Effect of Different Kernels on the Performance of an SVM Based Classification

Deepika Kancherla, Jyostna Devi Bodapati, Veeranjaneyulu N

Abstract--- According to the literature Support Vector Machines (SVM) is one of the robust classification models which guarantees reasonable per-formance even with small training datasets. Though the deep learning models are able to produce the state of the art performance large volumes of training data is required to achieve that. SVMs are basically designed to be binary classifiers and can be extended to multiple classes that are very common in many real world applications. In this paper we are trying to prove that generalization ability of support vector machines (SVM's) is good on difficult real world problems. We also try to analyze the effect of different features and different types of kernels on their performance. For the illustrations we have used different types of features like gist, HOG, histogram. In this work we show how the types of features extracted from the data can affect the performance of the classifier. The original version of SVMs is designed for linear classification tasks which can be applied to non-linear classification by projecting the data into a non-linear space using kernel trick. In this paper we even try to analyze the effect of kernels like linear, polynomial, Gaussian, sigmoidal and user defined kernels and how the type of kernel effect the performance of the support vector machine based classification task. Based on the studies we have conducted, it is observed that type of features and type of kernels used have a great impact on the performance of an SVM based classification task. Type of the features we can use is solely dependent on the problem on hand. On the other side impact of the kernel is dependent on the data set. Our Studies show that RBF kernel and histogram intersection kernel leads to better performance than others.

Keywords: Histogram Intersection Kernel; Kernel trick; SVM; Types of Kernels; User-defined kernels

INTRODUCTION

With the advent of Machine learning there are many advances in the human life in the recent past. Machine learning applications have the capability to transform the world to an unbelievable state in near future too. As per the literature Machine learning is a field of computer science where the computer systems must be able to learn based on the given data without much of the programming efforts. Pattern recognition is a subject that concentrates on the recognition of patterns and regularities in the data. Classification is the process of identifying the label of the given object from the available set of labels. Emotion recognition, speaker recognition are few applications of the classifications task to mention a few. For humans classifying a given object like pen, pencil, toy, book etc., is very obvious. Actually humans can process more complex data. The applications of classification even include brain tumor detection and classification [1]. Many techniques using different kinds of features can be used for this classification task [2] [3]. Face Recognition is yet another important application that can be realized using Classification [4]. Another important application is scene classification task, in which given an image that contains a picture of the scene such as mountain, forest, city, street etc.; the model has to assign it one of the scene types [5]. Most of the classification models are able to classify the data that is linearly separable. Linear separable classes are those in which data is mutually exclusive and there is no overlapping between classes. These problems are very trivial and can be easily solved but most of the real world problems are non-linear in nature. SVM has the ability to model the non-linearly separable data by using kernel trick.

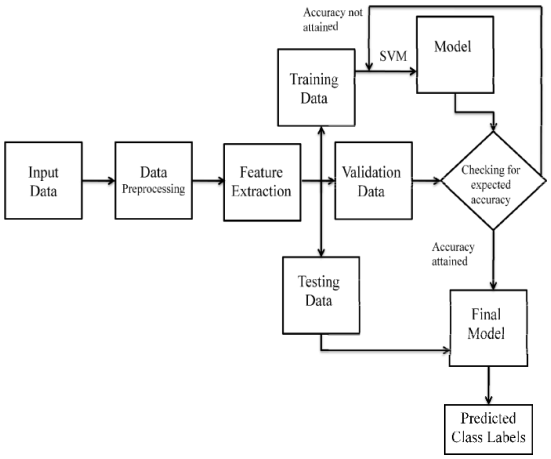

Fig. 1: Flow diagram of an SVM Classification task

In most of the real world problems like scene classification and place retrieval are more challenging as they are non-linearly separable. Our experiments prove that the SVM with non-linear kernels is suitable for scene classification and exhibits very good generalization ability.

General Architecture of an SVM classification task: Following figure shows the general sequence of operations that are to be followed in any of the classification task. Broadly speaking any classification task comprises of the tasks like: Acquire the data, Preprocess the data, Feature extraction, Normalization, partition the data into training, validation and test data sets. Data in the training set along with the labels is passed to the training model. Model parameters are tuned as long as the intended accuracy is obtained on validation data. Once the model is stable performance of the test data is analyzed using that model.

Revised Manuscript Received on February 11 , 2019.

Deepika Kancherla, Student, CSE Dept, Vignan's University, Guntur, Ap, India.
Jyostna Devi Bodapati, Asssistant Professor, CSE Dept, Vignan's University, AP, India.(jyostna.bodapati82@gmail.com)
Veeranjaneyulu N, Professor, IT Dept, Vignan's University, AP, India.

Accuracy of the model is considered as the performance of that model.

SUPPORT VECTOR MACHINES BASED CLASSIFICATION MODEL

SVM is one of the robust classification models with good generalization ability on unseen data. If the data is linearly separable it is possible to draw infinite number of decision boundaries that separates the data. On applying any linear classification model like perceptron the model returns one of these decision boundaries. Support Vector Machine is a binary classifier that works on the concept of identifying the separating hyper plane with maximum margin between the two classes.

If the data is two-dimensional then the decision boundary returned by SVM is a line and the decision boundary is a plane in case of three-dimensional data. The decision boundary is a hyper plane for the data with 4 or more dimensions. Objective of SVM is to find the decision boundary with the maximum margin. Margin can be formally defined as the distance from the nearest training example to the decision boundary.

The basic SVM classifier version is a two class classifier which can be used to classify data of the two given classes. SVMs can be easily adapted multiple classes by using either the one vs. rest or one vs. one classification techniques. For the classification task which involves multiple classes, the one vs. one classification method leads to many binary classifiers compared to one vs. rest. So in our studies we have used one vs. rest method to classify multiple classes using SVM.

The basic SVM classifier can result a linear classifier which can be used to separate a linearly separable data. SVMs can be extended to draw a non-linear decision boundary by transforming the input from its original space to a high dimensional space. As the relation between the input space and transformed space is non-linear in nature, the objective is to obtain a non-linear decision boundary.

The functions used for transforming data to a non-linear space are called as Kernel functions. The objective of SVM is to determine optimal separating hyper plane. Other models like perceptron, simply finds one of the possible hyper planes. Optimal hyper plane is the one with the maximum distance from the margins. This optimal hyper plane guarantees to give better generalization rate than others on unseen patterns. Given the training data $\{x_i, y_i\}_{i=1}^{n}$ where $\{x_i\}_{i=1}^{n}$ is the data and $y_i \in \{-1,1\}$ is the associated label for x_i, the objective of SVM is to identify the hyper plane that can separate the data with maximum distance. This hyper plane can be defined by the equation $W^T X + b$, which has maximum distance from both the classes. The data points for which $W^T X + b \geq 1$ are expected to be belonging to the positive class (class1) and are classified as positive class examples. The set of data points, for which $W^T X + b \leq -1$ are expected to belong to the negative class and are classified as negative class (class2) data points. The set of data points, for which $W^T X + b = 0$ are expected to be within the margin. The optimizing function is designed such that the distance between the two hyper-planes is maximum, which is a maximization problem given by $\frac{1}{\|w\|}$.

The same can be posed as minimization problem as minimize $\|w\|$ subject to the constraint that : $y_i({w_i}^T x_i + w_0) \geq 1, \forall i$

The objective function of SVM can be formulated as: $L_p = \frac{1}{2}\|w\|^2 - \sum_{i=1}^{n} \alpha_i \{y_i({w_i}^T x_i + w_0) - 1\}$

By getting the dual of the above equation, the solution to the optimization problem is computed easily. However, complications arise when the data is linearly non-separable or overlapping. Data representing most of the real time applications is not that simple, and is often require more complex boundaries to accurately classify the unseen data points (test cases) on the basis of the examples that are available (train cases). The degree of non-linearity of the decision boundary depends on the transformation function that is being applied.

SVMs are proved to be very robust and efficient on the data that is non-linearly separable and also on overlapping data. To address this, a slack variable is introduced ξ_i. The above optimization problem can be represented as follows: $\frac{\|w\|^2}{2} + C(\sum_i \varepsilon_i)^k$ subject to the constraint that $(w^T x_i + w_0) \geq 1 - \varepsilon_i, \forall y_i = 1\ (w^T x_i + w_0) \geq -1 + \varepsilon_i, \forall y_i = -1$ and $\varepsilon_i \geq 0$

The above equation is in the primal form and can be easily represented as an equivalent dual form $\sum \alpha_i - (1/2) * \sum \alpha_i \sum \alpha_j y_i y_j {x_i}^T x_j$ subject to $0 \leq \alpha_i \leq C$ and $\sum \alpha_i y_i = 0$. In this equation, ${x_i}^T x_j$ is termed as the kernel function and can be represented as $K(x_i, x_j)$. The term $K(x_i, x_j)$ represents the similarity value between x_i and x_j.

$K(x_i, x_j) = \emptyset(x_i)^T \emptyset(x_j)$ where $\emptyset(x_i)$ is the representation of x_i, in the high dimensional space.

KERNEL FUNCTIONS

The objective is to obtain a separating plane that separates the data such that the data belonging to different classes fall on either sides of the separating plane. A schematic example is shown in the illustration below.

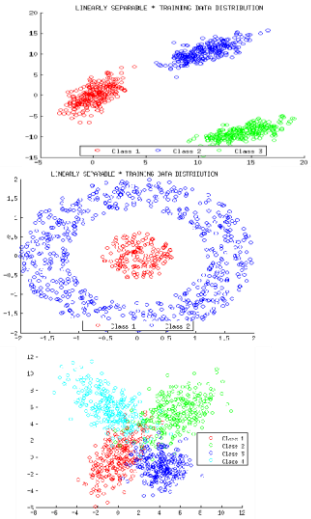

Fig. 2: (a) Linearly separable (b) non-linearly separable (c) overlapping data

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

Based on the complexity of classification task data can be broadly classified into: linearly separable, linearly non-separable and overlapping. If the data of two classes can be separated using linear boundary then such data is called as linearly separable. If the data can't be separated by a linear boundary and can be separated using a non-linear boundary then such data is called as non-linearly separable data. If the data can neither separated by a linear boundary and nor separated by a non-linear boundary, such data is called as overlapping data. In simple terms overlapping data is the one which overlaps in the original data. Fig 2 represents three different types of data in two dimensional space. Fig 2.a shows the data that can be linearly separable where the data can be separable easily with the help a planes. Fig 2.b shows the data that cannot be separable by a plane but can be separable by a non-linear structure. In this example an oval is sufficient to separate the data of the classes. Fig 2.c shows the data that is overlapping and is neither separable by a plane nor separable by a non-linear shape. But the data may be separable in the higher dimensional space either by a plane or by a non-linear shape. Challenging part is to figure out the type of suitable transformation. Fig 3 shows an example that shows projection of data from original space to transformed space. This transformation is a non-linear transformation and the space is higher dimensional. In the original space data is non-linearly separable and after projecting it a different space it can be separable by a plane.

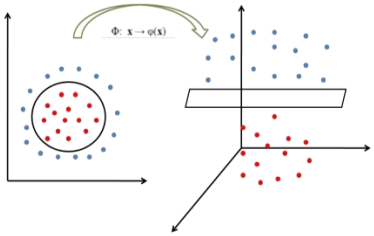

Fig. 3: Non-linear Transformation

In the literature there are different types of kernels to achieve this task. Following are the different kernels that are used for the task of classification.

Kernel Type	$K(X_i, X_j)$	Remarks
Linear	$(X_i^T . X_j)$	Linear kernel and used on the data if the class boundaries are linear
Polynomial	$(X_i . X_j + 1)^h$	Non-linear kernels and used on the data if the class boundaries are non-linear or overlapping
Radial Basis (RBF)	$e^{-\|X_i - X_j\|^2 / 2\sigma^2}$	
Sigmoidal	$tanh(kX_i . X_j - \delta)$	

Table 1: Types of widely used kernels

FEATURES

GIST Features: These are global image features and they assist in characterizing various important statistics of a scene. These features are computed by convoluting the filter with an image at different scales and orientations. Thus, high and low frequency repetitive gradient directions of an image can be measured. The scores for filter convolution at each orientation and scale are used as Gist features for an image. These features are currently being used for scene classification.

HOG Features: HOG breaks up an image into small cells, computes the HOGs for each cell, normalizes HOGs using block pat-tern, and provides a descriptor for each cell. These features are generally used for object detection in an image. The basic idea behind HOG features is that shape and appearance of local objects within an image can be described by the intensity gradients distribution. This method involves counting the occurrence of gradient orientation and thereby maintains photo metric transformations and geometric invariance. The descriptor generation is comprised of four main steps: gradient computation, orientation binning, descriptor blocks generation, and block normalization.

Histogram Features: An image histogram is a type of histogram that acts as a graphical representation of the tonal distribution in a digital image. It plots the number of pixels for each tonal value. By looking at the histogram for a specific image a viewer will be able to judge the entire tonal distribution at a glance.

These are the features that were used in our work. Many other features exist which can be used efficiently for various applica-tions like the Local Binary Pattern Features [6] [7]. Apart from feature extraction, features can also be r educed. Feature Reduction is performed to ensure that the unimportant features do not affect the performance of the classifier [8].

PROPOSED WORK

In addition to the kernels listed in Table.1, Histogram intersection kernel (HIK) proved to be very efficient for the classification tasks.

Let x and y be two different images and h_x and h_y be the histograms of x and y respectively with d bins. The Histogram intersection kernel on the data points x and y can be computed as:

$$K(h_x, h_y) = \sum_{j=1}^{d} \min(h_x(j), h_y(j))$$

Kernel gram matrix: Kernel gram matrix is an n x n matrix that helps to select the kernel that better classifies the data. In the kernel gram matrix each cell with indices i, j gives the similarity value between i^{th} and j^{th} data points. The ideal kernel matrix must have higher values in the diagonal positions and low values as we move towards off-the-diagonal. It gives an idea of how the block diagonal matrix looks like for the different values of kernel parameters. We will select specific parameters of kernel which has non-zero values along the block diagonals.

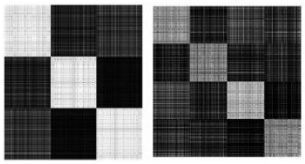

Fig. 4: Sample kernel gram matrices for 3-class and 4-class problems

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

In this work we have applied SVM on various datasets to verify the effect of the kernel on classification accuracy and also to verify the effect of type of features on the accuracy.

EXPERIMENTAL RESULTS

Following are the different data sets used for our experimental results:

Dataset	No.of examples	Number of classes	Description
ORL	400	40	Facial data
MIT	2688	8	Scene image data
Yale	165	15	Facial data
Brain Tumor	216	2	Brain Tumor
MNIST	5000	10	Digit recognition

Table 2: Summary of proposed datasets

a. Face Recognition: For the application of Face Recognition using ORL datasets high accuracy has been obtained for any kernel using the Nu-SVC and the best features that work for this data have been identified as GIST and Histogram Features. Table 3 and Table 4 show the results of face recognition task using ORL and Yale datasets respectively.

For Face Recognition using Yale dataset 100% accuracy has been obtained using any kernel by taking into consideration the GIST Features.

b. Scene Recognition: For this task MIT Dataset is used. For the Scene Recognition Application the best results are obtained using the combination of GIST Features with the Radial Basis Function Kernel when compared to the others for the MIT Dataset. Table 5 shows performance of the MIT dataset using different kernels.

c. Tumor Detection: For this task we have opted Brain Tumor dataset. The highest accuracy for Brain Tumor Detection has been obtained using HOG and Histogram Features when compared to the GIST Features. The use of any kernel with any type of SVC yields the same result. Table 6 shows performance of the tumor dataset using different kernels.

Features	Type of SVM	Type of kernel				
		Linear	Polynomial	Sigmoid	RBF	HIK
GIST Features	C-SVC	91.25%	91.25%	91.88%	95.00%	NA
	Nu-SVC	98.13%	98.75%	98.75%	98.75%	NA
HOG Features	C-SVC	93.75%	82.50%	82.50%	93.13%	NA
	Nu-SVC	93.75%	93.75%	94.38%	93.75%	NA
Histogram Features	C-SVC	NA	NA	NA	NA	98.75%
	Nu-SVC	NA	NA	NA	NA	98.75%

Table 3: Performance of different kernels on ORL dataset

Features	**Type of SVM**	**Type of kernel**				
		Linear	**Polynomial**	**Sigmoid**	**RBF**	**HIK**
GIST Features	C-SVC	97.33%	97.33%	97.33%	97.33%	NA
	Nu-SVC	100.00%	100.00%	100.00%	100.00%	NA
HOG Features	C-SVC	89.33%	85.33%	85.33%	89.33%	NA
	Nu-SVC	89.33%	93.33%	89.33%	93.33%	NA
Histogram Features	C-SVC	NA	NA	NA	NA	82.67%
	Nu-SVC	NA	NA	NA	NA	82.67%

Table 4: Performance of different kernels on Yale dataset

Features	Type of SVM	Type of kernel				
		Linear	Polynomial	Sigmoid	RBF	HIK
GIST Features	C-SVC	78.56%	58.53%	68.28%	82.06%	NA
	Nu-SVC	79.44%	78.56%	79.10%	82.19%	NA
HOG Features	C-SVC	69.49%	47.04%	50.54%	70.16%	NA
	Nu-SVC	69.49%	69.76%	69.96%	70.09%	NA
Histogram Features	C-SVC	NA	NA	NA	NA	22.65%
	Nu-SVC	NA	NA	NA	NA	26.14%

Table 5: Performance of different kernels on MIT dataset

Features	Type of SVM	Type of kernel				
		Linear	Polynomial	Sigmoid	RBF	HIK
GIST Features	C-SVC	98.61%	98.61%	98.61%	98.61%	NA
	Nu-SVC	100.00%	100.00%	100.00%	100.00%	NA
HOG Features	C-SVC	100.00%	100.00%	100.00%	100.00%	NA
	Nu-SVC	100.00%	100.00%	100.00%	100.00%	NA
Histogram Features	C-SVC	NA	NA	NA	NA	100.00%
	Nu-SVC	NA	NA	NA	NA	100.00%

Table 6: Performance of different kernels on Tumor dataset

Features	Type of SVM	Type of kernel				
		Linear	Polynomial	Sigmoid	RBF	HIK
GIST Features	C-SVC	92.40%	97.20%	72.90%	97.20%	NA
	Nu-SVC	96.20%	95.90%	95.00%	97.50%	NA
HOG Features	C-SVC	96.30%	95.30%	85.80%	96.700%	NA
	Nu-SVC	95.70%	96.00%	95.60%	95.70%	NA
Histogram Features	C-SVC	NA	NA	NA	NA	22.40%
	Nu-SVC	NA	NA	NA	NA	23.40%

Table 7: Performance of different kernels on MNIST dataset

d. Digit Recognition: MNIST dataset is used for this task. For digit recognition using MNIST dataset almost all the features with any kind of kernel yield similar kind of results except for the Histogram features used in combination with the Histogram Intersection Kernel. The highest accuracy amongst all has been obtained using GIST Features with the Radial Basis Function Kernel. Table 7 shows performance of the MNIST digit dataset using different kernels.

On a general note even for a specific application there doesn't exist a specific kernel or a specific kind of feature that will always yield the best result. The outcome of any application is always subject to the kind of data but doesn't depend entirely on the kind of features or the type of kernel being used.

CONCLUSION

After analyzing our experimental studies, we conclude that among the other features gist gives best results and Nu-SVC with RBF kernel outperforms other kernels. Difference between C-SVC and Nu-SVC is it is to set the Nu value in case of Nu-SVC because its value is bounded between 0 and 1.

FUTURE SCOPE

These days every application has a scope for improvement with deep neural network. As a future extension to this work, we propose to use deep features instead of the handcrafted features and we are also planning to analyze the performance of different models with deep features.

REFERENCES

1. Shil, SK.et al., "An improved brain tumor detection and classification mechanism", In proceedings of IEEE conference on *Information and Communication Technology Convergence (ICTC), 2017*.
2. Zhang et al, "A novel method for magnetic resonance brain image classification based on adaptive chaotic PSO", *Progress In Electromagnetics Research* 109 (2010): 325-343.
3. Islam et al, "A new hybrid approach for brain tumor classification using BWT-KSVM", In Proceedings of *Advances in Electrical Engineering*, 2017.
4. Jyostna Devi Bodapati et al, "A novel face recognition system based on combining eigenfaces with fisher faces using wavelets", *Procedia Computer Science*2 (2010): 44-51.
5. Jyostna devi Bodapati et al, "SCENE CLASSIFICATION USING SUPPORT VECTOR MACHINES WITH LDA", *Journal of Theoretical & Applied Information Technology. 2014 May 31;*63(3).
6. Harris, Samuel, et al. "LBP features for hand-held ground penetrating radar." *Detection and Sensing of Mines, Explosive Objects, and Obscured Targets XXII*. Vol. 10182. International Society for Optics and Photonics, 2017.
7. Pradhan, Debasish "Enhancing LBP Features for Object Recognition using Spatial Pyramid Kernel." (2017).
8. Jyostna devi Bodapati et al, "An intelligent authentication system using wavelet fusion of K-PCA, R-LDA", *IEEE International Conference on Communication Control and Computing Technologies (ICCCCT), 437-441, 2010,*