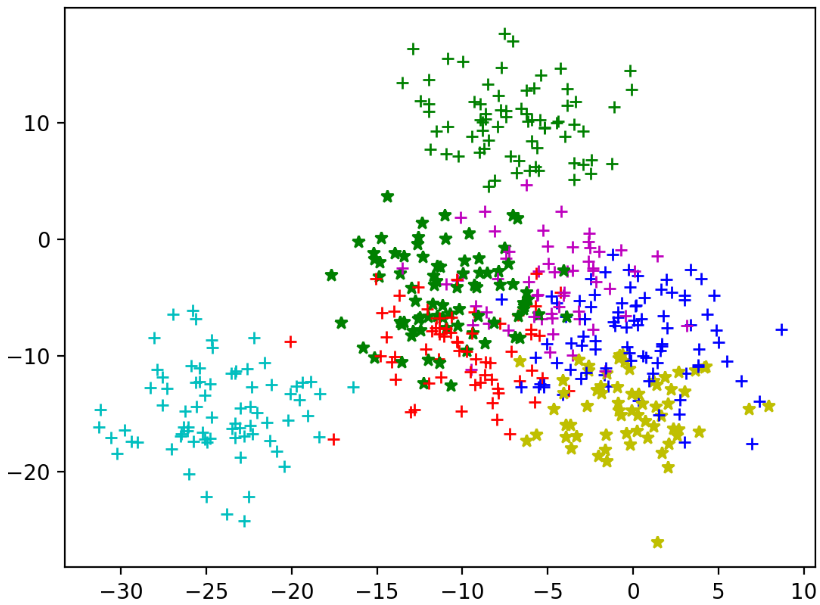

Figure 1: The distributions of data points in training set. Different colors and shapes indicate different current labels.

1.2 Step 2

In this step, we calculate conditional probabilities of $P(V|C)$ and prior $P(C)$ after the clustering in Step.1. By Multinomial Naive Bayes [1, 2], the prior $P(C)$ for each lable C_i is as follows:

$$P(C_i) = \frac{N_i + \alpha}{N + |C| * \alpha}$$

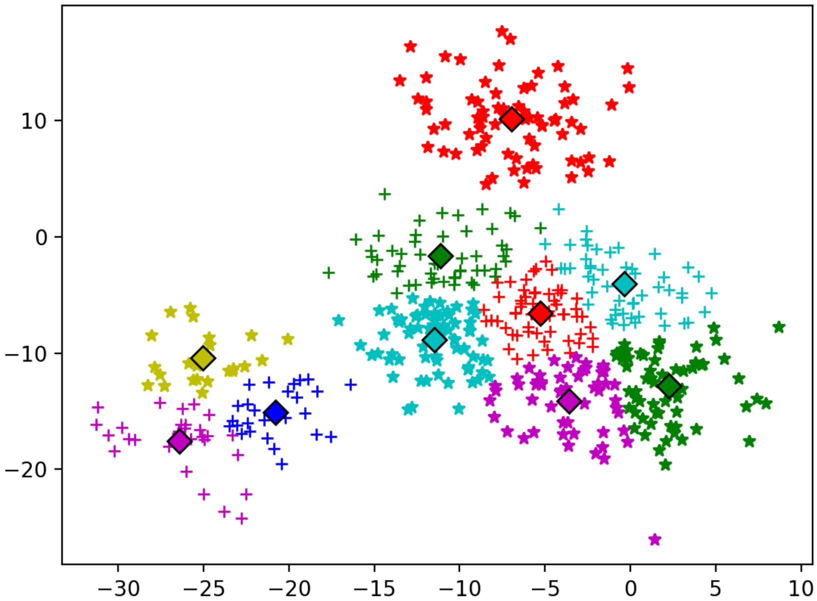

Figure 2: Clustering result of k-means clustering with k = 10 means. The rhombuses with black marker edge are the centroids of each cluster.

where N is total number of the sample points, N_i is the number of points belongs to label C_i, $|C|$ is the number of categories, and α is the smoothing prior. Here we set α to 0 in our further experiments. The conditional probability $P(V_j|C_i)$ is as follows:

$$P(V_j|C_i) = \frac{N_j + \alpha}{N_i + dim * \alpha}$$

where N_i is the number of points belongs to label C_i, N_j is the number of points belongs to V_j within C_i, dim is the dimensional of data points, and α is the smoothing prior. Table. 1 and Table. 2 show the prior table and conditional probabilities table, respectively.

$P(C_0)$	$P(C_1)$	$P(C_2)$	$P(C_3)$	$P(C_4)$	$P(C_5)$	$P(C_6)$
0.17	0.128	0.142	0.116	0.152	0.13	0.162

Table 1: The prior $P(C)$ for each label C_i

1.3 Step 3

In this step, we determine the class of each point by using $P(C|V) = \alpha P(V|C)P(C)$ in the testing set. In the implementation, the whole framework includes 3 stage:

1. Convert continuous features into discrete classes for each samples in the testing set. In this stage, we compare the distance between each sample and the centroid of cluster V_j. Then assign the

	$P(V_j\|C_0)$	$P(V_j\|C_1)$	$P(V_j\|C_2)$	$P(V_j\|C_3)$	$P(V_j\|C_4)$	$P(V_j\|C_5)$	$P(V_j\|C_6)$
V_0	0.0	0.0156	0.0	0.0	0.3289	0.0	0.0
V_1	0.1529	0.1406	0.0	0.3966	0.0	0.0154	0.1605
V_2	0.0	0.0625	0.0	0.1897	0.0	0.0	0.4568
V_3	0.0	0.0	0.0	0.0	0.3816	0.0	0.0
V_4	0.3765	0.0	0.0	0.3276	0.0	0.0	0.0123
V_5	0.0	0.0156	0.0	0.0	0.2895	0.0	0.0
V_6	0.3176	0.0	0.0	0.0	0.0	0.5385	0.0
V_7	0.1529	0.1563	0.0	0.0172	0.0	0.4462	0.0
V_8	0.0	0.6094	0.0	0.0517	0.0	0.0	0.3704
V_9	0.0	0.0	1.0	0.0172	0.0	0.0	0.0

Table 2: The conditional probability $P(V_j|C_i)$

samples to the closeted centroid.

2. Predict labels for each samples using $P(C|V) = \alpha P(V|C)P(C)$. Note, the prior $P(C)$and conditional probabilities $P(V|C)$ have been calculated in Step. 2

Fig. 3 shows the groundtruth of testing set. As shown in Fig. 4, one time predict result achieves error rate 21.6 %. Note, here the number of clusters $k = 10$, smoothing item $\alpha = 0$, and the training set and testing set are the original source data.

Next, we run the program 20 times get the average and standard deviation of the classification error rate. Fig. 5 shows classification error rate of 20 different runs of the k-means algorithm. The average is 26.58 % and the standard deviation is 3.17.

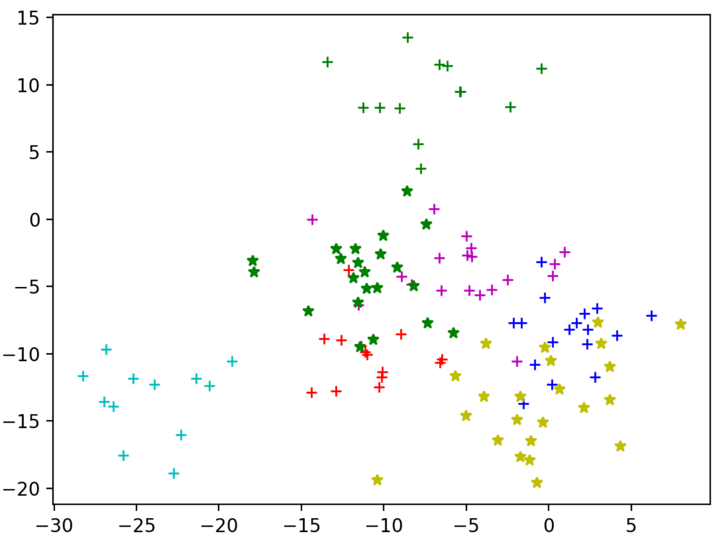

Figure 3: The distributions of data points in test set. Different colors and shapes indicate groundtruth.

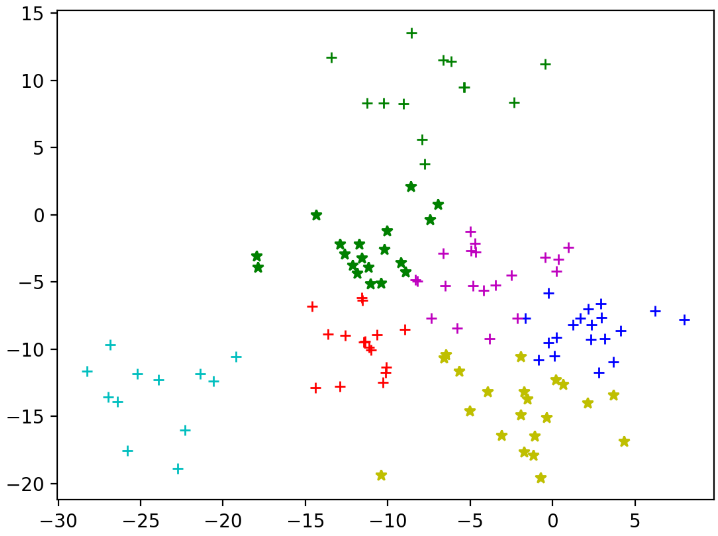

Figure 4: One time predict result of testing set. Error rate is 21.6 %

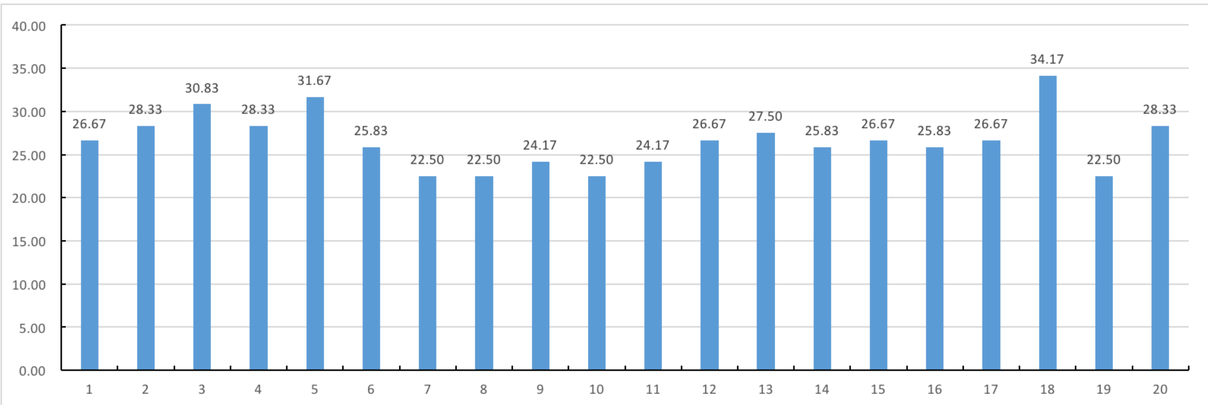

Figure 5: Classification error rate of 20 different runs of the k-means algorithm. The average is 26.58 % and the standard deviation is 3.17.

1.4 Step 4

In this section, we discuss classification results within different clustering number k. Table. 3 shows classification error rate (%) of 20 different runs of the k-means algorithm with different k values. Fig. 6 shows classification error rate of 20 different runs of the k-means algorithm with different k values. Fig. 7 and Fig. 8 show the average and standard deviation of error rate with 20 runs of k-means algorithm according to different k value, respectively.

According to the results, we have following conclusion:

1. For these specific training and testing data, the best cluster k is around 10.

2. The convergence time increases with the increasing of k. (Running time is not shown here, but it is indeed so)

3. Given an appropriate cluster number k, such as the number of groundtruth categories (here the original number of categories is 7), the results tend to be better. While if k is larger than the number of clusters in the real cluster, the classification accuracy will degrade sharply.

runs	$k = 2$	$k = 5$	$k = 6$	$k = 8$	$k = 12$	$k = 15$	$k = 20$	$k = 50$
1	75.83	47.50	30.83	30.00	25.83	25.83	29.17	27.50
2	75.83	39.17	30.83	29.17	30.83	29.17	25.00	27.50
3	75.83	37.50	32.50	25.00	28.33	25.00	31.67	26.67
4	75.83	37.50	30.83	34.17	27.50	25.00	25.00	29.17
5	75.83	39.17	39.17	27.50	22.50	30.00	27.50	22.50
6	75.83	39.17	39.17	30.83	26.67	25.00	29.17	30.83
7	75.83	37.50	30.83	37.50	26.67	33.33	25.83	27.50
8	75.83	39.17	30.83	35.83	20.83	27.50	30.00	27.50
9	75.00	37.50	28.33	35.83	27.50	28.33	30.83	26.67
10	75.83	47.50	39.17	25.00	26.67	26.67	25.00	22.50
11	75.00	39.17	39.17	40.00	30.00	24.17	25.00	25.83
12	75.00	39.17	30.83	29.17	21.67	26.67	26.67	32.50
13	79.17	39.17	30.83	35.83	25.83	30.83	28.33	25.00
14	75.83	39.17	30.83	30.00	26.67	25.00	27.50	30.83
15	79.17	37.50	30.83	33.33	28.33	30.00	25.00	26.67
16	75.83	39.17	28.33	24.17	28.33	29.17	29.17	29.17
17	75.83	39.17	39.17	22.50	25.83	27.50	26.67	28.33
18	75.83	49.17	28.33	30.83	31.67	21.67	30.00	24.17
19	75.83	39.17	39.17	25.00	25.83	29.17	26.67	25.00
20	75.83	39.17	30.83	31.67	27.50	26.67	28.33	26.67

Table 3: Classification error rate (%) of 20 different runs of the k-means algorithm with different k values.

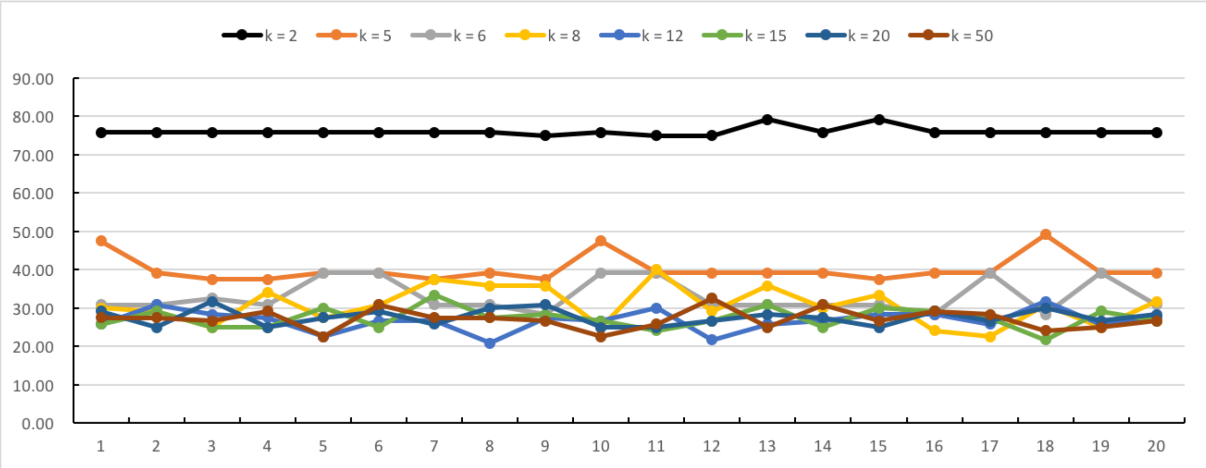

Figure 6: Classification error rate of 20 different runs of the k-means algorithm with different k values.

Figure 7: Average error rate of 20 different runs of the k-means algorithm with different k values.

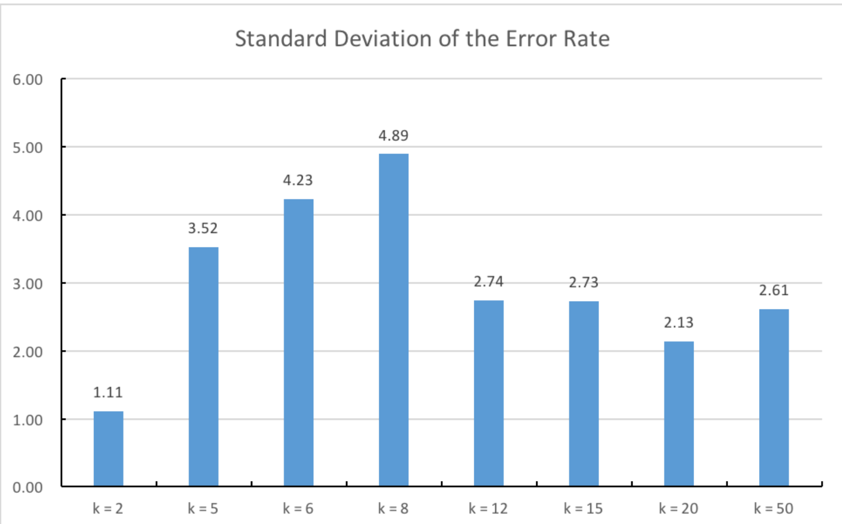

Figure 8: Standard deviation of error rate of 20 different runs of the k-means algorithm with different k values.

1.5 Bonus 1

In this section, we modify the data generator script to generate higher dimensional data (3d, 4d, 5d). Keep the number of classes ($nmeans = 7$) and the Gaussian widths ($width = 3.2$) the same. Modify the program to repeat the experiment for k = 10. Fig. 9, Fig. 10 and Fig. 11 show experimental results of 20 different runs of the k-means algorithm with higher dimensional data in aspect of error rate, average and standard deviation.

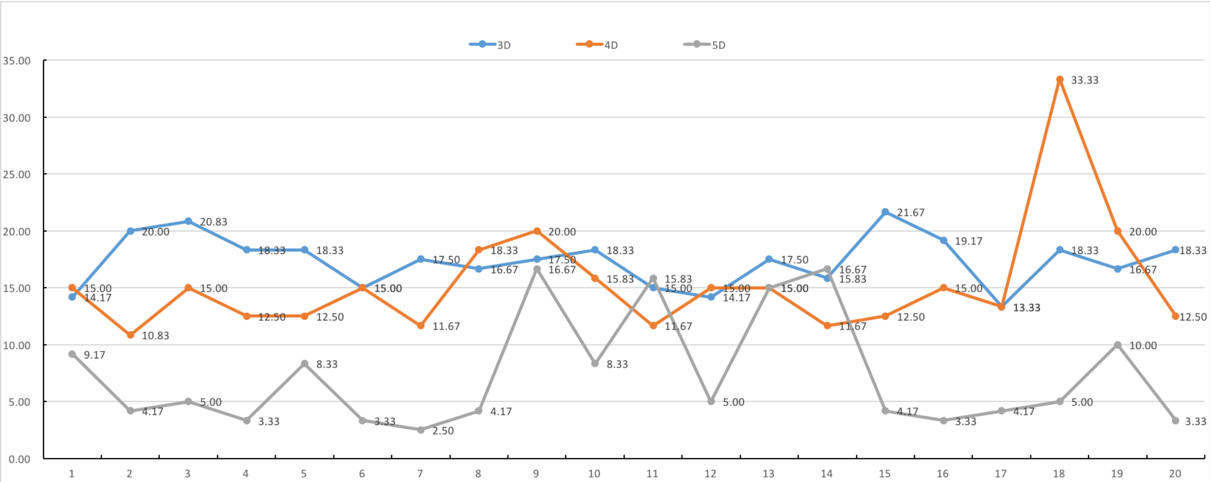

Figure 9: Classification error rate of 20 different runs of the k-means algorithm with higher dimensional data.

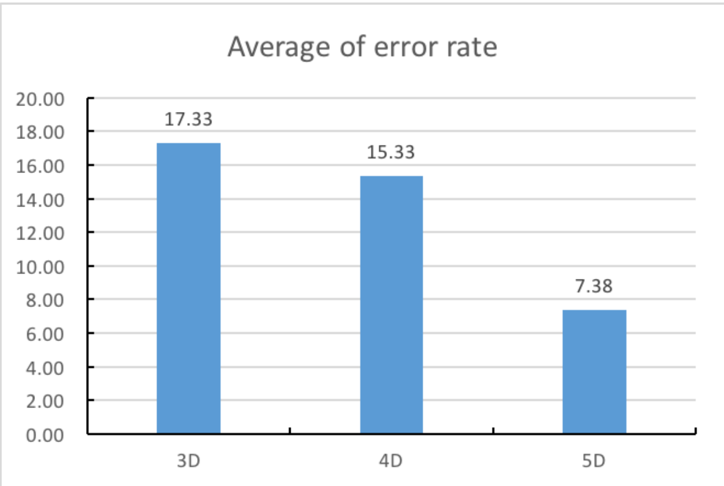

Figure 10: Average of error rate of 20 different runs of the k-means algorithm with higher dimensional data.

1.6 Bonus 2

Next, we modify the data generator script to increase the size of the Gaussian widths of the data generated for 2d data (so that they overlap more). Fig. 12, Fig. 13 and Fig. 14 show experimental results

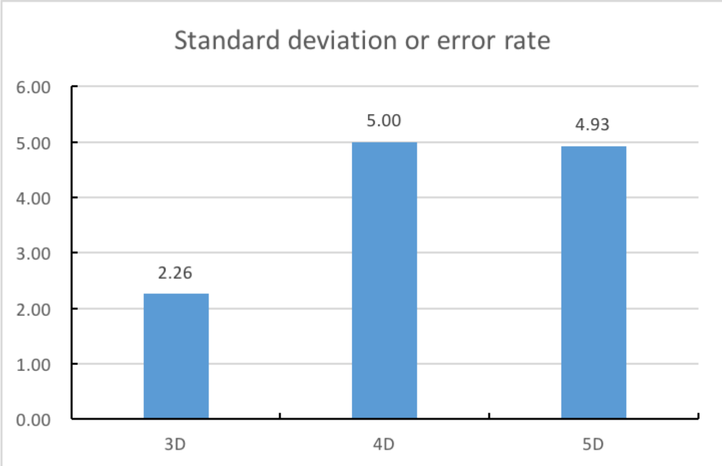

Figure 11: Standard deviation of error rate of 20 different runs of the k-means algorithm with higher dimensional data.

of 20 different runs of the k-means algorithm with higher dimensional data in aspect of error rate, average and standard deviation.

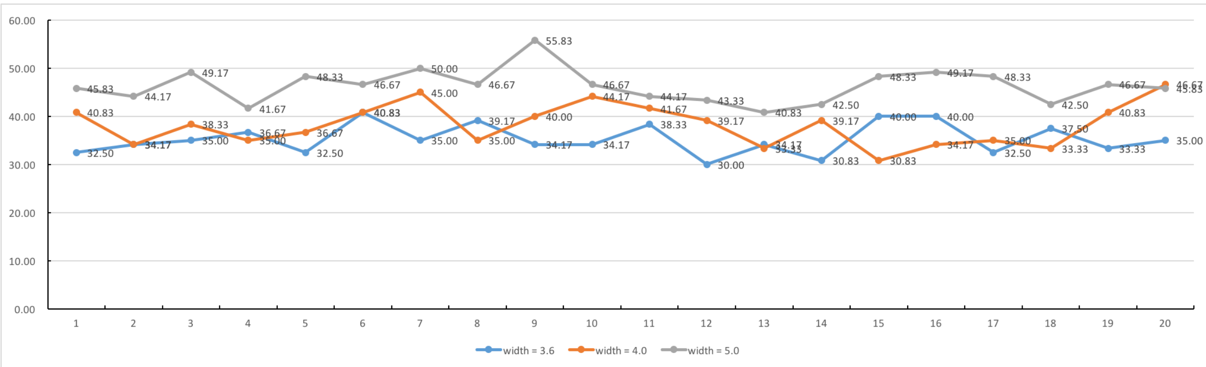

Figure 12: Classification error rate of 20 different runs of the k-means algorithm as we increase Gaussian widths of the data for 2D data.

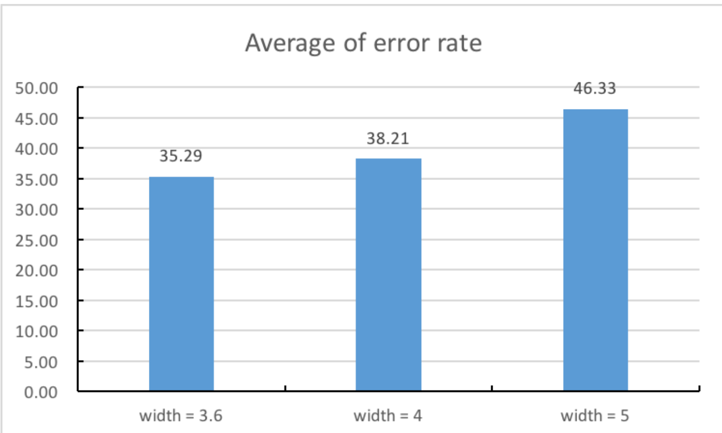

Figure 13: Average of error rate of 20 different runs of the k-means algorithm as we increase Gaussian widths of the data for 2D data.

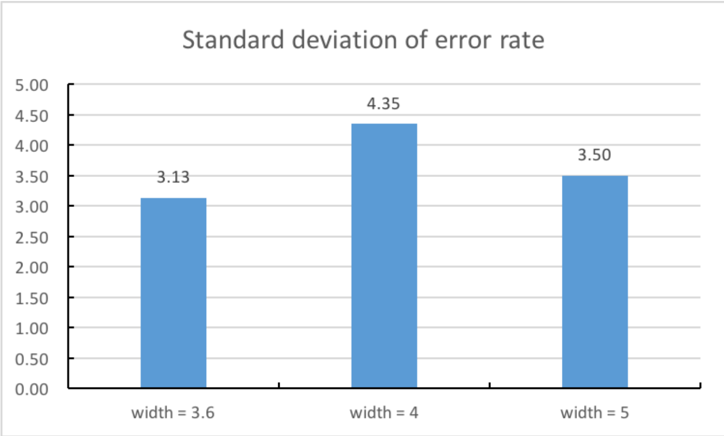

Figure 14: Standard deviation of error rate of 20 different runs of the k-means algorithm as we increase Gaussian widths of the data for 2D data.

1.7 Bonus 3

2 Problem 2

2.1 Bayesian network

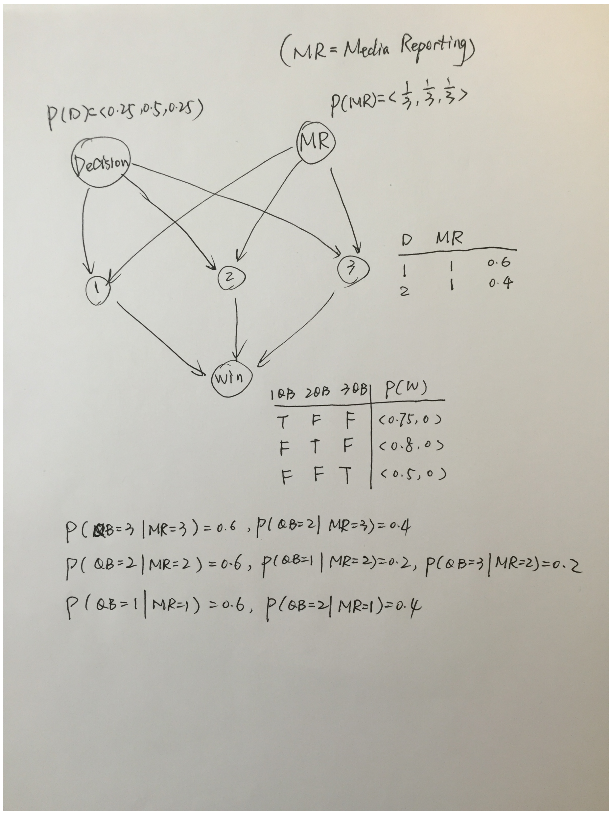

Figure 15: Bayesian network.

2.2 Calculate the probability

References

[1] Scikit-learn. Naive bayes. http://scikit-learn.org/stable/modules/naive_bayes.html#naive-bayes.

[2] Wikipedia. Naive bayes classifier. https://en.wikipedia.org/wiki/Naive_Bayes_classifier.