

Project Report

Rishabh Aryan Das
UFID: 0536 - 9273

Amit Bhadra
UFID: 6649 - 4087

October 2020

Implementation

- Implemented three types of classifiers namely *linearSVM*, *KNN* & *MLP* on the two datasets that were provided to us *tictac-toe_single.txt* and *tictactoe_final.txt*.
- Implemented three types of regressors namely *LinearRegression*, *MLPRegressor* & *KNNRegressor* on the multi-label dataset *tictactoe_multi.txt*. The output of the regressors were converted to binary form using one hot encoding based on a threshold value.
- Implemented an interactive tictactoe game. The game is designed to be played between a human and the computer where the computer uses the parameters of the model to predict the probabilities of each grid cells and make the next move.
- The datasets were split in the ratio 8:2 for training sets and testing sets. The final accuracies are calculated based on the test sets.

Evaluation

- The evaluation metrics of the classifiers are given below as mentioned in *classifiers_log.txt* and *regressors_log.txt* files.

Dataset	Model	Hyperparameters	Accuracy
tictac_final	SVM	penalty: 'l2' loss: 'squared_hinge' dual: True tol: 1e-4 C: 1.0	97.9
	KNN	n_neighbors: 5 weights: 'uniform' algo: 'auto' leaf_size: 30 metric: 'minkowski'	97.3
	MLP	hidden_layer_sizes: 100 activation: 'relu' solver: 'adam' learning_rate: 'constant'	100
tictac_single	SVM	'C': 12 'gamma': 1	39
	KNN	algorithm: 'brute', leaf_size: 1, n_neighbors: 10, weights: 'distance'	100
	MLP	activation: 'tanh' hidden_layer_sizes: (100, 50, 25), learning_rate: 'constant' solver: 'lbfgs'	100
tictac_multi	LR	NA	72.3
	KNN	algorithm: 'brute' leaf_size: 1 n_jobs: -1, n_neighbors: 10, weights: 'distance'	100
	MLP	activation: 'tanh' hidden_layer_sizes: (100, 50, 25), learning_rate: 'constant' solver: 'lbfgs'	100

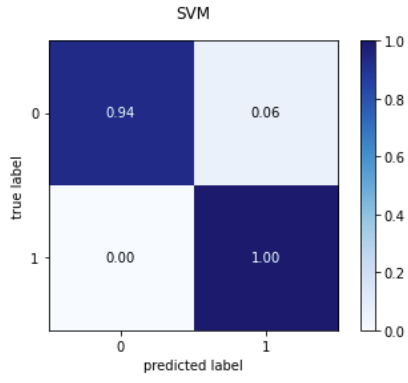
No hyperparameters are provided for linear regression as we used the normal equation to calculate the weights

Conclusion

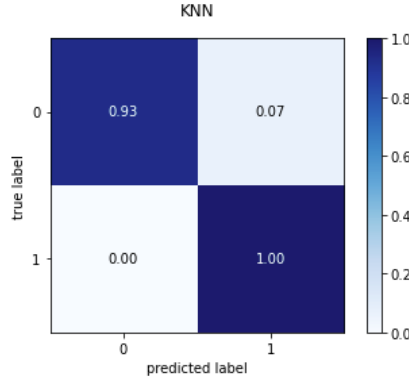
- Among all the classifiers the best accuracy is reported by KNN and MLP
- Among all the regressors the best accuracy was reported by KNN and MLP
- Hence we have used the MLP regressor model for the game

Instructions

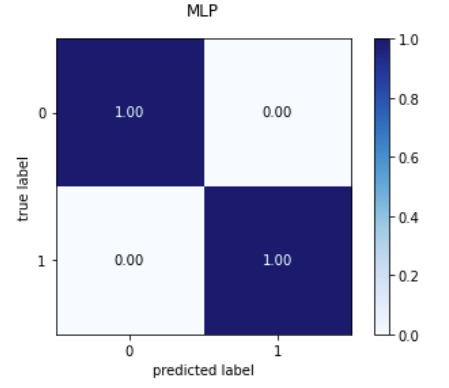
- Train the classifiers using command **python classifiers.py**
- Train the regressors using command **python regressors.py**
- Start the game using command **python playKataKuti.py**



(a) SVM

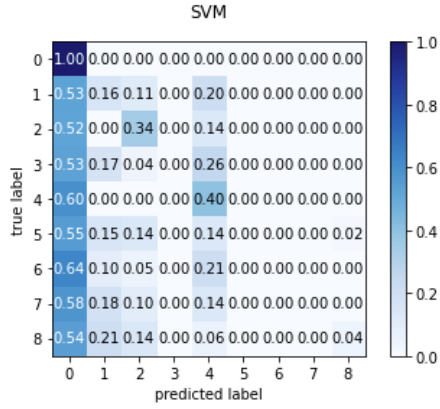


(b) KNN

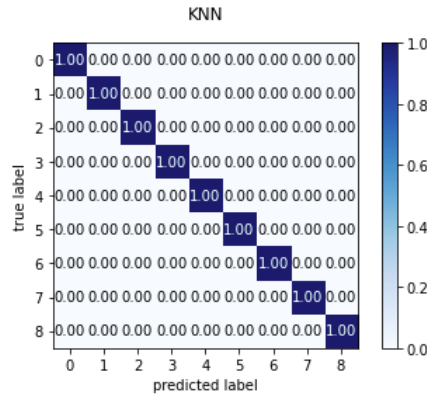


(c) MLP

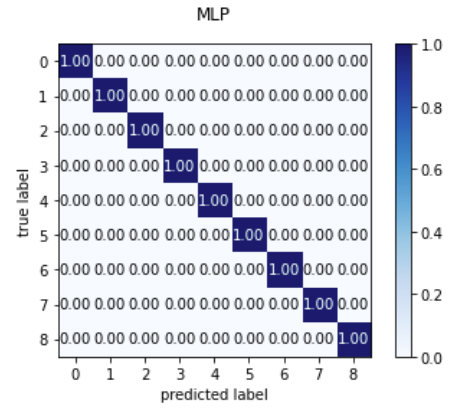
Figure 1: Confusion matrices of tictac_final.txt dataset



(a) SVM

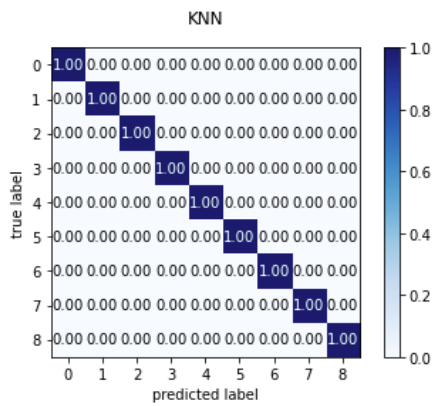


(b) KNN

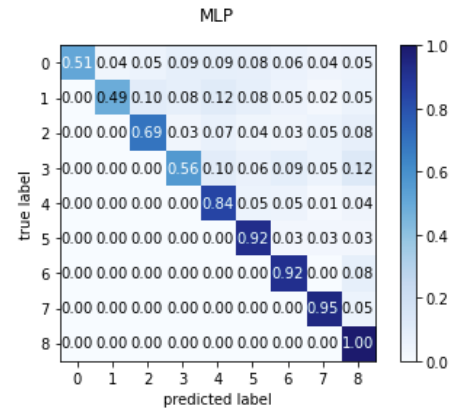


(c) MLP

Figure 2: Confusion matrices of tictac_single.txt dataset



(a) KNN



(b) MLP

Figure 3: Confusion matrices of tictac_multi.txt dataset