E1-313:TIPR-Assignment1 Report **Kishalay Das**
TA: Anirban Biswas Student Id-15938
Instructor: M NARASIMHA MURTY Submission Date: 11/02/2019

Python version 3 is used to solve all problems.

1 Task-I: Implement Random Projection Algorithm

Random projection is a technique used to reduce the dimensionality of a set of points which lie in Euclidean space.the original D-dimensional data is projected to a K-dimensional (K <<D) subspace, using a random $K \times D$ - dimensional matrix R whose columns have unit lengths.
let we have Data Matrix :

$$A_{n\times l} = \begin{bmatrix} u_1 \\ \\ . \\ . \\ . \\ u_2 \\ \end{bmatrix}_{n\times l}$$

And we choose an Random matrix from Normal Distribution of 0 mean and unit variance:

$$R_{l\times k} = \begin{bmatrix} r_1 & .....r_k \end{bmatrix}_{l\times k}$$

Then Random Projection of A is :

$$\text{E}_{n\times k} = 1\frac{}{\sqrt{K}}.\text{A.R}$$

For each data set we have experiment with K=2,4,8...D/2 and save them for in .csv format.

2 Task-II/III: Classification

This section I will report the accuracy and F1-score of different Data-sets on K-NN and Bayes Classifier(On custom designed code).

2.1 Nearest Neighbour Classifier

I have implemented K-Nearest Neighbour Classifier with K=5 and here is the results obtained for Dolphin and Pubmed Data Set for Original as well as Random Projected Dimensions:.

Dolphin Dataset

D-Value	32(Original)	16	8	4	2
Accuracy	92.30	92.30	92.30	92.30	76.92
F1-Score(Macro)	0.727	0.727	0.727	0.727	0.581
F1-Score(Micro)	0.9230	0.9230	0.9230	0.9230	0.7692

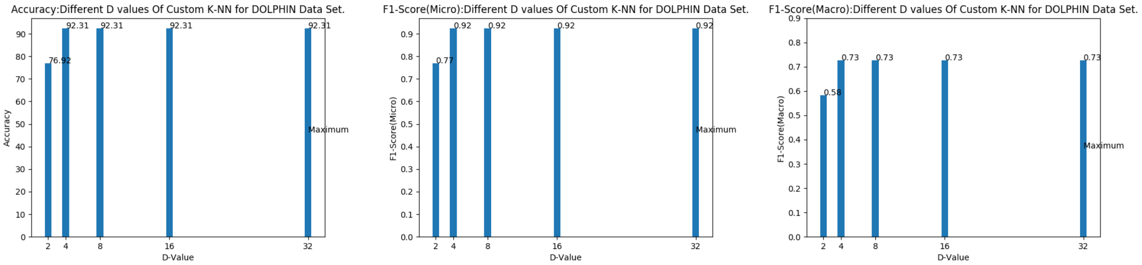

Fig1:-Accuracy,F1-Score(Micro/Macro)in KNN-Classifier on Dolphin Dataset.

Best Accuracy is found in Dimension 32(Original) as well as reduced dimension 16,8 and 4.Plots are as follows

Pubmed Dataset

D-Value	128(Original)	64	32	16	8	4	2
Accuracy	37.09	36.80	36.80	38.61	38.83	36.22	36.77
F1-Score(Macro)	0.3362	0.3338	0.3356	0.3513	0.3499	0.3271	0.3343
F1-Score(Micro)	0.3709	0.3680	0.3680	0.3861	0.3883	0.3622	0.3677

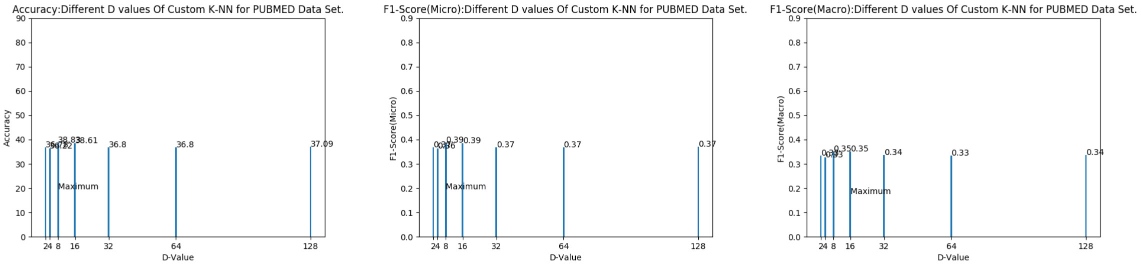

Fig2:-Accuracy,F1-Score(Micro/Macro)in KNN-Classifier on Pubmed Dataset.

Best Accuracy is found in reduced dimension 8 and 16, which is slighly higher than Original Dimensions.Even most of the other reduced dimensions give close results compared to original dimension.

2.2 Bayes Classifier

Dolphin Dataset

D-Value	32(Original)	16	8	4	2
Accuracy	92.30	92.30	92.30	92.30	76.92
F1-Score(Macro)	0.7272	0.7272	0.7272	0.7272	0.5814
F1-Score(Micro)	0.9230	0.9230	0.9230	0.9230	0.7692

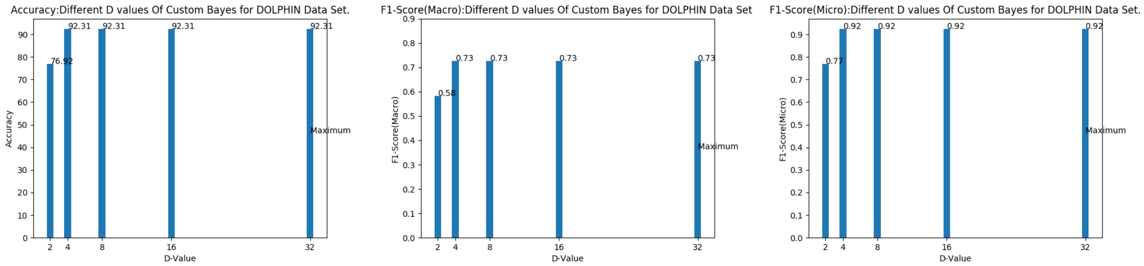

Fig3:-Accuracy,F1-Score(Micro/Macro)in Bayes-Classifier on Dolphin Dataset.

Best Accuracy is found in Dimension 32(Original) as well as reduced dimension 16, 8 and 4.

Pubmed Dataset

D-Value	128(Original)	64	32	16	8	4	2
Accuracy	42.47	43.10	43.44	42.23	42.18	41.02	41.94
F1-Score(Macro)	0.4009	0.3855	0.3637	0.3169	0.3136	0.2967	0.3092
F1-Score(Micro)	0.4247	0.4310	0.4344	0.4223	0.4218	0.4102	0.4194

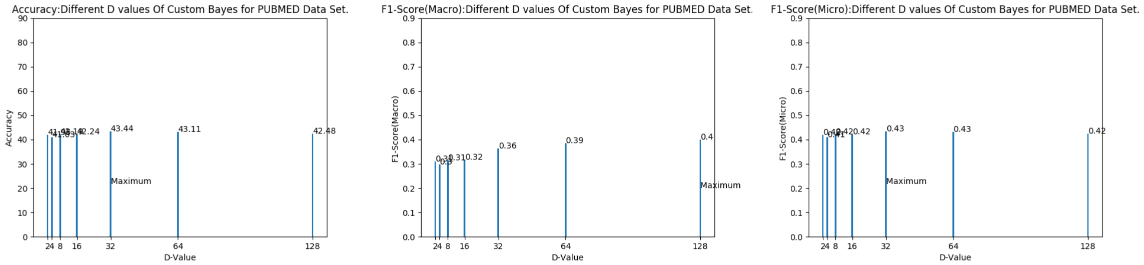

Fig4:-Accuracy,F1-Score(Micro/Macro)in Bayes-Classifier on Pubmed Dataset.

Best Accuracy is found in reduced dimension 32 and 64(Again which is very close to Original Dimension).Even most of the other reduced dimensions gives results close to original dimension.

2.3 Twitter Data-Set

This data set was text file with sentences in it.So first I represented it as Bag-of-words Representations.There are two ways of this representations : **Multivariate Bernoulli model** and **Multinomial model**.I followed the later one.Multinomial model of representation is defined as :

Pr(X_i=w |C$_j$)=**fraction of times in which word w appears among all words in documents of topic C_j**

Following the results found in Naive Bayes applied on Bag-of-words of Twitter Dataset:

D-Value	Original	512	256	128	64	32	16	8	4	2
Accuracy	57.41	12.16	12.16	33.16	54.66	12.16	12.16	54.66	54.66	12.16
F1-Score(Macro)	0.50	0.072	0.072	0.16	0.23	0.072	0.072	0.23	0.23	0.072
F1-Score(Micro)	0.5741	0.121	0.121	0.331	0.546	0.121	0.121	0.546	0.546	0.1216

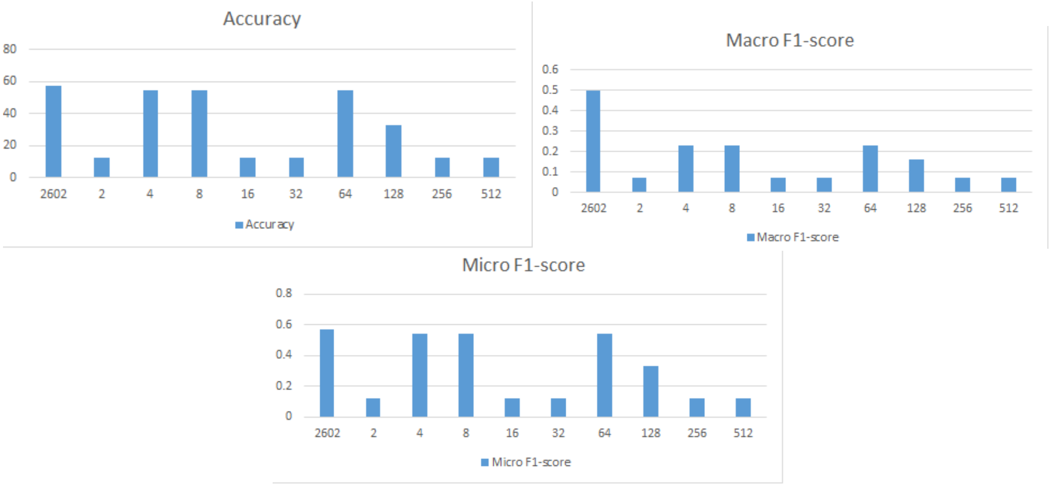

Fig5:-Accuracy,F1-Score(Micro/Macro)in Bayes-Classifier on Twitter Dataset.

Best Accuracy is found in Original dimension.But with reduced dimension 4,8,64 we got very close to it.

Following the results found in K-NN(k=5) applied on Bag-of-words of Twitter Dataset:

D-Value	Original	512	256	128	64	32	16	8	4	2
Accuracy	53.16	42.91	46.08	44.25	47.33	46	45.25	42.66	44.33	45.91
F1-Score(Macro)	0.43	0.31	0.32	0.30	0.33	0.34	0.33	0.31	0.31	0.32
F1-Score(Micro)	0.5316	0.4291	0.4608	0.4425	0.4733	0.46	0.4525	0.4266	0.4433	0.4591

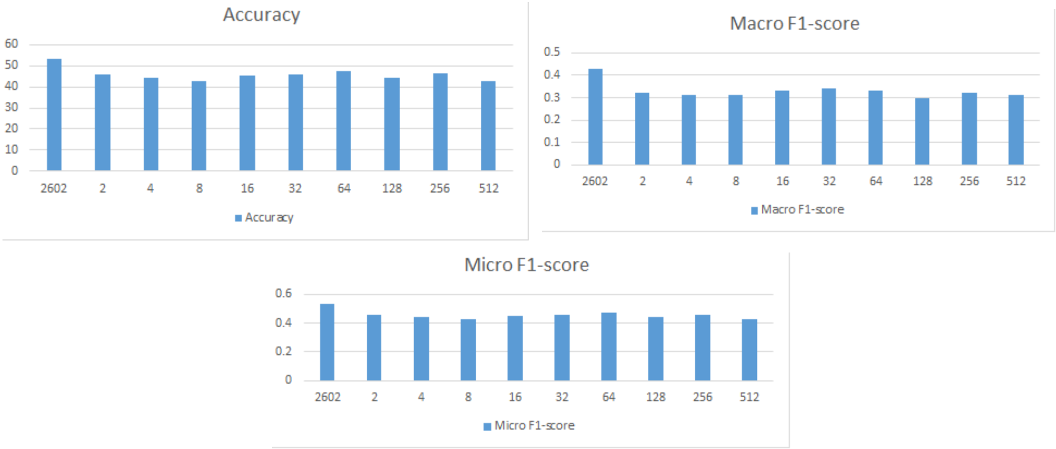

Fig6:-Accuracy,F1-Score(Micro/Macro)in Knn-Classifier on Twitter Dataset.

Again Best Accuracy is found in Original dimension.So it turns out that random projection can't affect much in this data set.

So we can conclude that even with low dimension we get higher accuracy in some cases(like PUBMED),still its not a significant gain.It is very close to what we get in higher dimension.On the other side data sets like Dolphin/Twitter we lost information due to dimension reduction and hence we didnot get accuracy better that Original Dimension.So I believe High Dimension is more useful in classification.I will go with Amar.

3 Task-IV/V: Classification(Library Code)

This section I will report the accuracy and F1-score of different Data-sets on K-NN and Bayes Classifier(On scikit-learn library code).and compare results with custom designed code.

3.1 Nearest Neighbour Classifier

I have implemented K-Nearest Neighbour Classifier with K=5 and here is the results obtained for Dolphin and Pubmed Data Set for Original as well as Random Projected Dimensions:.
Dolphin Dataset

D-Value	32(Original)	16	8	4	2
Accuracy	92.30	92.30	92.30	92.30	84.61
F1-Score(Macro)	0.727	0.727	0.727	0.727	0.65
F1-Score(Micro)	0.9230	0.9230	0.9230	0.9230	0.8461

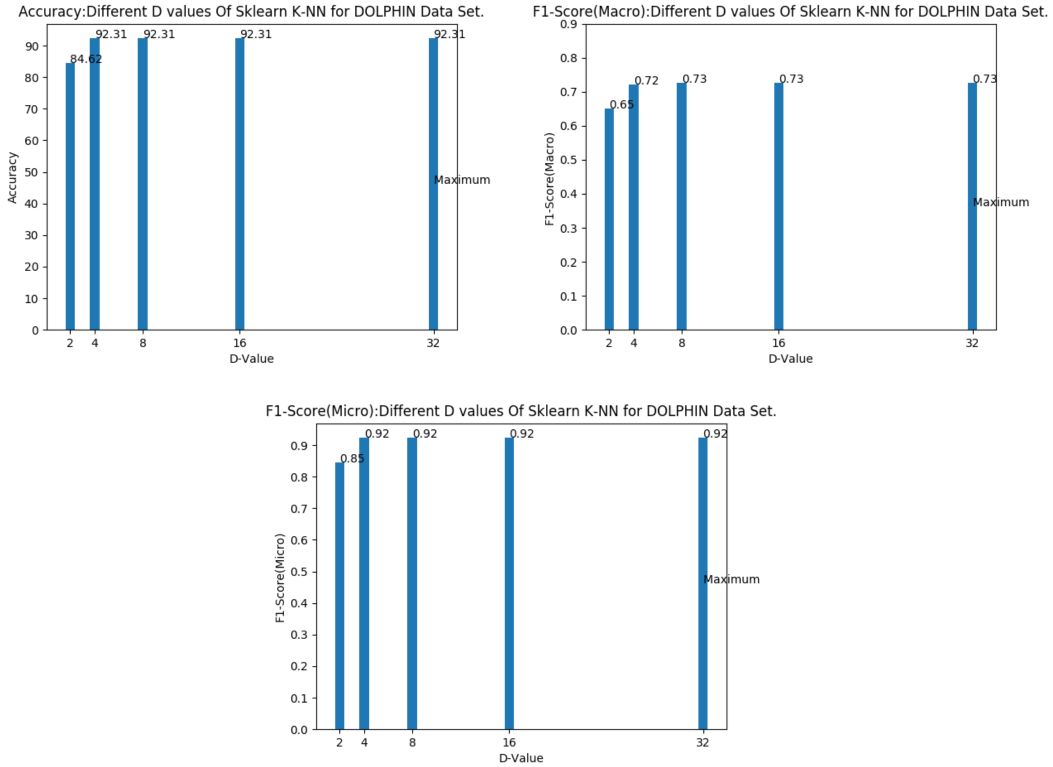

Fig7:-Accuracy,F1-Score(Micro/Macro)in Sklearn Knn-Classifier on Dolphin Dataset.

Best Accuracy is found in Dimension 32(Original) as well as reduced dimension 16,8 and 4.Exactly same I got in my own custom K-NN classifier.But in case of dimensionality 2 Library KNN gives better pwerformance that my own code.

Pubmed Dataset

D-Value	128(Original)	64	32	16	8	4	2
Accuracy	35.426	35.64	36.17	34.75	35.54	34.84	36.15
F1-Score(Macro)	0.334	0.3372	0.3405	0.3285	0.3339	0.3277	0.3421
F1-Score(Micro)	0.3542	0.3564	0.3617	0.3475	0.3554	0.3484	0.3615

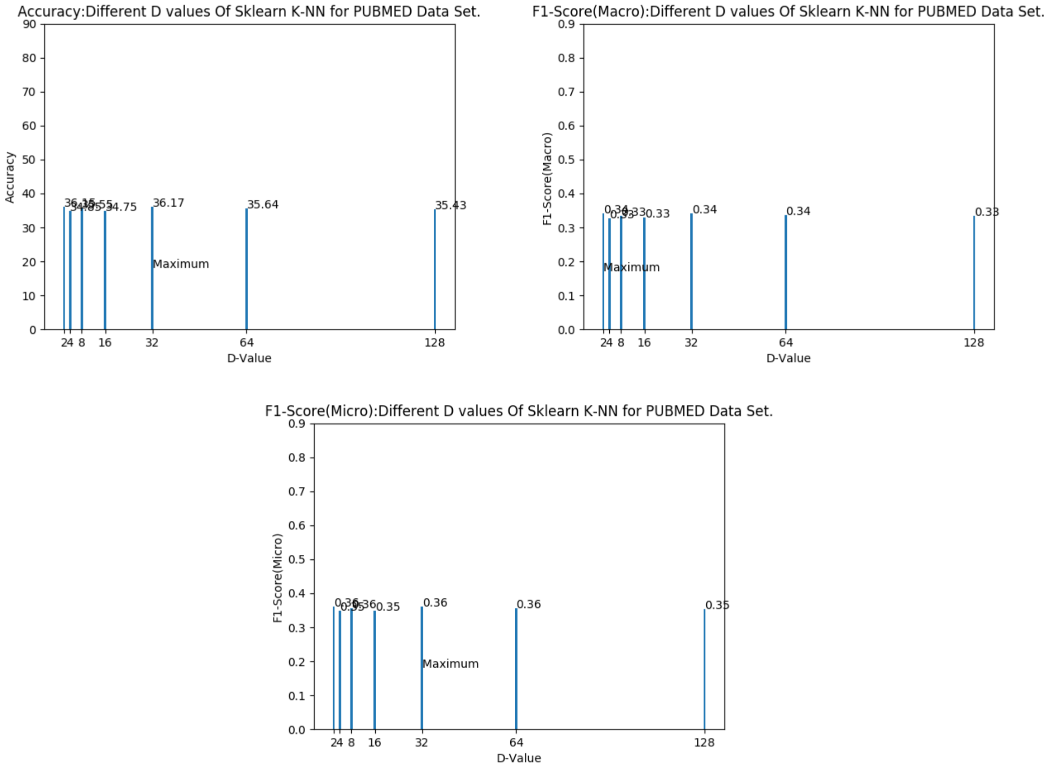

Fig7:-Accuracy,F1-Score(Micro/Macro)in Sklearn Knn-Classifier on Pubmed Dataset.

Best Accuracy is found in dimesion 2 and 32.In my own code result it was 8 and 16.But I believe that due to random projection effects the accuracy.Range of accuracy remains same in both the cases.

3.2 Bayes Classifier

Dolphin Dataset

D-Value	32(Original)	16	8	4	2
Accuracy	92.30	92.30	92.30	92.30	84.61
F1-Score(Macro)	0.727	0.727	0.727	0.727	0.65
F1-Score(Micro)	0.9230	0.9230	0.9230	0.9230	0.8461

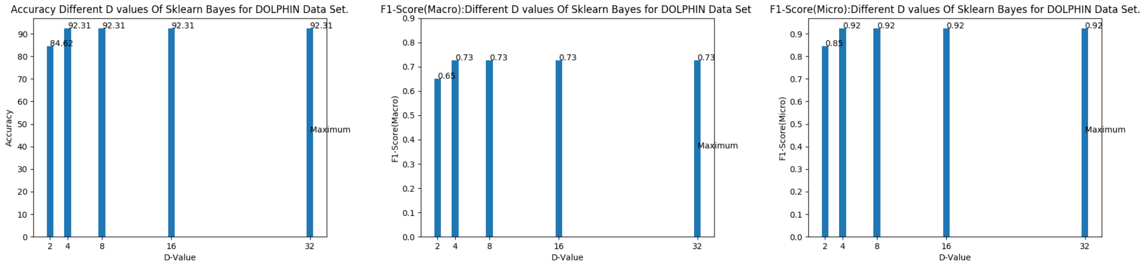

Fig9:-Accuracy,F1-Score(Micro/Macro)in Sklearn Bayes-Classifier on Dolphin Dataset .

Best Accuracy is found in Dimension 32 (Original) as well as reduced dimension 16, 8 and 4.In my own custom code also I got higher accuracy in these dimensions only.

Pubmed Dataset

D-Value	128(Original)	64	32	16	8	4	2
Accuracy	42.47	42.83	43.61	43.37	41.63	40.88	39.94
F1-Score(Macro)	0.4009	0.3805	0.3852	0.3337	0.3087	0.3044	0.2876
F1-Score(Micro)	0.4247	0.4283	0.4361	0.4337	0.4163	0.4088	0.3994

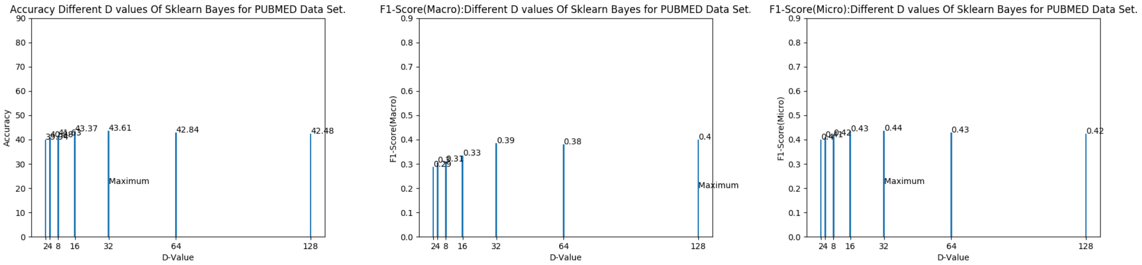

Fig9:-Accuracy,F1-Score(Micro/Macro)in Sklearn Bayes-Classifier on Pubmed Dataset .

Best Accuracy is found in reduced dimension 16.Even most of the other reduced dimensions gives r results close to original dimension.
My code is not completely outperforming the Library code.As we see from the results in Dolphin its almost same as the result we get by library function.And in Pubmed for some dimesions its getting better results but most of the cases library function gives better accuracy.

4 Task-VI: Locality Sensitive Hashing

Locality-sensitive hashing (LSH) reduces the dimensionality of high-dimensional data and is an approximation of KNN. LSH hashes Data-points so that similar points map to the same buckets with high probability (the number of buckets being much smaller than the universe of possible input items).Later we query test-points by hashing it into a bucket and checking the existing points.The Algorithm we used is as follows:

1.Pick a random projection of R^d onto a 1-dimensional line and chop the line into segments of length w, shifted by a random value $b \in [0, w)$.Formally the hash function is:

$$h_{r,b} = \text{Ceil}[(\text{r.x+b})/\text{w} \;]$$

where the projection vector $r \in R^d$ is constructed by picking each coordinate of r from the Gaussian distribution.

2.While Querying a test data point, we first hash the test point using the same hash function mentioned above(with same r and b used above).Once it is hashed to a hash-bucket, we take majority of all train points already hashed there.We will have 20 such hash tables.Finally we will take majority of all those labels returned by each hash-Tables.

Following results obtained using LSH:

D-Value	Dolphin	Pubmed	Twitter
Accuracy	53.84	38.20	50.167
F1-Score(Macro)	0.3303	0.1955	0.2227
F1-Score(Micro)	0.5384	0.3820	0.5016

Observation: As we mentioned LSH is an approximation of KNN,we can see here that Dataset like Pubmed and Twitter the accuracy is very close to what we find with KNN.Though with Dolphin the result is very Low.My observation in this case is all the train points are mostly mapped in two hash slots and hence with majority we are getting only two class labels which has most prior distribution.May be with some other hash function it will work better,

5 Task-VII: Classification on PCA

We have reduced the dimension of each data-set by PCA(K=2,4...D/2) and applied KNN(k=5)[Custom Code without using library] on that to see performance of each classifier.Results aee as follows:

5.1 Nearest Neighbour Classifier on PCA

Dolphin Dataset

D-Value	16	8	4	2
Accuracy	92.30	92.30	92.30	92.30
F1-Score(Macro)	0.727	0.727	0.727	0.7272
F1-Score(Micro)	0.9230	0.9230	0.9230	0.9230

All the reduced dimensions is giving same accuracy like the original dimension as reported before.

Pubmed Dataset

D-Value	64	32	16	8	4	2
Accuracy	37.16	37.84	38.17	38.17	38.87	37.04
F1-Score(Macro)	0.3351	0.3458	0.3462	0.3447	0.3521	0.3350
F1-Score(Micro)	0.3716	0.3784	0.3817	0.3817	0.3887	0.3704

Most of the reduce dimensions data matrix is giving better accuracy than the original dimension as reported before.

Twitter DataSet

D-Value	512	256	128	64	32	16	8	4	2
Accuracy	37.25	33.75	36.33	40.41	42.25	44.16	46.33	44.25	47.25
F1-Score(Macro)	0.3045	0.3104	0.3256	0.3475	0.3546	0.3534	0.3665	0.3380	0.3082
F1-Score(Micro)	0.3725	0.3375	0.3633	0.4041	0.4225	0.4416	0.4633	0.4425	0.4725

As we have seen in random projection also,in case of twitter dataset dimesion reduction makes no effect in terms of accuracy of the classifier.As you see both in LSH and PCA, we are not able to achieve the accuracy of original dimensions reported before.