Multi-class SVM with RBF kernel for activity classification from UCI HAR dataset

Amitojdeep Singh Brar
School of Optometry and Vision Science and Department of Systems Design Engineering, University of Waterloo
Waterloo, Canada

***Abstract*—Human activity recognition using smartphone is generally performed on smartphones in real-time and with limited resources. SVM classifier performs comparably to neural networks while using a fraction of resources. This study proposes SVM with RBF kernel to classify daily activities from UCI HAR dataset. SVM with RBF kernel is superior to other variants of SVM due to universal kernel and only two hyperparameters - C and γ. The system obtains an accuracy of 96.53% making it better than linear SVM, K-NN & decision trees and comparable to state-of-the-art methods in literature.**

***Index Terms*—SVM, RBF, Kernel, C, Gamma, Grid search, HAR**

I. Introduction

Human activity recognition (HAR) is an emerging field which deals with detection of human activities in scenarios like sports, everyday activity, fall or even community violence. It can be performed using different modalities like images, videos, and sensors [1]. UCI HAR dataset has accelerometer and gyroscope signals from a smartphone for 6 activities performed by each of the 30 participants. In this study the use of SVM using RBF kernel for classifying this dataset is studies. A review of SVM with RBF is provided along with parameter search methods. The results of the proposed method are compared with the literature.

II. SVM Classifier

Support vector machine (SVM) is a classifier that forms optimal hyperplanes to classify data. It is used for problems in various domains like computer vision, text categorization, finance and signal processing. There are mainly two types of SVM depending on the decision boundary - linear and non-linear. A linear SVM can not fit non linear decision boundaries efficiently, while a non-linear SVM has high computational costs due to high dimensionality. Kernel SVM uses an implicit mapping to reduce the complexity while retaining the non-linearity. There are several non-linear kernels like polynomial which require deciding on parameters like complexity (quadratic, cubic and so on) in addition to the penalty parameter C. SVM with Radial Basis Function (RBF) kernel has only two parameters - C and γ and has several kernels like linear and sigmoid as its special cases. This paper review discusses the advantages of RBF kernel and the parameter search methods. Further details of the analysis are provided in the paper [2].

III. RBF kernel for SVM

RBF kernel is considered a universal kernel function and can be applied to any distribution with the correct choice of parameters. A radial basis function is a real-value function whose value depends on distance of the given point from a center (often the origin). Euclidean distance is the most common metric used for RBF. The RBF kernel is used for non-linear mapping of SVM commonly and the centers are automatically determined by SVM once C and γ are specified. RBF kernel function is expressed as: $K(x_i, x_j) = exp(-\gamma||x_i, x_j||)^2$.

The minimization problem of the SVM is:

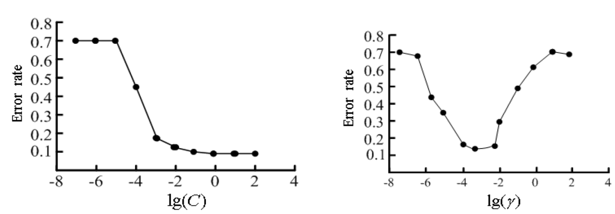
$$min_{\alpha_i} \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} y_i y_j \alpha_i \alpha_j exp(-\gamma||x_i, x_j||)^2 - \sum_{i=1}^{n} \alpha_i \quad (1)$$

such that, $\sum_{i=1}^{n} y_i \alpha_i = 0, 0 <= \alpha_i <= C$

The minimum value of the Lagrangian for SVM depends on choice of C add γ . So, the performance of the best classifier depends on their choice and hence it is important to optimize them.

IV. The effect of C and γ on performance on SVM with RBF

The parameter C in SVM with RBF kernel serves the same purpose as in linear SVM - to make the boundary softer and tolerate misclassification at the cost of complexity. The effect of adjusting the confidence range of C is varies in different data subspaces. On the other hand, γ changes the mapping function implicitly and the complexity of data distribution is the largest VC dimension. The kernel parameter (γ) and error punish factor (C) are the key influence over SVM performance as shown in figure 1 below [3].



(a) With the C changing trends of the error rate (b) With the γ changing trends of the error rate

Fig. 1. The effect of C and γ on the error rate **b1**

The error rate of a linear decreases rapidly with increase in C and then settles asymptotically such that there is no observable change in error on further increasing C. This flat part of the curve is where optimal value of γ can be searched by keeping C fixed. When the γ is varied linearly it is observed that certain values have lower error rates than values which are higher or lower than them.

The approach of finding a C and then varying γ as above is called the **double linear search method** and it is O(N) in complexity. A correction term δ can be introduced as it is dependent on C of the linear SVM, hence improving the parameter search.

Another common approach is **grid search method** which involves taking M values of C and N values of γ and then finding the error rates for MxN combinations. The grid search has higher accuracy but the complexity is $O(N^2)$. However, it can be parallelized as every SVM is trained independently. The search can be initially done with a few parameter and can be made more detailed with smaller steps in the region with low error. This approach is called **double grid search** and provides more precision while reducing the runtime.

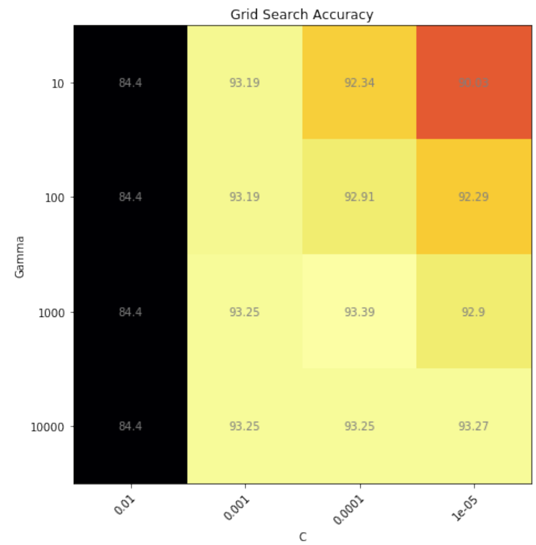


Fig. 2. The effect of C and γ on the error rate

V. Method

A. Implementation

In this project human activity classification task is performed on the UCI HAR dataset using SVM with RBF kernel. The parameters are optimized using 10 times 10 fold cross validation and the results are compared with other approaches including linear SVM, K-NN and decision trees. Since the dataset is from mobile device, computational efficiency is critical and hence the effect of PCA on the performance and run-time of each of the algorithms is also studied.

SVM with RBF kernel is an effective classifier as RBF can approximate several function including linear and sigmoid which are its special cases. Also, it is easier to train since it has only two parameters - C and γ. These can be searched using various methods as discussed in the earlier section. It is also faster than using feature extraction methods or more complex algorithms like neural networks with minimal difference in performance.

In this project, the SVM is implemented using sci-kit learn and the parameters are searched by using GridSearchCV method over 10 different sequences of the data with 10 fold cross validation.

B. Parameter search

A parameter search using loops over C and gamma was attempted but the run-time was very high (over 6 hours) due to lack of parallelization. The GridSearchCV function supports parallelization and brings down the runtime to 2 hrs using 8 threads on a system with 2.4 GHz (4.1 with turbo) quad core i5 processor and 16GB RAM.

In each of the 10 times, the data is shuffled and 4 values of gamma (0.01, 0.001, 0.0001, 0.00001) along with 4 values of C (10, 100, 1000, 10000) are used for each of the 10 folds. This leads to training for $10 \times 10 \times 4 \times 4 = 1600$ classifiers out of which the best one is selected.

VI. Observations

A. Training

The training accuracy of **98.94%** is obtained with **C = 100** and γ **= 0.001**. The corresponding model has an accuracy of **96.36%** on the test set. A slightly better set of parameters **C = 1000** and γ **= 0.0001** are obtained using 4 fold cross validation with a much faster run-time and a test accuracy of **96.53%**. The corresponding validation accuracy is 93.39%. The results from the grid search are plotted in the heatmap in Figure 2.

B. Test results

The best test accuracy of **96.53%** is achieved with C = 1000 and γ = 0.0001 as determined by grid search using 4 fold cross validation. This is comparable to the methods in the literature [4].The confusion matrix is as shown in Figure 3. The model performs quite effectively for most classes except some minor error for classifying between standing and sitting. Since, the smartphone with sensors is placed around the waist and stays in similar position and no motion during sitting and standing, it can make the signals similar. This problem can be solved by considering the change of signals temporally and keeping the transition between activities like sitting and standing in memory.

VII. Results

The proposed model is benchmarked against decision tree, K-NN (n=6), linear SVM and two of the studies in literature that used SVM on the same dataset. The results are as shown in the table I. The proposed model is able to outperform other standard classifiers as well as the original publication that used this data [5]. It performs very competitively to the recent study by Jain et. al. [4] which used SVM with descriptor based approach. However, neural network approaches like Stacked Autoencoders provide much better performance at the cost of added complexity and computational requirements.

TABLE I
ACCURACY OF VARIOUS APPROACHES FOR UCI HAR DATASET CLASSIFICATION

Activity	Proposed	Linear SVM	K-NN	DT	Jain [4]	Anguita [5]	Badem [6]
Walking	98.18%	98.13%	84.31%	86.87%	-	-	-
Upstairs	95.63%	94.32%	89.38%	82.88%	-	-	-
Downstairs	97.83%	95.41%	94.24%	82.59%	-	-	-
Sitting	95.72%	94.94%	88.35%	56.99%	-	-	-
Standing	92.25%	93.21%	81.98%	79.68%	-	-	-
Laying	100.00%	99.63%	99.61%	100.00%	-	-	-
Total	96.53%	95.94%	88.56%	81.50%	97.12%	96.33%	**97.90%**

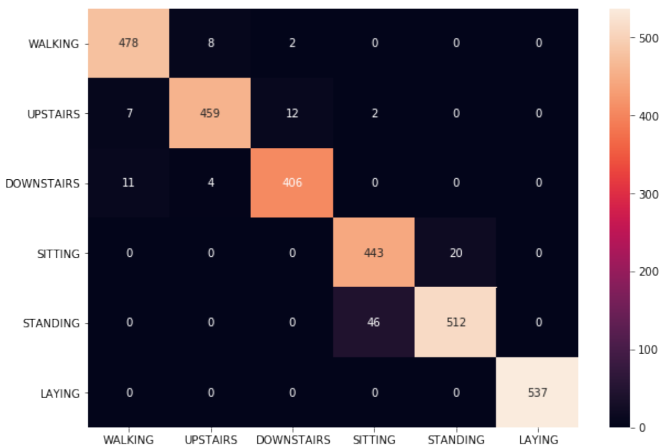


Fig. 3. The confusion matrix for SVM with RBF kernel - true labels on Y axis and predicted on X axis

TABLE II
AFFECT OF PCA WITH 20 DIMENSIONS ON VARIOUS APPROACHES FOR UCI HAR DATASET CLASSIFICATION

Measure	SVM RBF	Linear SVM	K-NN	DT
Accuracy	96.53%	95.94%	88.56%	81.50%
Accuracy PCA	88.63%	87.61%	84.49%	77.43%
Decrease	7.90%	8.33%	4.07%	4.07%
Test time (ms)	1,960	1,440	16,400	9
Test time PCA (ms)	241	104	266	6
Speedup	8.13×	13.84×	61.65×	1.5×

The table II shows the effect of using PCA to reduce the dimensionality to include the top 20 Eigenvectors (77.37% explained variance). The principal components are computed on the training set while the test time and the accuracy are for the test set. The proposed SVM with RBF approach gains 8.13× speed up at the cost of 7.9% accuracy decrease. PCA benefits computationally expensive methods like K-NN more while simpler methods like decision trees don't seem to benefit from speedup.

VIII. CONCLUSION

The presented model using SVM with RBF kernel achieves suitable performance comparable to state-of-the-art methods while keeping complexity lower than neural networks. It can be deployed on board devices like smartphones and smartwatches for activity detection.

In future works, efficient feature extraction techniques can be used to help improve the results and speed up the model. Other directions of research can include reduction in model complexity for improved on device performance or use of techniques like deep learning if computational constraints are lower. Tasks like fall detection and sports activity detection can also be explored where the sequence of signals is needed for a longer window of time.

REFERENCES

[1] S. K. Yadav, A Singh, A Gupta, *et al.*, "Real-time yoga recognition using deep learning," *Neural Computing and Applications*, vol. 31, no. 12, pp. 1–13, 2019.

[2] S. Han, C. Qubo, and H. Meng, "Parameter selection in svm with rbf kernel function," in *World Automation Congress 2012*, IEEE, 2012, pp. 1–4.

[3] B. E. Boser, I. M. Guyon, and V. N. Vapnik, "A training algorithm for optimal margin classifiers," in *Proceedings of the fifth annual workshop on Computational learning theory*, 1992, pp. 144–152.

[4] A. Jain and V. Kanhangad, "Human activity classification in smartphones using accelerometer and gyroscope sensors," *IEEE Sensors Journal*, vol. 18, no. 3, pp. 1169–1177, 2017.

[5] D. Anguita, A. Ghio, L. Oneto, *et al.*, "A public domain dataset for human activity recognition using smartphones.," in *Esann*, 2013.

[6] H. Badem, A. Caliskan, A. Basturk, *et al.*, "Classification of human activity by using a stacked autoencoder," in *2016 Medical Technologies National Congress (TIPTEKNO)*, IEEE, 2016, pp. 1–4.

[7] R. E. Schapire, Y. Freund, P. Bartlett, *et al.*, "Boosting the margin: A new explanation for the effectiveness of voting methods," *The annals of statistics*, vol. 26, no. 5, pp. 1651–1686, 1998.

[8] F. Pedregosa, G. Varoquaux, A. Gramfort, *et al.*, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.