

Kevin Hsieh
2078611
COSC4368
Dr. Eick

Problem Set 2 Task 3:

1. Accuracy of the four classification algorithms and the average Mean Absolute Error(MAE)

Linear Kernel accuracy: 17 %					Relu MLP accuracy: 14 %				
	precision	recall	f1-score	support		precision	recall	f1-score	support
0	0.14	0.20	0.17	5	0	0.00	0.00	0.00	5
1	0.25	0.30	0.27	10	1	0.25	0.20	0.22	10
2	0.25	0.22	0.24	9	2	0.17	0.22	0.19	9
3	0.17	0.17	0.17	6	3	0.20	0.17	0.18	6
4	0.00	0.00	0.00	3	4	0.00	0.00	0.00	3
5	0.00	0.00	0.00	3	5	0.00	0.00	0.00	3
6	0.00	0.00	0.00	5	6	0.20	0.20	0.20	5
7	0.00	0.00	0.00	3	7	0.00	0.00	0.00	3
accuracy			0.16	44	accuracy			0.14	44
macro avg	0.10	0.11	0.11	44	macro avg	0.10	0.10	0.10	44
weighted avg	0.15	0.16	0.15	44	weighted avg	0.14	0.14	0.14	44
RBF Kernel accuracy: 21 %					Tanh MLP accuracy: 10 %				
	precision	recall	f1-score	support		precision	recall	f1-score	support
0	0.00	0.00	0.00	5	0	0.00	0.00	0.00	5
1	0.23	0.70	0.34	10	1	0.18	0.20	0.19	10
2	0.15	0.22	0.18	9	2	0.18	0.22	0.20	9
3	0.00	0.00	0.00	6	3	0.00	0.00	0.00	6
4	0.00	0.00	0.00	3	4	0.00	0.00	0.00	3
5	0.00	0.00	0.00	3	5	0.00	0.00	0.00	3
6	0.00	0.00	0.00	5	6	0.00	0.00	0.00	5
7	0.00	0.00	0.00	3	7	0.00	0.00	0.00	3
accuracy			0.20	44	accuracy			0.09	44
macro avg	0.05	0.12	0.07	44	macro avg	0.05	0.05	0.05	44
weighted avg	0.08	0.20	0.11	44	weighted avg	0.08	0.09	0.08	44

```
linear SVC MAE testing score: 1.8863636363636365
rbf SVC MAE testing score: 1.9772727272727273
relu MLP MAE testing score: 1.9090909090909092
tanh MLP MAE testing score: 1.8863636363636365
```

2. After comparing the experimental results, write a paragraph which summarizes the experimental results and also tries to explain/speculate why, in your opinion one classification algorithm outperformed the other.

By default, SVM's usually have higher prediction accuracy than a multilayer perceptron. SVM's might have higher runtime as there are calculations it performs that are advanced like translating n-dimensional space using kernel functions. But it usually does a very good job in its predictions. In this dataset, we are classifying grade outcomes by various attributes that are observed with each student. The 8 classifications our models are predicting are from 0-8. This data specifically is prime for an SVM model, as it can easily find the perfect hyperplane after translating the data using its kernel function. As soon as you scale to large datasets, a MLP will perform better in almost all cases. People typically use pre-defined kernels in SVMs — and they usually work pretty well. But due to their one-size-fits-all nature, a SVM will perform worse than a MLP on tasks where it's better to let a MLP to construct its own non-linear projection.

3. If you conducted extra credit activities concerning minimizing the MAE of models, write 2-3 paragraphs which summarize these activities

```
linear SVC MAE testing score: 1.8863636363636365
rbf SVC MAE testing score: 1.9772727272727273
relu MLP MAE testing score: 1.9090909090909092
tanh MLP MAE testing score: 1.8863636363636365

improved linear SVC MAE testing score: 1.7727272727272727
improved rbf SVC MAE testing score: 1.6136363636363635
improved relu MLP MAE testing score: 1.6590909090909092
improved tanh MLP MAE testing score: 1.75
```

Finding the right parameters (like what C or gamma values to use) is a tricky task, but luckily, we can be a little lazy and just try a bunch of combinations and see what works best. This idea of creating a 'grid' of parameters and just trying out all the possible combinations is called a Gridsearch, this method is common enough that Scikit-learn has this functionality built in with GridSearchCV. The CV stands for cross-validation. GridSearchCV takes a dictionary that describes the parameters that should be tried and a model to train. The grid of parameters is defined as a dictionary, where the keys are the parameters and the values are the settings to be tested.

Here for linear and rbf SVM, I picked:

```
param_grid = {'C': [0.1,1, 10, 100, 1000], 'gamma': [1,0.1,0.01,0.001,0.0001]}  
grid = GridSearchCV(SVC(),param_grid,refit=True,verbose=3)
```

What fit does is a bit more involved than usual. First, it runs the same loop with cross-validation to find the best parameter combination. Once it has the best combination, it runs fit again on all data passed to fit (without cross-validation), to build a single new model using the best parameter setting.

Next I ran for this command both SVM models: **grid.best_params_**
Best Parameters Output:

```
{'C': 0.1, 'gamma': 1, 'kernel': 'linear'}  
{'C': 10, 'gamma': 0.01, 'kernel': 'rbf'}
```

And running predictions with those hyperparameters, I got an improvement of linear—20% and rbf—23% accuracy, an improvement from our previous of linear—17% and rbf—21%!

Next I ran the relu and tanh MLP through the GridSearchCV with these parameter spaces:

```
relu_parameter_space = {  
    'hidden_layer_sizes': [(50,50,50), (50,100,50), (100,,)],  
    'solver': ['sgd', 'adam'],  
    'alpha': [0.0001, 0.05],  
    'learning_rate': ['constant','adaptive'],  
}
```

Best Parameters Output:

```
{'activation': 'relu', 'alpha': 0.0001, 'hidden_layer_sizes': (100,),'learning_rate': 'constant',  
'solver': 'sgd'}  
  
{'activation': 'tanh', 'alpha': 0.0001, 'hidden_layer_sizes': (50, 100, 50), 'learning_rate':  
'constant', 'solver': 'sgd'}
```

And running predictions with those hyperparameters, I got an improvement of relu—23% and tanh—27% accuracy, an improvement from our previous of relu—14% and tanh—10%!

4. Finally, at the end of your report write a short paragraph which summarize the most important findings of this task.

With so many different classifications, it would make sense to have a low accuracy score. We can see that the estimator using the 'rbf' SVM kernel performed best, closely followed by 'linear' without using hyperparameter tuning, while 'relu' MLP performed better than 'tanh'. It's easy to understand that many machine learning problems benefit from either precision or recall as their optimal performance metric but implementing the concept requires knowledge of a detailed process. My first few attempts to fine-tune models for recall (sensitivity) were difficult, but overall I was able to lower the MAE metrics while significantly improved the MLP model's accuracy scores. In the end, I was able to show the practicality of optimizing a classifier for sensitivity. Education and related performance evaluation is a challenging task for human beings. Therefore computer assisted systems are developed in order to improve this evaluation. One of the most important approaches in this performance evaluation is Artificial Intelligence which can classify or predict huge data. Several experiments were conducted and different machine learning algorithms were implemented. Experimental results showed the classification of student performances according to any questionnaire type. With hyperparameter tuning, MLP seems to work best as the dataset becomes larger with precision.