# STA 561 Final Project
# Handwritten Digits Recognition

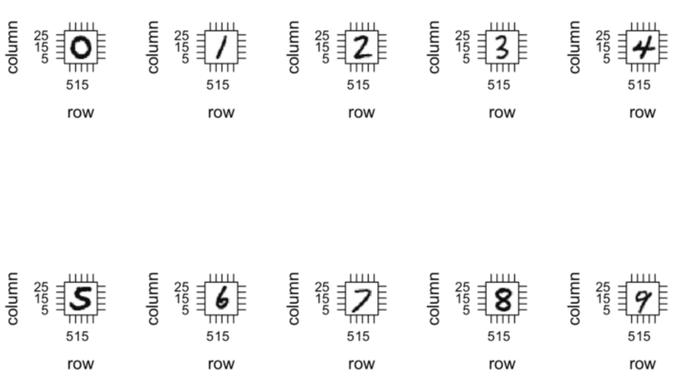
Minjung Park, Yue(April) Mu

## Abstract

In this final project, our goal is to classify images of handwritten single digits using the **Support Vector Machine**(SVM) method. SVM was originally designed and used for binary classification. However, it is also a great tool for multi-class classification, novelty detection, and regression. In our project, we will be using packages 'e1071' and 'kernlab' for SVMs with Gaussian kernel function. We will test the accuracy of such method and present the results through visualization.

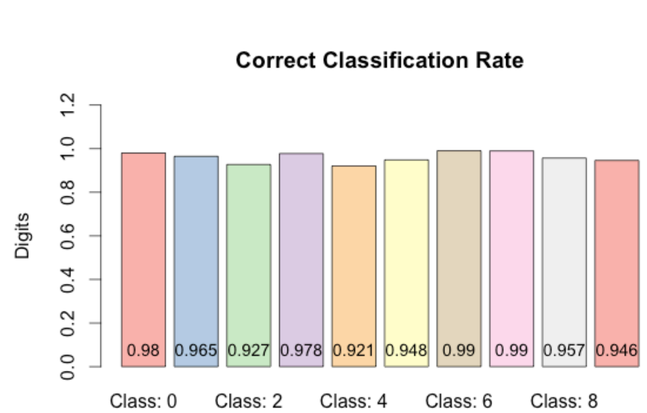
## Overview of Data

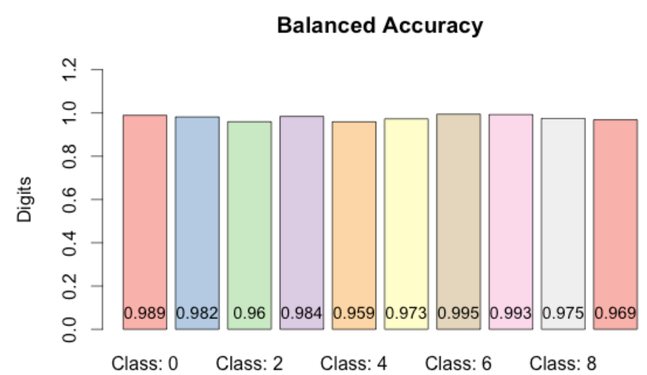
We used the data from Kaggle competition (link: https://www.kaggle.com/c/digit-recognizer). The data originally comes from the Modified National Institute of Standards and Technology (MNIST) database, which is a classic within the Machine Learning community. This dataset contains a total of 70000 observations split into training and testing datasets. Each observation is a 28 * 28 = 784 pixels image, and each pixel contains a grey-scale value from 0 to 255. The only difference between training and testing dataset is that the training data contains a label column with values of 0-9, which indicates the actual handwritten digit.
This is a sample of 10 digits from the training set




## SVM & Training Accuracy

In order to assess the accuracy of the model, we decided to use the K-fold cross-validation approach. In K-fold cross-validation, the original dataset is randomly partitioned into K subsets of equal size. One of the K subsets is retained as the validation data for testing the model, and the remaining K-1 subsets are used as training data. This cross-validation process is then repeated for K times, with each of the K subsets used exactly once as the validation data. For this dataset, we will implement cross validation on a subset of the training data with 5000 observations and K = 5 folds.
The average accuracy we got from the K-fold cross validation is about 94.6%.

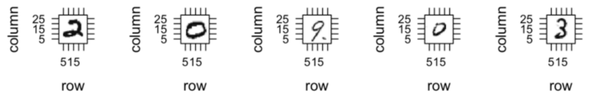


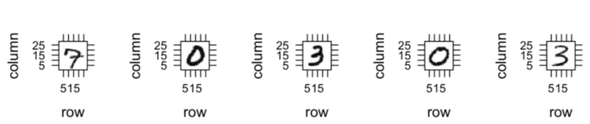




When looking at the Sensitivity Rate (True Positive Rate), it seems that the model was performing a bit better for some specific classes. However, for some other classes, the True Positive Rate is a bit lower.

## Prediction

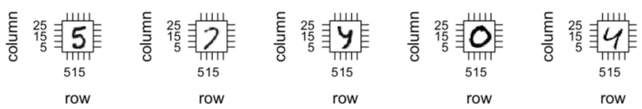
With an overall accuracy of 94.6%, we can then train the model on the full training set and predict the label based on the full testing set. In our sample of 20 observations, except that the 4th digit '0' is misread as '9', the rest 19 predictions are correct, which roughly gives a 95% accuracy. This result is consistent with the average accuracy we calculated previously.

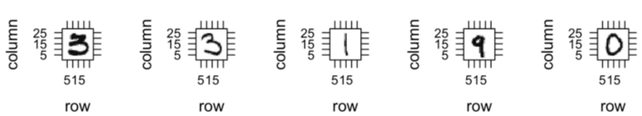



```
as.character(prediction[1:10,1])
```

```
##  [1] "2" "0" "9" "9" "3" "7" "0" "3" "0" "3"
```




```
as.character(prediction[11:20,1])
```

```
##  [1] "5" "7" "4" "0" "4" "3" "3" "1" "9" "0"
```

## Conclusion

The Support Vector Machines (SVMs) algorithm has been widely used for classification purposes. Our analysis demonstrates that SVM works well for the digit recognition problem.