Introduction:

In this assignment we are working on Artificial Neural Network and K Nearest neighbors. I have used two data sets:

- 1) Best GPU Processor Predictor
- 2)Brisbane weather predictor

DATASET 1

Best GPU Processor Predictor

In part 1, We have applied classification algorithms on GPU Processor Predictor dataset and classified the combination of processors as good and bad. We have checked for null values and made a new column i.e average runtime. We have checked for outliers and have split the dataset into training set and testing set with the ratio of 30% testing set and 70% training set. Standard scaling is done on all the feature variables.

ARTIFICIAL NEURAL NETWORK

1) Download and use any neural network package to classify your classification problems. Experiment with number of layers and number of nodes, activation functions(sigmoid,tanh etc), and may be couple of other parameters.

EXPERIMENTS

A) Trying with different number of epochs:

NUMBER OF EPOCHS	ACTIVATION FUNCTION	NO. OF HIDDEN LAYERS	NUMBER OF NODES	ACCURACY	CONFUSION MATRIX
200	RELU	1	6	0.78516	[[39768 3416] [10430 10836]]
100	RELU	1	6	0.75351	[[39066 4118] [11768 9498]]
400	RELU	1	6	0.86844	[[7918 861] [1128 5212]]

B) Trying with different number of layers:

NUMBER OF EPOCHS	ACTIVATION FUNCTION	NO. OF HIDDEN LAYERS	NUMBER OF NODES	ACCURACY	CONFUSION MATRIX
400	RELU	2	6	0.86308	[[7983 796] [1274 5066]]
400	RELU	3	6	0.58066	[[8779 0] [6340 0]]
400	RELU	4	6	0.58066	[[8779 0] [6340 0]]

C) Trying with different Activation Function:

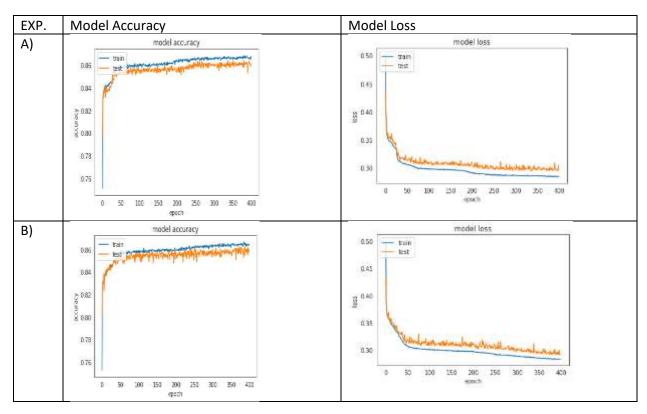
NUMBER OF EPOCHS	ACTIVATION FUNCTION	NO. OF HIDDEN LAYERS	NUMBER OF NODES	ACCURACY	CONFUSION MATRIX
400	RELU	2	6	0.86308	[[7983 796] [1274 5066]]
400	SIGMOID	2	6	0.86057	[[7717 1062] [1046 5294]]
400	SOFTMAX	2	6	0.41933	[[0 8779] [0 6340]]

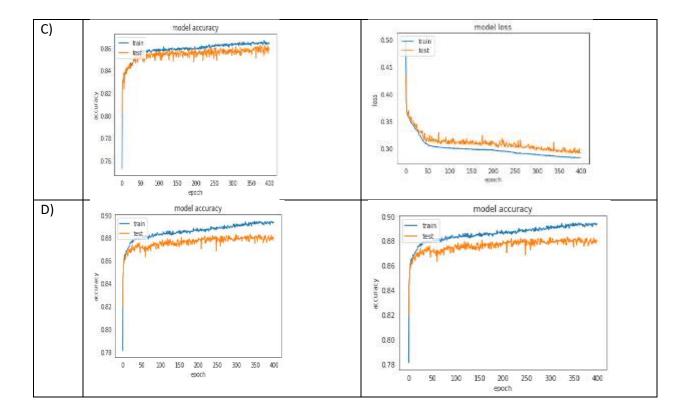
D) Changing the number of nodes to 16

NUMBER OF EPOCHS	ACTIVATION FUNCTION	NO. OF HIDDEN LAYERS	NUMBER OF NODES	ACCURACY	CONFUSION MATRIX
400	RELU	2	16	0.88385	[[7906 873] [883 5457]]

Hence the above combination is the best model with the appropriate parameters. Below are the graphs for the above experiment.

GRPAHS:

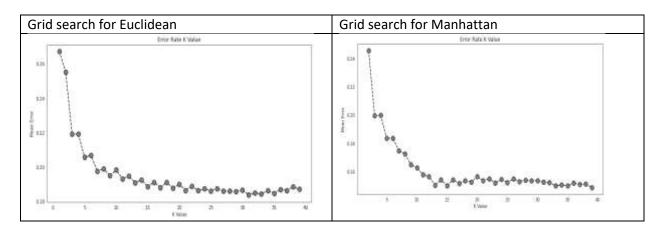




2) K NEAREST NEIGHBOURS

Download and use any KNN package to classify your classification problems. Experiment with number of neighbors. You can use any distance metric appropriate to your problem. Just be clear to explain why you used the metric that you used.

In this algorithm, I have tried with random values of k such as 5 and 3 and then applied grid search algorithm for Euclidean and Manhattan distance to find appropriate value of k. Below are the results and graphs.



Value of K	Confusion	Accuracy	Mean	Standard Deviation	Learning curve
	Matrix		Accuracy		
			with CV=5		

				1	0858090813290621
5(Eucldiean)	[[7359 1420]	79.42324	78.85754	0.57531	KNNClassifier
	[1691 4649]]				10
	[1031 4043]]				on
					200
					1
					R os
					92 Naming score
					0.0 Cross-addition score
					5000 10000 15000 20000 25000 30000 35000 40000
					Facing examples
3	[[7218 1561]	78.06733	77.42026	0.31985	KNNClassifier
	[1755 4585]]				10
	[1755 1565]]				0.8
					0.6
					B 0.4
					0.2
					- Naming State
					RG — Cress-validation score
					5000 30000 25000 30000 25000 30000 35000 40000
					having countries.
32(Best	[[7823 956]	81.49348	80.43090	0.51173	kNYClassifie*
value from	[1842 4498]]				1.0 Substitution score —— Cross velidation score
	[1042 4430]]				40
grid search)					
					1
					N
					921
					0.0
					5200 10000 25000 20000 35000 36000 36000 40000 Thirting examples
					Twing examples
5(Manhatta	[[7518 1261]	81.64561	81.16513	0.54125	RNINClassifier
n)	[1514 4826]]				10
'''	[1314 4020]]				
					- 1.
					0.6
					0.4
					07
					Training score
					0.0 - Cross-validation score
					5000 10000 15000 20000 25000 10000 15000 40000
3					hunning dramptics
	[[7347 1432]	80.0515	79.16654	0.34034	NNClassifier
	[[7347 1432]	80.0515	79.16654	0.34034	7.37.7.33.7.3
	[[7347 1432] [1584 4756]]	80.0515	79.16654	0.34034	KNNClassifier 10
		80.0515	79.16654	0.34034	KNNClassifier
		80.0515	79.16654	0.34034	KNNClassifier 18 08
		80.0515	79.16654	0.34034	KNNClassifier 10 03 06
		80.0515	79.16654	0.34034	KNNClassifier 18 08
		80.0515	79.16654	0.34034	KNNClassifier 10 03 06
		80.0515	79.16654	0.34034	NNNClassifier 18 08 06 8
		80.0515	79.16654	0.34034	NNNClassifier 18 08 06 10 04 02 10 10 10 10 10 10 10 10 10 10 10 10 10
		80.0515	79.16654	0.34034	0.8 0.6 0.4 0.4 0.2 no
15/host	[1584 4756]]				NNNClassifier
15(best	[1584 4756]] [[7804 975]	80.0515 84.9857	79.16654 84.06236	0.34034	NNNClassifier
T	[1584 4756]] [[7804 975]				NNNClassifier
value from	[1584 4756]]				NNNClassifier
value from manhattan	[1584 4756]] [[7804 975]				NNNClassifier
value from manhattan	[1584 4756]] [[7804 975]				NNNClassifier
value from	[1584 4756]] [[7804 975]				NNNClassifier
value from manhattan	[1584 4756]] [[7804 975]				NNNClassifier
value from manhattan	[1584 4756]] [[7804 975]				NNNClassifier
value from manhattan	[1584 4756]] [[7804 975]				NNNClassifier

DATASET 2

Brisbane weather dataset contains 24 variables. It contains 56240 rows. In this I have applied classification algorithm to decide whether it will rain tomorrow or not. There are total 7 categorical variables including date column. We will create dummy variables for categorical variables. We have checked for outliers and have split the dataset into training set and testing set with the ratio of 30% testing set and 70% training set. Standard scaling is done on all the feature variables.

ARTIFICIAL NEURAL NETWORK

EXPERIMENTS

A) Trying with different number of epochs:

NUMBER OF EPOCHS	ACTIVATION FUNCTION	NO. OF HIDDEN LAYERS	NUMBER OF NODES	ACCURACY	CONFUSION MATRIX
200	RELU	1	6	0.85655	[[12154 1004] [1424 2344]]
100	RELU	1	6	0.85708	[[12267 891] [1528 2240]]
400	RELU	1	6	0.85684	[[12353 805] [1618 2150]]

B) Trying with different number of layers:

NUMBER OF EPOCHS	ACTIVATION FUNCTION	NO. OF HIDDEN LAYERS	NUMBER OF NODES	ACCURACY	CONFUSION MATRIX
100	RELU	2	6	0.85749	[[12349 809] [1603 2165]]
100	RELU	3	6	0.85767	[[12320 838] [1571 2197]]
100	RELU	4	6	0.85773	[[12375 783] [1625 2143]]
100	RELU	10	6	0.85501	[[12181 977] [1477 2291]]

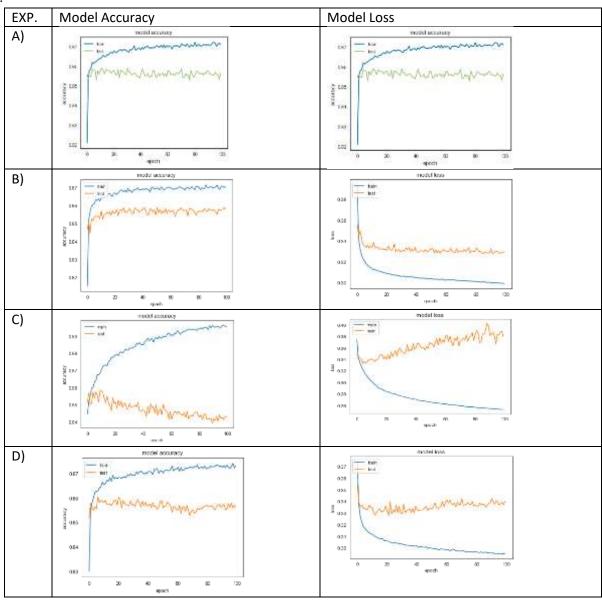
C) Trying with different Activation Function:

NUMBER OF EPOCHS	ACTIVATION FUNCTION	NO. OF HIDDEN LAYERS	NUMBER OF NODES	ACCURACY	CONFUSION MATRIX
100	SIGMOID	4	6	0.84012	[[11888 1270] [1436 2332]]
100	SOFTMAX	4	6	0.83752	[[12358 800] [1950 1818]]
100	TANH	4	6	0.84414	[[12008 1150] [1488 2280]]

D) Changing the number of nodes to 16

NUMBER OF EPOCHS	ACTIVATION FUNCTION	NO. OF HIDDEN LAYERS	NUMBER OF NODES	ACCURACY	CONFUSION MATRIX
100	RELU	2	16	0.85288	[[12248 910] [1580 2188]]

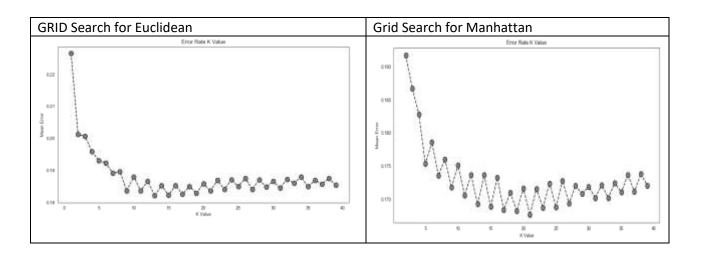
GRPAHS:



3) K NEAREST NEIGHBOURS

Download and use any KNN package to classify your classification problems. Experiment with number of neighbors. You can use any distance metric appropriate to your problem. Just be clear to explain why you used the metric that you used.

In this algorithm, I have tried with random values of k such as 5 and 3 and then applied grid search algorithm for Euclidean and Manhattan distance to find appropriate value of k. Below are the results and graphs.



Value of K	Confusion Matrix	Accuracy	Mean Accuracy with CV=5	Standard Deviation	Learning curve
5(Eucldiean)	[[12387 771] [2495 1273]]	80.70424	81.12624	0.32931	WANCIssasifier I livering pore Disabindisher cents OS OS OS OS OS OS OS OS OS O
3	[[12130 1028] [2367 1401]]	79.94210	80.32359	0.30965	### NONC 2000 2000 4000 5000
13(Best value from grid search)	[[12727 431] [2650 1118]]	81.79723	81.936486	0.243646	NNNClassifier Tuening wave Crisis-4-bidedon screen 0.6 0.4 0.2 D.0 5000 10000 10000 20000 20000 30000 30000 40000 40000 Training examples
5(Manhattan)	[[12429 729] [2238 1530]]	82.47075	82.534052	0.3279131	100 Training examples Training examples

2	[[12107 071]	01 22040	01 7162	0.10636	KNINCIsseifier
3	[[12187 971] [2189 1579]]	81.33049	81.7162	0.19626	0.6 0.6 0.4 0.4 0.2 0.4 0.2 0.4 0.2 0.4 0.5 0.4 0.5 0.4 0.5 0.4 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5
21(best value from grid search)	[[12787 371] [2466 1302]]	83.23880	83.2987	0.26177	2000 15000 20000 20000 20000 20000 40000 40000 40000 Training searchies KNINClassifier Liverage score Creativalidation score 100 100 1000 10000 10000 20000 30000 30000 40000 40000 40000 700000 70000 70000 70000 70000 70000 70000 70000 70000 70000 70000 700000 700000 70000 70000 70000 70000 700000 700000 70000 70000 7

ACCURACY COMPARISON:

DATASETS	SVM	Decision Trees	Boosting	ANN(Relu)	KNN(Manhattan)
Dataset	78.7323506	68.2932	75.4584	88.3854	84.9857
1(GPU)	594259				
Dataset	86.4409783	80.1784	85.4011	085.7733	83.2388
2(Brisbane	7646225				
weather					
predictor)					

ANN did better for dataset 1 and SVM did better for dataset 2.

IMPROVEMENTS AND SUGESSTIONS:

- 1) More combinations of activation functions , number of layers and number of nodes could have been tried to achieve better accuracy and results.
- 2) Plotting accuracy curve for different algorithms could have helped us to compare the different algorithms better.