Data Mining and Warehousing Assignment 1

Indian Institute of Information Technology, Allahabad

IIT2018176	IIT2018178	IIT2018179	IIT2018202
Milan Bhuva	Manav Agarwal	Mohammed Aadil	Ankit Rauniyar

Abstract: Support vector machines (SVMs) appeared in the early nineties as optimal margin classifiers and since then they've been consistently used to solve real life problems often providing improved results as compared to other classification algorithms.

SVM is a supervised machine learning algorithm that can be used for both classification and regression. However, it is mostly used in classification problems. SVM works on the concept of finding the optimal hyperplane that best classifies the given data. An optimal hyperplane is the one that maximizes margins from both classes. SVMs can be used against both linear and non-linear data. SVM makes use of tuning parameters namely Kernels, Regularization, Gamma, and Margin. These parameters can be tuned in order to receive high accuracy.

I. INTRODUCTION

Basically, support vector machines are a way of classifying points by building separating hyperplanes between them. Kernels determine how SVMs work because they allow the SVM to consider candidates separating hyperplanes in very high dimensions. A **kernel** is a specialized kind of similarity function. It takes two points as input, and returns their similarity as output, just as a similarity metric does.

This is what our assignment mainly focuses on. We, as a group, were given the task to classify and compute the accuracy of KT_SVM (K-Times Markov Sampling) performed on two datasets: PASCAL VOC 2012 and Letter Recognition. It is important to get a grip of Markov Chains before moving on. Markov Chain Monte Carlo sampling provides a class of algorithms for systematic random sampling from high-dimensional probability distributions. Unlike Monte Carlo sampling methods that are able to draw independent samples from the distribution, Markov Chain Monte Carlo methods draw samples where the next sample is dependent on the existing sample, called a Markov Chain. This allows the algorithms to narrow in on the quantity that is being approximated from the distribution, even with a large number of random variables. K times is a small change from the original MCMC sampling where the MCMC Algorithm is run only k times.

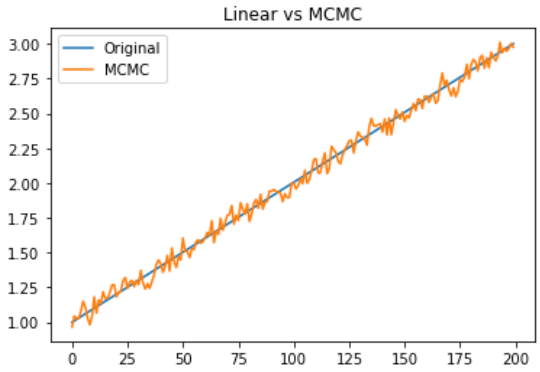
We use K-Times Markov chain Monte carlo sampling for the following advantage:

1. Misclassification rates are smaller.
2. The total time of sampling and training is less.
3. The obtained classifiers are more sparse.

II. DATASET DESCRIPTION

We have used 2 datasets for training and classification purposes. One of these papers is provided by UCI and the other one dataset was created for Visual object classes challenge and is commonly referred to as Pascal2.

The first dataset consists of 20000 instances of 26 Capital letters in the english alphabet. The images are based on 20 different fonts where each letter within these 20 fonts is randomly distorted to produce 20,000 unique stimuli. We train our SVM on the first 16000 stimuli and use the rest 4000 to test our Model.



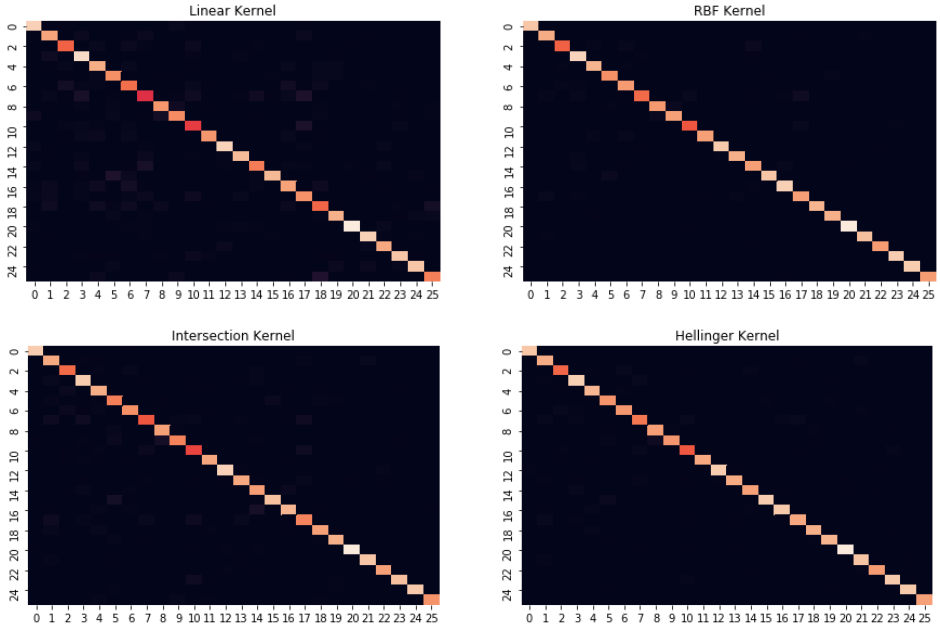
The second dataset consists of random objects in images. This dataset ‘s main purpose is to teach the model to recognise objects in an image. The training dataset consists of 20 classes of images and 11,530 images in total and 27,450 ROI annotated objects with 6,929 segmentations.

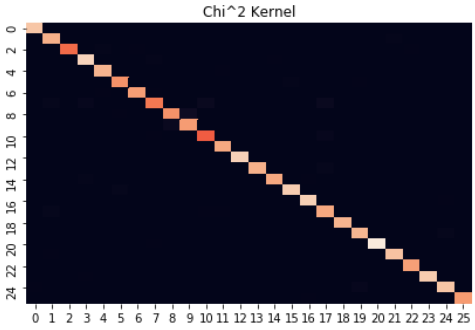
IV. RESULTS

Kernel	KPCA	SVDD	OCSVM	OCSSVM	OCSSVM with SMO	KT_SVM
Linear	0.02	0.09	0.01	0.07	0.04	0.15
RBF	0.05	0.07	0.14	0.09	0.04	0.04
Intersection	0.18	0.01	0.04	0.26	0.22	0.08
Hellinger	0.01	0.02	0.02	0.13	0.1	0.05
χ2	0.18	0	0.02	0.18	0.17	0.03

KT_SVM is the model demonstrated in this paper which involves using k times markov sampling, generally for this dataset we used **k=2** as mentioned in the research paper. After using that algorithm we notice the accuracy as mentioned in the last column of the figure above.

V. CONCLUSION





The above image confusion matrices obtained from using different kernels on an SVM. As visible, the linear kernel contains a lot of false positives and false negatives, however the chi squared kernel gives a confusion matrix with very few number of false positives and false negatives that it is more accurate than the linear kernel. Also, using Markov chaining along with chi squared kernel has improved the accuracy over traditional models as noticed in results.