K-Means, Expectation Max. with PCA, ICA, Decision Trees and ANN

AMIT V GOTTIPATI

Introduction

As part of this assignment unsupervised learning techniques like K means and Expectation Maximization were implemented. Furthermore, dimensionality reduction techniques like PCA, ICA, Decision Trees and Randomized Projections were implemented. Then binary classification using Artificial Neural Networks (ANN) was implemented in Python and Google colab using the SGEMM GPU kernel performance data set from UCI Machine Learning repository and on the Rain in Australia dataset from Kaggle. The project involved data pre-processing, data visualization, algorithm application, prediction and experimenting with different parameters like Neurons, Activation Functions, K, etc.

Dataset Description

Dataset1 - SGEMM GPU :

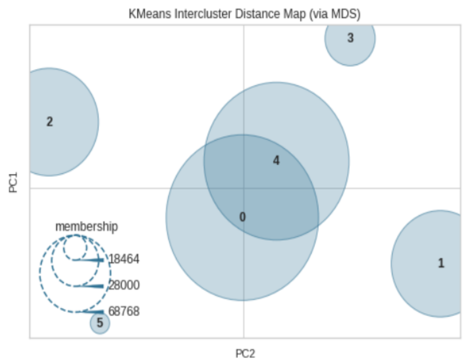
- The First dataset contains 241600 observations and 18 variables with no missing values.
- There are 18 variables, the first 14 are parameters upon which we will train our model. The first 10 parameters are ordinal with 4 levels, the next 4 are binary and last 4 variables are 4 different run times for the SGEMM GPU
- The description for the independent variables is given in the dataset link.
- The dataset was divided into train and test with a 70:30 split.
- For dependent variable 'avg' was created using average of the 4 run times provided which tells us the average running time for matrix-matrix product of the four runs. For classification the dependent variable 'avg' was converted into a binary variable using median as the threshold. (High/Low)

Dataset2:

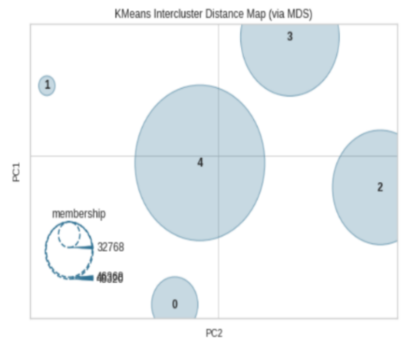
- This Dataset contains 142193 observations with 24 variables
- Here the Target variable is Rain Tomorrow, we need to predict if it will rain tomorrow or not.
- It is given that Risk_MM variable is a very good predictor of the outcome and is dropped.
- Variables with more than 35% missing values were dropped and Last observation carried forward LOCF was used to impute the remaining missing values

K-means: Dataset1

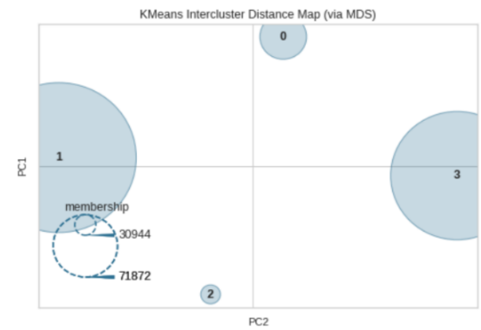
K-means



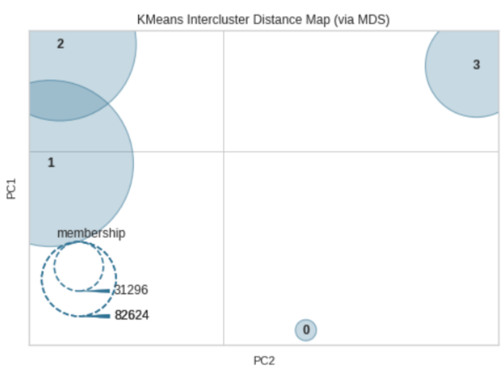
PCA



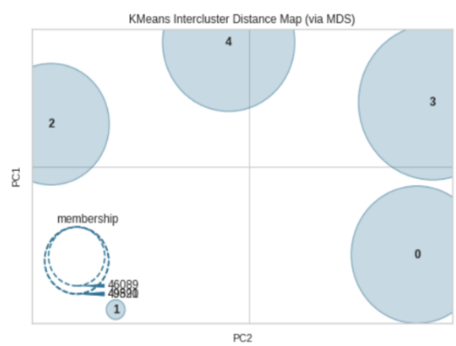
Decision Trees

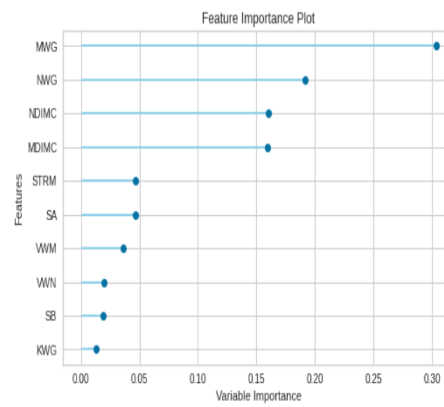


ICA



Randomized Projections





Cluster Output

K-means

avg	0	1
Cluster		
Cluster 0	37902	42098
Cluster 1	19372	8628
Cluster 2	18898	9550
Cluster 3	4000	14464
Cluster 4	34109	34659
Cluster 5	6520	11400

PCA

avg	0	1
Cluster		
Cluster 0	6791	25977
Cluster 1	10610	21518
Cluster 2	27249	19119
Cluster 3	26725	21595
Cluster 4	49426	32590

Decision Trees

avg	0	1
Cluster		
Cluster 0	17864	54008
Cluster 1	19667	10829
Cluster 2	50043	15045
Cluster 3	11863	31337
Cluster 4	21364	9580

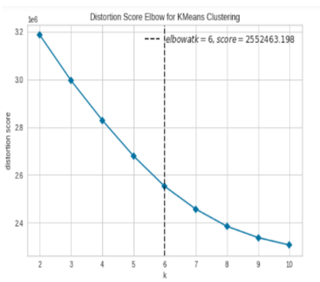
ICA

avg	0	1
Cluster		
Cluster 0	6520	12424
Cluster 1	53573	55163
Cluster 2	40570	42054
Cluster 3	20138	11158

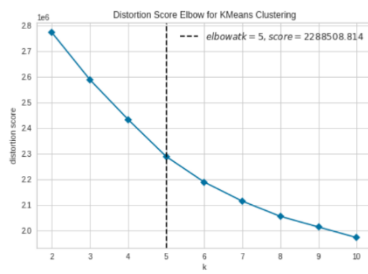
Randomized Projections

avg	0	1
Cluster		
Cluster 0	19927	29663
Cluster 1	25128	15573
Cluster 2	21096	24993
Cluster 3	36063	19336
Cluster 4	18587	31234

Kmeans



Kmeans +PCA



From the following experiment we saw the best K =6 for kmeans and k=5 for kmeans with PCA.The Best K chosen thorugh the elbow plot. The clusters were very close when kmeans without any dimensionality reduction was applied. We see the distribution of labels being different for kmeans with different dimensionality reduction. No best K was observed for ICA. When Using Decision Trees the best features were MWG,NWG,NDIMC, MDIMC. All features of ICA and Random Projections were used. Cluster plots were produced but when opening the jupyter notebook, did not resurface and notebook was crashing for this dataset.

The kmeans intercluster difference inndistance was maximum for kmeans and least for Randomized Projections

Expectation Maximization

PCA

EM_pred	avg 0	1
0	4826	4349
1	4078	4328
2	21671	16331
3	7170	17247
4	2940	7062
5	2851	5227
6	23495	14828
7	23709	12643
8	12352	18439
9	8286	14057
10	9423	6288

Decision Trees

EM_pred	avg 0	1
0	21598	22882
1	48533	78891
2	19433	10711
3	31237	8315

ICA

EM_pred	avg 0	1
0	2353	2794
1	897	1330
2	16299	7899
3	2982	3151
4	5189	16954
5	6654	11154
6	9346	11945
7	6118	4407
8	360	725
9	1768	1353
10	12842	5717
11	52800	48919
12	246	1597
13	2947	2854

Randomized Projections

EM_pred	avg 0	1
0	4910	5434
1	27636	14708
2	1517	3711
3	4531	2721
4	2445	4363
5	2968	4889
6	30308	38635
7	3380	6058
8	9607	2985
9	228	826
10	15857	9272
11	1841	704
12	7838	16166
13	7735	10327

When we applied Expectation maximization, we saw the time to train and fit on our data was longer than kmeans and kmeans with other dimensionality reduction techniques. Here we see that our label is classified with majority in many of the clusters.

Dataset1	Train Accuracy	Test Accuracy	Input Dimensions	Batch Size	Epoch	Activation Functions	Neurons In Each Layer
Orignal Dataset1	97.2593	97.0778	14	5	5	R,T,T,T,T	72,72,36,18,4
PCA Dataset	92.8736	92.7773	11	5	5	R,T,T,T,T	72,72,36,18,4
Decision Trees	88.3952	88.2188	4	5	5	R,T,T,T,T	72,72,36,18,4
ICA	50.0354	49.915	14	5	5	R,T,T,T,T	72,72,36,18,4
RP	95.62	95.3	14	5	5	R,T,T,T,T	72,72,36,18,4

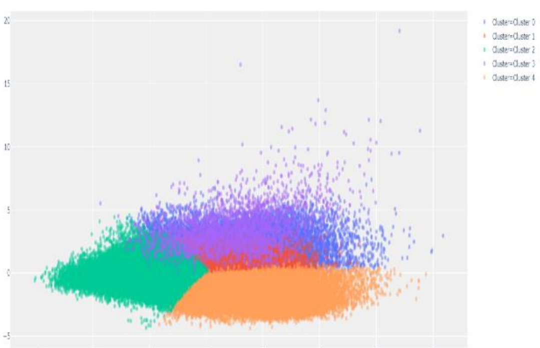
NN with cluster labels	58.5737	58.3981	2	5	5	R,T,T,T,T	72,72,36,18,4

Neural Networks

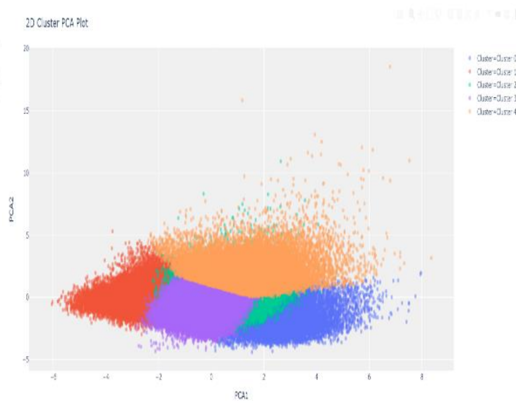
From this experiment we saw that when neural network was applied with dimensionality reduction the performance did not improve. Best performance among dimensionality reduction was observed for Randomized projections and poor performance for ICA. When we use cluster labels of kmeans and expectation maximization as our input features to classify our target label, we observed a accuracy of 58%. RP performed very similar to our original dataset. The speed of these new neural networks was faster. Tuning our neural network may help us get better accuracy. Including more cluster labels may help us achieve better results.

K means :Dataset 2

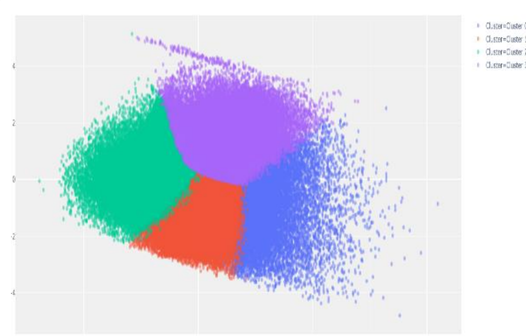
K-means



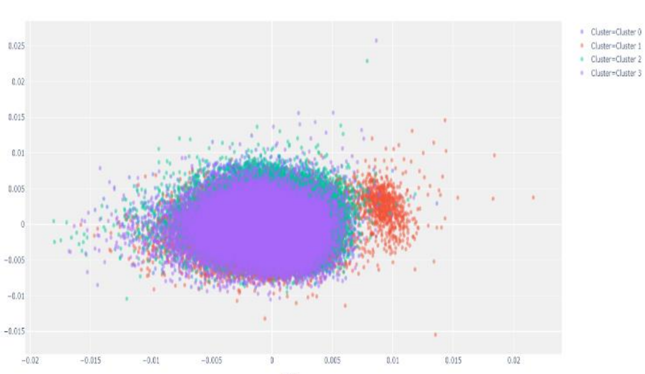
PCA



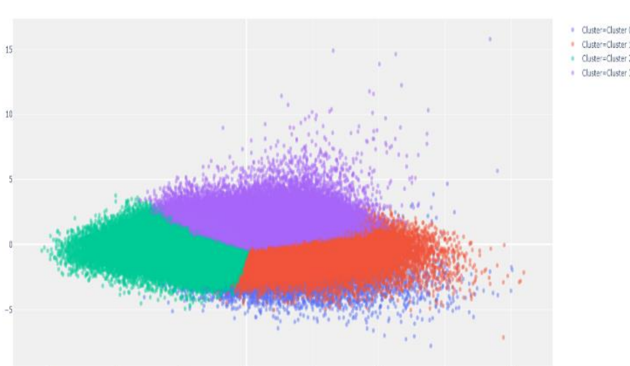
Decision Trees



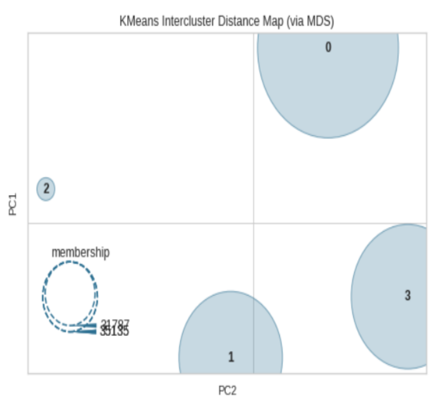
ICA

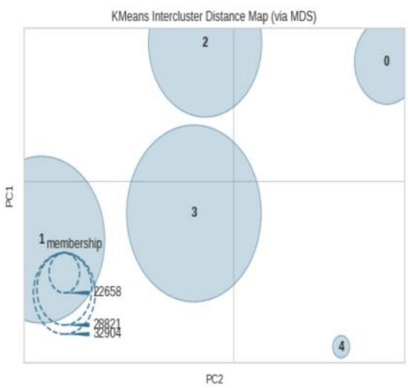


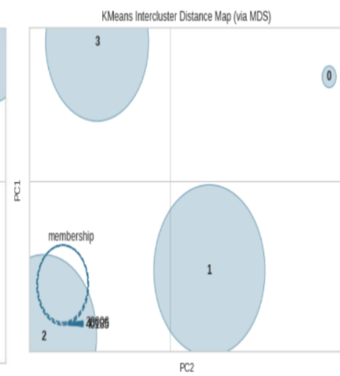
Randomized Projections

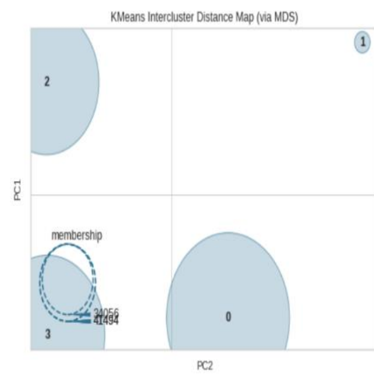


K-means PCA Decision Trees ICA









Cluster Outputs

K-means PCA Decision Trees ICA Randomized Projections

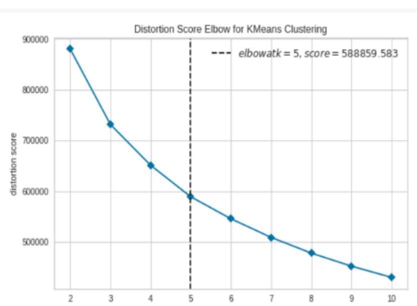
RainTomorrow	0	1
Cluster		
Cluster 0	15070	10983
Cluster 1	24149	8959
Cluster 2	42970	7775
Cluster 3	906	2000
Cluster 4	27221	2160

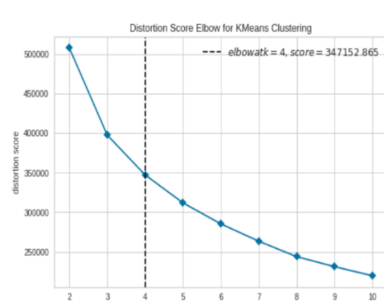
RainTomorrow	0	1
Cluster		
Cluster 0	20399	2259
Cluster 1	26465	6439
Cluster 2	19676	9145
Cluster 3	32669	3137
Cluster 4	11107	10897

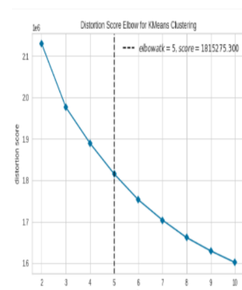
RainTomorrow	0	1
Cluster		
Cluster 0	9625	9601
Cluster 1	29786	14990
Cluster 2	35415	4770
Cluster 3	35490	2516

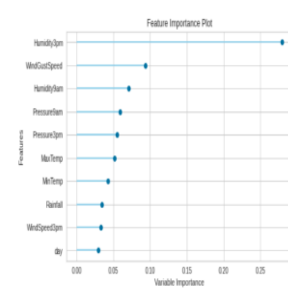
RainTomorrow	0	1
Cluster		
Cluster 0	37656	11765
Cluster 1	12671	4551
Cluster 2	27289	6767
Cluster 3	32700	8794

RainTomorrow	0	1
Cluster		
Cluster 0	18191	9757
Cluster 1	31612	2665
Cluster 2	40467	7424
Cluster 3	20046	12031









PCA Decision Trees kmeans

From the following experiment we saw the best K=4 for kmeans and k=5 for kmeans with PCA.The Best K chosen thorugh the elbow plot. The clusters became seperated when kmeans with dimensionality reduction was applied.We see the distribution of labels being different for kmeans with different dimensionality reduction. No best K was observed for ICA. When Using Decision Trees the best features were Humidity3pm, WingustSpeed, Humidity9am, Pressure9am, Pressure3pm. All features of ICA and Random Projections were used.

We observe elongated clusters for randomized projections. Clusters were very compact for ICA. For Decision Trees the clusters seem seperated the most.

The kmeans intercluster difference inndistance was maximum for PCA and least for Decision Trees

Expectation Maximization

PCA

RainTomorrow	0	1
EM_pred		
0	3540	846
1	74382	12374
2	17025	7233
3	15369	11424

Decision Trees

RainTomorrow	0	1
EM_pred		
0	9429	4760
1	14585	4411
2	14479	5671
3	16202	10098
4	55621	6937

ICA

RainTomorrow	0	1
EM_pred		
0	1560	579
1	2739	2549
2	18504	1435
3	185	621
4	24833	5789
5	957	1421
6	1718	599
7	21055	6763
8	6668	2349
9	1621	2585
10	733	346
11	19529	2985
12	534	933
13	9680	2923

Randomized Projections

RainTomorrow	0	1
EM_pred		
0	7576	2213
1	7515	2693
2	6650	1266
3	6337	2632
4	9118	2381
5	7091	1241
6	12894	3897
7	6454	1013
8	10667	3241
9	3317	937
10	8243	2370
11	9860	3116
12	7968	2749
13	6626	2128

When we applied Expectation maximization,We can see in ICA and kmeans that some of the clusters are having most of the points and other having very less points, the clusters are not unifromly distributed. We also saw the time to train and fit on our data was longer than kmeans and kmeans with other dimensionality reduction techniques.

Neural Networks

Dataset2	Train Accuracy	Test Accuracy	Input Dimensions	Batch Size	Epoch	Activation Functions	Neurons In Each Layer
Orignal Dataset1	87.04	86.1	111	5	5	R,S,S,R	64,32,16,8
PCA Dataset	82.43	82.45	4	5	5	R,S,S,R	64,32,16,8
Decision Trees	83.79	84.04	5	5	5	R,S,S,R	64,32,16,8

ICA	83.69	83.83	14	5	5	R,S,S,R	64,32,16,8
RP	85.9	85.44	111	5	5	R,S,S,R	64,32,16,8
NN with cluster labels	78.39	78.17	2	5	5	R,R,R,R	8,8,4,4

From this experiment we saw that when neural network was applied with dimensionality reduction the performance did not improve. Best performance among dimensionality reduction was observed for Randomized Projection RP and no dimensionality reduction technique performed poorly. When we used cluster labels of K-means and expectation maximization as our input features to classify our target label, we observed very good accuracy of 78%. Tuning our dimensionality reduction neural networks, especially Randomized projections may help us get better results. Also, if we use cluster labels of our dimensional reduction clustering, we may observe even better performance. We observed better performance for dataset2 than dataset1 for classification using Neural Networks with dimensionality reduction.