CS312 - Artificial Intelligence Lab

Assignment 6

Arun R Bhat (190010007), Mohammad Sameer (190010024)

1 Introduction

We are assigned the task of classifying emails as spam and non-spam using Support Vector Machines. We experimented thoroughly trying out multiple parameters and kernel functions in order to achieve our best results.

2 Dataset Description

Various features like frequency of occurrences of certain keywords, length of capitalized words etc. help us in classifying it as spam or not. The data set given to us contains 4601 samples. We are asked to divide it into two partitions - a training set (70%) and a test set (30%).

3 Libraries

We used `pandas` library for smooth working of our model. We imported `train_test_split` package from `sklearn.model_selection` library in order to split the dataset as per the requirements. We also used `StandardScaler` package from `sklearn.preprocessing` library for feature scaling (normalizing) the data. For implementing SVM, we took help from `SVC` package of `sklearn.svm` library. We calculated results using `precision_recall_fscore_support` and `accuracy_score` packages of `sklearn.metrics` library.

```
from sklearn.metrics import precision_recall_fscore_support, accuracy_score
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from IPython.display import clear_output
from sklearn.svm import SVC
import pandas as pd
```

3.1 Pandas

Pandas is a Python library that provides quick, versatile, and expressive data structures for working with"relational" or "labelled" data. Its goal is to serve as the foundation for undertaking realistic, real-world data analysis in Python. Furthermore, it aspires to be the most powerful and flexible open source data analysis and manipulation tool available in any language.

3.2 Scikit-Learn

Scikit-learn (or sklearn) is a free Python machine learning package. [3] It includes support vector machines, random forests, gradient boosting, k-means, and

DBSCAN, among other classification, regression, and clustering techniques, and is designed to work with the NumPy and SciPy python numerical and scientific libraries.

4 Details of the SVM Package used

We used `SVC` package of `sklearn.svm` library. It mainly used two parameters - regularization parameter C and function parameter *kernel* and trained the Support Vector to classify emails using the given data-set. It finds the hyperplane that separates spam and non-spam emails with maximum margin.

5 Kernel functions

5.1 Linear Kernel

When the data is linearly separable, that is, when it can be separated using a single line, the Linear Kernel is utilised. It is one of the most often adopted kernels. It is most generally used when a data set contains a rich set of features.

5.2 Polynomial Kernel

The polynomial kernel is a kernel function in that embodies the similarity of vectors (training samples) in a feature space over polynomials of the original variables, allowing learning of non-linear models. It is commonly used with support vector machines (SVMs) and other kernelized models. To estimate their similarity, the polynomial kernel looks not only at the specified attributes of input samples, but also at combinations of these. Such combinations are referred to as interaction features in regression analysis. A polynomial kernel's feature space is the same as that of polynomial regression, but without the combinational blowup in the number of parameters to learn. For binary inputs, the features correspond to their logical conjunctions.

5.3 RBF Kernel

Due to its resemblance to the Gaussian distribution, RBF kernels are the most generic type of kernelization and one of the most extensively used kernels. For any two points, the RBF kernel function computes their similarity, or how near they are to each other. The RBF Kernel is well-known because of its resemblance to the K-Nearest Neighbor Algorithm. Because RBF Kernel Support Vector Machines only need to store the support vectors during training and not the complete dataset, it has the advantages of K-NN and avoids the space complexity problem.

6 Results

We varied C and kernel in our experiments to obtain optimal results. Training set accuracy and testing set accuracy can be seen for various C values corresponding to each kernel. Let's consider test accuracy as our resulting metric. For linear model, train accuracy increases with C upto C = 5000 but test accuracy decreases

after C = 100. This shows optimal C is somewhere around 100. Similarly for other poly kernel, optimal C is found to be around 5000 and optimal C for RBF kernel is 5 when we saw test accuracy. But RBF model always outperformed other kernels.

	C_Value	Linear_Train	Linear_Test	Poly_Train	Poly_Test	RBF_Train	RBF_Test
0	0.000500	0.863354	0.847212	0.610559	0.608979	0.605590	0.606807
1	0.000100	0.728882	0.713251	0.607453	0.608255	0.605590	0.606807
2	0.005000	0.912112	0.904417	0.617081	0.615496	0.605590	0.606807
3	0.001000	0.885714	0.880521	0.612112	0.609703	0.605590	0.606807
4	0.050000	0.927640	0.920348	0.677329	0.666908	0.890683	0.888487
5	0.010000	0.918012	0.912382	0.633540	0.623461	0.722981	0.690080
6	0.500000	0.929814	0.921796	0.760248	0.725561	0.937578	0.929761
7	0.100000	0.929193	0.922520	0.702484	0.688631	0.911491	0.909486
8	1.000000	0.931056	0.924692	0.793789	0.753802	0.946584	0.940623
9	5.000000	0.932609	0.922520	0.865217	0.813179	0.959938	0.942795
10	10.000000	0.932919	0.923244	0.893168	0.841419	0.967702	0.937002
11	50.000000	0.933851	0.924692	0.950000	0.900072	0.981056	0.931933
12	100.000000	0.933540	0.925416	0.963043	0.916003	0.985714	0.929761
13	500.000000	0.934472	0.921796	0.981988	0.917451	0.992547	0.926140
14	1000.000000	0.933540	0.917451	0.986646	0.918175	0.993789	0.923244
15	5000.000000	0.930435	0.916003	0.993168	0.921796	0.995963	0.905141

Figure 1: Results with various parameters

7 Conclusion

RBF kernel showed much better accuracy as compared to other models when lower amount of regularization was applied. Still it is possible that few examples are misclassified as all data-sets contain outliers. So we can conclude that the most points of the data-set can be classified using RBF kernel after comparing different values of regularization parameter C across different kernels.