DMW Assignment-3

Submitted By - [Akhil Shukla, IIT2018112] [Akhil Singh, IIT2018198][Javed Ali, IIT2018501][Manan Bajaj, IIT2018502][Lokesh, IIT2018503]

6 th Semester, B.Tech, Department of Information Technology, IIIT Allahabad

You have to understand the algorithm proposed in the paper "Generalization ability of SVM classification based on Markov Sampling ".

Run the algorithm on the shared pascal and Letter dataset and show the accuracy in terms of the attached image table: (make one more column in the last name MS_SVM with the new algorithm and give the result.

Markov Sampling Algorithm Implementation

We use Letter Dataset[2], it has 16 different features relating to alphabets A and B (for forming a binary classifier as given in paper) to be recognized. First we segment the dataset into a train and test set with 1088 samples for training and 467 for testing. We use markov sampling (explained next) to choose samples from the training set that forms a markov chain.

Markov Sampling Algorithm

Step 1: Let m be the size of training samples and m%2 be the remainder of m divided by 2. m+ and m− denote the size of training samples which label are +1 and −1, respectively. Draw randomly N1(N1 ≤ m) training samples {zi}N1 i=1 from the dataset Dtr. Then we can obtain a preliminary learning model f0 by SVMC and these samples.
Set m+ = 0 and m− = 0.

We used N1 = 800, m = 1088

Step 2: Draw randomly a sample from Dtr and denote it the current sample zt. If m%2 = 0, set m+ = m+ + 1 if the label of zt is +1, or set m− = m− + 1 if the label of zt is −1.

Step 3: Draw randomly another sample from Dtr and denote it the candidate sample z*.

Step 4: Calculate the ratio P of e−(f0,z) at the sample z* and the sample zt, P = e−(f0,z*) / e−(f0,zt)
.
Step 5: If P = 1, yt = −1 and y* = −1 accept z* with probability P = e−y*f0 /e−ytf0 . If P = 1, yt = 1 and y* = 1 accept z* with probability P = e−y*f0 /e−ytf0 . If P = 1 and yty* = −1 or P < 1, accept z* with probability P. If there are k candidate samples z* can not be accepted continuously, then set P = qP and with probability P accept z*. Set zt+1 = z*, m+ = m+ +1 if the label of zt is +1, or set m− = m− + 1 if the label of zt is −1 [if the accepted probability P (or P’, P’’) is larger than 1, accept z* with probability 1].

Step 6: If m+ < m/2 or m− < m/2 then return to Step 3, else stop it.

Then we train the SVM Classifier with different kernels using the markov samples. The final classifier is tested against the test dataset and performance recorded.

Various SVM kernels (Hellinger, Intersection and Chi Squared) that are not directly implemented in SVC are implemented using custom kernels which follow the mentioned kennel functions-

kernel	$k(x, y)$
Hellinger's	$\sqrt{xy}$
χ^2	$2\frac{xy}{x+y}$
intersection	$\min\{x, y\}$

Figure1 : Simplified Kernel functions from [6]

Hellinger is a simple square root of dot product, Chi Squared is implemented using “AdditiveChi2Sampler” and “SGDClassifier” of sklearn. SGD classifier follows SGD training for SVM cited[3, 4]. Intersection kernel is implemented as a simple inner product of X and Y matrices cited[5, 7].

Observation
N1 = 800 samples (Initial training sample for initial SVMC model in sampling)
Accuracy on Linear Kernel SVM - 99.1434 %
Accuracy on RBF Kernel SVM - 99.3576 %
Accuracy on Polynomial Kernel SVM - 99.1434 %
Accuracy on Hellinger Kernel SVM - 98.2869 %
Accuracy on Chi Squared Kernel SVM - 99.4542 %
Accuracy on Intersection Kernel SVM - 99.1434 %

Misclassification Rate on Linear Kernel SVM - 0.8566 %
Misclassification Rate on RBF Kernel SVM - 0.6424 %
Misclassification Rate on Polynomial Kernel SVM - 0.8566 %
Misclassification Rateon Hellinger Kernel SVM - 1.7131 %
Misclassification Rate on Chi Squared Kernel SVM - 0.5458 %
Misclassification Rate on Intersection Kernel SVM - 0.8566 %

A3 SVM Misclassification						
Kernel	KPCA	SVDD	OCSVM	OCSSVM	OCSSVM with SMO	MS_SVM
Linear	0.02	0.09	0.01	0.07	0.04	0.0085
RBF	0.05	0.07	0.14	0.09	0.04	0.0064
Intersection	0.18	0.01	0.04	0.26	0.22	0.0085
Hellinger	0.01	0.02	0.02	0.13	0.1	0.0171
Chi Squared	0.18	0	0.02	0.18	0.17	0.0054

References

[1] Xu, Jie, et al. "The generalization ability of SVM classification based on Markov sampling." *IEEE transactions on cybernetics* 45.6 (2014): 1169-1179.

[2] Letter Dataset - https://archive.ics.uci.edu/ml/datasets/Letter+Recognition

[3] sklearn kernel approximations AdditiveChi2Sampler
https://scikit-learn.org/stable/modules/generated/sklearn.kernel_approximation.AdditiveChi2Sampler.html#sklearn.kernel_approximation.AdditiveChi2Sampler

[4] sklearn Linear Model SGD Classifier
https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.SGDClassifier.html

[5] A. Barla, F. Odone and A. Verri, "Histogram intersection kernel for image classification," Proceedings 2003 International Conference on Image Processing (Cat. No.03CH37429), Barcelona, Spain, 2003, pp. III-513.
doi: 10.1109/ICIP.2003.1247294

[6] Vedaldi A, Zisserman A. Efficient additive kernels via explicit feature maps. IEEE Trans Pattern Anal Mach Intell. 2012 Mar;34(3):480-92. doi: 10.1109/TPAMI.2011.153. PMID: 21808094.

[7] Intersection Kernel Function
https://stats.stackexchange.com/questions/48506/what-function-could-be-a-kernel