CSC411: Assignment 3

Due on Sunday, December 3$^{\text{rd}}$, 2017

Student Name: **Gokul K. Kaushik**

Student Number: **999878191**

Table of Contents

1 - 20 Newsgroup Predictions

3 methods chosen

I chose to use the *tf-idf* feature representation. I felt that logically it would give me a better result as it uses frequency. **All subsequent information is based using this feature set**.

The 3 models (other than the baseline model) that were chosen are:

1. SGD
2. Logistic Regression
3. Random Forest

Train and Test Accuracies and Losses

1. Baseline Model
    (a) Train accuracy: 59.8727
    (b) Test accuracy: 45.7912
    (c) Train loss: 0.4012
    (d) Test loss: 0.5420
2. SGD
    (a) Train accuracy: 95.6248
    (b) Test accuracy: 69.6096
    (c) Train loss: 0.0437
    (d) Test loss: 0.3039
3. Logistic Regression
    (a) Train accuracy: 89.5704
    (b) Test accuracy: 67.7509
    (c) Train loss: 0.1042
    (d) Test loss: 0.3224
4. Random Forest
    (a) Train accuracy: 97.2777
    (b) Test accuracy: 41.5825
    (c) Train loss: 0.0272
    (d) Test loss: 0.5841

Picking the Best Hyper Parameters

1. **SGD**: Hinge Loss was used as the loss function with an L2 penalization function. In order to get the best hyper parameters, the optimal regularized constant was determined through trial and error.

2. **Logistic Regression**: The hyper parameters were selected by a randomized search. The range of parameters were specified with a random state of 1 with no cross validation.
3. **Random Forest**: The default parameters were used. Increasing the randomness did not increase the training or test accuracy significantly. It was therefore assumed that the default parameters (provided by the library) was satisfiable.

Explain why you picked these 3 methods

1. **SGD**: It has been known to work well with natural language processing problems (e.g. the given data set, where text needs to be classified). It was expected to perform better than the baseline model as there were many parameters that could be tuned to optimize performance. It worked as expected: it had the highest performance among models based on the test set accuracy of 69%.
2. **Logistic Regression**: Logistic regression does not over fit for the data and the loss functions give it flexibility for penalization. Hence it seemed a good choice when correlated features are being used. The expectation was that it would work better than the base line model for the same settings (penalization, etc.) It works as expected as it did not assume independence among features given a class (the baseline predictor does).
3. **Random Forest**: Random forest models are good with complex classification tasks. The classification could involve taking advantage of the meaning behind text (natural language processing). It performed poorer than the baseline model. The Random forest had a high training accuracy which is expected (as it memorizes the data and has a tendency to over fit). However, this did not translate into test cases perhaps because the feature information was not continuous to provide for an obvious trend or perhaps due to the over fit.

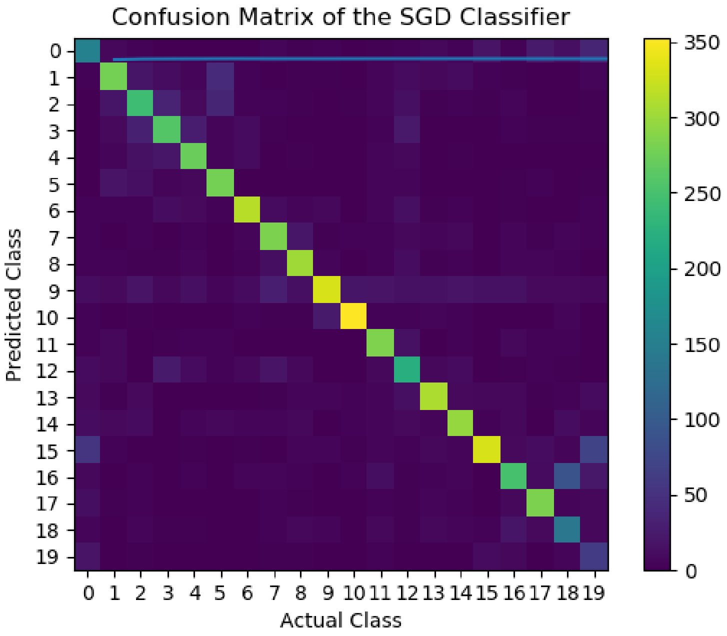
Best Classifier and Confusion Matrix

The best classifier was the **SGD classifier** with a test accuracy of 69% and a training accuracy of 95%.

The class with the most confusion (i.e. which had the highest number of false predictions associated with it) was **talk.religion.misc (feature 19)**.

The two classes that the classifier were most confused about were **talk.politics.guns (feature 16)** and **talk.politics.misc (feature 18)**.

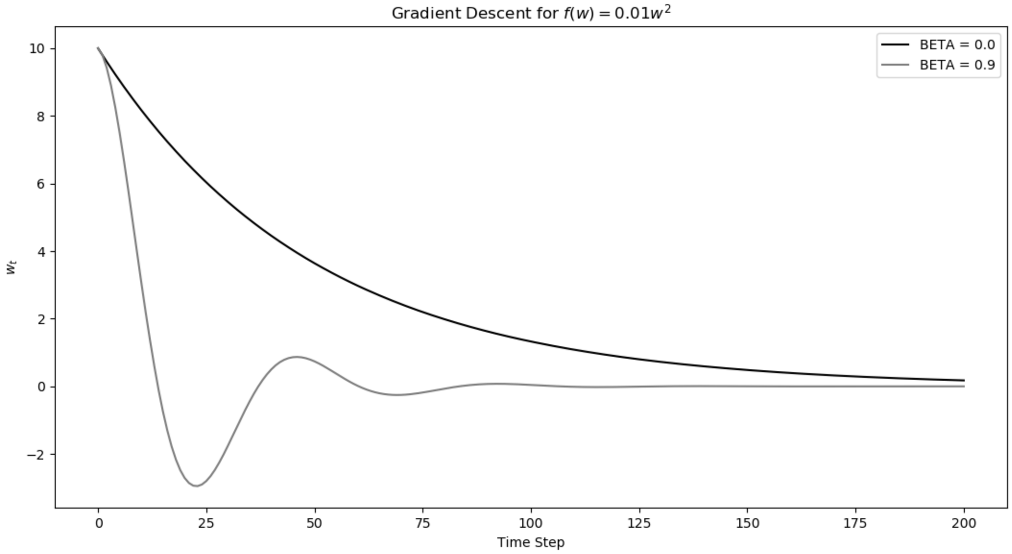
The confusion matrix is plotted below:



2 - Training SVM with SGD

2.1 - SGD with Momentum

Plotting w_t for each time step t by applying the iterative stochastic gradient-descent on $f(w) = 0.01w^2$, we get the following graph (for $\beta = 0.1$ and $\beta = 0.9$ for upto 200 time-steps):



2.2 -Training SVM

2.3 - Apply 4-vs-9 Digits on MNIST

Two SVM models were trained using gradient descent. Both models had the following properties:

1. A learning rate of $\alpha = 0.05$
2. A penalty of $C = 1.0$
3. Mini-batch sizes of $m = 100$ and $T = 500$

Model 1 and 2 had different β values.

1. **Model 1**: $\beta = 0$
2. **Model 2**: $\beta = 0.1$

For the first model use = 0 and for the second use = 0.1. For both of the trained models report the following:

2.3.1 - Training Loss

1. **Model 1** ($\beta = 0$) **Training Loss**: 0.3829
2. **Model 2** ($\beta = 0.1$) **Training Loss**: 0.3743

2.3.2 - Test Loss

1. **Model 1** ($\beta = 0$) **Test Loss**: 0.3748
2. **Model 2** ($\beta = 0.1$) **Test Loss**: 0.3672

2.3.3 - Classification Accuracy on the Training Set

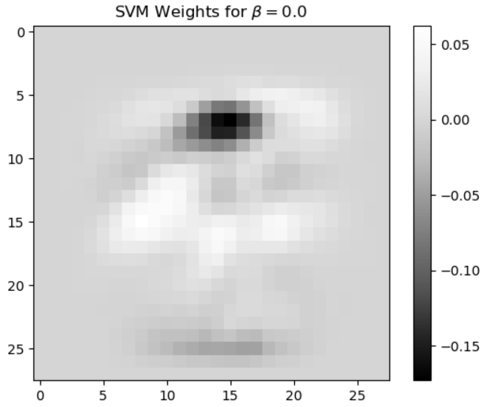
1. **Model 1** ($\beta = 0$) **Classification Accuracy on Training Set**: 0.9139
2. **Model 2** ($\beta = 0.1$) **Classification Accuracy on Training Set**: 0.9127

2.3.4 - Classification Accruacy on the Test Set

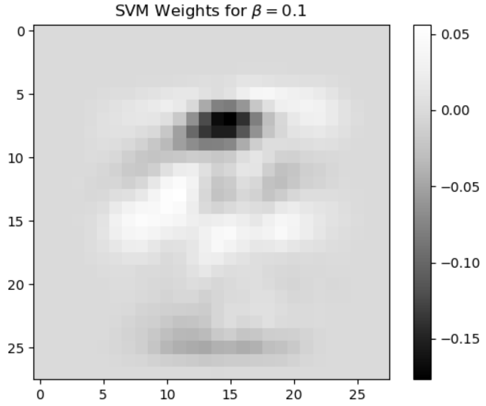
1. **Model 1** ($\beta = 0$) **Classification Accuracy on Test Set**: 0.9154
2. **Model 2** ($\beta = 0.1$) **Classification Accuracy on Test Set**: 0.9125

2.3.5 - Plot w as a 28 × 28 image

w for Model 1 ($\beta = 0$):



w for Model 2 ($\beta = 0.1$):



3 - Kernels

3.1 - Positive Semi definite and Quadratic Form

Prove that a symmetric matrix $K \in \mathbb{R}^{d \times d}$ is a positive semi definite iff for all vectors x we have $\mathbf{x}^T K \mathbf{x} \geq 0$.

Proof:

$$K\mathbf{x} = \lambda \mathbf{x}$$

where λ is the eigenvalue and $\mathbf{x}$ is the eigenvector.

Suppose $\mathbf{x}$ is an eigenvector of K and replacing $K\mathbf{v}$ with $\lambda\mathbf{v}$ (from the definition of an eigenvector and eigenvalue above):

$$\mathbf{x}^T K \mathbf{x} = \mathbf{x}^T \mathbf{x} \lambda$$

$$\mathbf{x}^T K \mathbf{x} = |\mathbf{x}|^2 \lambda$$

Therefore, as $|\mathbf{x}|^2 \geq 0$, for the equation $\mathbf{x}^T K \mathbf{x} \geq 0$, the eigenvalue must be: $\lambda \geq 0$. Since it is a semi-definite matrix, where the eigenvalues ≥ 0, this holds true.

3.2 - Kernel Properties

3.2.1 - Prove Property $k(\mathbf{x}, \mathbf{y}) = \alpha$ is a kernel for $\alpha > 0$

$$\phi(\mathbf{x}) = \sqrt{\alpha}$$

$$k(\mathbf{x}, \mathbf{y}) = \langle \phi(\mathbf{x}), \phi(\mathbf{y}) \rangle$$

$$k(\mathbf{x}, \mathbf{y}) = \sqrt{\alpha}\sqrt{\alpha}$$

$$k(\mathbf{x}, \mathbf{y}) = \alpha$$

3.2.2 - Prove Property $k(\mathbf{x}, \mathbf{y}) = f(\mathbf{x}) \cdot f(\mathbf{y})$ is a kernel for $f : \mathbb{R}^d \to \mathbb{R}$

$$\phi(\mathbf{x}) = f(\mathbf{x})$$

$$k(\mathbf{x}, \mathbf{y}) = \langle \phi(\mathbf{x}), \phi(\mathbf{y}) \rangle$$

$$k(\mathbf{x}, \mathbf{y}) = f(\mathbf{x}) \cdot f(\mathbf{y})$$

3.2.3 - Prove Property If $k_1(\mathbf{x}, \mathbf{y})$ and $k_2(\mathbf{x}, \mathbf{y})$ are kernels then $k(\mathbf{x}, \mathbf{y}) = a \cdot k_1((\mathbf{x}, \mathbf{y}) + b \cdot k_2((\mathbf{x}, \mathbf{y})$ for $a, b > 0$ is a kernel

Let $\mathbf{K}_1$ and $\mathbf{K}_2$ be gram matrices

Therefore:

$$\mathbf{x}^T\mathbf{K}_1\mathbf{x} \geq 0$$

and

$$\mathbf{x}^T\mathbf{K}_2\mathbf{x} \geq 0$$

Let:

$$\mathbf{K}_3(\mathbf{x}, \mathbf{y}) = a\mathbf{K}_1(\mathbf{x}, \mathbf{y}) + b\mathbf{K}_2(\mathbf{x}, \mathbf{y})$$

$$\mathbf{K}_3 = a\mathbf{K}_1 + b\mathbf{K}_2$$

$$\mathbf{x}^T\mathbf{K}_3\mathbf{x} = \mathbf{x}^T(a\mathbf{K}_1 + b\mathbf{K}_2)\mathbf{x}$$

$$\mathbf{x}^T\mathbf{K}_3\mathbf{x} = a\mathbf{x}^T\mathbf{K}_1\mathbf{x} + b\mathbf{x}^T\mathbf{K}_2\mathbf{x}$$

and using 3.2.1's proof, $\mathbf{K}_3(\mathbf{x}, \mathbf{y})$ is a kernel, we get:

$$a\mathbf{x}^T\mathbf{K}_1\mathbf{x} + b\mathbf{x}^T\mathbf{K}_2\mathbf{x} \geq 0 \text{ for } a, b > 0$$

3.2.4 - Prove Property If $k_1(\mathbf{x}, \mathbf{y})$ is a kernel then $k(\mathbf{x}, \mathbf{y}) = \frac{k_1(\mathbf{x}, \mathbf{y})}{\sqrt{k_1(\mathbf{x}, \mathbf{x})}\sqrt{k_1(\mathbf{y}, \mathbf{y})}}$ is a kernel

Let:

$$k_1(\mathbf{x}, \mathbf{y}) = \langle \phi(\mathbf{x}), \phi(\mathbf{y}) \rangle$$

$$\langle \phi(\mathbf{x}), \phi(\mathbf{y}) \rangle = \phi(\mathbf{x}) \cdot \phi(\mathbf{y})$$

In equation $k_1(\mathbf{x}, \mathbf{y})$, replacing all the instances of $k_1(\mathbf{x}, \mathbf{y})$ with $\phi(\mathbf{x}) \cdot \phi(\mathbf{y})$:

$$k(\mathbf{x}, \mathbf{y}) = \frac{k_1(\mathbf{x}, \mathbf{y})}{\sqrt{k_1(\mathbf{x}, \mathbf{x})}\sqrt{k_1(\mathbf{y}, \mathbf{y})}}$$

$$k(\mathbf{x}, \mathbf{y}) = \frac{\phi(\mathbf{x}) \cdot \phi(\mathbf{y})}{\sqrt{\phi(\mathbf{x}) \cdot \phi(\mathbf{x})}\sqrt{\phi(\mathbf{y}) \cdot \phi(\mathbf{y})}}$$

$$k(\mathbf{x}, \mathbf{y}) = \frac{\phi(\mathbf{x}) \cdot \phi(\mathbf{y})}{|\phi(\mathbf{x})||\phi(\mathbf{y})|}$$

$$k(\mathbf{x}, \mathbf{y}) = \frac{k_1(\mathbf{x}, \mathbf{y})}{|\phi(\mathbf{x})||\phi(\mathbf{y})|}$$

Let $\mathbf{K}_1$ for matrix $k_1(\mathbf{x}, \mathbf{y})$ be a gram matrix. Therefore:

$$\mathbf{x}^T\mathbf{K}_1\mathbf{x} \geq 0$$

Multiplying each entry of k_1 will result in a positive value. Therefore, $k(\mathbf{x}, \mathbf{y})$ is a kernel.