CS 6375

Scikit Lab 2

Names of students in your group:

Ankita Patil (asp160730)

Abhilash Gudasi (abg160130)

Number of free late days used: 0

Please list clearly all the sources/references that you have used in this assignment.

<http://scikit-learn.org/stable/tutorial/text_analytics/working_with_text_data.html>

**Q. 1 Parameter Tuning of Models on the Digits Dataset**

--> **Decision Tree**

Best parameters:

{'criterion': 'entropy',

'max\_depth': 14,

'max\_features': 7,

'min\_samples\_leaf': 1,

'min\_samples\_split': 2}

Accuracy: 82.2%

Detailed classification report:

The model is trained on the full development set.

The scores are computed on the full evaluation set.

precision recall f1-score support

0 0.90 0.96 0.93 27

1 0.70 0.80 0.75 35

2 0.94 0.83 0.88 36

3 0.70 0.90 0.79 29

4 0.76 0.83 0.79 30

5 0.92 0.82 0.87 40

6 0.93 0.86 0.89 44

7 0.84 0.82 0.83 39

8 0.81 0.64 0.71 39

9 0.77 0.80 0.79 41

avg / total 0.83 0.82 0.82 360

--> **Neural Net:**

Best parameters:

{'activation': 'relu',

'alpha': 0.1,

'hidden\_layer\_sizes': (10, 10, 5),

'learning\_rate': 'constant',

'learning\_rate\_init': 0.01}

Accuracy: 76.11%

Detailed classification report:

The model is trained on the full development set.

The scores are computed on the full evaluation set.

precision recall f1-score support

0 0.93 0.96 0.95 27

1 0.85 0.80 0.82 35

2 0.97 0.94 0.96 36

3 0.84 0.90 0.87 29

4 1.00 0.97 0.98 30

5 0.91 0.97 0.94 40

6 0.98 0.93 0.95 44

7 0.90 0.97 0.94 39

8 0.45 0.26 0.33 39

9 0.71 0.66 0.68 41

avg / total 0.85 0.83 0.83 360

--> **Support Vector Machine:**

Best parameters:

{'C': 100, 'degree': 3, 'gamma': 0.0001, 'kernel': 'poly', 'max\_iter': 100}

Accuracy :98.95%

Detailed classification report:

The model is trained on the full development set.

The scores are computed on the full evaluation set.

precision recall f1-score support

0 1.00 1.00 1.00 27

1 0.95 1.00 0.97 35

2 1.00 1.00 1.00 36

3 1.00 1.00 1.00 29

4 1.00 1.00 1.00 30

5 0.97 0.97 0.97 40

6 1.00 0.98 0.99 44

7 1.00 1.00 1.00 39

8 1.00 0.97 0.99 39

9 0.98 0.98 0.98 41

avg / total 0.99 0.99 0.99 360

--> **Gaussian Naive Bayes** (Since this has only one parameter we ran directly without grid search)

Accuracy: 82.5%

Detailed classification report:

The model is trained on the full development set.

The scores are computed on the full evaluation set.

precision recall f1-score support

0 1.00 1.00 1.00 27

1 0.66 0.89 0.76 35

2 0.94 0.47 0.63 36

3 0.89 0.83 0.86 29

4 1.00 0.73 0.85 30

5 0.97 0.88 0.92 40

6 1.00 1.00 1.00 44

7 0.74 1.00 0.85 39

8 0.54 0.82 0.65 39

9 0.96 0.63 0.76 41

avg / total 0.87 0.82 0.83 360

--> **Logistic Regression:**

Best parameters:

{'C': 1.0,

'fit\_intercept': False,

'max\_iter': 1000,

'penalty': 'l1',

'tol': 0.01}

Accuracy: 96.79%

Detailed classification report:

The model is trained on the full development set.

The scores are computed on the full evaluation set.

precision recall f1-score support

0 1.00 1.00 1.00 27

1 0.89 0.94 0.92 35

2 0.97 0.94 0.96 36

3 0.94 1.00 0.97 29

4 1.00 1.00 1.00 30

5 0.97 0.97 0.97 40

6 0.98 0.98 0.98 44

7 0.95 0.97 0.96 39

8 0.92 0.92 0.92 39

9 0.97 0.88 0.92 41

avg / total 0.96 0.96 0.96 360

--> **k-Nearest Neighbors**:

Best parameters:

{'algorithm': 'auto', 'n\_neighbors': 2, 'p': 2, 'weights': 'distance'}

Accuracy: 98.88%

Detailed classification report:

The model is trained on the full development set.

The scores are computed on the full evaluation set.

precision recall f1-score support

0 1.00 1.00 1.00 27

1 1.00 1.00 1.00 35

2 1.00 0.97 0.99 36

3 0.94 1.00 0.97 29

4 1.00 1.00 1.00 30

5 0.97 0.97 0.97 40

6 1.00 1.00 1.00 44

7 1.00 1.00 1.00 39

8 1.00 1.00 1.00 39

9 0.97 0.95 0.96 41

avg / total 0.99 0.99 0.99 360

**Which method(s) performed best and why do you think so? What could be done to improve the results?**

Models Accuracy

Decision Tree 82.2%

Neural Net 76.11%

Support Vector Machine 98.95%

Gaussian Naive Bayes 82.5%

Logistic Regression 96.79%

k-Nearest Neighbors 98.88%

- Since the accuracy of Support Vector Machine, Logistic Regression, k-Nearest Neighbors seems to be fairly high, they seem to have performed best on given data. But again, it all depends on the type of data, features in the dataset and data size.

- Support Vector Machine is more robust, it has simple decision boundary. Due to optimal margin gap between separating hyperplanes, it can do predictions better with test data and that's why we observe that with test data, SVM has done better predictions

- When dealing with multi-dimensional data, logistic regression performs better

- k-Nearest neighbors has pretty good accuracy which indicates that classes are quite separable from each other and overall, the accuracy also depends on the value you choose for k

From above table, we observe that the accuracy of Decision Tree, Neural Net, Gaussian Naive Bayes is comparatively lower than other models, because the accuracy depends on various factors such as the type of data, data size, noise in the data, features in the data and the machine learning model you are applying.

To improve the model performance or to enhance the results, we can do following things:

- Use ensemble methods such as Bagging, Boosting

- We should perform feature engineering

- We can treat missing data

**Q. 2 Working with Text Data**

Dataset chosen: Twitter U.S. Sentiment Analysis

Best parameters MNB Algorithm

alpha with value 0.01

With MNB, the best score/accuracy we got, 77.45%

We are not satisfied with the accuracy.

Before preprocessing and after preprocessing the tweet text data, we did not see much improvement in the accuracy.

We can improve the accuracy with a linear support machine(SVM) by playing with parameters such as loss, alpha, penalty, maximum iterations, etc. For text processing, SVM seems to be better than MNB.