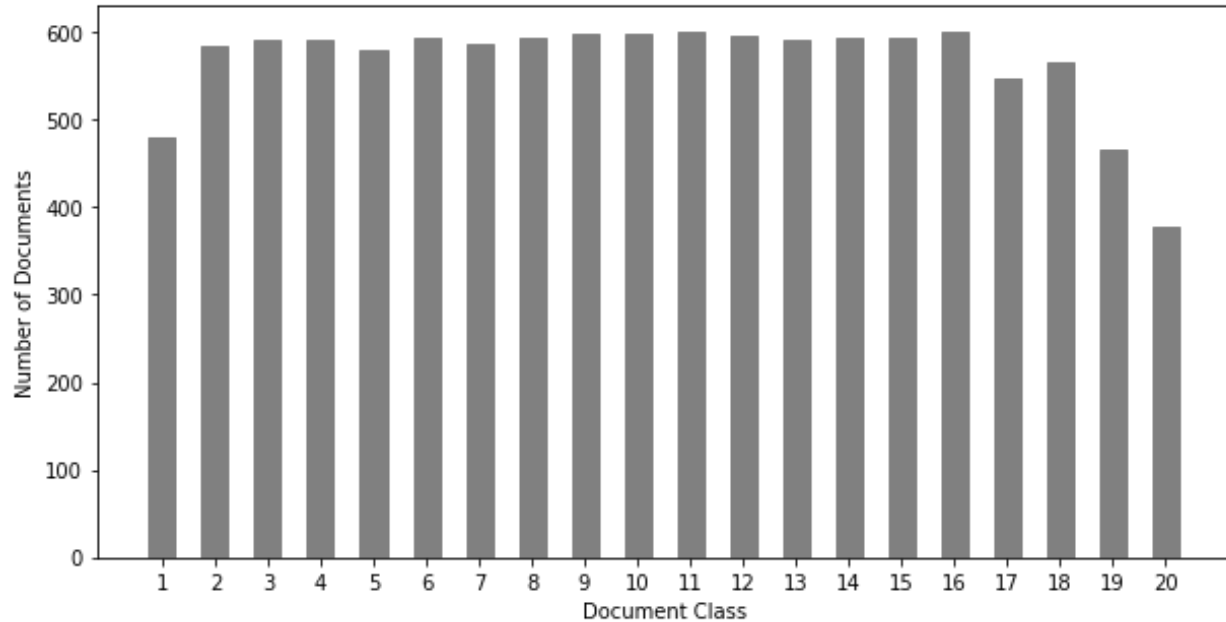


## ECE219 Project Report 1

Mahmoud Essalat, Sherry He, Weitang Sun, Siqu Huang  
ID: 005034839, 805040110, 904946260, 504490530

### Question 1.

The histogram of the numbers of training documents for each of the 20 categories is shown in the following. It is easy to see that the numbers of training documents are evenly distributed across categories.



I list the category label of each number in the following:

1 - alt.atheism, 2 - comp.graphics, 3 - comp.os.ms-windows.misc, 4 - comp.sys.ibm.pc.hardware, 5 - comp.sys.mac.hardware, 6 - comp.window.x, 7 - misc.forsale, 8 - rec.autos, 9 - rec.motorcycles, 10 - rec.sport.baseball, 11 - rec.sport.hockey, 12 - sci.crypt, 13 - sci.electronics, 14 - rec.sport.baseball, 15 - sci.space, 16 - soc.religion.christian, 17 - talk.politics.guns, 18 - talk.politics.mideast, 19 - talk.politics.misc, 20 - talk.religion.misc.

### Question 2.

X\_train.shape: (4732, 20297)

X\_test.shape: (3150, 20297)

### Question 3.

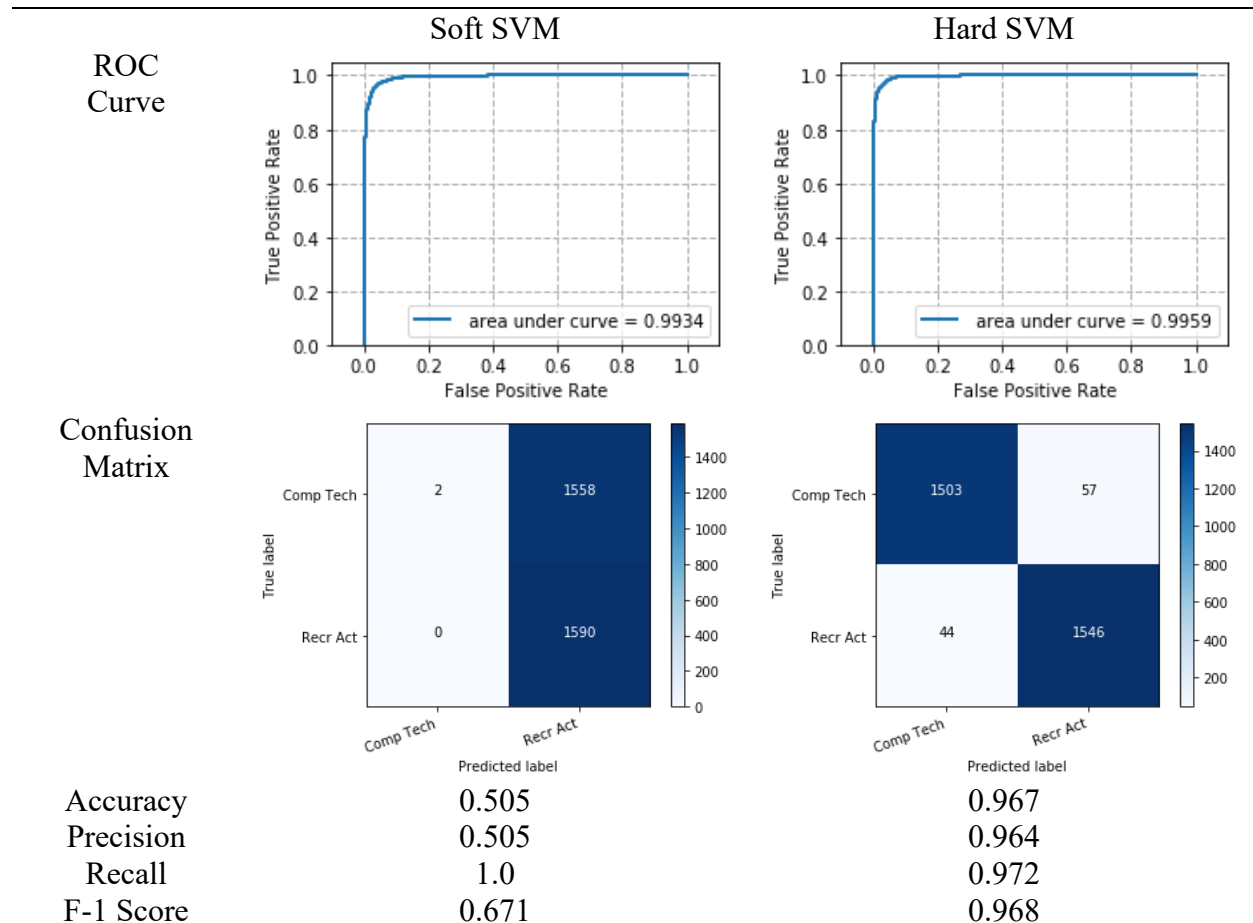
In NMF:  $\|X - WH\|_F^2 = 4200.36$

In LSI:  $\|X - U_k \Sigma_k V_k^T\|_F^2 = 4171.32$

The error of NMF is larger. It is because SVD finds the best k-rank approximation of the TFIDF matrix whereas NMF doesn't find the best k-rank approximation and it just find a k-rank approximation with non-negative matrix constraint. Therefore, SVD can better approximate the original matrix.

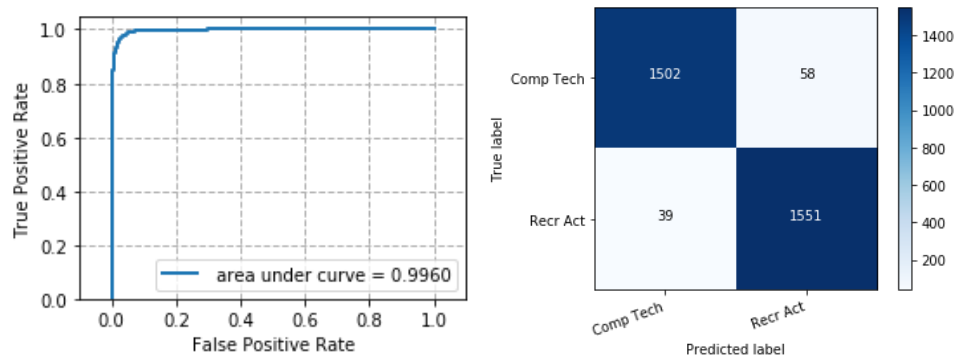
#### Question 4. SVM

ROC curve and confusion matrix for soft margin and hard margin SVM are reported in the following:



It is easy to see that hard margin SVM performs better. From the confusion table, soft margin SVM classify most data points to one class, so it performs poorly. It is because soft margin SVM doesn't regret much from putting the data points in the wrong class, as the extreme case  $\gamma = 0$  any hyper plane that goes through two data points of different class can be the solution. Thus, the coefficients we obtained from soft margin SVM can't classify the data points well. The extreme case would be

After using a 5-fold cross validation, we found the best value of the parameter  $\gamma$  is 100. For this best SVM, accuracy is 0.969, precision is 0.964, recall is 0.975, F-1 Score is 0.970. The ROC curve and confusion matrix are reported in the following.



### Question 5. Logistic Regression

I report the ROC curve and confusion matrix of logistic regression without regularization, with L1 regularization and best regularization strength, with L2 regularization and best regularization strength in the following. Accuracy, precision, recall, F-1 score and best regularization strength are reported in the table too.

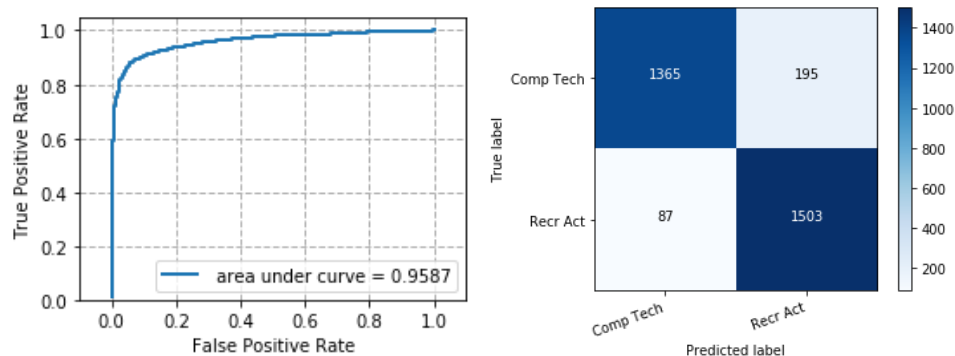
	without regulation	with L1 regulation	With L2 regulation
ROC curve			
Confusion Matrix			
Accuracy	0.9663	0.9664	0.9686
Precision	0.9603	0.9603	0.9616
Recall	0.9736	0.9736	0.9767
F-1 Score	0.9669	0.9689	0.9691
Best $k$	N/A	1000	100

Logistic regression with L2 regularization has the less errors than with L1 regularization. This is because L2 shrinks the coefficients more than L1. Regularization mitigated the overfitting problem and thus L1 and L2 have less test errors than the one without regularization. L1 and L2 are similar to Lasso and Ridge. L1/Lasso penalizes excessive number of parameters. It helps with picking up a few important variables and set the coefficients of unimportant variables to 0. L2/Ridge tries to pick up information from all the variables. L1 shrinks some coefficients to 0. L2 shrinks coefficients in general. L1 is suitable when we want to pick up a few important variables (a sparse vector). L2 is suitable when each variable contains small information.

Both logistic regression and linear SVM are trying to classify data points using a linear decision boundary (In fact SVM can perform nonlinear classification using kernel). Their performances are different because their loss function is different: SVM minimize hinge loss while logistic regression minimizes logistic loss. Logistic loss diverges faster than hinge loss. Thus Logistic classifier is more sensitive to outliers.

### Question 6. Naïve Bayes

The ROC curve and confusion matrix of GaussianNB classifier training is reported in the following. Accuracy is 0.910, Precision is 0.885, Recall is 0.945, F-1 Score is 0.914.

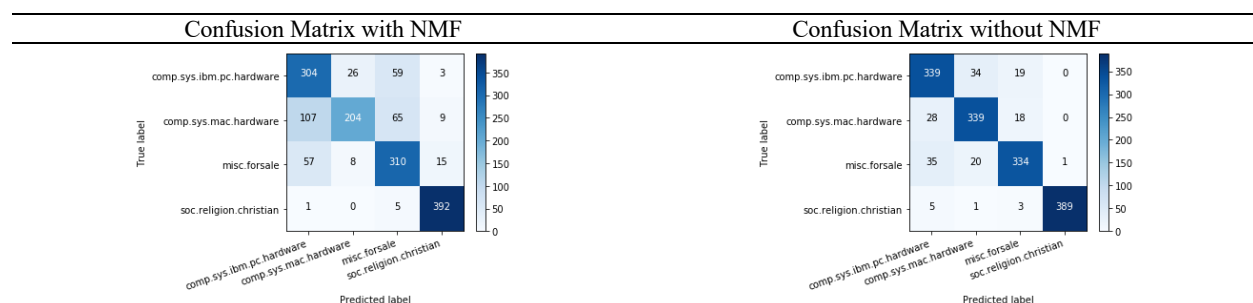


### Question 7. Pipeline and Grid Search

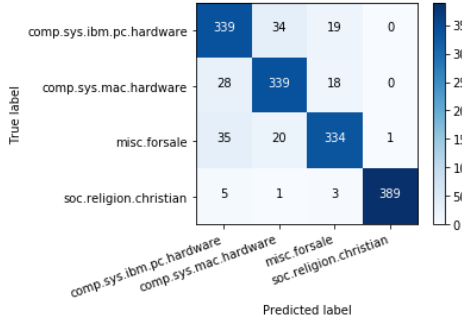
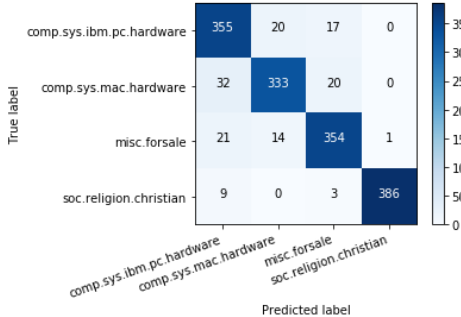
The best combination is SVM ( $\gamma = 100$ ), min\_df = 3, LSI, using lemmatization, with data removing the headers and footers. The average score is 0.96.

### Question 8. Multiclass classification

As we only have four categories here, we don't have as many words (variables) as in the 20 categories case. Our confusion matrix using NMF and without using NMF shows that, the classifying results are better with the original matrix, than with the reduced matrix. Therefore, we use the original matrix for the classification.



The confusion matrix and corresponding accuracy, recall, prevision and F-1 score of Naïve Bayes classification and Multiclass SVM classification are reported in the following

		Naïve Bayes (OvO)				SVM (OvO)					
Confusion Matrix											
True label	comp.sys.ibm.pc.hardware	339	34	19	0	True label	comp.sys.ibm.pc.hardware	355	20	17	0
	comp.sys.mac.hardware	28	339	18	0		comp.sys.mac.hardware	32	333	20	0
	misc.forsale	35	20	334	1		misc.forsale	21	14	354	1
	soc.religion.christian	5	1	3	389		soc.religion.christian	9	0	3	386
		Predicted label						Predicted label			
Accuracy		0.895						0.912			
Precision		0.833, 0.860, 0.893, 0.997,						0.851, 0.907, 0.898, 0.997			
Recall		0.865, 0.881, 0.856, 0.977						0.906, 0.865, 0.908, 0.970			
F-1 Score		0.849, 0.870, 0.874, 0.987						0.878, 0.886, 0.903, 0.983			